الجـزء الثانى

PART TWO

سـمات أسـاسـية

BASIC FEATURES

قـوائم الأدلـة والأوامـر المصاحبـة
DIRECTORY LISTING AND ASSOCIATED COMMANDS

نظراً لأنك انتجت الدليل الفرعى \DB4\STUDY فى الفصل الثالث.. فإنه يكون فارغاً حالياً. إلا أنك تدخل فى بعض الأوقات، فى أدلة، بها أنواع عديدة مختلفة من الملفات وتسمح لك الأوامر التالية برؤية الملفات الموجودة – فعلاً – فى أحد الأدلة الجزئية التى تتصل بها.

بافتراض وجود بعض الملفات فى الدليل الحالى.. فإن :

.DIR <cr> يبين كل الملفات الموجودة فى الدليل الحالى، ولها الاسم الثانوى DBF.. وفى كلمات أخرى يبين لك هذا الأمر ملفات بيانات dBASE بصورة تقليدية.

.DIR * .NDX <cr> يبين كل الملفات التى تنتهى بـ NDX.

.DIR * .TXT <cr> يبين كل الملفات التى تنتهى بـ TXT.

.DIR * .DPT <cr> يبين كل الملفات التى تنتهى بـ DPT. وهكذا.

.DIR * . * <cr> يبين كل الملفات الموجودة فى الدليل الحالى بغض النظر عن أسمائها.

DIR P * . * <cr>. يبين كل ملفات بيانات dBASE الموجودة فى الدليل الحالى، ولها اسم أولى، يبدأ بالحرف P.

. DIR * . F * <cr> يبين كل الملفات الموجودة فى الدليل الحالى، ولها أى اسم أولى، إلا أن اسمها الثانوى، لابد أن يبدأ بالحرف F:. وغيرها، وغيرها.

فى حالة ما إذا كان مطلوبا اخلاء أماكن على القرص أو تنظيمه.. يمكنك استخدام الأوامر التالية:

لحذف ملف استخدم DELETE FILE XXX. YYY، يليها الضغط على مفتاح الادخال، وهذا الأمر يحذف الملف المسمى فى الأمر.

لا يمكنك حذف ملف إلا إذا كان الملف له رموز حالة status-codes، أو خواص قراءة وكتابة read/write فقط. لا يمكن حذف الملف الذى له رموز حالة أو خواص قراءة فقط /read

only (أى أنه ملف محمى)، رموز الحالة والخواص مشروحة فى دليل نظام التشغيل المتاح لك استخدامه، وكذلك إذا حذفت ملف dBASE فى تكوينه حقل مذكرة memo-field.. فلاحظ أن ملف المذكرة المصاحب (ملف DBF.) لا يحذف تلقائياً، إلا أن حقل المذكرة القائم بذاته لا يمكن استخدامه، إلا إذا أشير له من خلال ملف dBASE. (ملفات المذكرة Memo files وحقول المذكرة موضحة فى المواقع المناسبة فى الكتاب.)

لاعادة تسمية ملف.. استخدم RENAME ABC. F1 TO XYZ. F2، يليها الضغط على مفتاح الادخال، ويعيد هذا الأمر تسمية الملف كما هى محددة فيه تماماً. إذا أعدت تسمية ملف dBASE فى تكوينه حقل ذاكرة.. فلا تعاد تسمية ملف المذكرة المصاحب له تلقائياً، وتفشل على ذلك محاولتك استخدام ملف dBASE، الذى أعيدت تسميته؛ لأن نظام dBASE يذكر لك فى هذه الحالة أنه لا يمكن فتح ملف المذكرة. وعلى هذا.. يجب أن تعيد تسمية الاسم الأولى لملف المذكرة، كذلك قبل أن تستطيع استخدام ملف dBASE الذى أعيدت تسميته. (ملفات المذكرة وحقول المذكرة موضحة فى المواقع المناسبة من الكتاب.)

كما يمكنك أن تستخدم كذلك أمر RUN؛ للحصول على قائمة الدليل، وإزالة الملفات المعاد تسميتها عن طريق DOS :

RUN DIR *.* . يبين كل الملفات الموجودة فى الدليل الحالى، بغض النظر عن اسمائها.

RUN DEL XXX.YYY . يحذف الملف المسمى XXX.YYY.

RUN REN ABC. F1 XYZ. F2 . يعيد تسمية الملف ABC. F1؛ ليأخذ الاسم XYZ.F2.

لاحظ الاختلاف الطفيف فى تشكيل أمرى RENAME و DELETE؛ طبقاً لما إذا كانا محددين على مستوى DOS أو على مستوى dBASE. وكما سبق توضيحه فى قسم سابق.. يمكنك أن تنفذ أى أمر من أوامر نظام التشغيل DOS، وذلك باستخدام أمر التداخل RUN.

عملية الإنتاج THE CREATION PROCESS

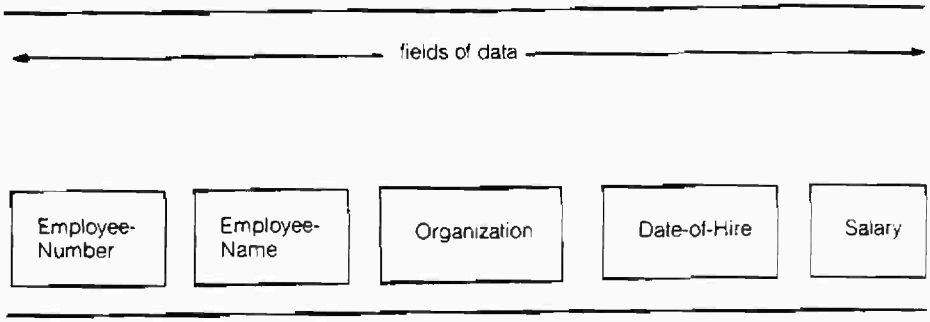
لقد انتجت فعلاً عند هذه المرحلة أول قاعدة بيانات باستخدام dBASE، ويتطلب هذا – كما هو الحال مع معظم أنشطة تشغيل البيانات – بعض التخطيط المسبق.

من المهم جداً أن تفهم كيف تتخيل البيانات التى تريد تتبعها، وذلك قبل أن يمكنك أن تبدأ العملية الفعلية لانتاج قاعدة بيانات باستخدام dBASE.

انواع الحقول FIELD TYPES

افرض انك تريد تتبع أحد العاملين فى شركتك وأنك متخيل، سواء كان هذا فى ذهنك أو على ورق، انك تريد تتبع رقم تعريف العامل Employee-Number، واسمه Name، ومؤسسته Organization، وتاريخ تعيينه Date-of-Hire، وراتبه Salary، وذلك لكل عامل من العاملين. عند هذه النقطة.. تكون قد تخيلت هيكل أو تكوين قاعدة البيانات؛ أى إنه بتحديدك حقول المعلومات التى تريد تتبعها.. فإنك تكون قد أخذت أول خطوة، فى اتجاه تحديد تكوين قاعدة البيانات المقترحة.

وقل الآن أنك تخيلت كيف يبدو هيكل البيانات لكل من العاملين، الذى يشبه شكل II-1.



شكل II-1 : هيكل البيانات لكل من العاملين.

يتم تحديد الأسماء الفعلية للحقول الموجودة فى الهيكل الافتراضى فيما بعد، إلا أن الأسماء سالفة الذكر كافية الآن. والآن.. عليك أن تأخذ خطوة ثانية للامام، وتترك dBASE يعرف ما ستكون عليه محتويات الحقول. وبعد كل شىء.. فإن الأسماء الفعلية لا تعنى أى شىء لقاعدة البيانات، وتحتاج إلى إعدادها، مع الحذر من أنواع البيانات التى تتعامل معها فى كل حقل.

يمكنك أن تعرف الحقل بأنه من أحد الأنواع الستة : حرفى Character، أو عددى صحيح Numeric BCD، أو عددى له نقطة (علامة عشرية) متحركة Numeric Floating Point، أو تاريخ Date، أو منطقى Logical، أو مذكرة Memo.

الحقول الحرفية والعددية CHARACTER AND NUMERIC FIELDS

أساساً.. فإن الحقل المعرف بأنه حقل حرفى يقبل أى رمز أو بيانات يتم ادخالها وراء بعضها، بينما لايقبل الحقل المعرف أنه من النوع العددى إلا الأرقام من 0 إلى 9، والعلامة العشرية، واشارة السالب فقط كبيانات. (محاولة ادخال بيانات حرفية فى حقل عددى تبطل عمل لوحة المفاتيح.) إلا أنه قد تحدث مواقف مبهمة، ويكون عليك أن تقوم بحلها. خذ - مثلاً - حقل رقم تعريف العامل، وافرض انك تعرف بالنسبة لادخال البيانات الفعلية - فى التكوين - أن البيانات التي يجب ادخالها فى هذا الحقل، يجب ألا تشتمل إلا على الأرقام من 0 إلى 9 فقط؛ أى إنه لن تكون هناك رموزاً خاصة (مثل $ أو @ أو * أو غيرها) أو حروفاً كمدخلات لهذا الحقل. هل تعرف حقل رقم تعريف العامل على أنه عددى الآن؟

الاجابة لا... لأن الاختبار الوحيد اللازم لتحديد ما إذا كان الحقل عددياً على عكس كونه حرفياً، هو أن تسأل نفسك السؤال التالى : هل أتوقع أن أجرى أى نوع من أنواع الحسابات على هذا الحقل؟ هل سأضيف أرقام تعريف العاملين على بعضها، أو هل سآخذ أحد سجلات العاملين، وأضيف رقم التعريف الموجود فيه إلى رقم الضمان الاجتماعى الموجود فيه؟ ونظراً لأن إجابة هذه الأسئلة هى... لا؛ فيجب أن تعرف مثل هذه الحقول (رقم تعريف العامل ورقم الضمان الاجتماعى) بأنها حقول حرفية، حتى إذا كان متوقع لها أن تحتوى على أرقام فقط.

وحقل مثل حقل الراتب.. يجب أن يعرف على أنه من النوع العددى.

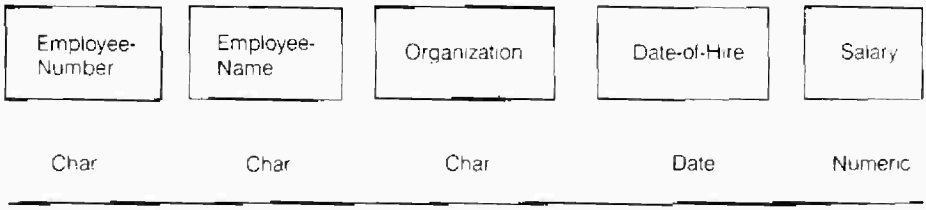
حقول التاريخ DATE FIELDS

عندما يعرف الحقل بأنه من نوع التاريخ.. فيفترض نظام dBASE أنك ستدخل التاريخ على الصورة MM/DD/YY، وعندما تدخل فعلاً البيانات فى هذا الحقل.. يتأكد نظام dBASE تلقائياً من دقة البيانات التى تدخلها؛ فمثلاً يؤدى ادخال التاريخ على الصورة التالية

12/35/88 إلى أن تصدر عنه صفارة من الجهاز مع رسالة خطأ. ويتأكد التنقيح المبنى داخلياً من السنة الكبيسة leap year كذلك! وحقول التاريخ مفيدة جداً فى تقليلها من مجهود البرمجة اللازم لإعداد المقاطع التى تحسب الأوقات المنقضية؛ لانك تستطيع جمع اعداد عليها أو طرح أعداد منها (معطية لك حقول تاريخ أخرى نتيجة لذلك)، أو يمكنك جمع أو طرح حقلين تاريخ مباشرة (مع اعطاء النتيجة باليوم). ويمكن أن تتراوح قيم التواريخ من 01/01/0100 إلى 12/31/9999، وتوضح الآلية الفعلية لاستخدام حقول التاريخ فيما بعد.

فى هيكلنا التخيلى.. يعرف حقل تاريخ التعيين Date-of-Hire بأنه من نوع التاريخ.

وعند هذه المرحلة.. يشبه هيكلنا التخيلى، ما هو موجود فى شكل II-2 :



شكل II-2 : أنواع الحقول فى هيكل البيانات.

الحقول المنطقية LOGICAL FIELDS

الحقل المنطقى هو الحقل الذى لا يقبل إلا حرف واحد من الأربعة حروف : إما الحرف Y، أو T (ليناظر YES أو TRUE)، أو الحرف N، أو F (يناظر NO أو FALSE). وتخزن البيانات الفعلية كما أدخلت تماماً، إلا أنها تعرض على الشاشة، أو على الطابع فى الصورة .F.، أو الصورة .T. فقط... فإذا ما لم يتم ادخال بيانات، فالقيمة التقليدية هى .F..

فمثلاً.. ذلك إذا عرف حقل اسمه EXEMPT بأنه حقل منطقى.. فقد تريد ادخال T فى حقول EXEMPT لكل العاملين المستثنيين، مع ترك حقول EXEMPT للعاملين الآخرين، دون إدخال شىء ليحدد لهم .F. تقليديا. ونظراً لأن طبيعة الحقل أنه منطقى.. فمن الممكن الاتصال بمثل هذه الحقول لمعرفة محتوياتها، دون استخدام أى مؤثر علاقى.

وكمثال فعلى.. فإن أمر dBASE يستخدم حقلاً منطقياً اسمه EXEMPT كما سبق ذكره (سيرد العديد من الأمثلة فيما بعد) لنقل أنك تريد قائمة بكل العاملين المستثنيين. فالأمر التالى :

DISP FOR EXEMPT <cr>

يؤدى العمل.

قارن هذا بالموقف الذى يعرف فيه حقل EXEMPT بأنه حقل من النوع الحرفى، مع إدخالك T كبيانات فى الحقل لكل العاملين المستثنيين. وفى هذه الحالة.. يجب أن يتغير الأمر سالف الذكر ليأخذ الشكل التالى :

. DISP FOR EXEMPT = 'T' <cr>

لاحظ أن المؤثر العلاقى (=) والرمز 'T'، استخدما الآن لتوضيح المعنى لنظام dBASE.

حقول المذكـــرة MEMO FIELDS

تسمح حقول المذكرة للمستفيد بحفظ مذكرات (ملاحظات أو وثائق) لسجلات فردية. ويمكن أن يصل حجم كل محتوى لحقل المذكرة إلى 512 كيلوبايت (أو 512000 رمزاً أو خانة). اعتبر - على سبيل المثال - مكتب طبيب أسنان؛ حيث يريد مدير المكتب استخدام نظام dBASE فى إعداد مذكرة موجزة للمريض، وتظهر هذه المذكرة بعد ذلك على الفاتورة التالية للمريض، ويمكنك باستخدام مثل حقول المذكرة هذه... حفظ مذكرات مختلفة لسجلات فردية فى الملف الرئيسى master.

يستخدم نظام dBASE ملفات خارجية، تخزن فيها محتويات المذكرات الفردية، ويمكن لكل مذكرة فردية أن تتسع لتشمل 512 كيلوبايت، ويكون لهذا الملف الخارجى نفس الاسم الاولى لملف dBASE الرئيسى، إلا أنه يكون له الاتساع DBT. كاسم ثانوى له.

فاذا كان لديك - على سبيل المثال - ملف اسمه PERSNL . DBF، ومعرف فى تكوينه حقل مذكرة فينتج نظام dBASE ملفاً اسمه PERSNL . DBT بصورة تلقائية، يدعم حقول المذكرة الموجودة فى الملف PERSNL . DBF.

يحفظ نظام dBASE هذا الملف فى صورة صالحة للاستخدام الداخلى، وتوضح الآلية الفعلية لاستخدام حقول المذكرة فيما بعد.

أحجام الحقول FIELD WIDTHS

لقد تعلمت عند هذه النقطة أنك إذا أردت تخيل هيكل قاعدة بيانات.. فيجب أن تحدد حقول المعلومات المختلفة، التى تريد حفظها لكل كينونة أو مفردة entity، ويجب أن تحدد نوع البيانات التى تشغل كل حقل. وتكون هناك حاجة إلى تقديم جزء اضافى من المعلومات؛ لإكمال هيكل قاعدة البيانات المقترحة. وكما يمكن أن تكون قد خمنت.. فعليك أن تخطر نظام dBASE بحجم الحقول: ماعدد رموز المعلومات الذى تتوقعه كبيانات لرقم تعريف كل عامل أو لاسم كل عامل؟ و.. هكذا.

الحقول الحرفية : دعنا نقول إن أى محتوى لحقل رقم تعريف العامل لن يزيد عن 6 رموز بيانات، وأن حقل اسم العامل يحتوى حتى 15 رمزاً من البيانات، وأن حقل المؤسسة يحتوى حتى 3 رموز بيانات، ويمكنك أن تختار عدد الخانات اللازمة لكل حقل، كما يتراءى لك، وبحد أقصى 254 خانة للحقل الواحد.

الحقول العددية : الحقول - مثل حقل الراتب - تكون فى حاجة إلى عناية خاصة، وافرض أنك تخيلت أن البيانات المراد ادخالها فى حقل الراتب تكون على الصورة 99999:99 (أى خمس خانات للأرقام، التى تسبق العلامة العشرية، وخانتان بعد العلامة العشرية). بالنسبة للرواتب التى يمكن أن تصل إلى $99,999.99 فى السنة.. يكون هذا كافياً، ويخدم كذلك فى تخزين متوسط الراتب. ويجب أن يعرف حجم هذا الحقل الآن على الصورة التالية 8,2 (الطول الاجمالى 8 خانات، منها خانة للعلامة العشرية، وخانتان للكسر العشرى).

كتوضيح أكثر لهذه النقطة.. أدخل حقلاً ليس له أى معنى منطقى فى نوع قاعدة البيانات التى تستخدمها، إلا أنه نو تطبيق محدد فى مكان آخر. واعتبر حقلاً مثل الحقل من نوع المدين والدائن debit/credit المستخدم فى التطبيقات المالية، ولكى تحدد حجم هذا الحقل..

عليك أن تعتبر – مرة أخرى – نوع البيانات التى تشغل الحقل فعلاً. افرض – كمثال – أن الحقل مصمم ليحتوى على 3 خانات قبل العلامة العشرية، وخانتين بعدها، ويمكن أن يشتمل على قيم سالبة، يتم إدخالها أثناء ادخال البيانات؛ أى إنه من الممكن أن يكون لديك رقماً مثل 999.99، ويجب أن يعرف حجم هذا الحقل على الصورة التالية : 7,2 (إجمالى 7 خانات: ثلاثة منها لأرقام صحيحة تسبق العلامة العشرية، وخانتان بعد العلامة العشرية، وخانة للعلامة العشرية، وخانة أخرى لاشارة السالب الممكنة). كل الحقول العددية محددة بالنسبة للنوع "F"، ويشمل هذا وجود العلامة العشرية والاشارة.

حقول التاريخ : دائماً ما يكون لكل الحقول المعرفة بأنها من نوع التاريخ حجم سابق تعريفه فى الصورة YYYYMMDD، والقرن التقليدى هو 19XX.

الحقول المنطقية : لكل الحقول المنطقية حجم سابق التعريف، وهو خانة واحدة؛ لأن مثل هذا الحقل لا يمكن أن يقبل إلا حرف واحد من الحروف الأربعة /T/F/Y/N فقط كمدخلات له.

حقول المذكرة : لكل حقول المذكرة حجم سابق تعريفه كذلك، وهو 10 خانات فى ملف DBF.. ونظراً لأن نظام dBASE يخزن المحتويات الفعلية للمذكرات فى ملف منفصل (له الاتساع DBT.).. فانه يستخدم هذا الحقل الذى حجمه 10 خانات – فى ملف DBF. – فى تخزين موقع المذكرة فى ملف DBT..

وفى مثالنا يشبه الهيكل النهائى الآن شكل II-3.

Employee-Number	Employee-Name	Organization	Date-of-Hire	Salary (5).(2)
C	C	C	D	N
6	15	3	8	8,2

شكل II-3 : هيكل البيانات النهائى.

أسماء الحقول FIELD NAMES

كل حقل موجود فى هيكل قاعدة بيانات.. يجب أن يكون له اسم فريد به، ومختلف عن بقية أسماء الحقول الموجودة فى نفس الهيكل. ويمكن لأسماء الحقول أن تكون أى أسماء يقع عليها اختيارك، إلا أننا ننصح بشدة بأن تكون الأسماء ممثلة للبيانات التى صمم الحقل ليحتويها. فمثلاً يمكنك أن تعرف ENO كإسم لحقل رقم تعريف العامل، إذا رغبت فى ذلك، على أساس أن هذا الاسم له انطباع معين لديك، إلا أنه لا يعطى أى انطباع لأى شخص آخر، يعمل بعدك مع نفس قاعدة البيانات. وعلى هذا.. فقد تستخدم EMPNO كاسم لهذا الحقل، وهذا بالطبع سهل فى دلالته بالمقارنة بالاسم ENO لرقم تعريف العامل Employee-Number، إلا أنك إذا أردت دقة أكثر.. فيمكنك استخدام الاسم EMP-NUM.

كمثال آخر لحقل تاريخ التعيين.. يمكنك أن تستخدم DT-OF-HIRE بدلاً من DTHR. ولتجزئة الكلمات لتسهيل القراءة.. استخدم الشرطة التى توضع تحت الحرف، وإذا استخدم فراغ بدلاً من ذلك.. فإنه يتسبب فى حدوث خطأ.

القاعدة الوحيدة هى :

يجب ألا يزيد الطول الكلى لاسم الحقل نفسه عن 10 خانات، بما فى ذلك الخانة الخاصة بالشرطة، التى توضع تحت الحرف، والمستخدمة كفاصل لتسهيل القراءة.

لا تخلط بين طول اسم الحقل وحجم الحقل نفسه؛ فمثلاً... ذلك الحقل المسمى EMP-NAME معرف بأن طوله 15 خانة، إلا أن EMP-NAME نفسها لا تشغل إلا 8 خانات فقط.

الفصل السابع

إنتاج هيكل قاعدة بيانات

CREATING A DATABASE STRUCTURE

نتناول الآن ميكانيكية تعريف هيكل قاعدة البيانات لنظام dBASE. بعد دخولك نظام dBASE.. فإن ظهور ملقن النقطة يعنى أن النظام ينتظر منك أن تدخل أحد الأوامر.

أمر انتاج الهيكل هو أمر CREATE.

تحذير : إذا طلبت من نظام dBASE أن ينتج ملفاً، إلا أنك ذكرت عن طريق الخطأ اسماً لملف موجود فعلاً.. فيقدم لك النظام رسالة تحذيرية. وتحدد الرسالة التحذيرية أن الملف موجود بالفعل، ويعطيك النظام الفرصة إما لإلغاء الملف الموجود والكتابة فيه، أو لإلغاء أمر CREATE. ولسوء الحظ.. إن الاجراء التقليدى هو إلغاء الملف الموجود فعلاً! وهذا يعنى أنك إذا كنت عجولا.. فمن الممكن أن تلغى ملفاً موجوداً فعلاً بضغطك على مفتاح الإدخال! وقد يكون هذا الملف مهماً بل ومطلوباً فيما بعد؛ لذلك يجب أن تكون حذراً جداً مع هذا البديل.

إذا استخدمت نفس اسم الملف الموجود حالياً كاسم لملف جديد (سواء كان هذا بمحض اختيارك أم لا) وإذا غيرت بعد ذلك من رأيك.. فيكون قد فات الأوان. لاحظ أنه عند هذه النقطة.. فإن تغييرك رأيك لا يعيد لك الملف الذى كان موجوداً من قبل، وعلى هذا.. فعليك أن تفكر أكثر من مرة، قبل أن تفعل ذلك.

عندما تختار اسماً لملف.. يمكنك أن تكتب أمر الانتاج على النحو التالى :

. CREATE <cr>

أو على النحو التالى :

. CREATE <filename> <cr>

فإذا لم تدخل اسم الملف فى سطر الأمر نفسه.. فسوف يطلب منك نظام dBASE أن تدخله فى خطوة تالية.

افرض أنك سميت الملف الاسم PERSNL. وبعد انتاجك هذا التكوين.. سيصبح الاسم الفعلى له فى النظام PERSNL . DBF، وذلك لأن نظام dBASE يضيف DBF. كاسم ثانوى للملف.

. CREATE PERSNL <cr>

تأثير هذا الاجراء هو إظهار نظام dBASE شكل 1-7 على الشاشة، متوقعاً إدخالك لمحتويات تعريف الملف.

```
Layout   Organize   Append   Go To   Exit

Bytes remaining:   3946
Num  Field Name  Field Type  Width  Dec  Index

  1  EMP_NUM     Character       3         N
  2  EMP_NAME    Character      13         N
  3  TOWN        Character       9         N
  4  ORG         Character       3         N
  5  DT_OF_HIRE  Date            8         N
  6  SALARY      Numeric         8     2   N
  7  NOTE        Memo           10         N
  8              Character                 N
```

شكل 1-7 : تعريفات الحقول للهيكل المقترح.

يريد نظام dBASE منك - اساساً - أن تحدد اسماً لكل حقل، تريد إدخاله، ونوع هذا الحقل، وحجمه، وعدد الخانات المخصصة للكسر العشرى إذا كان الحقل عددياً (له نقطة ثابتة أو نقطة متحركة)، وماإذا كان هذا الحقل مستخدماً كحقل رئيسى للاتصالات السريعة التالية بالسجلات، وذلك لأن dBASE يدعوك - الآن - إلى تحويل الهيكل التخيلى لقاعدة البيانات إلى واقع فعلى.

يمكن ادخال المحتويات الفعلية للحقول، إما بحروف كبيرة، أو بحروف صغيرة.

ملاحظة :

عندما تطلب إنتاج CREATE ملفاً.. فقد تغير بالطبع رأيك بالنسبة لهذا الأمر، وفى هذه الحالة ما عليك إلا أن تضغط على مفتاح Esc، ويطلب منك dBASE أن تؤكد هذا الاجراء.

عرف حقول المعلومات للهيكل المقترح الآن. من فضلك استمر فى انتاج الملف المسمى PERSNL (.DBF)، كما أوضحها هنا، لاننى سأستخدم هذا الملف فى الأمثلة المستخدمة فى هذا الكتاب.

ملاحظات على امر الانتاج

OBSERVATIONS FOR THE CREATE COMMAND

١ - يمكنك أن تطلب تعريف الملف، إما على قرص مرن، أو على القرص الثابت. ارجع إلى الصفحات السابقة لمراجعة هذا الجزء إذا لزم الأمر.

٢ - لا تكتب على لوحة المفاتيح بسرعة مع أمر CREATE! فإذا طلبت انتاج ملف باسم معين، وكان هناك ملف بنفس الاسم فى نفس الدليل المستخدم.. فسوف يقدم لك نظام dBASE رسالة تحذيرية، والتى نأمل أن تراها؛ فالضغط المتسرع فى هذه اللحظة على مفتاح الادخال، سوف يخفى الرسالة التحذيرية، ويلغى كذلك الملف الموجود حالياً بنفس هذا الاسم فى الدليل الحالى، حتى إذا غيرت رأيك بالنسبة لإلغاء الملف الجديد.

٣ - يمكنك أن تغير رأيك بالنسبة إلى إنتاج الملف الجديد، وذلك بالضغط على مفتاح Esc فى أى لحظة أثناء إنتاج الملف. وبالرغم من أن هذا لا ينتج ملفاً جديداً، فإذا كنت قد استخدمت اسماً لملف قديم... فلا يمكن استعادة هذا الملف القديم أبداً.

٤ - النوع التقليدى للحقول فى نظام dBASE هو الحرفى، فإذا كنت تعرف حقلاً حرفياً فما عليك إلا أن تضغط ببساطة على مفتاح الادخال؛ لنقل نقطة البداية إلى حقل WIDTH، واستمر فى التعريف.

إذا أردت تعريف حقلاً عددياً.. اكتب ببساطة N كأول رمز لتعريف حقل عددى، له نقطة ثابتة، أو أكتب F لتعريف حقل عددى له نقطة متحركة**، وهذا يغير التعريف إلى Numeric أو Float، وتنتقل نقطة البداية إلى حقل WIDTH.

إذا كتبت L لحقل على أنه حقل منطقى.. فسوف يظهر لك نظام dBASE الحجم التقليدى للحقل، وهو 1، وينقل نقطة البداية إلى تعريف الحقل التالى.

بالمثل... تجعل كتابة D لحقل من نوع التاريخ النظام يقدم 8 لحجم تقليدى للحقل، كما أن كتابة M لحقل من نوع المذكرة، تجعل النظام يقدم 10 كحجم تقليدى للحقل، وينقل نقطة البداية إلى تعريف الحقل التالى.

لاحظ أنك تستطيع استخدام قضيب المسافات - كذلك - فى الحركة، خلال تعريفات الحقول المختلفة، وتضغط بعد ذلك على مفتاح الإدخال لاختيار نوع الحقل المطلوب.

٥ - مع انتاجك الهيكل.. يمكنك أن تحرك نقطة البداية حول الشاشة، باستخدام مفاتيح الأسهم، مع إجراء تغييرات على الحقول التى سبق ادخالها إذا كانت هناك حاجة لذلك.

استخدم Ctrl-N لبدء تعريف حقل جديد، فى موقع وجود نقطة البداية.

استخدم Ctrl-U لحذف تعريف حقل فى موقع نقطة البداية.

إذا كنت قد عرفت هيكلاً طويلاً (بحد أقصى 255 حقلاً).. يمكنك استخدام مفاتيح PgUp و PgDn للدوران للأمام أو للخلف.

سبق تقديم مفاتيح التحكم الأخرى، إلا أنك قد تجدها مفيدة جداً أو لا تجدها كذلك؛ طبقاً لامكانياتك الخاصة فى الكتابة باستخدام لوحة المفاتيح. ويمكن استخدام ذلك فى عمل تغييرات على أسماء الحقول الموجودة.

** الحقل العددى الذى له نقطة ثابتة هو الحقل المتوقع ان يشتمل على ارقام صحيحة فقط كمحتويات له. اما الحقل العددى الذى له نقطة متحركة فهو الحقل المتوقع ان تكون بياناته من النوع المشتمل على علامة عشرية وكسر عشرى (المترجم)

أ – *الضغط على Ctrl-G أو على مفتاح الحذف* : في كل مرة تضغط على هذين المفتاحين، وأنت في منتصف الحقل.. يحذف الرمز الموجود نقطة البداية تحته (ولا تتحرك نقطة البداية)، وتقل الرموز الموجودة على يمين نقطة البداية بمقدار رمز واحد.

ب – *الضغط على Ctrl-Y* : الضغط على هذا الخليط – أثناء وجودك في منتصف الحقل – يمحو محتويات الحقل من عند موقع نقطة البداية، حتى نهايته؛ مستبدلاً أياه بفراغات (تأثير الإخلاء حتى نهاية الحقل).

جـ – *الضغط على Ctrl-V، أو على مفتاح الاقحام Ins* : يؤدى ذلك وظيفة الادخال. افرض أن أحد الحقول سمى EMP-NM، بدلاً من EMP-NUM، وذلك عن طريق الخطأ، وتريد الآن إدخال الحرف U قبل الحرف M مباشرة (يمكنك بالطبع اعادة كتابة الاسم كله مرة أخرى إلا أن الفكرة الحالية، هي توضيح استخدام مفاتيح الاقحام (الادخال). انقل نقطة البداية إلى الحرف M، واضغط على Ctrl-V، أو على مفتاح Ins. لاحظ أن كلمة INSERT تظهر في قمة الشاشة، في سطر لوحة الاصابات، وذلك يعنى أنك في حالة الادخال insert mode، على عكس وجودك في حالة إعادة الكتابة على ما هو موجود over-write mode (وهى بالطبع الحالة التقليدية). اكتب الآن الحرف U، وسوف تجد أنه تم ادخاله في الحقل؛ أى إن الرموز الأخرى تتحرك لليمين؛ لتترك مكاناً فارغاً لهذا الحرف، وإذا دخلت رموز كثيرة داخل الحقل... فيفيض الحقل. بما فيه. حاول ذلك لرؤية التأثير الكلى لوظيفة الادخال. وللخروج من حالة الادخال.. اضغط على Ctrl-V، أو على مفتاح Ins مرة أخرى؛ أى إن الضغط على Ctrl-V أو على مفتاح Ins ينتقل من وإلى حالة الإدخال.

د – *الضغط على مفتاح العودة للخلف backspace* : الضغط على هذا المفتاح يتسبب في حركة نقطة البداية، ويتحرك الحرف الموجودة تحته موقعاً واحداً ناحية اليسار، مع حذف هذا الحرف، كما تتحرك كل الحروف الأخرى الموجودة على اليمين طبقاً لذلك، ويؤدى الاستمرار في الضغط على هذا المفتاح إلى حذف الحرف التالى والتالى له، وهكذا مخرجاً نقطة البداية ناحية اليسار.

٦ – إذا أدخلت نفس الحقل فى سطرين مختلفين (أى إنك تحاول انتاج نفس الحقل مرتين) فيصدر النظام صفارة لك.

٧ – اقصى حجم مسموح به لحقل حرفى هو 254 خانة.

٨ – أقصى حجم لعنصر عددى هو 20 خانة لدقة الحقول المعرفة بأنها N، و 15.9 لدقة الحقول المعرفة بأنها F، وتشمل هذه الأحجام العلامة العشرية واشارة السالب إذا كانت موجودة.

٩ – يمكنك أن تعـرف 255 حقلاً، كحد أقصى لعدد الحقول المعرفة فى هيكل أى قاعدة بيانات.

١٠ – داخل هذا القيد – بألا يزيد عدد الحقول المعرفة فى أى قاعدة بيانات – يجب ألا تعرف رموز بيانات أكثر من 4000 رمزاً. ومثال ذلك أنك عرفت فى الحقول السبع سالفة الذكر 54 رمزاً من البيانات. (3 لحقل EMP-NUM، و 13 لحقل EMP-NAM، و 9 لحقل TOWN، و 3 لحقل ORG، و 8 لحقل DT-OF-HIRE، و 8 لحقل SALARY و 10 لحقل NOTE).

١١ – عندما تعرف حقلاً أو أكثر بأنه حقل رئيسى.. ينتج نظام dBASE ملفاً آخر اسمه PE-RSNL.MDX، يدعم ملف PERSNL. DBF الذى سبق انتاجه، وذلك بصورة لقائية. ويعنى الاتساع MDX. ملفاً متعدد الفهرسة multiple index file، والحقول المحددة فى تكوين PERSNL. DBF بأنها حقول رئيسية، تستخدم كإشارات فهارس index-tags فى ملف PERSNL. MDX.

افرض أن الحقل TOWN محدد بأنه حقل رئيسى. ينتج dBASE الآن ملفاً اسمه PE-RSNL. MDX، وينتج إشارة فهرس فى حقل TOWN، داخل ملف PERSNL. MDX هذا.

ودائماً ما يكون dBASE ملماً بالحقول الموجودة فى الملف PERSNL. DBF المعرفة بأنها رئيسية، وذلك خلال الملف المصاحب له والمسمى PERSNL. MDX. فإذا أحضرت– بعد ذلك – ملف PERSNL للاستخدام USE.. فإن كذلك dBASE يفتح

كذلك- بصورة تلقائية - الملف MDX. المصاحب، وينشط إشارات الفهارس الموجودة فى ملف MDX.

وتحفظ الفهارس الملفات منظمة بترتيب معين، إلا أنه هناك كثير جداً عن الفهرسة عن هذه العبارة الواحدة، وسوف تدرس الفهرسة وتأثيراتها بتفاصيل كبيرة، فى قسم لاحق من هذا الكتاب، كما تدرس كذلك ملف MDX. وإشارات الفهرسة فى ذلك القسم.

١٢ - عندما تعرف حقلاً أنه حقل مذكرة.. فإن dBASE ينتج بصورة تلقائية ملفاً آخر اسمه PERSNL. DBT، يدعم ملف PERSNL. DBF. وتعنى DBT. نص قاعدة البيانات data-base text، ويستخدم ملف PERSNL. DET لأنه يحتوى على المحتويات الفعلية للمذكرات الفردية.

فاذا أحضرت بعد ذلك ملف PERSNL للاستخدام USE.. فيفتح dBASE كذلك ملف DBT. المصاحب له بصورة تلقائية.

١٣ - عندما تريد الانتهاء من تعريف الهيكل.. اضغط على مفتاح الإدخال عند سطر فارغ.

والآن.. بعد أن انتهيت من تعريف الهيكل بضغطك على مفتاح الادخال.. فإن dBASE ينتظر منك الضغط مرة أخرى على مفتاح الادخال لتأكيد هذا الهيكل، كما أنه يسألك إذا ما أردت أن تدخل بيانات فى هذا الهيكل الآن أم لا. وعادة ما تكون الاجابة لا... نظراً لأنك فى العادة تريد أن تراجع الهيكل الذى عرفته أولاً، وإجراء أى تعديلات عليه قبل أن تستمر فعلاً فى إدخال البيانات داخله.

ادخل من فضلك N كإجابة بأنك لا تريد إدخال بيانات الآن؛ فسوف تدخل البيانات فيما بعد.

بعد إنتاجك تكوين قاعدة بيانات.. قد تكون الآن فى حاجة إلى مراجعة دقة التكوين، قبل أن تستمر فعلاً بوضع بيانات فيه، كما تريد كذلك الاحتفاظ بنسخة دائمة من التكوين لأغراض التوثيق الخاصة بك.

عرض هيكل الملف وتوثيقه

DISPLAYING AND DOCUMENTING THE STRUCTURE OF THE FILE

نظراً لأن الملف PERSNL. DBF قد انتج لتوه (الهيكل فقط، فلم يتم ادخال بيانات فيه حتى الآن) فانه يكون تحت الاستخدام USE فعلاً، وتحت تحكم dBASE كذلك. وعلى هذا.. يمكنك أن تستمر بإعطاء مزيد من الأوامر لتنفذ على هذا الملف مباشرة، وإذا كنت تعمل مع ملف آخر عادة، ثم أردت إحضار ملف PERSNL للاستخدام USE.. فعليك أن تدخل الأمر التالي: USE PERSNL..

وعند هذه اللحظة، كما سبق ذكره، يصبح الملف مستخدماً USE؛ لأن هيكله سبق إعداده.

والآن لعرض هيكل هذا الملف.. عليك بكتابة ما يلى :

. DISPLAY STRUCTURE <cr>

أو كتابة ما يلى :

. DISPLAY STRU <cr>

هذا يظهر الملف على الشاشة، والذى يكون له الشكل التالى :

```
Structure for database: C:\PROJECTS\DB4\DATA\PERSNL.DBF
Number of data records:       5
Date of last update    : 01/29/89
Field  Field Name  Type       Width     Dec     Index
    1  EMP_NUM     Character      3               N
    2  EMP_NAME    Character     13               N
    3  TOWN        Character      9               N
    4  ORG         Character      3               N
    5  DT_OF_HIRE  Date           8               N
    6  SALARY      Numeric        8       2       N
    7  NOTE        Memo          10               N
** Total **                      55
```

(تذكرة : يمكن فى dBASE اختصار أى أمر، أو أى مؤشر لأى أمر إلى الحروف الأربعة الأولى منه.)

وفى حالة الهيكل الطويل للملف.. فإن dBASE يعرض 16 حقلاً على الشاشة فى نفس الوقت، وينتظر منك أى إدخال قبل أن يعرض 16 حقلاً التالية وهكذا. إذا أردت طباعة هذا

الهيكل.. يمكنك أن تكتب ما يلى :

. DISP STRU TO PRINT <cr>

لاحظ أن عرض الهيكل يبين كذلك عدد السجلات الموجودة فى الملف، وتاريخ آخر تعديل أدخل على الملف.

تعديل (تغيير) الهياكل MODIFYING STRUCTURES

بعد مراجعة هيكل الملف الذى انتجته لتوك.. فمن المنطقى لك أن تأخذ الوقفة التالية : إذا أردت أن تجرى بعض التعديلات على هيكل قاعدة البيانات، فيكون الوقت مناسباً - الآن - لعمل ذلك، قبل أن تبدأ فى وضع البيانات. وربما ترغب فى ادخال حقل آخر جديد بين حقلى ORG و DT-OF-HIRE، أو ربما تريد تغيير اسم أو نوع أو حجم حقل، أو حذف حقل أو أكثر من الحقول التى سبق تعريفها. وبغض النظر عن نوع التغيير الذى تريده.. فمن المنطقى أن تجرى التغيير الآن، إلا أن هذا لايعنى أنك لا تستطيع تغيير هيكل أى قاعدة بيانات موجودة فعلاً ومحتوية على سجلات بيانات؛ إذ يمكنك أن تغير الهياكل فى أى وقت لأى قاعدة بيانات.

كلمة تحذير A WORD OF CAUTION

١ - إذا كنت تعدل هيكل قاعدة بيانات موجودة فعلاً - وبها سجلات - لاحظ أنه يجب عليك اختيار تعديلك بحذر. اعتبر الموقف التالى... افرض أن لديك حقلاً حرفياً اسمه EMP-NUM، وبياناته خليط من الأرقام والحروف، ويبدأ بأرقام، وتريد أن تغير نوعه إلى النوع العددى. ونظراً لأن dBASE لا يمكن منطقياً أن يحفظ بيانات حرفية فى حقل عددى؛ فعند الانتهاء من إجراء التعديل تفقد كل البيانات الحرفية التى كانت موجودة فى الحقل EMP-NUM! ولايبقى بالحقل العددى الجديد سوى الأرقام التى كانت فى بدايته فقط، أما إذا وجدت فى بداية الحقل الحرفى حروفاً بدلاً من أرقام.. فبعد التعديل لا تصبح فى الحقل الجديد أية بيانات.

٢ – لا يستطيع dBASE معالجة تغييرات مزدوجة لاسم حقل واحد، أو لأسماء عدة حقول، ومعالجة معظم التغييرات الأخرى فى نفس الوقت. وإذا لم تهتم بهذا التحذير ففى نهاية إجرائك التعديلات.. تجد أنك قد فقدت كل البيانات الموجودة فى كل السجلات لهذا الحقل مع تغيير الاسم! إذا أردت أن تغير اسم حقل وحجمه فى نفس الوقت (سواء كان هذا فى نفس الحقل أو فى حقول مختلفة) قم بإجراء أحد التغييرات أولاً (أى واحد منهما)، ثم احفظ الهيكل واستخدم بعد ذلك الهيكل الجديد فى عمل التغيير الآخر.

كملاحظة مفيدة.. إذا نتج عن تعديل الهيكل تغيير فى الطول الكلى للسجل.. فسوف ينتج عن التعديل فقدان فى بيانات الحقل الذى تغير اسمه.

٣ – إذا غيرت اسم حقل واحد أو أكثر (كتغيير وحيد)، وأردت أن تحفظ الهيكل الجديد.. يسألك dBASE (بالقرب من سطر الحالة) السؤال التالى :

هل يجب نسخ البيانات من الاحتياطى لكل الحقول؟ والإجابة إما نعم أو لا.

Should data be COPIED from backup for all fields? (Y/N)

ما يعنيه هذا هو أنك عندما غيرت اسم الحقل.. لم يصبح فى مقدور dBASE أن يقرر، وهذا أمر طبيعى جداً، هل كان هذا مكافئاً لحذف حقل موجود فعلاً وتعريف حقل جديد، أم أنه يغير ببساطة اسم الحقل الموجود فعلاً. وباجابتك بنعم على هذا السؤال.. تستطيع الاحتفاظ بكل بياناتك لكل السجلات لهذا الحقل، عند الانتهاء من اجراء تغيير الاسم. (هذا له علاقة بالاضافة APPEND الداخلية التى يجريها dBASE. أمر الإضافة APPEND موضح بالتفصيل فى المواقع المناسبة من الكتاب.)

٤ – إذا قللت حجم حقل عددى (على أنه هو التغيير الوحيد) تحفظ البيانات الأصلية من الحقل العددى، إذا كان من الممكن لها أن تحتوى فى الحقل الأصغر فقط. ومثال ذلك أنه إذا كان حجم الحقل الأصلى هو 5 خانات، وقللته إلى 3 خانات.. فإن القيمة الأصلية 12345 تفقد، أما إذا كانت القيمة الأصلية هى 123.. فسوف يحتفظ بها فى الحقل الجديد.

٥ – إذا قللت حجم حقل حرفى (على أنه هو التغيير الوحيد) فيمكنك أن تتوقع حذفاً فى البيانات من ناحية اليمين، والذى يمكن أن ينتج، أو لا ينتج، عنه فقدان للبيانات؛ طبقاً لمحتويات الحقل الحرفى الأصلية.

٦ – من الواضح أنك إذا وسعت حجم الحقل (على أنه هو التغيير الوحيد).. فلن تفقد أى بيانات.

٧ – يعتمد تعديل الهيكل على النسخ الاحتياطية التى انتجت، والبيانات التى يعاد نسخها فى الهيكل المعدل؛ فإذا أثارت interrupt جهاز الكمبيوتر فى منتصف هذه العملية.. فقد تفقد بياناتك.

٨ – السماح بالوقت والمكان، عندما تريد إجراء تعديل على أحد الملفات؛ فننصحك بعمل نسخة احتياطية لتكوين الملف المراد اجراء تعديل عليه؛ بحيث يمكنك أن تقارنه مع الهيكل المعدل، وتتأكد من سلامة البيانات.

فى هذه اللحظة.. فإنك انتجت الملف PERSNL. DBF، ولا توجد به سجلات بيانات، وتستطيع – بالطبع – تعديل هيكله دون أى تأثير ضار.

لتعديل هيكل قاعدة البيانات التى اعددتها لتوك.. استخدم ما يلى :

. USE PERSNL <cr> ليس ضرورياً فى الواقع؛ لأن الملف تم انتاجه على التو، وموجود فعلاً فى الاستعمال USE.

. MODIFY STRUCTURE <cr>

أو . MODI STRU <cr>

تحضر هذه الأوامر الهيكل على الشاشة، ويكون تشكيله متطابقاً مع التشكيل الذى سبق لك رؤيته، عندما انتجت الهيكل، كما أن التحكم فى نقطة البداية متطابق كذلك مع ما سبقت مناقشته فى قسم الانتاج creation.

بعد إجراء التعديلات المطلوبة على الهيكل.. يمكنك أن :

تضغط على Ctrl-W أو Ctrl-End لحفظ الهيكل الجديد، أو تضغط على Ctrl-Q أو Ctrl-Esc؛ لتغيير رأيك بالنسبة إلى التغييرات التى حدثت.

باستخدام خليط التحكم فى نقطة البداية الذى رأيته حتى الآن.. عدل هيكل الملف المسمى PERSNL. DBF؛ ليحتوى على حقل جديد اسمه EXEMPT (وهو حقل منطقى) بعد حقل ORG. يستخدم هذا الحقل لتتبع ما إذا كانت حالة العامل مستثناة أم لا، ولاحظ أن هذا الحقل المنطقى له حجم مشمول، هو خانة واحدة. اضغط على Ctrl-W؛ لحفظ الهيكل الجديد للملف. (الضغط على Ctrl-Q، أو على مفتاح الهروب يحتفظ بالهيكل الأصلى.) بعد انتهاء عملية الحفظ.. يعود dBASE للظهور مرة أخرى بملقن النقطة. للتأكد من أن التغييرات قد حدثت فعلاً، ولأغراض التوثيق كذلك.. اكتب أياً من الأمرين التاليين :

. DISP STRU <cr>

. DISP STRU TO PRINT <cr> أو

فيظهر لك الهيكل التالى :

```
Structure for database: C:\PROJECTS\DB4\DATA\PERSNL.DBF
Number of data records:        5
Date of last update    : 01/29/89
Field  Field Name  Type       Width     Dec     Index
    1  EMP_NUM     Character      3                 N
    2  EMP_NAME    Character     13                 N
    3  TOWN        Character      9                 N
    4  ORG         Character      3                 N
    5  EXE         Logical        1                 N
    6  DT_OF_HIRE  Date           8                 N
    7  SALARY      Numeric        8       2         N
    8  NOTE        Memo          10                 N
```

عند هذه النقطة فى دراستك لنظام dBASE.. يجب أن تكون قادراً على إنتاج هيكل CREATE وتعديله MODIFY، إذا دعت الحاجة لذلك؛ لتصل إلى شكل قاعدة البيانات الذى تريده بالضبط.

الحقل الإضافى THE EXTRA FIELD

أنظر إلى هيكل الملف مرة أخرى... هل هناك تعارض مبين فى الهيكل المسرود؟ (حاول أن تجده قبل أن تستمر فى القراءة.)

إذا لاحظت.. فإن dBASE يفيدك بأن الطول الكلى لهيكل السجل هو 56 رمزاً، إلا أنك متأكد أن مجموع أحجام الحقول الفردية لايمكن أن يزيد عن 55، وهذا يعنى أن dBASE يقول أن طول سجلات البيانات التالية سيكون 56 رمزاً، إلا أنك متأكد تماماً أن طول سجلات البيانات التالية لا يمكن أن يزيد عن 55 رمزاً.

لماذا يعطيك dBASE موقع رمز إضافى فى هيكل الملف بوضوح، وبالتالى فى كل سجل يأتى تباعاً؟

افرض أنك أدخلت بعض سجلات بيانات فى الهيكل، وقررت بعد ذلك فى لحظة معينة أنه لايعجبك شكل معين، ويمكنك أن تطلب من dBASE أن يحذف DELETE السجل، ولا يؤدى طلبك حذف السجل حذفاً له الآن، إلا أنه يحذفه منطقياً، أى إنه توضع إشارة على السجل بأنه سيحذف. ويستخدم dBASE نجمة * تسبق السجل، كإشارة لأنه سيحذف، ودون أى موقع لرمز إضافى لوضع النجمة فيه.. لن يكون هناك خيار آخر لنظام dBASE، إلا أن يضع النجمة فى أى موقع من مواقع الحقول الموجودة بالسجل.

تقدم إمكانية حذف السجلات منطقياً خدمة كبيرة للمستفيد، إذا غير رأيه بالنسبة إلى حذف السجل، ويستطيع المستفيد دائماً أن يطلب من dBASE أن يعيد تنشيط أى سجل من السجلات المراد حذفها، أو يعيد تنشيط كل السجلات المراد حذفها، ويحدث هذا بحذف النجمة الموجودة فى الموقع الذى يسبق السجل مباشرة.

وتلخيصاً.. فإن موقع الرمز الاضافى يقدمه نظام dBASE لمعالجة مؤشر الحذف (*) الممكن، إذا كانت هناك حاجة لحذف أو لإعادة تنشيط سجلات. (يجب أن تتأكد أن هذه الحاجة سوف تتواجد.)

لقد سبق أن ذكرت أنك تستطيع تعريف حتى 4000 رمزاً من البيانات - كحد أقصى - لتكوين السجل، وهذه الخانات لاتشمل الموقع الاضافى الذى ينتجه نظام dBASE لمعالجة حذف السجلات. وأنت كمستفيد.. يمكنك أن تعرف تكويناً من 4000 رمز كهيكل لقاعدة البيانات، والحقل الذى يعالج مؤشر الحذف هو ميزة إضافية من dBASE.

الفصل الثامن

أمـر الاضافـة

APPEND COMMAND

والآن... بعد أن أصبحت معتاداً على ميكانيكية انتاج هيكل وتعديله؛ ليناسب أغراضك.. يمكنك أن تستمر مع ميكانيكية إدخال بيانات فى هيكل الملف.

يستخدم أمر الاضافة APPEND فى ادخال بيانات فى قاعدة البيانات، ويستخدم بغض النظر عما إذا كانت قاعدة بياناتك تحتوى على بيانات حالياً أم لا؛ أى إنك تستطيع أن تبدأ إدخال البيانات للمرة الأولى باستخدام أمر الاضافة، ثم تستمر فى ادخال بيانات أكثر فى مرات لاحقة؛ باستخدام نفس الأمر كذلك. وتضاف السجلات الجديدة ببساطة إلى نهاية سجلات البيانات الموجودة قبل الاضافة.

ولاستخدام الأمر يكتب ما يلى :

. USE PERSNL <cr>

إذا لم يكن موجوداً فى الاستخدام فعلاً

. APPEND <cr>

يحضر هذا هيكلاً فارغاً (قناع mask للملف PERSNL) على الشاشة على النحو التالى :

```
                                                        Caps
Record No       1
EMP_NUM
EMP_NAME
TOWN
ORG
EXE
DT_OF_HIRE     /  /
SALARY              .
NOTE          memo
```

هذا هو قناع أو هيكل السجل الجديد. ويمكنك الآن أن تبدأ فى ادخال البيانات فيه، وأسماء الحقول هى بالطبع نفس الأسماء، التى سبق أن قدمتها عندما أنتجت هيكل الملف، ويحدد حجم الحقول المختلفة بعرض الرؤية المعكوسة inverse video، ويحدد رقم السجل الموجود فى القمة السجل الذى تقوم بإدخال البيانات فيه الآن.

أبدأ من فضلك إدخال زوجاً من السجلات، وادخل أى بيانات تريدها الآن؛ حتى تكتسب شعوراً بأمر الإضافة APPEND الآن.

تلاحظ بالطبع أنه بمجرد اتمام ادخال سجل.. يحضر dBASE القناع على الفور للسجل التالى على الشاشة، ادخل زوجاً من السجلات، ثم اضغط على مفتاح Esc عند أى سجل تالٍ للعودة إلى ملقن النقطة.

يمكنك إدخال أى عدد من السجلات من خلال هذه العملية.

اضافة بيانات مذكرة APPENDING MEMO DATA

لادخال بيانات فى حقل مذكرة.. انقل نقطة البداية إلى الحقل، ولاحظ أن مؤشر المذكرة memo marker يحتوى على كلمة memo بحروف صغيرة. اضغط الآن على Ctrl-Home؛ لتجد نفسك فى مشغل كلمات نظام dBASE، وادخل المذكرة فى لوحة مسودات تشغيل الكلمات word-processing scratch pad المقدمة. اضغط على Ctrl-End لحفظ محتويات المذكرة، أو اضغط على مفتاح Esc إذا غيرت رأيك بالنسبة إلى تغيير المذكرة. وبمجرد ادخالك المذكرة، وتأكيدك لها.. يتغير مؤشر المذكرة إلى MEMO بالحروف الكبيرة! وهذه هى الكيفية التى تميز بها بين السجلات التى بها مذكرة، والأخرى التى لاتوجد بها مذكرة.

بالرغم من أن حقل المذكرة – نفسه – لايزيد حجمه فى هيكل الملف عن 10 خانات.. إلا أن محتويات المذكرة تخزن فى ملف مذكرة منفصل، ومثال ذلك حيث إن الملف PERSNL. DBF له حقل مذكرة.. فإن المحتويات الفعلية للمذكرات تخزن فى ملف آخر اسمه PE-RSNL. DBT.

APPENDING COMMON DATA — اضافة بيانات مشتركة

إذا أردت ادخال معلومات مشتركة فى السجلات (مثل أن معظم العاملين يعملون فى نفس المؤسسة، ويسكنون فى نفس المدينة، ويكون من الممل تكرار كتابة GSD و ROCHESTER فى 300 سجل).. يمكنك أن تستخدم سمة، تساعدك فى حفظ معلومات مشتركة فى السجلات، وتستدعى هذه السمة قبل أن تدخل فى حالة الإضافة APPEND.

```
.USE PERSNL      <cr>
.SET CARRY ON    <cr>
.APPEND    <cr>
```

والآن.. بدلاً من ظهور الهيكل الفارغ للسجل التالى على الشاشة.. يظهر القناع مملوءاً بما سبق إدخاله فى السجل السابق! (وبالنسبة لأول سجل المضاف إلى هيكل قاعدة بيانات فارغ.. يكون القناع فارغاً دائماً.) يمكنك الآن الحركة خلال السجل، مجرياً التغييرات على البيانات، ويمكنك أن تتخطى الحقول، التى تحتوى على المعلومات المشتركة المطلوبة.

(لاحظ أنك إذا بدأت بمعلومات مشتركة خطأ.. فمن الممكن أن تنتهى بزوج من مئات السجلات، التى يجب أن يعاد تنقيحها!)

ادخل زوجاً من السجلات فى هذه الحالة، ثم اضغط بعد ذلك على مفتاح Esc.

اضافة بيانات مشتركة مختارة
APPENDING SELECTED COMMON DATA

يمكنك أن تختار الحقول التى تريد الاستمرار معها، ولست فى حاجة إلى حمل السجل كله للأمام.

```
.SET CARRY TO TOWN,ORG      <cr>
.APPE    <cr>
```

ومن الآن.. فالبيانات الموجودة فى الحقلين المذكورين فى الأمر، هى التى تحمل إلى الأمام من سجل لآخر، أما بقية السجل فتكون خالية.

ادخل زوجاً من السجلات فى هذه الحالة، ثم اضغط على مفتاح Esc.

يمكنك أن تضيف حقولاً أكثر إلى قائمة الحقل الموجودة، وذلك كما يلى :

```
.SET CARRY TO EXEMPT  ADDI     <cr>
.APPE     <cr>
```

فى الواقع.. أنت تطلب من dBASE أن يشمل حقل EXEMPT فى عملية الحمل، ويعنى مؤشر ADDI الاضافة، أى إن حقل EXEMPT يضاف إلى المؤشرات، التى سبق وجودها، ORG, TOWN.

يمكنك الحصول على نفس النتيجة من خلال العبارة :

```
.SET CARRY TO TOWN,ORG,EXEMPT     <cr>
.APPE     <cr>
```

ادخل زوجاً من السجلات فى هذه الحالة، ثم اضغط على مفتاح Esc.

للعودة إلى حالة الحمل التقليدية.. يمكنك أن تكتب ما يلى :

```
.SET CARRY TO     <cr>
.APPE     <cr>
```

لاحظ أن هذا يبدأ مرة أخرى فى حمل كل الحقول للأمام؛ لأن الأمر سالف الذكر نفى قائمة الحقل، التى سبق تحديدها، إلا أن حالة الحمل "Set Carry On" تظل كما هى دون تغيير!

ملاحظة :

تعيد عبارة SET CARRY ON <field-list> بنفسها حالة الحمل "on"، وعلى هذا.. فأنت لست فى حاجة إلى تحديد وضع حالة الحمل، قبل هذه العبارة. مثال ذلك:

```
.SET CARRY ON     <cr>
.SET CARRY TO TOWN,ORG     <cr>
.APPE     <cr>
```

فى المثال السابق.. عبارة SET CARRY ON زيادة عن الحاجة؛ لأنها نفسها تعيد حالة الحمل إلى الوضع on.

إذا لم ترد تأثير الحمل CARRY ON.. يجب أن تكتب SET CARRY OFF <cr>

الإضافة إلى بيانات تاريخية APPENDING FOR HISTORICAL DATA

قد تلاحظ أنه عندما تحضر الاضافة APPEND القناع أو الهيكل على الشاشة؛ بغرض ادخال البيانات.. فإن كل التواريخ تظهر فى ثمانى خانات (MM/DD/YY)؛ أى إنه مسموح لك بخانتين اثنتين فقط للسنة، مع استخدام 19 كرقم تقليدى للقرن.

إذا كنت تدخل بيانات تاريخية.. فقد تريد الاضافة APPEND؛ للحصول على تاريخ كامل فى 10 خانات مثل MM/DD/CCXYY (حيث CC تمثل القرن). ولعمل ذلك.. يجب أن تخطر نظام dBASE بما تريد، قبل أن تدخل حالة APPEND، وذلك من خلال استخدام أمر SET:

```
.SET CENTURY ON     <cr>
.APPE     <cr>
```

ومنذ الآن.. ففى الأوامر التى تشمل الحالة الكاملة للشاشة، تظهر كل التواريخ فى حالة العشرة خانات، ويمكنك أن تدخل الآن القرن أو تنقحه.

ولإنهاء هذه الحالة.. اكتب العبارة التالية :

```
.SET CENTURY OFF     <cr>
```

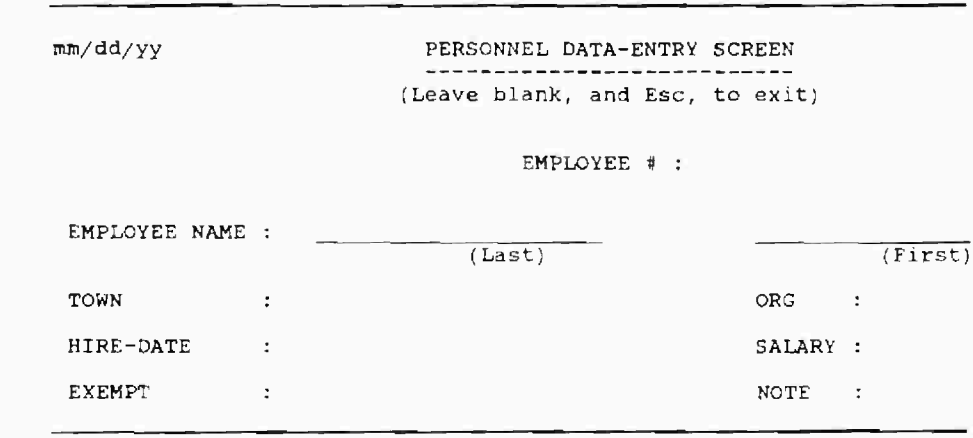
الإضافة بتشكيل خاص للشاشة

APPENDING WITH A SPECIAL SCREEN FORMAT

حتى الآن.. لاتبدو الشاشة فى منظر منمق؛ إذ أن كل أسماء الحقول تكون مضبوطة من ناحية اليسار فى الشاشة كلها، إلا أنك تستطيع تقديم شاشة أكثر رونقاً فى حالة الاضافة، ويحدث هذا من خلال إنتاج ملف تشكيل خاص للشاشة.

ملف تشكيل الشاشة screen-format file – كما يبدو من اسمه – هو ملف يصف موقع كل حقل بالضبط من قاعدة البيانات الحالية على الشاشة، وأى ثوابت يجب أن تنثر عبر الحقول، وأين يجب وضع التعليمات الخاصة، إذا كانت هناك تعليمات خاصة، على الشاشة.

كمثال... بملف تشكيل الشاشة، يمكنك أن تحدد أن تخطيط الشاشة يبدو مثل شكل 1-8.



شكل 1-8 : مثال لتخطيط الشاشة.

وبعيداً عن العدد اللانهائي، للمواقع الممكنة للحقول.. يمكنك أن تقدم تعليمات خاصة بل، وتضيء بعض العناصر عن طريق تلوين صناديق حول أحد الحقول أو أكثر من حقل واحد، وقذف قطرات متناثرة من الألوان هنا وهناك.

تذكر العملية الدقيقة لإنتاج ملف تشكيل الشاشة بالتفصيل في الملحق ب. وللحظة الحالية.. افرض أنك أنتجت ملف تشكيل شاشة (أو تخطيط شاشة) اسمه LAYOUTIN (.FMT) كدعم لأمر APPEND للملف PERSNL.

لتنشيط ملف التخطيط هذا للاضافة.. يجب أن تكتب ما يلي :

```
.USE PERSNL      <cr>
.SET FORMAT TO LAYOUTIN      <cr>
.APPEND      <cr>
```

والآن.. يتولى أمر APPEND العمل، وكل شيء سبقت مناقشته يظل كما هو، والاختلاف الوحيد هو أن تخطيط الشاشة يظهر، كما هو معرف في ملف التخطيط الذي سبق تنشيطه.

ويمكن بالطبع أن تكون لديك ملفات تشكيل متعددة، تدعم ملف بيانات واحداً لأسباب مختلفة. فيمكن أن يكون هناك أي عدد من ملفات التشكيل موجودة فعلاً على قرصك.

ملاحظة :

عندما تكون مستخدماً ملف تشكيل لإنعاش الشاشة.. تذكر دائماً أن توقف نشاط ملف التشكيل، قبل انتهاء عملك بالنظام؛ فإذا لم تتبع هذه النصيحة.. فمن الممكن أن تتوقع حدوث أشياء غريبة؛ خاصة أثناء تنفيذ أحد برامج الكمبيوتر. وبالرغم من أنك مازلت بعيداً عن كتابة برنامج كمبيوتر، مستخدماً نظام dBASE.. إلا أنه لا يوجد وقت أفضل من هذا؛ لتتعود على عادات جيدة فى استخدام نظام dBASE.

فى المثال السابق.. عندما تضيف بيانات مستخدماً ملف تشكيل، يجب أن تكتب :

.SET FORMAT TO <cr>

أمر SET FORMAT TO. دون أى مؤشرات يغلق (يوقف نشاط) ملف التشكيل.

ملاحظات على الإضافة OBSERVATIONS ON APPEND

١- إذا حاولت ادخال بيانات حرفية (حروف أبجدية أو رموز خاصة.. إلخ) فى حقل عددى.. فان ذلك يبطل عمل لوحة مفاتيحك، كما تصدر صفارة من النظام. وإذا لم يتم ادخال أى بيانات.. فإن البيانات التقليدية التى يحتويها الحقل، هى الصفر.

٢ - يمكنك باستخدام SET CARRY ON توفير كثير من مجهود ادخال البيانات، إذا كانت لديك معلومات مشتركة، يراد ادخالها فى عديد من السجلات أو فى السجلات كلها. وبمجرد إدخال حالة الحمل SET CARRY ON.. يظل الحمل سارياً فى قاعدة البيانات كلها، ويمكنك أن تختار الحقول المراد إجراء الحمل عليها أى : SET CARRY ON <field-list>، وتذكر أن تكتب SET CARRY OFF، إذا لم ترد استخدام هذه السمة.

٣ - فى حالة استخدام SET CENTURY ON.. تقدم كل التواريخ على الصورة MM/ DD/CCYY، التى تيسر لك الاتصال ببيانات القرن. تذكر أن تكتب ما يلى : SET CENTURY OFF، إذا لم تكن هناك حاجة إلى الاتصال بالقرن فى كل التواريخ.

٤ – عندما تصل نقطة البداية إلى نهاية أى حقل.. فإنها تنتقل تلقائياً لأسفل إلى الحقل التالى، مع اصدار صفارة.

٥ – إذا انتقلت نقطة البداية أبعد من آخر حقل للسجل الحالى.. يحفظ السجل الحالى، ويظهر القناع للسجل التالى على الشاشة لادخال بياناته.

٦ – يمكنك أن تجرى أى عدد من التغييرات على بيانات السجل الحالى على الشاشة، وذلك بنقل نقطة البداية للأمام وللخلف عبر الحقول؛ مستخدماً مفاتيح التحكم فى نقطة البداية، التى سبق توضيحها.

٧ – لست فى حاجة إلى احضار نقطة البداية إلى نهاية آخر حقل؛ لكى تحفظ السجل؛ إذ يمكنك ادخال بيانات فى عدة حقول مختارة فقط، إذا رغبت فى ذلك، وعندما تريد حفظ السجل.. عليك بالضغط على Ctrl-End (أو على مفتاح PgDn)، ويحفظ هذا السجل الحالى الذى تم ملؤه جزئياً، ويقدم القناع للسجل التالى.

٨ – يمكن أن يعالج dBASE حتى بليون سجل (أو 2 بليون رمز إيهما يحدث أولاً) فى أى قاعدة بيانات، ودون الحاجة إلى القول.. فليست لدى الفرصة لإثبات أو لعدم إثبات هذا التأكيد من أشتون تيت.

٩ – بعد أن يحفظ السجل ويقدم الهيكل الفارغ (أو الهيكل المملوء إذا استخدمت SET CARRY ON) للسجل التالى.. يمكنك أن تعود دائماً إلى السجل السابق، باستخدام مفتاح PgUp، ثم الرجوع بعد ذلك للأمام باستخدام مفتاح PgDn، وتستطيع العودة إلى سجلات سابقة حتى السجل رقم 1 فقط. وإذا حاولت الضغط على PgDn بعد آخر سجل، ولم تكن مستخدماً SET CARRY ON.. فإن dBASE يسألك إذا كنت تريد اضافة سجلات أكثر أم لا، ويمكنك أن تضغط على N، أو على مفتاح Esc للعودة إلى ملقن النقطة. إلا أنك إذا كنت مستخدماً SET CARRY ON.. فإن محاولتك الضغط على PgDn بعد آخر سجل، تخدم فى انتاج سجلات متكررة فقط من آخر سجل.

١٠ – عندما تكون فى حالة الاضافة.. فقد سبق أن ذكرت أنك تستطيع استخدام مفتاحى PgUp وPgDn، خلال السجلات الموجودة، وجعل أى سجل منها هو السجل الحالى.

إذا قررت حذف سجل معين.. فيمكنك استخدام Ctrl-U، عندما يكون السجل على الشاشة. وإذا قررت إخلاء محتويات هذا السجل؛ بحيث يمكنك ادخال بيانات أخرى فى نفس السجل.. اضغط على F10 لفتح قضيب القائمة menu bar فى أعلى سطر من أسطر الشاشة، والذى يكون معداً تقليدياً فى حالة RECORDS. باستخدام مفتاح حركة السهم لأسفل.. انقل الاضاءة إلى BLANK RECORD، واضغط على مفتاح الادخال، بما يخلى السجل الحالى على الشاشة. يمكنك الآن ادخال بيانات أخرى فى السجل، ومن الواضح أنك قد لا تريد تفريغ سجل لتركه فارغاً فقط فى الملف، ويجب أن تدخل فى السجل الفارغ بيانات، أو تضغط على Ctrl-U لحذفه.

تساعدك البدائل الأخرى فى قضيب القائمة فى حذف سجلات (انت تعرف أن Ctrl-U يؤدى نفس العمل)، وتساعدك - كذلك - على نقل مشير السجل إلى أى سجل تريده، كما تساعدك على البحث عن سلاسل بيانات فى السجلات الموجودة.

إلا أننى بسبب هدفى بتوضيح كيفية عمل كل ذلك، وأكثر من ذلك عند ملقن النقطة فلن استخدم قضيب القائمة هذا فى نقل مشير السجل هنا وهناك، أو فى البحث عن سجلات.. أو فى حذف سجلات... فهناك طرق أكثر قوة لعمل ذلك عند ملقن النقطة، وسوف تدرس هذه البدائل مع استمرارك فى الكتاب، وتظهر دراسة القوائم بالقرب من نهاية الكتاب.

١١ - يمكن - نظرياً - أن تكون لكل سجل فى قاعدة البيانات مذكرة، سبق ادخالها فيه، ويمكن أن يصل لحجم الفعلى لأى مذكرة واحدة إلى 512 كيلو، التى تناظر حوالى 56 صفحة من الحجم 8-1/2 x 11 مملؤة تماماً.

إلا أنك إذا كنت تتوقع أن تكتب عدة أسطر بسيطة فقط كمذكرات لسجلات فردية.. فمن الأفضل أن تعرف الحقل بأنه من النوع الحرفى، بدلاً من أنه من نوع المذكرة. (وبعد كل شىء.. يمكن للحقل الحرفى أن يصل طوله إلى 254 خانة، ويمكنك ببساطة أن تعرف حقلين حرفيين مثل COMMENT1 و COMMENT2.. إلخ للمذكرة، إذا كانت هناك حاجة لذلك). والميزة هى أنك لست فى حاجة إلى الدخول فى مشغل كلمات نظام dBASE؛ لادخال بعض البيانات فى هذا الحقل. والمميزات الأخرى هى أنه يمكن عرض

البيانات من الحقل الحرفى فى أى صورة من صور أمر DISPLAY (الذى يدرس فيما بعد)، كما أن بعض الأوامر تنفذ بصورة أسرع، إذا لم تكن هناك حاجة لتشغيلها على ملفات DBT. (ملفات المذكرات).

١٢ - لايسمح الحقل المنطقى إلا بأى حرف من الحروف الأربعة T / F / Y / N فقط كمدخلات له، ولايقل هذا الحقل، ولايعرض إلا أى من الحرفين T أو F فقط. ومثال ذلك... سبق أن عرفت حقلاً منطقياً فى الملف الذى انتجته اسمه EXEMPT، ويجب أن تحتوى سجلات العاملين المستثنين على الحرف T (أى حقيقى TRUE) فى حقل EXEMPT. أما العاملين غير المستثنين.. فيجب أن يحتوى حقلهم EXEMPT على F (أى خطأ FALSE)، والقيمة التقليدية إذا لم يتم إدخال أى حرف هى F.

١٣ - تنقح البيانات التى يتم ادخالها بأنها تواريخ بغرض الدقة، كما أن dBASE يتأكد كذلك من السنة الكبيسة، ويفترض فى المحتوى أن يكون فى الصورة MM/DD/YY، التى يجب أن يتم إدخاله بها.

١٤ - عندما تنتهى من آخر اضافة إلى اليوم، ووضع dBASE الهيكل الفارغ (أو المملوء) للسجل التالى على الشاشة.. يمكنك أن تخرج من حالة الاضافة بإحدى طريقتين :

أ - الضغط على مفتاح الادخال، عندما تكون نقطة البداية فى أول موقع من أول حقل، يخرجك من حالة الاضافة، وهذا صحيح كذلك إذا كنت تستخدم سمة SET CARRY ON.

ب - يمكنك أن تضغط دائماً على Ctrl-Q أو على Esc (مغيراً رأيك) للسجل التالى المقدم، وتخرج من حالة الاضافة، وهذا صحيح كذلك مع سمة SET CARRY ON.

١٥ - هذا يحضر الامكانية الحتمية - أنك لم تحفظ آخر سجل تم العمل معه - لكى تخرج من حالة الاضافة. وإذا كنت قد أدخلت بيانات فى سجل معين، وتريد الخروج دون حفظ هذا السجل فدائماً ما تستطيع عمل ذلك، مستخدماً الضغط على Ctrl-Q، أو الضغط على Esc عند هذا السجل.

١٦ - إذا كان بهيكل السجل عديد من الحقول (بحد أقصى 255).. فمن الممكن ألا تكون كل محتويات السجل مرئية على الشاشة فى نفس الوقت، وعندما تدخل آخر رمز، من آخر

حقل مرئى على الشاشة.. تظهر بقية هيكل السجل على الشاشة، أو إذا أردت أن تتخطى بضع حقول عند ادخالك البيانات.. يمكنك أن تستخدم PgDn أو PgUp للدوران للأمام والخلف؛ لرؤية الحقول الأخرى.

١٧ - لا تنس عند الانتهاء من عملية الاضافة أن تضع SET CENTURY OFF. و SET CARRY OFF. إذا كان هذا ضرورياً، وإلا.. فإن هذه الحالات تظل نشطة، بغض النظر عن الملف المستخدم USE، بعد الخروج من حالة الاضافة فى جلسة العمل الحالية.

١٨ - أثناء إدخال البيانات إذا سمعت صفيراً مزعجاً.. يجب أن تخرج من حالة الاضافة، عائداً إلى ملقن النقطة (انظر الملاحظة رقم ١٤) ثم تكتب SET BELL OFF، وتضغط على مفتاح الادخال، وتعود مرة أخرى إلى حالة الاضافة.

١٩ - أثناء عملية ادخال البيانات... إذا وجدت نفسك خارج حالة الاضافة بطريق الخطأ، يمكنك العودة مرة أخرى إلى حالة الاضافة؛ مستخدماً أمر APPEND.

هناك بعض الصيغ الأخرى من أمر الاضافة، إلا أنها مقدمة فى أقسام لاحقة.

عند هذه النقطة.. احذف كل السجلات التى سبق لك ادخالها؛ بغرض تدريبك حتى الآن مستخدماً ما يلى :

ZAP <cr>. يسألك dBASE عما إذا كنت تريد إخلاء الملف (حذف كل السجلات). اكتب Y.

يدرس أمر ZAP بالتفصيل فى قسم لاحق.

APPEND <cr>.

استمر مع أمر APPEND من فضلك، وادخل سجلات بيانات لهيكل قاعدة البيانات المسماة PERSNL . DBF . استخدم سجلات البيانات البالغ عددها 15 التالية كمدخلات؛ لأنك ستستخدم هذه البيانات فى أمثلة لاحقة.

أتوقع أنك تستغرق بعض الوقت، لكى تعتاد على فكرة ملف التشكيل (فى ملف تخطيط الشاشة)، كما أنك تحتاج قليلاً من الخبرة لاستخدام سمة تصميم الصيغ فى dBASE؛

لإنتاج ملفات تخطيط الشاشة. وعلى هذا.. استخدم شاشة الإضافة العادية (أو التقليدية) فى إدخال البيانات، فى الوقت الحالى.

لاحظ أن بعض البيانات تظهر كخليط من الحروف الكبيرة والحروف الصغيرة. من فضلك ادخلها كما هى تماماً. (الأرقام الموجودة على اليسار هى أرقام السجلات، التى يقدمها dBASE تلقائياً مع استمرارك فى إضافة بيانات.)

أننى افترض أنك ستدخل بيانات خاصة بك فى أحد حقول المذكرات، أو فى أكثر من حقل من هذه الحقول طبقاً لما ترغبه. سوف تبين أمثلة أمر DISPLAY التالية بعض المذكرات التى استخدمتها أنا كعينات.

تذكر أنك إذا أدخلت مذكرة لأى سجل، وأكدت ادخالها.. فإن مؤشر المذكرة memo marker يتغير لهذا السجل إلى الحروف الكبيرة.

سجلات البيانات المراد إدخالها فى الملف

DATA RECORDS TO BE ENTERED IN THE FILE

Record#	EMP_NUM	EMP_NAME	TOWN	ORG	EXE	DT_OF_HIRE	SALARY	NOTE
1	005	NINA BHARUCHA	WEBSTER	BSG	.T.	05/24/80	25000.00	MEMO
2	010	PETE JOHNSON	brighton	BSG	.T.	02/03/76	27590.00	MEMO
3	015	GLORIA PATEL	FAIRPORT	RMG	.T.	07/16/82	27500.00	memo
4	020	MAX LEVINSKY	HENRIETTA	RMG	.F.	04/13/69	27550.00	MEMO
5	025	KIM BRANDT	FAIRPORT	RMG	.F.	04/04/77	36000.00	MEMO
6	030	TIM MONTAL	ROCHESTER	RBG	.F.	07/07/81	41900.00	memo
7	035	WILLIAM PATEL	penfield	GSD	.F.	08/17/71	28900.00	memo
8	040	JAMES JAMESON	ROCHESTER	GSD	.T.	10/21/77	29800.00	memo
9	045	MORRIS KATZ	webster	BSG	.F.	09/14/80	23450.00	memo
10	050	PAUL BHARUCHA	BRIGHTON	BSG	.T.	05/23/73	29100.00	MEMO
11	055	PHIL MARTIN	WEBSTER	RMG	.F.	07/19/80	31000.00	memo
12	060	JOHN PETERSON	BRIGHTON	RBG	.T.	04/17/79	31480.00	MEMO
13	065	JOY HARDY	fairport	RBG	.F.	01/19/79	34200.00	MEMO
14	070	JAN MOREY	ROXY	GSD	.T.	04/23/67	18190.00	memo
15	075	JOHN JONES	rochester	GSD	.T.	04/04/70	25100.00	memo

عند هذه النقطة من دراستك dBASE.. يجب أن تكون قادراً على انتاج CREATE هيكل وتعديله MODIFY؛ إذا دعت الحاجة لذلك وعرضه DISPLAY، وتوثيقه، وإضافة APPEND بيانات داخله.

الفصل التاسع

أمر العرض

DISPLAY COMMAND

بعد ادخالك بعض البيانات فى قاعدة بياناتك.. فإنك تستطيع الآن أن ترى هذه البيانات فى صور وأشكال متعددة، وعند هذا الحد.. يمكنك أن تتعرف على أمر قوى جداً من أوامر dBASE، وهو أمر العرض DISPLAY.

من خلال استخدام هذا الأمر.. يمكنك أن تعرض (على الشاشة أو على ورق الطباعة) مايلى :

- كل البيانات من قاعدة بيانات واحدة، أو أكثر من واحدة.
- سجلاً واحداً من قاعدة بيانات واحدة، أو أكثر من واحدة.
- مجموعة سجلات من قاعدة بيانات واحدة، أو أكثر من واحدة.
- حقلاً محدداً من سجلات مختارة من قاعدة بيانات واحدة، أو أكثر من واحدة.
- سجلات من قاعدة بيانات واحدة، أو أكثر من واحدة، تحقق شرطاً بسيطاً.
- سجلات من قاعدة بيانات واحدة، أو أكثر من واحدة، تحقق شرطاً مركباً.
- خليطاً مما سبق ذكره.

كما يمكنك أن ترى فهذا الأمر مفيد جداً فى كل مما يؤديه. اتبع من فضلك هذا القسم بدقة؛ نظراً لأنه بالرغم من أنه مفيد جداً، وممتع جداً إلا أن المؤشرات التى تدرسها فى هذا الأمر تظهر كذلك فى عديد من أوامر dBASE الأخرى، و يضمن فهمها هنا لك التعود السريع على الأوامر الأخرى كذلك.

الصيغة العامة لأمر DISPLAY هى :

```
[FOR <condition>]
[WHILE <condition>]
[TO PRINT]
[TO FILE <filename>]
[OFF]
```

كما يمكنك أن تقدر.. فكل مؤشرات الأمر اختيارية، وهذا الأمر مرن، ولاحظ أن عدد الفراغات بين المؤشرات لا معنى له، بافتراض أنك تترك فراغاً واحداً على الأقل.

أول نقطة تميز... هى أن أمر DISP – دون أى مؤشرات – يعرض دائماً محتوى السجل الذى يشير إليه نظام dBASE، عندما يتم ادخال الأمر، ويسمى السجل الذى يشير إليه dBASE حاليا السجل الحالى current record.

إذا كان طول السجل أكبر من حجم الشاشة .. فإن السجل يلتف حول wrap around الشاشة.

إذا استخدمت الخليط التالى :

```
.USE PERSNL     <cr>
.DISP     <cr>
```

فيتسبب أمر USE فى فتح قاعدة البيانات المسماة PERSNL، ويعطى dBASE التحكم فى أول سجل من سجلاتها (أى إن dBASE يشير إلى أول سجل)، ويعرض أمر DISP أول سجل.

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      1  005     NINA BHARUCHA WEBSTER    BSG .T. 05/24/80   25000.00 MEMO
```

تذكر أنك إذا رأيت مؤشر المذكرة بحروف كبيرة.. فمعنى ذلك أن السجل يحتوى على مذكرة.

نقل مشير السجل MOVING THE RECORD POINTER

نستطيع أن ننقل مشير السجل إلى أى سجل نريده من سجلات قاعدة البيانات، وذلك على النحو التالى :

SKIP <cr>. ينقل المشير إلى السجل التالى (أى إلى السجل رقم 2)

..DISP <cr>

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      2  010     PETE JOHNSON   brighton   BSG .T. 02/03/76   27590.00 MEMO
```

SKIP 5 <cr>. ينقل المشير من السجل رقم 2 إلى السجل رقم 7.

..DISP <cr>

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE    SALARY NOTE
      7  035     WILLIAM PATEL  penfield  GSD .F. 08/17/71    28900.00 memo
```

SKIP - 3 <cr>. ينقل المشير من السجل رقم 7 إلى السجل رقم 4.

.DISP <cr>

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE    SALARY NOTE
      4  020     MAX LEVINSKY   HENRIETTA RMG .F. 04/13/69    27550.00 MEMO
```

لاحظ أن أمر DISP يعرض ببساطة محتوى السجل الموجود عنده المؤشر.

إذا كنت تعرف رقم السجل الذى تريد نقل التحكم إليه، وليكن السجل رقم 15 –على سبيل المثال– فإنه – بغض النظر عن الموقع الحالى للمشير – يمكنك أن تكتب مايلى :

```
.GOTO 15    <cr>
.DISP       <cr>

Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE    SALARY NOTE
     15  075     JOHN JONES     rochester GSD .T. 04/04/70    25100.00 memo

.GO 10      <cr>
.DISP       <cr>

Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE    SALARY NOTE
     10  050     PAUL BHARUCHA  BRIGHTON   BSG .T. 05/23/73    29100.00 MEMO
```

وهذه هى أسهل طريقة !!

```
.7      <cr>
DISP    <cr>

Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE    SALARY NOTE
      7  035     WILLIAM PATEL  penfield  GSD .F. 08/17/71    28900.00 memo
```

أى مما هو مذكور أعلاه.. ينقل التحكم إلى السجل المناسب، ويعرض أمر DISPLAY الذى يصدر بعد ذلك السجل الحالى. وأسرع طريقة للوصول إلى رقم سجل محدد.. هى إدخال رقم السجل نفسه.

لنقل التحكم بسرعة جداً إلى أول سجل فى قاعدة البيانات.. اكتب ما يلى :

```
GO TOP     <cr>
DISP       <cr>
```

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      1  005     NINA BHARUCHA  WEBSTER    BSG .T. 05/24/80   25000.00 MEMO
```

```
GOTO TOP     <cr>
DISP       <cr>
```

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      1  005     NINA BHARUCHA  WEBSTER    BSG .T. 05/24/80   25000.00 MEMO
```

هذه هى أسهل طريقة.

.1 <cr>

.DISP <cr>

نظراً لأن أول سجل دائماً ما يكون له رقم سجل 1.. فإن هذه هى أسهل طريقة للوصول إلى قمة الملف.

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTBE
      1  005     NINA BHARUCHA  WEBSTER    BSG .T. 05/24/80   25000.00 MEMO
```

وللوصول إلى نهاية الملف بسرعة كبيرة.. تستخدم صيغة شبيهة على النحو التالى :

```
GO BOTTOM     <cr> or
GO BOTT       <cr> or
GOTO BOTT     <cr> or
.DISP         <cr>
```

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
     15  075     JOHN JONES     rochester  GSD .T. 04/04/70   25100.00 memo
```

مرة أخرى.. فالنقطة التى يجب أن تميز، هى أنه فى غياب أى من المؤشرات، دائماً ما يعرض أمر DISPLAY محتوى السجل الذى يشير إليه dBASE، عند اصدار الأمر.

إذا حاولت ما يلى :

10 <cr>. يضع هذا المشير عند السجل رقم 10.

SKIP 9999 <cr>. نظراً لعدم وجود عديد من السجلات، لديك فى قاعدة بياناتك.. ينقل dBASE المشير إلى ما بعد آخر سجل موجود فى الملف، وهذا هو شرط نهاية الملف end-of-file.

DISP.

```
Record#  EMP_NUM EMP_NAME      TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
```

تأثير ذلك هو عدم عرض أى شىء. أما إذا حاولت إصدار الأوامر التالية :

10 <cr>.

SKIP - 9999 <cr>. يشير dBASE الآن إلى السجل رقم 1.

DISP <cr>.

```
Record#  EMP_NUM EMP_NAME      TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      1  005     NINA BHARUCHA WEBSTER    BSG .T. 05/24/80   25000.00 MEMO
```

لاحظ من فضلك أن أمر DISPLAY يعرض بصورة تقليدية البيانات، ومعها عناوين الأعمدة. فإذا لم ترد إظهار عناوين الأعمدة لأى سبب من الأسباب يمكنك أن تكتب ما يلى :

SET HEADING OFF <cr>.

ومن الآن.. تعرض البيانات دون عناوين الأعمدة، وفى هذه اللحظة.. إفرض أنك تريد عرض الأعمدة. أكتب فى هذه الحالة ما يلى:

SET HEADING ON <cr>.

استمر الآن مع بقية مؤشرات أمر DISPLAY، وقد سبق تقديم الصيغة العامة للأمر.

دعنى أذكرك عند هذه النقطة باستخدام مفتاح (الهروب) Esc. وعندما تكتب أمراً عند ملقن النقطة... فإما أن تستخدم مفتاح العودة للخلف للعودة إلى أى مؤشر تريد تصحيحه، أو أنه يمكنك أن تعيد كتابة الأمر. ولاعادة كتابة الأمر.. اضغط على Esc لحذف الأمر الموجود حالياً فى سطر الأمر، وأعد كتابة الأمر الجديد.

THE SCOPE PARAMETER — مؤشر المدى

يحدد مؤشر المدى (scope) مدى العملية لنظام dBASE؛ أى إنه يحدد عدد السجلات المراد اعتبارها فى أمر DISPLAY.

يمكنك استخدام أحد المحتويات الأربعة التالية :

١ – ALL من الواضح أن هذا يعنى عرض كل السجلات.

إلا أن أمر DISPLAY ALL يعرض 19 سجلاً فقط – فى نفس الوقت – على الشاشة، وينتظر dBASE أن تضغط على مفتاح ليعرض لك مجموعة السجلات التالية وعددها19 سجلاً كذلك وهكذا.

إذا كان طول السجل أكبر من حجم الشاشة.. فإن كل سجل يشغل أكثر من سطر واحد على الشاشة.

تسمى حركة السجلات عبر الشاشة دوران scrolling، ولإيقاف الدوران أو لإعادة تنشيطه.. اضغط على Ctrl-S.

```
.DISP ALL

Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      1  005     NINA BHARUCHA  WEBSTER    BSG .T. 05/24/80   25000.00 MEMO
      2  010     PETE JOHNSON   brighton   BSG .T. 02/03/76   27590.00 MEMO
      3  015     GLORIA PATEL   FAIRPORT   RMG .T. 07/16/82   27500.00 memo
      4  020     MAX LEVINSKY   HENRIETTA  RMG .F. 04/13/69   27550.00 MEMO
      5  025     KIM BRANDT     FAIRPORT   RMG .F. 04/04/77   36000.00 MEMO
      6  030     TIM MONTAL     ROCHESTER  RBG .F. 07/07/81   41900.00 memo
      7  035     WILLIAM PATEL  penfield   GSD .F. 08/17/71   28900.00 memo
      8  040     JAMES JAMESON  ROCHESTER  GSD .T. 10/21/77   29800.00 memo
      9  045     MORRIS KATZ    webster    BSG .F. 09/14/80   23450.00 memo
     10  050     PAUL BHARUCHA  BRIGHTON   BSG .T. 05/23/73   29100.00 MEMO
     11  055     PHIL MARTIN    WEBSTER    RMG .F. 07/19/80   31000.00 memo
     12  060     JOHN PETERSON  BRIGHTON   RBG .T. 04/17/79   31480.00 MEMO
     13  065     JOY HARDY      fairport   RBG .F. 01/19/79   34200.00 MEMO
     14  070     JAN MOREY      ROXY       GSD .T. 04/23/67   18190.00 memo
     15  075     JOHN JONES     rochester  GSD .T. 04/04/70   25100.00 memo
```

لاحظ أن NOTE يبين إما MEMO أو memo لتوضيح وجود محتويات مذكرة أو عدم وجودها. وبعد فترة بسيطة.. فإنك ترى كيفية عرض DISPLAY المحتويات الفعلية لحقول المذكرات.

لاحظ كذلك أنه لكل حقل معروض.. فإن dBASE يقدم عرض العمود، الذى يمكن أن يحتوى الأكبر من اسم الحقل، أو من البيانات، وهناك عمود واحد بالضبط كفراغ بين الأعمدة المعروضة.

٢ – RECO n لعرض سجل محدد برقمه يمكنك أن تكتب ما يلى :

```
DISP RECO 5     <cr>

Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      5  025     KIM BRANDT     FAIRPORT   RMG .F. 04/04/77   36000.00 MEMO
```

يعرض هذا السجل رقم 5 كاملاً، إلا أنك قد تتذكر أن الخليط التالى يؤدى نفس العمل :

هذا ينقل المشير إلى السجل رقم 5 .5 <cr>

هذا يعرض السجل المشار إليه. .DISP <cr>

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      5  025     KIM BRANDT     FAIRPORT   RMG .F. 04/04/77   36000.00 MEMO
```

٣ – NEXT n لعرض مجموعة من السجلات، ولتكن 5 – بدءًا بالسجل رقم 5 – يمكنك أن تكتب ما يلى :

<cr> 5. ينقل المشير إلى السجل رقم 5.

.DISP NEXT 5

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      5  025     KIM BRANDT     FAIRPORT   RMG .F. 04/04/77   36000.00 MEMO
      6  030     TIM MONTAL     ROCHESTER  RBG .F. 07/07/81   41900.00 memo
      7  035     WILLIAM PATEL  penfield   GSD .F. 08/17/71   28900.00 memo
      8  040     JAMES JAMESON  ROCHESTER  GSD .T. 10/21/77   29800.00 memo
      9  045     MORRIS KATZ    webster    BSG .F. 09/14/80   23450.00 memo
```

لاحظ أن بديل NEXT N يشمل السجل الحالى كجزء من مؤشر NEXT، كما أن بديل NEXT n يشير إلى سجلات واقعية، وليس إلى حدوثات منطقية!! يتم توضيح هذه النقطة فيما بعد.

يعرض أمر DISP شاشة واحدة فقط فى نفس الوقت، وينتظر منك الضغط على أى مفتاح، قبل أن يعرض مجموعة السجلات التالية.

REST – ٤

. 10 <cr>

..DISP REST <CR>

يعرض هذا المؤشر بقية الملف بدءاً بالسجل الحالى.

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
     10  050     PAUL BHARUCHA  BRIGHTON   BSG .T. 05/23/73   29100.00 MEMO
     11  055     PHIL MARTIN    WEBSTER    RMG .F. 07/19/80   31000.00 memo
     12  060     JOHN PETERSON  BRIGHTON   RBG .T. 04/17/79   31480.00 MEMO
     13  065     JOY HARDY      fairport   RBG .F. 01/19/79   34200.00 MEMO
     14  070     JAN MOREY      ROXY       GSD .T. 04/23/67   18190.00 memo
     15  075     JOHN JONES     rochester  GSD .T. 04/04/70   25100.00 memo
```

الأمر سالف الذكر متطابق تماماً مع مدى (a high number). مثال ذلك فان الأمر التالى له نتيجة متطابقة تماماً مع نتيجة الأمر السابق

```
.10
.DISP NEXT 9999

Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
     10  050     PAUL BHARUCHA  BRIGHTON   BSG .T. 05/23/73   29100.00 MEMO
     11  055     PHIL MARTIN    WEBSTER    RMG .F. 07/19/80   31000.00 memo
     12  060     JOHN PETERSON  BRIGHTON   RBG .T. 04/17/79   31480.00 MEMO
     13  065     JOY HARDY      fairport   RBG .F. 01/19/79   34200.00 MEMO
     14  070     JAN MOREY      ROXY       GSD .T. 04/23/67   18190.00 memo
     15  075     JOHN JONES     rochester  GSD .T. 04/04/70   25100.00 memo
```

تذكر أن أمر DISPLAY لا يعرض إلا شاشة كاملة من السجلات فى نفس الوقت. قد تكون لاحظت أن كل البيانات من الحقل المنطقى قد سبقها نقطة وتبعها نقطة فى هذه العروض مع ترك تمييز البيانات المنطقية من البيانات الحرفية لك.

مؤشر قائمة الحقل وقائمة التعبير

THE FIELD-LIST / EXPRESSION-LIST PARAMETER

المؤشر التالى هو مؤشر قائمة الحقل field-list. تستطيع بهذا المؤشر أن تحدد أى الحقول التى تريد سردها من السجلات المختارة من مؤشر المدى. (فى غياب هذا المؤشر يتم اختيار كل الحقول لعرضها).

ملاحظة : تعرض الحقول بترتيب اسمائها فى الأمر. لاحظ أن الفواصل فى قائمة الحقل فى الأمر اختيارية!

مثال ذلك ما يلى :

```
.DISP  ALL            EMP_NUM,EMP_NAME,TOWN     <cr>
       <---scope---> <---------- field-list ---------->
```

أنت تطلب عرض بيانات من كل السجلات لكنها للحقول المذكورة فى الأمر فقط. نتيجة ذلك هو القائمة التالية :

```
Record#  EMP_NUM EMP_NAME       TOWN
      1  005     NINA BHARUCHA  WEBSTER
      2  010     PETE JOHNSON   brighton
      3  015     GLORIA PATEL   FAIRPORT
      4  020     MAX LEVINSKY   HENRIETTA
      5  025     KIM BRANDT     FAIRPORT
      6  030     TIM MONTAL     ROCHESTER
      7  035     WILLIAM PATEL  penfield
      8  040     JAMES JAMESON  ROCHESTER
      9  045     MORRIS KATZ    webster
     10  050     PAUL BHARUCHA  BRIGHTON
     11  055     PHIL MARTIN    WEBSTER
     12  060     JOHN PETERSON  BRIGHTON
     13  065     JOY HARDY      fairport
     14  070     JAN MOREY      ROXY
     15  075     JOHN JONES     rochester

   DISP  ALL           TOWN,ORG,SALARY     <cr>
         <--scope--> <---- field-list ---->

Record#  TOWN       ORG   SALARY
      1  WEBSTER    BSG 25000.00
      2  brighton   BSG 27590.00
      3  FAIRPORT   RMG 27500.00
      4  HENRIETTA  RMG 27550.00
      5  FAIRPORT   RMG 36000.00
      6  ROCHESTER  RBG 41900.00
      7  penfield   GSD 28900.00
      8  ROCHESTER  GSD 29800.00
      9  webster    BSG 23450.00
     10  BRIGHTON   BSG 29100.00
     11  WEBSTER    RMG 31000.00
     12  BRIGHTON   RBG 31480.00
     13  fairport   RBG 34200.00
     14  ROCHESTER  GSD 18190.00
     15  rochester  GSD 25100.00

   DISP  ALL           EMP_NAME,NOTE     <cr>
         <--scope--> <-field-list->
```

فى هذه الحالة ذكر الحقل NOTE فى قائمة الحقل. وعلى هذا فانك ترى المحتويات الفعلية للمذكرات وليست كلمة memo.

```
Record#  EMP_NAME       NOTE
      1  NINA BHARUCHA  To my daughter: Trust you have no
                        objection to my including your name in
                        the book.
      2  PETE JOHNSON   You will be eligible for 3 weeks
                        vacation as of 12-01-85.

      3  GLORIA PATEL
      4  MAX LEVINSKY   You will be eligible for 5 weeks
                        vacation as of 12-01-85
      5  KIM BRANDT     Hope you're enjoying your retirement !!
      6  TIM MONTAL
      7  WILLIAM PATEL
      8  JAMES JAMESON
      9  MORRIS KATZ
     10  PAUL BHARUCHA  To my son: How could your date of hire
                        show 1973 when I know you were born in
                        1980 ?
     11  PHIL MARTIN
     12  JOHN PETERSON  Thanks to you, John Peterson, between
                        you and Pete Johnson we have managed to
                        make a complete mess of the payroll.
     13  JOY HARDY      Any relation to Oliver ?
     14  JAN MOREY
     15  JOHN JONES
```

يقدم نظام dBASE حجماً تقليدياً سعته 50 عمود لحقل المذكرة. وطبقاً لعدد الحقول الأخرى التى تريد عرضها فقد تحتاج إلى أن تتحكم فى حجم المذكرة memo-width هذا على النحو التالى :

.SET MEMOIDTH TO 30 (cr)

والآن تتقلص محتويات المذكرات، إذا ما وجدت، فى هذا الحيز المحدد. وأقل حيز لحجم المذكرة هو 8 كما أن أقصى حيز تستطيع تحديده هو 32,000.

```
.DISP  RECO   5     EMP_NAME,TOWN,ORG     <cr>
       <--scope-->  <------- field-list ------->

Record#  EMP_NAME       TOWN        ORG
      5  KIM BRANDT     FAIRPORT    RMG
```

ملاحظة : مؤشرات قائمة الحقل يمكن أن تشتمل كذلك على قوائم تعبيرات إذا كان هناك حاجة لذلك. يمكنك أن تعرف تعبيراتك الحسابية الخاصة بك.

فى المثال التالى تطلب من dBASE أن يعرض الرواتب كما هى تماماً ويعرض كذلك ما يمكن أن تكون عليه الرواتب إذا ما ازدادت قيمتها بنسبة 10%. التعبير SALARY * 1.1 يعنى زيادة 10% فى الراتب.

```
.DISP  ALL      EMP_NAME, SALARY,SALARY*1.1     <cr>
       <scope>  <------------ field-list ------------>

Record#  EMP_NAME        SALABRY     SALARY
      1  NINA BHARUCHA  25000.00      27500
      2  PETE JOHNSON   27590.00      30349
      3  GLORIA PATEL   27500.00      30250
      4  MAX LEVINSKY   27550.00      30305
      5  KIM BRANDT     36000.00      39600
      6  TIM MONTAL     41900.00      46090
      7  WILLIAM PATEL  28900.00      31790
      8  JAMES JAMESON  29800.00      32780
      9  MORRIS KATZ    23450.00      25795
     10  PAUL BHARUCHA  29100.00      32010
     11  PHIL MARTIN    31000.00      34100
     12  JOHN PETERSON  31480.00      34628
     13  JOY HARDY      34200.00      37620
     14  JAN MOREY      18190.00      20009
     15  JOHN JONES     25100.00      27610
```

ملاحظة : أمر DISPLAY ALL السابق يبين كيف يبدو الراتب إذا ما ازداد 10% ولا تتغير بيانات الراتب فى الملف نفسه بأى طريقة لأن هذا لا يزيد عن كونه أمر عرض DISPLAY.

يمكنك أن تبدأ من موقع معين فى قاعدة البيانات، وذلك كما يلى :

```
.5    <cr>
DISP  NEXT 5      EMP_NAME,ORG,DT_OF_HIRE    <cr>
      <--scope--> <---------- field-list ---------->

Record#  EMP_NAME       ORG DT_OF_HIRE
      5  KIM BRANDT     RMG 04/04/77
      6  TIM MONTAL     RBG 07/07/81
      7  WILLIAM PATEL  GSD 08/17/71
      8  JAMES JAMESON  GSD 10/21/77
      9  MORRIS KATZ    BSG 09/14/80
```

حيث أنك استخدمت SET HEADING ON فتظهر عناوين الأعمدة فى المخرجات. يجب أن تلاحظ أن عناوين الأعمدة هذه تظهر بالضبط عندما تكتب اسماء الحقول المناظرة لها فى عبارة الأمر نفسها. فإذا ما كان الأمر على النحو التالى:

```
.5    <cr>
.DISP  NEXT 5       Emp_Name,Org,DL_Of_Hire    <cr>
       <--scope--> <--------- field-list--------->
```

فتظهر عناوين الأعمدة على النحو التالى :

```
Record#  Emp_Name       Org Dt_Of_Hire
      5  KIM BRANDT     RMG 04/04/77
      6  TIM MONTAL     RBG 07/07/81
      7  WILLIAM PATEL  GSD 08/17/71
      8  JAMES JAMESON  GSD 10/21/77
      9  MORRIS KATZ    BSG 09/14/80
```

اعداد الحقول SET FIELDS

أريد أن أذكر سمة مفيدة لاختيار الحقول من قاعدة البيانات والتى يمكن أن توفر الكثير من مجهود الكتابة. افرض أنه فى إحدى جلسات العمل مع dBASE أردت العمل مع مجموعة من الحقول وأردت أن تتجنب تحديد مؤشر حقول FIELDS طويل فى كل عبارة تكتبها.

يقدم dBASE أمر SET FIELDS حيث يمكنك من تحديد الحقول التى تريد استخدامها مرة وبعد ذلك يؤثر كل أمر على هذه الحقول التى اعددتها. مثال ذلك :

```
.USE PERSNL
.SET FIELDS TO TOWN,ORG,SALARY,DT_OF_HIRE
```

من الآن يبدو الملف كما لو كان محتوياً على هذه الحقول التى أعدت فقط. ويوفر لك هذا الكثير من تكرار كتابة مؤشر الحقول FIELDS فى كل عبارة تكتبها.

.SET FIELDS TO EXE

يضيف هذا الأمر إلى قائمة الحقول الموجودة والتى سبق إعدادها.

.SET FIELDS OFF

يوقف هذا الأمر نشاط قائمة الحقول مؤقتاً. وتأثير ذلك هو إمكانية الاتصال بكل حقول الملف الآن.

.SET FIELDS ON

ينشط هذا الأمر قائمة الحقول التى سبق إعدادها مرة أخرى.

ملاحظة : إذا كنت تعمل مع أكثر من ملف واحد فى نفس الوقت (وهذا الأمر موضح فى المواقع المناسبة من الكتاب) فيمكنك أن تحدد عبارات SET FIELDS TO مختلفة للملفات المختلفة.

أمر SET FIELDS TO (field-ist) تأثيره وقتى فقط. فإذا ما أحضرت ملفاً آخراً للاستخدام (فى نفس منطقة العمل) فالحقول FIELDS الأصلية التى أعدت للملف السابق لا وجود لها الآن.

شرط من أجل THE FOR CONDITION

مؤشر FOR (condition) قوى جداً. فهو يتركك تحدد شروطاً بسيطة أو مركبة يتم اختيار السجلات تحتها بغرض عرضها. أننى أقترح أنه يجب عليك أن تتفهم ذلك جيداً. ففى

نهاية هذه المناقشة يجب أن تكون قادراً على استخلاص بيانات من قاعدة بيانات لأى شرط ممكن تخيله.

ملاحظة : إذا لم يتحدد مؤشر المدى ALL, RECO (n), NEXT (n), REST فى الأمر فأن ALL يكون هو الاجراء التقليدى. أى أن كل السجلات التى تحقق الشرط يتم اختيارها للعرض إلا إذا ذكر جزء المدى (scope). فإذا ماحقق الشرط عدد من السجلات يزيد عما يمكن أن يظهر على شاشة واحدة فأن النظام ينتظر منك الضغط على أى مفتاح لعرض الشاشة التالية من السجلات.

عندما تحدد قيماً لتتحقق فى أحد الشروط ففى حالة الحقول الحرفية يجب أن تقدم هذه القيم بين علامتى تنصيص (علامتى التنصيص الفردية أو الزوجية مقبولتان). إلا أنه فى حالة الحقول العددية يجب ألا توضع القيم المحددة بين علامتى تنصيص. وبالنسبة إلى الحقول المنطقية فلا يكون هناك حاجة لأى قيم. وبالنسبة إلى حقول التاريخ فيجب أن تكون على الصورة MM/DD/YY. ويوضح هذا باستفاضة فيما بعد.

أمثلة على العرض مع استخدام شروط مكونة من حقول حرفية

EXAMPLES OF DISPLAY FOR WITH CHARACTER FIELDS

. DISPLAY FOR TOWN = 'ROCHESTER' (cr)

نظراً لأنه لم يتحدد مدى scope للعملية إلا أن شرط FOR تم ذكره فان الوضع التقليدى للمدى هو ALL.

لاحظ أن كل السجلات التى يوجد فى حقلها المسمى TOWN كلمة ROCHESTER قد اختيرت للعرض.

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      6  030     TIM MONTAL     ROCHESTER RBG .F. 07/07/81    41900.00 memo
      8  040     JAMES JAMESON ROCHESTER GSD .T. 10/21/77    29800.00 memo

          DISP FOR TOWN = 'rochester'     <cr>

Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
     15  075     JOHN JONES     rochester GSD .T. 04/04/70    25100.00 memo
```

ملاحظة : يمكنك أن تقدم مفتاحاً شاملاً. أى أن قيمة المفتاح الكامل لا تحتاج إلى تقديمها. لكن مرة أخرى القيمة المقدمة يجب أن توضع بين علامتى تنصيص وذلك لأن الحقل TOWN من النوع الحرفى.

فى الأمر التالى كل السجلات التى لها الحروف الكبيرة R O C H فى أول أربع خانات من حقل FIELD هى التى يقع عليها الاختيار لعرضها.

```
.DISP FOR TOWN = 'ROCH'     <cr>

Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE    SALARY NOTE
      6  030     TIM MONTAL     ROCHESTER RBG .F. 07/07/81    41900.00 memo
      8  040     JAMES JAMESON  ROCHESTER GSD .T. 10/21/77    29800.00 memo
```

فى المثال التالى تقوم بتقديم مفتاح أكثر شمولاً. فأنت تريد كل السجلات التى يبدأ فيها حقل TOWN بالحرف الكبير R.

```
.DISP FOR TOWN = 'R'     <cr>

Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE    SALARY NOTE
      6  030     TIM MONTAL     ROCHESTER RBG .F. 07/07/81    41900.00 memo
      8  040     JAMES JAMESON  ROCHESTER GSD .T. 10/21/77    29800.00 memo
     14  070     JAN MOREY      ROXY       GSD .T. 04/23/67    18190.00 memo

.DISP FOR TOWN = 'R'     <cr>
```

تحصل على نفس النتيجة التى سبق الحصول عليها. وهذا يظهر الحقيقة بأن الأوامر مرنة. الفراغات قبل وبعد المؤشرات أو قبل وبعد المؤثرات العلاقية (فى هذه الحالة علامة التساوى =) لا تعنى شيء.

ملاحظة : فى كل الحالات السابقة واللاحقة تلاحظ أننى أقارن اسم الحقل مع ثابت وليس العكس. احتفظ دائماً بهذا التسلسل. إذا حاولت أن تكتب العبارة على النحو التالى : DISP FOR 'ROCH' = TOWN فلن تجد أى شيء يحقق هذا الشرط!

```
.DISP RECO 5 FOR TOWN = 'R'   (cr)
```

الأمر السابق يطلب عرض سجل محدد أنه رقم 5 إذا كان ينتمى إلى المدينة 'R' فقط.

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE    SALARY NOTE
```

يمكن للسجل رقم 5 أن يحقق أو لا يحقق شرط العرض وذلك طبقاً للبيانات الموجودة فى السجل. وفى حالتنا هذه لا يحقق السجل رقم 5 الشرط.

لاحظ أنه لا يوجد أى سجل يحقق الشرط وأنك تحصل على عناوين الأعمدة فقط (نظراً لأنك استخدمت SET HEADING ON) مع عدم ظهور بيانات كمخرجات.

ادرس المثال التالى بعناية :

```
.USE PERSNL     <cr>
.DISP ALL       <cr>

Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE    SALARY NOTE
      1  005     NINA BHARUCHA  WEBSTER    BSG .T. 05/24/80   25000.00 MEMO
      2  010     PETE JOHNSON   brighton   BSG .T. 02/03/76   27590.00 MEMO
      3  015     GLORIA PATEL   FAIRPORT   RMG .T. 07/16/82   27500.00 memo
      4  020     MAX LEVINSKY   HENRIETTA  RMG .F. 04/13/69   27550.00 MEMO
      5  025     KIM BRANDT     FAIRPORT   RMG .F. 04/04/77   36000.00 MEMO
      6  030     TIM MONTAL     ROCHESTER  RBG .F. 07/07/81   41900.00 memo
      7  035     WILLIAM PATEL  penfield   GSD .F. 08/17/71   28900.00 memo
      8  040     JAMES JAMESON  ROCHESTER  GSD .T. 10/21/77   29800.00 memo
      9  045     MORRIS KATZ    webster    BSG .F. 09/14/80   23450.00 memo
     10  050     PAUL BHARUCHA  BRIGHTON   BSG .T. 05/23/73   29100.00 MEMO
     11  055     PHIL MARTIN    WEBSTER    RMG .F. 07/19/80   31000.00 memo
     12  060     JOHN PETERSON  BRIGHTON   RBG .T. 04/17/79   31480.00 MEMO
     13  065     JOY HARDY      fairport   RBG .F. 01/19/79   34200.00 MEMO
     14  070     JAN MOREY      ROXY       GSD .T. 04/23/67   18190.00 memo
     15  075     JOHN JONES     rochester  GSD .T. 04/04/70   25100.00 memo

.1      <cr>  This moves the record-pointer to record #1.
.DISP NEXT 4 FOR TOWN = 'FAIR'     <cr>

Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE    SALARY NOTE
      3  015     GLORIA PATEL   FAIRPORT   RMG .T. 07/16/82   27500.00 memo
```

نظراً لأن مؤشر NEXT n يشير إلى سجلات واقعية وليس إلى حدوثات منطقية فإن dBASE يختبر السجلات الواقعية الأربعة التالية NEXT 4 بدأ بالسجل الحالى ويحدد ما يحقق شرط TOWN = 'FAIR'. وهذا ما أعنيه عندما أقول أن مؤشر NEXT n يشير إلى سجلات واقعية وليس إلى حدوثات منطقية. أى أنك تستطيع أن تقول أن مدى إحدى العمليات له أولوية على شرط For.

تأثير التتابع EFFECT OF SEQUENCING

بالنسبة إلى الميكروكمبيوتر إذا كانت البيانات متتابعة فى ترتيب تصاعدى معتاد فتأتى الأرقام أولاً (فهى أقل منطقياً logically lower من كل الحروف). كما أن مجموعة الحروف الأبجدية الكبيرة تأتى قبل مجموعة الحروف الأبجدية الصغيرة (فهى أقل منطقياً logically lower منها). أى أنك إذا طلبت عرضاً تتابعياً طبقاً للمدينة لكل سجلات ملفك فأنك تحصل على قائمة كالمبينة أدناه. (أرقام السجلات تظل كما هى الأرقام الأصلية من قاعدة البيانات، إلا أن السجلات أعيد ترتيبها لتبين نتيجة الترتيب التصاعدى. والعمليات الدقيقة للتتابع مغطاة بالتفصيل فى قسم لاحق من الكتاب.)

Record#	EMP_NUM	EMP_NAME	TOWN	ORG	EXE	DT_OF_HIRE	SALARY	NOTE
10	050	PAUL BHARUCHA	BRIGHTON	BSG	.T.	05/23/73	29100.00	MEMO
12	060	JOHN PETERSON	BRIGHTON	RBG	.T.	04/17/79	31480.00	MEMO
3	015	GLORIA PATEL	FAIRPORT	RMG	.T.	07/16/82	27500.00	memo
5	025	KIM BRANDT	FAIRPORT	RMG	.F.	04/04/77	36000.00	MEMO
4	020	MAX LEVINSKY	HENRIETTA	RMG	.F.	04/13/69	27550.00	MEMO
6	030	TIM MONTAL	ROCHESTER	RBG	.F.	07/07/81	41900.00	memo
8	040	JAMES JAMESON	ROCHESTER	GSD	.T.	10/21/77	29800.00	memo
14	070	JAN MOREY	ROXY	GSD	.T.	04/23/67	18190.00	memo
1	005	NINA BHARUCHA	WEBSTER	BSG	.T.	05/24/80	25000.00	MEMO
11	055	PHIL MARTIN	WEBSTER	RMG	.F.	07/19/80	31000.00	memo
2	010	PETE JOHNSON	brighton	BSG	.T.	02/03/76	27590.00	MEMO
13	065	JOY HARDY	fairport	RBG	.F.	01/19/79	34200.00	MEMO
7	035	WILLIAM PATEL	penfield	GSD	.F.	08/17/71	28900.00	memo
15	075	JOHN JONES	rochester	GSD	.T.	04/04/70	25100.00	memo
9	045	MORRIS KATZ	webster	BSG	.F.	09/14/80	23450.00	memo

أمثلة على العرض مع استخدام شروط مكونة من حقول حرفية

(تابع ما سبق)

EXAMPLES OF DISPLAY WITH CHARACTER FIELDS-Continued

حاول الآن إجراء التجربة التالية، مع استخدام ملف PERSNL الأصلى كمدخلات.

. DISP FOR TOWN < 'R' <cr>

حيث < هى أقل من.

كل المدن التى تبدأ أسماؤها بأى حرف أقل من الحرف R – فى مجموعة الحروف الأبجدية – يقع عليها الاختيار للعرض. (تذكر علامتى التنصيص، وقواعد أسبقية الحروف الكبيرة، والحروف الصغيرة.)

Record#	EMP_NUM	EMP_NAME	TOWN	ORG	EXE	DT_OF_HIRE	SALARY	NOTE
3	015	GLORIA PATEL	FAIRPORT	RMG	.T.	07/16/82	27500.00	memo
4	020	MAX LEVINSKY	HENRIETTA	RMG	.F.	04/13/69	27550.00	MEMO
5	025	KIM BRANDT	FAIRPORT	RMG	.F.	04/04/77	36000.00	MEMO
10	050	PAUL BHARUCHA	BRIGHTON	BSG	.T.	05/23/73	29100.00	MEMO
12	060	JOHN PETERSON	BRIGHTON	RBG	.T.	04/17/79	31480.00	MEMO

. DISP FOR TOWN > "R" <cr>

حيث > هى أكبر من.

كل المدن التى تبدأ أسماؤها بأى حرف أعلى من الحرف R – فى مجموعة الحروف الأبجدية – يقع عليها الاختيار للعرض. (علامتى التنصيص الفردية أو الزوجية مقبولة.)

Record#	EMP_NUM	EMP_NAMBE	TOWN	ORG	EXE	DT_OF_HIRE	SALARY	NOTE
1	005	NINA BHARUCHA	WEBSTER	BSG	.T.	05/24/80	25000.00	MEMO
2	010	PETE JOHNSON	brighton	BSG	.T.	02/03/76	27590.00	MEMO
7	035	WILLIAM PATEL	penfield	GSD	.F.	08/17/71	28900.00	memo
9	045	MORRIS KATZ	webster	BSG	.F.	09/14/80	23450.00	memo
11	055	PHIL MARTIN	WEBSTER	RMG	.F.	07/19/80	31000.00	memo
13	065	JOY HARDY	fairport	RBG	.F.	01/19/79	34200.00	MEMO
15	075	JOHN JONES	rochester	GSD	.T.	04/04/70	25100.00	memo

DISP FOR TOWN <> 'R' <cr>

. DISP FOR TOWN <> 'R' <cr>

حيث <> هو رمز عدم التساوى. ولا يمكنك إدخال هذا الرمز على الصورة <> (لا يوجد فراغ داخل الرمز!). إشارة <> بالرغم من أنها مكونة من رمزين، إلا أنها تعتبر رمزاً واحداً.

كل المدن التى تبدأ أسماؤها بأى حرف لا يساوى R – فى مجموعة الحروف الأبجدية – يقع عليها الاختيار لعرضها.

Record#	EMP_NUM	EMP_NAME	TOWN	ORG	EXE	DT_OF_HIRE	SALARY	NOTE
1	005	NINA BHARUCHA	WEBSTER	BSG	.T.	05/24/80	25000.00	MEMO
2	010	PETE JOHNSON	brighton	BSG	.T.	02/03/76	27590.00	MEMO
3	015	GLORIA PATEL	FAIRPORT	RMG	.T.	07/16/82	27500.00	memo
4	020	MAX LEVINSKY	HENRIETTA	RMG	.F.	04/13/69	27550.00	MEMO
5	025	KIM BRANDT	FAIRPORT	RMG	.F.	04/04/77	36000.00	MEMO
7	035	WILLIAM PATEL	penfield	GSD	.F.	08/17/71	28900.00	memo
9	045	MORRIS KATZ	webster	BSG	.F.	09/14/80	23450.00	memo
10	050	PAUL BHARUCHA	BRIGHTON	BSG	.T.	05/23/73	29100.00	MEMO
11	055	PHIL MARTIN	WEBSTER	RMG	.F.	07/19/80	31000.00	memo
12	060	JOHN PETERSON	BRIGHTON	RBG	.T.	04/17/79	31480.00	MEMO
13	065	JOY HARDY	fairport	RBG	.F.	01/19/79	34200.00	MEMO
15	075	JOHN JONES	rochester	GSD	.T.	04/04/70	25100.00	memo

يمكن أن تستخدم إشارة # على أنها رمز لعدم التساوى أيضاً.

. DISP FOR TOWN # 'R' <cr>

وتحصل على نفس النتيجة التى سبق لك الحصول عليها، وقد يكون هذا أسهل فى الكتابة عن شرط عدم التساوى.

. DISP FOR TOWN < = 'R' <cr>

رمز => هو رمز دال على خليط أقل من أو يساوى، ومرة أخرى.. لايمكنك أن تكتب هذا على الصورة => (لايوجد فراغ داخل الرمز!). ويعتبر => رمزاً واحداً.

ملاحظة : يمكنك استخدام =>، أما استخدام >= فغير مسموح به.

Record#	EMP_NUM	EMP_NAME	TOWN	ORG	EXE	DT_OF_HIRE	SALARY	NOTE
3	015	GLORIA PATEL	FAIRPORT	RMG	.T.	07/16/82	27500.00	memo
4	020	MAX LEVINSKY	HENRIETTA	RMG	.F.	04/13/69	27550.00	MEMO
5	025	KIM BRANDT	FAIRPORT	RMG	.F.	04/04/77	36000.00	MEMO
6	030	TIM MONTAL	ROCHESTER	RBG	.F.	07/07/81	41900.00	memo
8	040	JAMES JAMESON	ROCHESTER	GSD	.T.	10/21/77	29800.00	memo
10	050	PAUL BHARUCHA	BRIGHTON	BSG	.T.	05/23/73	29100.00	MEMO
12	060	JOHN PETERSON	BRIGHTON	RBG	.T.	04/17/79	31480.00	MEMO
14	070	JAN MOREY	ROXY	GSD	.T.	04/23/67	18190.00	memo

. DISP FOR TOWN >= 'R' <CR>

رمز =< هو خليط أكبر من أو يساوى. مرة أخرى لا يمكنك كتابة على الصورة = > (لايوجد فراغ داخل الرمز!). يعتبر =< رمزاً واحداً.

ملاحظة ، يمكنك استخدام =<، وغير مسموح باستخدام => .

Record#	EMP_NUM	EMP_NAME	TOWN	ORG	EXE	DT_OF_HIRE	SALARY	NOTE
1	005	NINA BHARUCHA	WEBSTER	BSG	.T.	05/24/80	25000.00	MEMO
2	010	PETE JOHNSON	brighton	BSG	.T.	02/03/76	27590.00	MEMO
6	030	TIM MONTAL	ROCHESTER	RBG	.F.	07/07/81	41900.00	memo
7	035	WILLIAM PATEL	penfield	GSD	.F.	08/17/71	28900.00	memo
8	040	JAMES JAMESON	ROCHESTER	GSD	.T.	10/21/77	29800.00	memo
9	045	MORRIS KATZ	webster	BSG	.F.	09/14/80	23450.00	memo
11	055	PHIL MARTIN	WEBSTER	RMG	.F.	07/19/80	31000.00	memo
13	065	JOY HARDY	fairport	RBG	.F.	01/19/79	34200.00	MEMO
14	070	JAN MOREY	ROXY	GSD	.T.	04/23/67	18190.00	memo
15	075	JOHN JONES	rochester	GSD	.T.	04/04/70	25100.00	memo

أمثلة على العرض مع استخدام حقول عددية

EXAMPLES OF DISPLAY WITH NUMERIC FIELDS

ملاحظة ، عندما تحدد قيماً تتحقق فى أحد الشروط.. ففى حالة الحقول العددية، يأخذ dBASE القيمة التى تقدمها فى معناها الجبرى، كما أن القيمة التى تتحدد، يجب ألا توضع بين علامتى تنصيص كذلك.

. DISP FOR SALARY = 25 <cr>

لاحظ أنه فى المثال السابق.. لا يتحقق الشرط لأى من العاملين؛ لأن dBASE يبحث عن راتب يتساوى تماماً مع 25 دولار. وبالنسبة للحقول العددية.. تؤخذ القيمة الجبرية لها، فإذا كنت تبحث عن القيمة 25000.. فيجب أن تدخلها على نفس صورتها :

. DISP FOR SALARY = 25000 <cr>

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      1  005     NINA BHARUCHA WEBSTER    BSG .T. 05/24/80   25000.00 MEMO
```

حاول مع بقية هذا، مثل ما هو مبين، إذا ما رغبت؛ فقد اخترت ألا أعرض القوائم التفصيلية.

```
DISP FOR SALARY < 25000      <cr>
DISP FOR SALARY # 25000      <cr>
.DISP FOR SALARY > 25000     <cr>
.DISP FOR SALARY >= 25000    <cr>
.DISP FOR SALARY <= 25000    <cr>
DISP FOR SALARY > 25000 50   <cr>
```

لاحظ أعلاه أنك قد تبين – إذا كان ضرورياً – المواقع العشرية فى القيم العددية.

امثلة على العرض مع استخدام حقول منطقية

EXAMPLES OF DISPLAY WITH LOGICAL FIELDS

. DISP FOR EXE <cr>

لاحظ انك لا تحتاج إلى أى مؤثر علاقى فى هذه الحالة؛ لأنك تتعامل مع حقول منطقية. والتأثير هو أنك تطلب عرض كل السجلات، التى يكون فيها الحقل EXEMPT حقيقياً؛ أى :

.EXEMPT = <true>

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      1  005     NINA BHARUCHA WEBSTER    BSG .T. 05/24/80   25000.00 MEMO
      2  010     PETE JOHNSON  brighton   BSG .T. 02/03/76   27590.00 MEMO
      3  015     GLORIA PATEL  FAIRPORT   RMG .T. 07/16/82   27500.00 memo
      8  040     JAMES JAMESON ROCHESTER  GSD .T. 10/21/77   29800.00 memo
     10  050     PAUL BHARUCHA BRIGHTON   BSG .T. 05/23/73   29100.00 MEMO
     12  060     JOHN PETERSON BRIGHTON   RBG .T. 04/17/79   31480.00 MEMO
     14  070     JAN MOREY     ROXY       GSD .T. 04/23/67   18190.00 memo
     15  075     JOHN JONES    rochester  GSD .T. 04/04/70   25100.00 memo
```

. DISP FOR . NOT . EXE <cr>

التأثير هو أنك تطلب عرض كل السجلات، التى يكون فيها EXEMPT = <false>

Record#	EMP_NUM	EMP_NAME	TOWN	ORG	EXE	DT_OF_HIRE	SALARY	NOTE
4	020	MAX LEVINSKY	HENRIETTA	RMG	.F.	04/13/69	27550.00	MEMO
5	025	KIM BRANDT	FAIRPORT	RMG	.F.	04/04/77	36000.00	MEMO
6	030	TIM MONTAL	ROCHESTER	RBG	.F.	07/07/81	41900.00	memo
7	035	WILLIAM PATEL	penfield	GSD	.F.	08/17/71	28900.00	memo
9	045	MORRIS KATZ	webster	BSG	.F.	09/14/80	23450.00	memo
11	055	PHIL MARTIN	WEBSTER	RMG	.F.	07/19/80	31000.00	memo
13	065	JOY HARDY	fairport	RBG	.F.	01/19/79	34200.00	MEMO

إذا كان الحقل EXEMPT معرفاً بأنه حقل حرفى معتاد.. فإن الحصول على القائمة السابقة يتطلب أوامر للعرض على النحو التالى :

. DISP FOR EXE = 'T' <cr>

أو . DISP FOR EXE = 'F' <cr>

طبقاً للبيانات، التى سبق إدخالها فى الحقل.

أمثلة على العرض مع استخدام حقول تاريخ

EXAMPLES OF DISPLAY WITH DATE FIELDS

تقدم أمثلة على أمر DISPLAY FOR، باستخدام حقول تاريخ، فيما بعد عند دراستك قسم الدوال Functions.

أمثلة على العرض مع استخدام حقول مذكرات

EXAMPLES OF DISPLAY WITH MEMO FIELDS

من فضلك.. لاحظ أن dBASE لا يقبل عبارة على النحو التالى :

. DISP FOR NOTES = 'T' <cr>

تحاول العبارة السابقة أن تسرد كل السجلات، التىّ تبدأ فيها المذكرات بالحرف T. ولسوء الحظ.. فإنك تحصل على رسالة، تفيد أن المؤثر غير صحيح invalid operator.

إلا أن هناك طرقاً أخرى لاستخلاص بيانات من حقول مذكرات، وبعض أمثلة أمر DI-SPLAY FOR، وذلك باستخدام حقول مذكرات، تقدم لك فيما بعد عند دراستك قسم الدوال Functions.

الشروط المركبة والشروط المتعدد

COMPLEX / MULTIPE CONDITIONS

يمكن أن يكون لديك أكثر من شرط واحد، يجب أن يتحقق؛ حتى يقع الاختيار على السجلات المراد عرضها.

```
DISP FOR TOWN = 'ROXY' .AND. SALARY < 25000  <cr>
```

ملاحظة ، يتطلب تكوين dBASE وجود نقطة، تسبق المؤثر المنطقي، وأخرى تلحقه، والمؤثرات المنطقية، هي : .AND. .OR. .NOT.

```
Record#  EMP_NUM EMP_NAME      TOWN      ORG EXE DT_OF_HIRE   SALARY NOTE
     14  070     JAN MOREY     ROXY      GSD .T. 04/23/67   18190.00 memo
```

```
. DISP FOR ORG = 'BSG' .AND. SALARY > = 25000  <cr>
```

```
Record#  EMP_NUM EMP_NAME      TOWN      ORG EXE DT_OF_HIRE   SALARY NOTE
      1  005     NINA BHARUCHA WEBSTER   BSG .T. 05/24/80   25000.00 MEMO
      2  010     PETE JOHNSON  brighton  BSG .T. 02/03/76   27590.00 MEMO
     10  050     PAUL BHARUCHA BRIGHTON  BSG .T. 05/23/73   29100.00 MEMO
```

```
DISP EMP-NUM, EMP-NAM, ORG FOR ORG = 'BSG' .OR. ORG = 'GSD'
```

```
Record#  EMP_NUM EMP_NAME      ORG
      1  005     NINA BHARUCHA BSG
      2  010     PETE JOHNSON  BSG
      7  035     WILLIAM PATEL GSD
      8  040     JAMES JAMESON GSD
      9  045     MORRIS KATZ   BSG
     10  050     PAUL BHARUCHA BSG
     14  070     JAN MOREY     GSD
     15  075     JOHN JONES    GSD
```

يمكن أن يكون لديك أى خليط متعدد من AND و OR.

```
.DISP FOR TOWN = 'ROCH' .AND. (ORG = 'BSG' .OR. ORG = 'GSD')
```

يجب أن تستخدم أقواساً؛ لكى توضح المنطق المطلوب من عبارة الأمر. وفى غياب الأقواس.. ينفذ dBASE الشروط طبقاً لمنطقه التقليدى، مع استمراره للحصول على مخرجات، قد لا تكون هى المطلوبة على الإطلاق.

فى المثال السابق.. تريد سرد السجلات التى تكون المدينة لها هى ROCH، والتى إما أن تكون المؤسسة لها هى BSG، أو GSD.

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      8  040     JAMES JAMESON ROCHESTER GSD .T. 10/21/77   29800.00 memo
```

فيما يلى.. بعض أمثلة أخرى للشروط المركبة :

```
.DISP FOR (ORG = 'B' .AND. SALARY > 35000) .OR.
          (ORG = 'G' .AND. SALARY > 50000)      <cr>
```

(لا يوجد أى من العاملين الذين يحققون العبارة السابقة.)

```
.DISP FOR (TOWN = 'R' .OR. TOWN = 'F') .AND. (ORG = 'G' .OR.
                                              ORG = 'B')      <cr>
```

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      8  040     JAMES JAMESON ROCHESTER GSD .T. 10/21/77   29800.00 memo
     14  070     JAN MOREY     ROXY       GSD .T. 04/23/67   18190.00 memo
```

وفيما يلى.. مثال شامل لشرط مركب :

```
.DISP FOR ( ..1.. .AND. ( ..2.. .OR. ..3.. )) .OR.
          ( ..4.. .AND. ( ..5.. .OR. ..6.. ))
```

حيث 1, 2, 3, 4, 5, 6 هى أى شروط.

تأكد أنك وضعت الأقواس المناسبة؛ لتوضيح المنطق المقصود تماماً! وتذكر - كذلك - أنك تستطيع استخدام Ctrl-Home فى تنشيط محتويات الشاشة، كنافذة لسطر الأمر.

THE WHILE PARAMETER

مؤشر بينما

فى حالة مؤشر FOR.. يقع الاختيار على كل السجلات التى تحقق شرط FOR لعرضها (إلا إذا استخدم بالطبع مدى SCOPE محدداً).

يختار مؤشر WHILE سجلات أثناء WHILE تحقق الشرط فقط. وبمعنى آخر.. بالرغم من أن أحد السجلات يكون محققاً للشرط، إلا أنه يكون فى المكان الخطأ من الملف؛ حيث لا يحقق سجلاً آخر يسبقه الشرط الخاص بـ (WHILE أى يجعله <false>).

انظر مرة أخرى إلى بيانات ملف PERSNL، ثم ادرس الفرق التالى بين استخدام شرط FOR، واستخدام شرط WHILE.

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE    SALARY NOTE
      1  005     NINA BHARUCHA  WEBSTER    BSG .T. 05/24/80    25000.00 MEMO
      2  010     PETE JOHNSON   brighton   BSG .T. 02/03/76    27590.00 MEMO
      3  015     GLORIA PATEL   FAIRPORT   RMG .T. 07/16/82    27500.00 memo
      4  020     MAX LEVINSKY   HENRIETTA  RMG .F. 04/13/69    27550.00 MEMO
      5  025     KIM BRANDT     FAIRPORT   RMG .F. 04/04/77    36000.00 MEMO
      6  030     TIM MONTAL     ROCHESTER  RBG .F. 07/07/81    41900.00 memo
      7  035     WILLIAM PATEL  penfield   GSD .F. 08/17/71    28900.00 memo
      8  040     JAMES JAMESON  ROCHESTER  GSD .T. 10/21/77    29800.00 memo
      9  045     MORRIS KATZ    webster    BSG .F. 09/14/80    23450.00 memo
     10  050     PAUL BHARUCHA  BRIGHTON   BSG .T. 05/23/73    29100.00 MEMO
     11  055     PHIL MARTIN    WEBSTER    RMG .F. 07/19/80    31000.00 memo
     12  060     JOHN PETERSON  BRIGHTON   RBG .T. 04/17/79    31480.00 MEMO
     13  065     JOY HARDY      fairport   RBG .F. 01/19/79    34200.00 MEMO
     14  070     JAN MOREY      ROXY       GSD .T. 04/23/67    18190.00 memo
     15  075     JOHN JONES     rochester  GSD .T. 04/04/70    25100.00 memo
```

```
.USE PERSNL     <cr>
.6      <cr>    Make 6 the current record.
.DISP WHILE TOWN = 'ROCH'
```

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE    SALARY NOTE
      6  030     TIM MONTAL     ROCHESTER  RBG .F. 07/07/81    41900.00 memo
```

دائماً ما يبدأ تنفيذ الأمر مع وجود مؤشر WHILE عند موقع السجل الحالى. ولاحظ أنه بالبدء بالسجل الحالى.. يعرض السجل رقم 6 فقط؛ لأنه يحقق الشرط، ولا يحقق السجل رقم 7 الشرط؛ فيتوقف العرض.

```
.DISP FOR TOWN = 'ROCHESTER'     <cr>

Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE    SALARY NOTE
      6  030     TIM MONTAL     ROCHESTER RBG .F. 07/07/81    41900.00 memo
      8  040     JAMES JAMESON ROCHESTER GSD .T. 10/21/77    29800.00 memo
```

دائماً ما يبدأ تنفيذ الأمر مع وجود مؤشر FOR عند بداية الملف، ولاحظ أعلاه أن كل السجلات التى تحقق شرط FOR وقع الاختيار عليها للعرض.

فى أى حالة من الحالتين.. إذا حققت سجلات أكثر مما تعرضه الشاشة الشرط.. فإن النظام ينتظر بين الشاشتين.

بالنظر مرة أخرى إلى السرد الكامل لملف PERSNL (الذى سبق عرضه من قبل).. ادرس المثال التالى؛ حيث إن WHILE و FOR موجودتان فى نفس العبارة.

```
.1
.DISP NEXT 10    FOR SALARY > 20000    WHILE EXE

Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE    SALARY NOTE
      1  005     NINA BHARUCHA  WEBSTER    BSG .T. 05/24/80    25000.00 MEMO
      2  010     PETE JOHNSON   brighton   BSG .T. 02/03/76    27590.00 MEMO
      3  015     GLORIA PATEL   FAIRPORT   RMG .T. 07/16/82    27500.00 memo
```

لاحظ أنه بالرغم من أن هناك أكثر من 3 سجلات تحقق شرط SALARY.. إلا أن أول سجل غير المستثنى (السجل رقم 4) انهى العرض، وهذا يعنى أن WHILE لها أولوية على FOR.

بمعرفة الحقيقة أن مؤشر WHILE يتطلب تجميع السجلات؛ لتعمل بطريقة صحيحة.. فقد تميل إلى أن تفكر أن مؤشر FOR أكثر فائدة فى نظام dBASE، إلا أنه تحت الظروف الصحيحة يكون WHILE أكثر فائدة جداً من FOR.

اعتبر الحالة التالية.

افرض أن عندى ملف عمليات جارية للمخزون؛ مرتباً طبقاً لحقل رئيسي، اسمه PART-NUM.، وأريد أن استخلص كل العمليات الجارية لرقم الجزء المساوى 'J123'. من الواضح

أن كل العمليات الجارية – فى هذا الملف التتابعى – للرقم 'J123' تأتى وراء بعضها؛ فإذا أدخلت :

. DISP FOR PART-NUM = 'J123' <CR>

يبدأ dBASE عند بداية هذا الملف، ويقرأ ويهمل بضع مئات (وربما بضع آلاف) من السجلات؛ حتى يجد مجموعة السجلات المطلوبة، التى يساوى رقم الجزء لها 'J123' فيقرأ ويعرض هذه السجلات ويعرضها، ثم يستمر فى قراءة واهمال بضع مئات (وربما بضع آلاف) من السجلات الأخرى، قبل أن يتوقف تشغيل الملف، وتأثير ذلك هو أنه لقراءة وعرض عدد محدود من السجلات.. يقوم dBASE بقراءة كل محتويات الملف، وهذا يعتبر ضياعاً كبيراً للوقت.

اعتبر نفس الحالة الآن – مع استثناء واحد – لقد رتبت نفسى لأجد أول حدوث لرقم الجزء المطلوب فوراً؛ أى إننى انتقلت إلى أول عملية جارية لها رقم الجزء 'J123'. وفى الموقع المناسب فى هذا الكتاب.. سأوضح لك كيف تجد بالضبط أى قيمة رئيسية مطلوبة فوراً.

أننى أدخل الأمر التالى الآن :

. DISP WHILE PART-NUM = 'J123' <cr>

لاحظ استخدام مؤشر WHILE. يبدأ dBASE فى الموقع الموجود فيه؛ أى عند أول رقم جزء يساوى 'J123'، ويعرض السجلات التالية أثناء WHILE تساوى رقم الجزء مع 'J123'.

لا يبدأ dBASE فى هذه الحالة من بداية الملف، ولكنه يقرأ ويستخلص السجلات التى تحقق الشرط فقط، ولا يكون فى حاجة إلى قراءة بقية الملف.

وفى ملف به عديد من الآلاف من السجلات يمكن أن يكون الفرق فى الوقت بين هاتين الطريقتين كبيراً جداً. ويتراكم التأثير مئات المرات، إذا احتجت أن تستخلص سجلات؛ معتمداً على قيمة حقل رئيسى فى حالة الوقت الحقيقى real-time وفى الواقع.. تعتمد برامج الكمبيوتر بشدة على مقدرتك على ايجاد أول سجل يحقق القيمة الرئيسية (قيمة الحقل الرئيسى)، وتشغيل سجلات أثناء WHILE، تساوى قيم الحقل الرئيسى. وقد تتكرر هذه العملية داخل البرنامج مئات من المرات، ويكون استخدام WHILE حاسماً للنظام.

يمكن أن يكون شرط WHILE بسيطاً أو معقداً؛ طبقاً لما تفعله به. وسوف ترى مزيداً عن مؤشر WHILE، فى قسم البرمجة من هذا الكتاب.

مؤشر إلى الطباعة THE TO PRINT PARAMETER

بالنسبة إلى كل الأمثلة التى سبق ذكرها.. إذا حددت جزء TO PRINT فى نهاية أى من العبارات.. فسوف تعرض المخرجات على الشاشة، كما أنها تتجه إلى الطابع كذلك. ومثال ذلك:

. DISP FOR TOWN = 'R' TO PRINT <cr>

هذا الأمر يرسل المخرجات إلى كل من الشاشة والطابع.

تتجه المخرجات المطبوعة إلى الطابع، مع عدم ترك هامش من ناحية اليسار -left margin؛ أى إن الطباعة تبدأ من عند أقصى موقع طباعة على اليسار، محدد بواسطة رأس الطباعة. وإذا أردت أن تحرك الهامش الأيسر قليلاً ناحية اليمين.. فيمكنك كتابة ما يلى :

SET MARGIN TO 10 <cr>

تبدأ المخرجات المطبوعة الآن بعد 10 مواقع من على يمين موقع الطباعة، المحدد بواسطة الطابع.

. SET MARGIN TO <cr>

يعيد الهامش الأيسر إلى القيمة صفر؛ أى عدم وجوده بالنسبة إلى أقصى موقع طباعة، محدد بواسطة الطابع.

مؤشر إلى ملف THE TO FILL PARAMETER

كما يمكن تخمينه بسهولة.. فكل العروض التى سبق اختيارها، يمكن أن ترسل إلى ملف، يسمى بأى اسم تحدده (filename of your choice)، مثل :

. DISP FOR SALARY > 3000 TO FILE SALARIES <cr>

يرسل هذا الأمر المخرجات إلى كل من الشاشة وملف مخرجات اسمه SALARIES.TXT، وهو ملف نصى حيث يقدم dBASE الاسم الثانوى التقليدى TXT. ويمكن أن يجرى أى تشغيل لاحق على هذا الملف، وليكن التشغيل بواسطة مشغل كلمات مثلاً، ويمكن جعل الملف جزءًا من خطاب أو وثيقة.

مؤشر الفصل THE OFF PARAMETER

المؤشر الأخير حالياً هو مؤشر OFF، وهو يحذف هذا المؤشر ببساطة أرقام السجلات من العرض. إذا كنت قد لاحظت.. فكل من العروض السابقة يبين أرقاماً للسجلات على اليسار. (أرقام السجلات هذه يقدمها dBASE، مع اضافة السجلات إلى قاعدة البيانات.) ويحدد مؤشر الفصل OFF أنك لا تريد أن ترى أرقام السجلات.

نظراً لأن أرقام السجلات، يمكن أن تكون مفيدة، للذهاب مباشرة إلى كل السجلات على سبيل المثال، فمن غير المستحب استخدامك لمؤشر OFF فى DISPLAY

أمر السرد LIST COMMAND

أنظر فصل 20 عن الأوامر المتنوعة Miscellaneous Commands؛ لمعرفة أمر السرد LIST.

الفصل العاشر

الـــدوال

FUNCTIONS

الدوال Functions هى مقاطع منفعة utility routines، مبنية داخلياً فى dBASE، وتساعد المستفيد أو المبرمج، فى أداء أنشطة متخصصة. وبدون هذه المقاطع على مستوى التجميع.. تصبح هذه الأنشطة المتخصصة صعبة جداً فى تنفيذها، إذا لم تكن مستحيلة.

من خلال استخدام هذه الدوال المبنية داخل dBASE.. يمكن دعم قوة أوامر العرض (وأوامر أخرى) فعلاً. وتأتى كل صيغة جديدة من صيغ dBASE بمزيد من الدوال المبنية داخلها، ويجب تذكرة الدوال على حساب إمكانياتها المتزايدة التى تقدمها.

وفى هذه اللحظة.. فإننى أركز على أكثر الدوال فائدة فى أداء أنشطة المقاطع، والبرامج، والدوال التى يلى ذكرها، هى امتدادات طبيعية لأمثلة العرض سالفة الذكر، وتوسع بشدة من مقدراتك على استخلاص البيانات ومعالجتها. ومجموعة الدوال الكاملة مذكورة فى الدليل التقنى لنظام dBASE.

قبل أن تبدأ تذكر ما يلى :

- الدوال عبارة عن مؤشرات وليست أوامر؛ أى إنها تساعدك فى تقليل الأمر، كما تفعل بقية المؤشرات الأخرى.
- يمكن تقليل اسم كل دالة إلى الأربعة حروف الأولى منه فقط، ولا تكون بحاجة إلى كتابة اسم الدالة كاملاً.
- تحتوى الدوال دائماً على أقواس، وإما أن تظهر الأقواس فى نهاية الدالة، أو أنها تحتوى على مؤشرات تعمل الدالة عليها.
- كل دالة تدرسها تطبق كذلك على مفردات، تسمى متغيرات ذاكرة memory variables فى مرحلة البرمجة فقط، وأنا لن أذكرها فى هذه المناقشة مرة أخرى.

THE LEN ( ) FUNCTION دالـة الطـول

تعيد الدالة طول الحقل الحرفى المحدد، أو طول التعبير الحرفى المحدد، أو طول حقل المذكرة المحدد.

.USE PERSNL <cr>
.LEN (TOWN) <cr>

وهى مـخـرجـات الدالة من dBASE. فيـقـدم dBASE طول الحـقل، بغض النظر عن محتوياته.

.? LEN (NOTE) <cr>

84 هى المخرجات من dBASE.. فيقدم dBASE القيمة 84، وهى طول محتويات المذكرة الخاصة بالسجل الحالى.

لاحظ أن طول الحقل ليس بالضرورة مثل طول البيانات الموجودة فى الحقل، ومثال ذلك حقل TOWN، يمكن أن يعرف بأن طوله هو 15 رمزاً، إلا أن حقل المدينة الموجود فيه Ro-chester تكون طول بياناته 9 رموز بيانات فقط.

لايجاد الطول الفعلى للبيانات داخل الحقل.. فإنك تكون بحاجة إلى استخدام خليط من دالتى ( ) LEN و ( ) TRIM. دالة ( ) TRIM موضحة فى هذا القسم فيما بعد.

THE SUBSTR ( ) FUNCTION دالـة السلسلة الجزئيـة

تطبق هذه الدالة على الحقول الحرفية وحقول المذكرة. وتستخدم إذا أردت التأكد من حدوث سلسلة string معينة داخل الحقل الحرفى أو داخل حقل المذكرة، مثل :

.DISP FOR TOWN = 'ROCH' <cr>

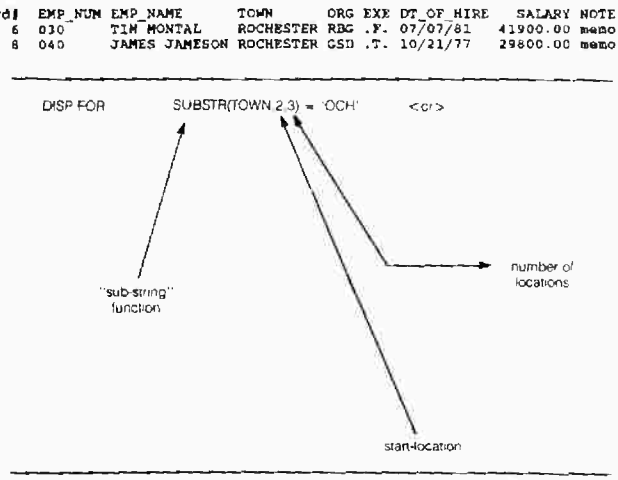
يختار السجلات التى لها الأربعة حروف الكبيرة R O C H فى أول أربع خانات لها، من حقل TOWN فقط. لقد سبق أن رأيت أمثلة عديدة لهذا فى الصفحات السابقة.

إلا أنك إذا أردت اختيار سجلات لها الحروف OCR فى الخانات الثانية والثالثة والرابعة من حقل TOWN.. فعليك أن تحدد ما يلى (انظر شكل 1-10) :

.DISP FOR SUBTR (TOWN, 2, 3) = 'OCH' <cr>

انك تحاول أن تنظر خلال السلسلة الجزئية لحقل المدينة، وتريد أن تحدد موقع البداية (موقع الحرف 2)، وتريد كذلك تحديد عدد المواقع (3 فى هذه الحالة). أى أن كل السجلات التى لها الحروف الثلاثة الكبيرة OCH فى أعمدتها 2, 3, 4 من حقل TOWN يقع الاختيار عليها لعرضها. وكما سبق ذكره.. فالقيمة التى تحددها (OCH) يجب أن توضع بين علامتى تنصيص؛ حيث إن حقل TOWN هو حقل حرفى.

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      6  030     TIM MONTAL     ROCHESTER RBG .F. 07/07/81   41900.00 memo
      8  040     JAMES JAMESON  ROCHESTER GSD .T. 10/21/77   29800.00 memo
```



شكل 1-10 : توضيح لكيفية تحديد دالة السلسلة الجزئية

إذا لم تحدد عدد المواقع.. فإن البديل التقليدى هو حتى نهاية السلسلة.

مثال ذلك :

. 6 <cr>
. ? SUBSTR (TOWN, 5) <cr>

ESTER هى المخرجات .

Sub-string on a Memo Field — جزئية من حقل مذكرة

لرؤية تأثير دالة ( ) SUBSTR عبر حقل مذكرة... لاحظ من فضلك أنك إذا حاولت نفس نوع عبارة الأمر :

```
. DISP FOR SUBSTR (NOTE, 7, 3) = 'son'  <CR>
```

ينتج من هذا خليط من سجلات صحيحة معروضة، ورسائل خطأ تنفيذ execution error، والسبب فى ذلك هو أن dBASE يحاول عمل سلسلة جزئية لكل حقول المذكرات المسماة NOTE، حتى إذا كانت سلاسل فارغة. وعلى حقل المذكرة الفارغ.. فإن مؤشر السلسلة الجزئية السابق 7 (موقع البداية) يقع خارج المدى بوضوح.

يمكن كتابة العبارة السابقة مختلفة بعض الشىء لسحب القائمة المطلوبة :

```
. DISP FOR LEN (NOTE) # 0 .AND. SUBSTR (NOTE, 7, 3) = 'son'   <cr>
```

هذا يجبر dBASE على تقويم دالة LEN (NOTE) قبل الجزء الآخر للعبارة، وتأثير ذلك هو تقويم السجلات التى لها مذكرات صحيحة فقط؛ لأن طول المذكرة الصحيحة لا يساوى صفراً.

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
     10  050     PAUL BHARUCHA BRIGHTON   BSG .T. 05/23/73   29100.00 MEMO
```

لإيجاد المذكرات التى تبدأ بالحرف 'T'.. يمكنك أن تكتب ما يلى :

```
.DISP FOR  LEN(NOTE) # 0    .AND.    SUBS(NOTE,1,1) = 'T'    <cr>

Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      1  005     NINA BHARUCHA WEBSTER    BSG .T. 05/24/80   25000.00 MEMO
     10  050     PAUL BHARUCHA BRIGHTON   BSG .T. 05/23/73   29100.00 MEMO
     12  060     JOHN PETERSON BRIGHTON   RBG .T. 04/17/79   31480.00 MEMO
```

THE $ ( ) FUNCTION — دالة فى اى مكان داخل

توجد صيغة بديلة من دالة السلسلة الجزئية أكثر قوة، وتجعلك تجد السلسلة الحرفية التى تريدها فى أى مكان داخل الحق؛ بدلا من مواقع محددة فقط. والكلمات الرئيسية هى أى مكان داخل anywhere within، والتى تعنيها دالة ( ) $.

. DISP FOR 'TER' $ (TOWN) <cr>

والتى تقرأ على النحو التالى : اعرض السجلات التى لها الرموز TER، فى أى مكان خلال الحقل TOWN.

لاحظ أنه يمكن ادخال هذه الدالة على النحو التالى كذلك :

```
.DISP FOR 'TER' $ TOWN     <cr>

Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE    SALARY NOTE
      1  005     NINA BHARUCHA  WEBSTER    BSG .T. 05/24/80   25000.00 MEMO
      6  030     TIM MONTAL     ROCHESTER  RBG .F. 07/07/81   41900.00 memo
      8  040     JAMES JAMESON  ROCHESTER  GSD .T. 10/21/77   29800.00 memo
     11  055     PHIL MARTIN    WEBSTER    RMG .F. 07/19/80   31000.00 memo

.DISP FOR 'JONES' $(EMP_NAME)     <cr>
```

يسرد هذا السجلات التى بها JONES، فى أى مكان فى حقل اسم العامل.

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE    SALARY NOTE
     15  075     JOHN JONES     rochester  GSD .T. 04/04/70   25100.00 memo
```

لاحظ مثالين آخرين لهذه الدالة. وأفرض أن عندى ملف تسجيل طلاب واحد. حقول هذا الملف اسمه CANCELCODE، ويشغل خانة واحدة. يمكن أن يحتوى الحقل على أى قيمة، ولتكن من 1 إلى 5 . هذه القيم تعتبر بالطبع بيانات حرفية.

وأريد أن أنتقى السجلات التى بها 1 أو 2 فى هذا الحقل المسمى CANCELCODE. ومن إحدى الطرق البسيطة لعمل ذلك ما يلى :

. LIST FOR CANCELCODE = '1' .OR. CANCELCODE = '2' <cr>

كما استطيع كذلك استخدام دالة $ فى هذه الحالة، بكتابة ما يلى :

. LIST FOR CANCELCODE $ ('1 2') <cr>

والتى تؤدى العمل!

تقرأ العبارة السابقة على النحو التالى : أسرد السجلات التى تكون قيمة CANCELCODE فى أى مكان السلسلة الحرفية فيها مساوياً 2 1. ولأن حقل

CANCELCODE يشغل خانة واحدة.. فقد اخترت السجلات التى تكون قيمة الخانة الواحدة فى أى مكان فى السلسلة الحرفية مساوية 12. أى إنى اخترت السجلات الموجود فى CANCELCODE لها القيمة 1 أو القيمة 2.

لاحظ من فضلك أنك إذا كتبت العبارة السابقة بطريقة مختلفة مثلما يلى :

. LIST FOR '12' $ (CANCELCODE) <cr>

فلا يوجد شىء يحقق هذا الشرط! السبب هو أن dBASE يترجم العبارة السابقة؛ لتقرأ على النحو التالى : اسرد السجلات التى توجد القيمة الحرفية 12 فى أى مكان فى CANCELCODE لها.

إلا أنه بسبب أن وجود قيمة من حرفين 12 فى حقل يشغل خانة واحدة CANCELCODE غير منطقى.. فلا يوجد أى سجل يحقق ذلك. إننى أقترح أن تعيد دراسة العبارتين السابقتين بعناية؛ لتفهم الفارق الأساسى بينهما.

دالة ( ) $ على حقل مذكرة $ ( ) on a Memo Field

لرؤية تأثير دالة ( ) $ على حقل ذاكرة.. لاحظ أنها تعمل معه مثل عملها مع الحقل الحرفى تماماً.

```
.DISP FOR  'son'  $(NOTE)     <cr>

Record#   EMP_NUM EMP_NAME       TOWN        ORG EXE DT_OF_HIRE    SALARY NOTE
     10   050      PAUL BHARUCHA  BRIGHTON    BSG .T. 05/23/73    29100.00 MEMO
     12   060      JOHN PETERSON  BRIGHTON    RBG .T. 04/17/79    31480.00 MEMO
```

يعمل هذا بطريقة مناسبة. إذا تذكرت بالنسبة إلى دالة ( ) SUBSTR... فعليك أن تكتب العبارة بحيث يمكن لنظام dBASE أن يهمل حقول المذكرة التى لا تحتوى على مذكرات، إلا أن دالة فى أى مكان خلال تعمل بطريقة جيدة مع حقل المذكرة. ومن الواضح أن لديك بعض التناقض هنا بين عمل دالتى ( ) SUBSTR و ( ) $ على حقل المذكرة.

كمثال آخر لاستخدام هذه الدالة على حقل مذكرة... افرض أن لديك أعداد نظام طبى؛ حيث يحفظ مدير المكتب حقل مذكرة للسجلات الرئيسية للمرضى. بافتراض أن اسم حقل الذاكرة هو NOTE، اكتب ما يلى :

.DISP FOR 'OVERDUE' $ (NOTE) <cr>

تتواجد قائمة بالمرضى الموجود فى أى مكان فى حقل الذاكرة، الذين لهم كلمة OVER-DUE.

دالة الحالة العليا (الحروف الكبيرة) THE UPPER ( ) FUNCTION

قد تكون لاحظت حتى الآن أنه فى حالة الحقول الحرفية.. يبحث dBASE عن قيمة حرفية تذكرها له فى التعبير. أى إذا كان إجراء إدخالك البيانات لا يتبعه اصطلاح فى الحالة العليا uppercase أو الحالة السفلى lowercase تحدده أنت بدقة.. فمن الممكن أن تنتهى البيانات بكونها خليط من رموز الحالة العليا والحالة السفلى؛ أى إن أوامر العرض DISPLAY تنتهى بعدم وجود بعض أو كل السجلات، التى تحقق الشرط فعلاً لوجود النوع الخطأ من الحالة للقيمة الثابتة.

إذا شككت أن هذا هو الحال.. يمكنك أن تطلب من dBASE أن يحاول إيجاد توافق، بغض النظر عن الحالة إذا كانت علوية أو سفلية.

.DISP FOR UPPER(TOWN) = 'ROCH' <cr>

Record#	EMP_NUM	EMP_NAME	TOWN	ORG	EXE	DT_OF_HIRE	SALARY	NOTE
6	030	TIM MONTAL	ROCHESTER	RBG	.F.	07/07/81	41900.00	memo
8	040	JAMES JAMESON	ROCHESTER	GSD	.T.	10/21/77	29800.00	memo
15	075	JOHN JONES	rochester	GSD	.T.	04/04/70	25100.00	memo

عند قراءة كل سجل من الملف.. ينفذ dBASE ترجمة داخلية لبيانات حقل TOWN إلى الحالة العليا، ثم يجرى بعد ذلك المقارنة مع ROCH. يشير اصطلاح الترجمة الداخلية inter-nal translation.. إلى ترجمة لحظية أو ترجمة منطقية أثناء تنفيذ الأمر فقط، أما البيانات الواقعية الموجودة فى الملف.. فتظل كما هى دون أى تغيير.

تأثير ذلك أنه يتم العثور على كل الخليط من رموز الحالة العليا والحالة السفلى للثابت الحرفى ROCH، الموجود فى أول أربعة خانات من الحقل TOWN.

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE    SALARY NOTE
      6  030     TIM MONTAL     ROCHESTER RBG .F. 07/07/81    41900.00 Memo
      8  040     JAMES JAMESON  ROCHESTER GSD .T. 10/21/77    29800.00 Memo
     14  070     JAN MOREY      ROCHESTER GSD .T. 04/23/67    18190.00 Memo
     15  075     JOHN JONES     rochester GSD .T. 04/04/70    25100.00 Memo

        .DISP FOR SUBSTR(UPPER(TOWN),2,3) = 'OCH'     <cr>
```

هذا يؤدى إلى العثور على كل الخليط من رموز الحالة العليا والحالة السفلى لـ OCH، الموجود فى الخانات 2, 3, 4 من الحقل المسمى TOWN الآن.

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE    SALARY NOTE
      6  030     TIM MONTAL     ROCHESTER RBG .F. 07/07/81    41900.00 memo
      8  040     JAMES JAMESON  ROCHESTER GSD .T. 10/21/77    29800.00 memo
     15  075     JOHN JONES     rochester GSD .T. 04/04/70    25100.00 memo

        .LIST FOR 'TER' $(UPPER(TOWN))     <cr>
```

هذا يؤدى إلى العثور على كل الخليط من رموز الحالة العليا والحالة السفلى لـ T E R، الموجودة فى أى مكان فى الحقل المدينة.

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE    SALARY NOTE
      1  005     NINA BHARUCHA  WEBSTER    BSG .T. 05/24/80    25000.00 MEMO
      6  030     TIM MONTAL     ROCHESTER RBG .F. 07/07/81    41900.00 memo
      8  040     JAMES JAMESON  ROCHESTER GSD .T. 10/21/77    29800.00 memo
      9  045     MORRIS KATZ    webster    BSG .F. 09/14/80    23450.00 memo
     11  055     PHIL MARTIN    WEBSTER    RMG .F. 07/19/80    31000.00 memo
     15  075     JOHN JONES     rochester GSD .T. 04/04/70    25100.00 memo
```

لقد وجدت أنه من الأسهل تذكر هذه الدوال، إذا قرأت العنصر العلوى UPPER كما لو لم يكن هناك أى اعتبار للحالة. وعلى هذا.. فإننى أقرأ مثال الأمر، سالف الذكر على النحو التالى : اسرد كل العناصر التى لها TER فى أى مكان فى الحقل TOWN، بغض النظر عن الحالة للحروف. والكلمات الرئيسية هنا هى «فى أى مكان فى» anywhere-in، و «بغض النظر عن الحالة» regardless-of-case.

ملاحظة : طبيعة دالة ( ) UPPER تفرض أن القيم التى تقدمها يجب أن تقدم فى الحالة العليا UPPERcase فقط؛ أى إن :

. DISP FOR UPPER (TOWN) = 'Roch' <cr>

ينتج عنها عدم وجود شىء يحقق الشرط؛ فيجب أن تكون القيمة التى تقدمها فى الحالة العليا UPPERcase؛ يبحث dBASE عند ذلك عن كل الخليط من الحالات، كما هو محدد فى الأمر.

دالة ( ) UPPER لا يمكن تطبيقها على حقول مذكرات؛ أى إنه يجب أن تقدم السلسلة الدقيقة المطلوب البحث عنها فى حقل المذكرة، إذا رغبت فى استخدام دالتى ( ) SUBSTR و $) (مع محتويات حقول مذكرات).

دالة الحالة السفلى (الحروف الصغيرة) THE LOWER ( ) FUNCTION

الدالة مقدمة كنوع من التوازن مع دالة ( )UPPER.

```
.USE PERSNL
.LIST FOR LOWER(town) = 'rochester'
```

مع قراءة كل سجل من الملف.. ينفذ dBASE ترجمة داخلية لبيانات حقل TOWN إلى الحالة السفلى، ثم يجرى المقارنة مع rochester. ويشير اصطلاح الترجمة الداخلية internal translation أو الترجمة المنطقية logical translation إلى الترجمة أثناء تنفيذ الأمر فقط. أما البيانات الواقعية الموجودة فى الملف فلا تتغير.

تأثير ذلك هو العثور على كل الخليط من حروف r o c h e s t e r فى الحالة العليا والحالة السفلى الموجودة، فى حقل المدينة.

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE    SALARY NOTE
      6  030     TIM MONTAL      ROCHESTER RBG .F. 07/07/81    41900.00 memo
      8  040     JAMES JAMESON   ROCHESTER GSD .T. 10/21/77    29800.00 memo
     15  075     JOHN JONES      rochester GSD .T. 04/04/70    25100.00 memo
```

.LIST FOR SUBSTR(LOWER(TOWN),2,3) = 'och'

ويجد هذا الآن كل الخليط من رموز och فى الحالة العليا والحالة السفلى، الموجودة فى المواقع 2, 3, 4 من حقل المدينة.

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      6  030     TIM MONTAL     ROCHESTER RBG .F. 07/07/81   41900.00 memo
      8  040     JAMES JAMESON  ROCHESTER GSD .T. 10/21/77   29800.00 memo
     15  075     JOHN JONES     rochester GSD .T. 04/04/70   25100.00 memo

    DISP FOR 'ter' $(LOWER(TOWN))
```

ويجد هذا كل الخليط من رموز t e r فى الحالة العليا والحالة السفلى، الموجودة فى أى مكان فى حقل المدينة.

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      1  005     NINA BHARUCHA  WEBSTER    BSG .T. 05/24/80   25000.00 MEMO
      6  030     TIM MONTAL     ROCHESTER RBG .F. 07/07/81   41900.00 memo
      8  040     JAMES JAMESON  ROCHESTER GSD .T. 10/21/77   29800.00 memo
      9  045     MORRIS KATZ    webster    BSG .F. 09/14/80   23450.00 memo
     11  055     PHIL MARTIN    WEBSTER    RMG .F. 07/19/80   31000.00 memo
     15  075     JOHN JONES     rochester GSD .T. 04/04/70   25100.00 memo
```

ملاحظة ، تفرض طبيعة دالة ( ) LOWER أن القيمة التى تقدمها، يجب أن تقدم كلها فى الحالة السفلى LOWERcase فقط؛ أى إن :

. DISP FOR LOWER (TOWN) = 'ROCH' <cr>

لا تجد شيئاً ! فيجب أن تكون القيمة التى تقدمها فى الحالة السفلى LOWERcase! يبحث dBASE عند ذلك عن كل الخليط من الحالات، كما هو محدد فى الأمر.

لا يمكن تطبيق دالة LOWERcase على حقول مذكرات؛ أى إنك يجب أن تقدم السلسلة الدقيقة المراد البحث عنها فى حقل الذاكرة، إذا أردت استخدام دالتى ( ) SUBSTR و ( ) $ مع محتويات حقول مذكرات.

اقتراحات لإستخدام حقول المذكرات مع الدوال

SUGGESTIONS FOR MEMO-FIELDS USAGE WITH FUNCTIONS

فى رأيى الشخصى.. يمثل عدم امكانية تطبيق دالتى ( ) UPPER و ( ) LOWER على حقول مذكرات عيباً خطيراً؛ فإذا أردت أن أبحث عن overdue فى حقل مذكرة.. تعمل عبارة الأمر التالية :

. DISP FOR 'OVERDUE' $ (NOTE) <cr>

إذا كان المحتوى الفعلى فى حقل الذاكرة هو OVERDUE فقط؛ أى خليطاً آخر من الحالة العليا والحالة السفلى لايجد السجل. والأسوأ من ذلك.. فإن المحتوى الفعلى لكلمة overdue فى المذكرات يمكن أن يكون قد أعد بطريقة مختلفة فى السجلات المختلفة؛ جاعلاً البحث الكامل عن overdue قريباً من المستحيل. فكر فى عدد المرات التى تستطيع ادخال عبارة الأمر بها؛ لاستخلاص كل الحدوثات، مثل :overdue و OVERdue و oVERDUE و OverDue.. الخ. يتعقد التأثير إذا كنت تبحث عن عبارة بدلاً من بحثك عن كلمة.

وبسبب هذا القصور المهم.. فاننى أوصى بشدة أنك إذا لم تكن فى حاجة إلى مذكرات طويلة.. يجب أن تعرف الحقول التى تريد استخدامها كحقول مذكرات على أنها حقول حرفية، وذلك لبضع أسباب جيدة :

١ – تميل الحقول الحرفية بنفسها إلى الدقة الكاملة للحقول الجزئية؛ فيمكنك أن تستخدم كل صيغ دوال ( ) SUBSTR و ( ) $ و ( ) UPPER و ( ) LOWER مع الحقول الحرفية.

٢ – بعض الأوامر تجرى تشغيلاً أسرع على ملفاتك (إذا لم يشتمل الملف على حقول مذكرات) لأنه لن يتواجد ملف DBT. مناظراً للملف.

٣ – يمكن أن يصل طول كل حقل حرفى إلى 254 خانة، ويمكنك أن تعرف دائماً حقولاً تسمى COMMENT2 و COMMENT1... وهكذا. وكل منها له طول يصل إلى 254 خانة، وهذا يعطيك ميزة سعة المذكرة كلها التى تريدها، دون الحاجة إلى تعريف حقول مذكرات فى هيكل الملف.

دالة التاريخ (تاريخ النظام) THE DATE ( ) FUNCTION

تاريخ نظام التشغيل متاح للاتصال به عن طريق dBASE، وذلك باستخدام دالة DATE ( ). مثال ذلك يقدم الأمر التالى :

```
. ? DATE ( )    <cr>
```

تاريخ اليوم الذى تعمل فيه.

عند أى نقطة أثناء التشغيل.. إذا رغبت فى التحقق من صحة تاريخ النظام الموجود أو أردت تعديله، يمكنك أن تكتب الأمر التالى :

```
. RUN DATE    <cr>
```

إذا تذكرت.. فإن أمر RUN هو خطاف dBASE، الذى ينقلك إلى مستوى نظام التشغيل. ينفذ الأمر السابق أمر DATE من أوامر نظام التشغيل، كما لو كنت تكتب عند ملقن DOS، ويقدم لك التاريخ الحالى، كما هو مخزن فى النظام مع الخيار لتغييره.

اشكال شاملة لدوال التاريخ Generic Forms of DATE Function

فى الأمثلة التالية.. يفترض أن أزواج الأقواس فى الأشكال الشاملة تحتوى على الحقل أو التعبير الذى تؤثر عليه الدالة. ويتبع ذلك أمثلة حقيقية. إلا أن هذه ليست إلا أشكال شاملة لدوال التاريخ ومعانيها لهذه اللحظة.

MONTH ( )	تقدم الشهر مكتوباً فى خانتين
DAY ( )	تقدم اليوم مكتوباً فى خانتين
YEAR ( )	تقدم السنة مكتوبة فى خانتين
CMONTH ( )	تقدم اسم الشهر مثل January و February ... الخ.
DOW ( )	تقدم رقم اليوم فى الاسبوع، مثل: 1 ليوم الأحد، و 2 ليوم الاثنين... الخ.
CDOW ( )	تقدم اسم اليوم فى الاسبوع، مثل: Monday و Tuesday... الخ.

CTOD ('mm/dd/yy') تقرأ على النحو التالى : من رمز إلى تاريخ؛ أى إن dBASE يبحث عن حقل حرفى (يجب أن يكون على هيئة (mm/dd/yy) كحقل تاريخ، ومثال ذلك لاظهار السلسلة الحرفية 06/01/88 كتاريخ.. يمكنك أن تدخل مايلى : CTOD ('06/01/88').

{mm/dd/yy} هذه الدالة متطابقة مع دالة ( ) CTOD، وقد تبدو أسهل فى إدخالها. ومثال ذلك لإظهار رموز السلسلة 06/01/88 كتاريخ.. يمكنك أن تدخل ما يلى : {06/01/88}

DTOC () تقرأ على النحو التالى : من تاريخ إلى رمز، وأحياناً تريد من dBASE أن يبحث عن حقل تاريخ لحقل حرفى.

تأتى توضيحات أكثر فيما يلى عن الأشكال الشاملة.

أنت تعرف أن تاريخ النظام (تاريخ اليوم لليوم الذى تعمل به مع النظام) مخزن على هيئة دالة، وهى ( ) DATE.

. ? MONTH (DATE ( ))	تقدم رقمين دالين على الشهر من تاريخ النظام
. ? DAY (DATE ( ))	تقدم رقمين دالين على اليوم من تاريخ النظام
. ? YEAR (DATE ( ))	تقدم أربعة أرقام دالة على السنة من تاريخ النظام
. USE PERSNL	تحضر السجل رقم 1 للاستخدام
. ? DT - OF - HIRE	تعرض تاريخ التعيين على الصورة 05/24/80
. ? MONTH (DT - OF - HIRE)	نتيجتها تأخذ الشكل التالى : 5
. ? DAY (DT - OF - HIRE)	نتيجتها تأخذ الشكل التالى : 24
. ? YEAR (DT - OF - HIRE)	نتيجتها تأخذ الشكل التالى : 1980
. ? DT - OF - HIRE + 35	ينتج عنها تاريخ جديد، وهو التاريخ الذى يأتى بعد 35 يوماً بالضبط من تاريخ التعيين.

. ? DATE ( ) - DT-OF-HIRE > 365 * 5 <cr>

السؤال المطروح هو : هل الفرق بين تاريخ اليوم وتاريخ التعيين أكبر من 5 سنين؟ أى إنه هل تم تعيين هذا الشخص لأكثر من 5 سنين؟ يجيب dBASE فى هذه الحالة بالاجابة T أى نعم.

. ? CMONTH (DT-OF-HIRE) ينتج عنها اسم الشهر May

. ? DOW (DT-OF-HIRE) ينتج عنها رقم 7، لأن اليوم هو يوم سبت

. ? CDOW (DT-OF-HIRE) ينتج عنها اسم اليوم، وهو Saturday

دالة من رمز إلى تاريخ CTOD تقدم رأياً مفيداً جداً، عندما تريد أن تقدم تاريخاً فى أى عبارة؛ لأنه يجب ادخال التاريخ كثابت فى الصورة mm/dd/yy.

كمثال... إذا أردت أن تجد الفرق بالأيام بين تاريخ تعيين العامل، والتاريخ 06/01/79.. فإنك تكتب:

. ? DT - OF - HIRE - CTOD ('06/01/79') <cr>

توضيح : يمكنك أن تطرح حقل تاريخ من حقل تاريخ آخر فقط. وعلى هذا.. يجب أن تحول السلسلة الحرفية 06/01/79 إلى حقل تاريخ؛ مستخدماً دالة من رمز إلى تاريخ.

ويمكن كذلك ادخال العبارة السابقة على النحو التالى :

. ? DT - OF - HIRE - {06/01/79}

لاحظ استخدام الأقواس { } الحالية؛ لتعريف دالة من رمز إلى تاريخ. ولاحظ كذلك أن علامتى التنصيص لا يتم ادخالهما، ويعرف المحتوى {06/01/79} متغير تاريخ.

ويجب أن يطرأ السؤال التالى على الذهن : نظراً لأننى ليس لدى طريقة لمعرفة ما إذا كان DT - OF - HIRE أكبر من أو أقل من التاريخ 06/01/79.. فمن الممكن أن تكون نتيجة عملية الطرح السابقة عدداً سالباً. إلا أن القيمة المطلقة (القيمة بدون أى اعتبار للاشارة الجبرية) وهى التى تهمنى، تكون دقيقة.

تذكر أنه فى أى عبارة من عبارات dBASE.. إذا أردت أن تقدم تاريخاً، فعليك أن تقدم ثابتاً فى الصورة mm/dd/yy، وإما أن تستخدم دالة CTOD على هذا الثابت كما فى الحالة : CTOD ('MM/DD/YY')، أو تستخدم دالة { } كما فى الحالة {MM/DD/YY}.

أمثلة للسرد تشمل حقول تاريخ

Examples of LISTS, Involving "DATE" Fields

١ – اسرد كل العاملين الذين عينوا عام 1980.

```
.LIST FOR YEAR(DT_OF_HIRE) = 1980     <cr>

Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      1  005     NINA BHARUCHA  WEBSTER    BSG .T. 05/24/80   25000.00 MEMO
      9  045     MORRIS KATZ    webster    BSG .F. 09/14/80   23450.00 memo
     11  055     PHIL MARTIN    WEBSTER    RMG .F. 07/19/80   31000.00 memo
```

٢ – اسرد كل العاملين الذين عينوا فى شهر ابريل؛ بغض النظر عن سنة التعيين.

```
.LIST FOR MONTH(DT_OF_HIRE) = 4     <cr>

Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      4  020     MAX LEVINSKY   HENRIETTA  RMG .F. 04/13/69   27550.00 MEMO
      5  025     KIM BRANDT     FAIRPORT   RMG .F. 04/04/77   36000.00 MEMO
     12  060     JOHN PETERSON  BRIGHTON   RBG .T. 04/17/79   31480.00 MEMO
     14  070     JAN MOREY      ROXY       GSD .T. 04/23/67   18190.00 memo
     15  075     JOHN JONES     rochester  GSD .T. 04/04/70   25100.00 memo
```

٣ – اسرد كل العاملين الذين عينوا فى اليوم الرابع من أى شهر، بغض النظر عن سنة التعيين.

```
.LIST FOR DAY(DT_OF_HIRE) = 4     <cr>

Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      5  025     KIM BRANDT     FAIRPORT   RMG .F. 04/04/77   36000.00 MEMO
     15  075     JOHN JONES     rochester  GSD .T. 04/04/70   25100.00 memo
```

٤ – اسرد كل العاملين الذين عينوا يوم الاثنين؛ بغض النظر عن تاريخ وسنة التعيين.

```
.LIST FOR DOW(DT_OF_HIRE) = 2     <cr>

Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      5  025     KIM BRANDT     FAIRPORT   RMG .F. 04/04/77   36000.00 MEMO
```

٥ – اسرد مرة أخرى كل العاملين الذين عيينوا يوم الاثنين، بغض النظر عن تاريخ وسنة التعيين.

```
LIST FOR CDOW(DT_OF_HIRE) = 'Mon'     <cr>

Record#  EMP_NUM EMP_NAME      TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      5  025     KIM BRANDT    FAIRPORT   RMG .F. 04/04/77   36000.00 MEMO
```

لاحظ أن المخرجات من dBASE دائماً ما تكون فى الصورة المبينة أعلاه، ويكون الحرف الأول فقط مكتوباً فى الحالة العليا؛ أى إنه حرف كبير.

٦ – اسرد مرة أخرى كل العاملين الذين عيينوا يوم الاثنين، بغض النظر عن تاريخ وسنة التعيين.

```
LIST FOR UPPER(CDOW(DT_OF_HIRE)) = 'MON'     <cr>

Record#  EMP_NUM EMP_NAME      TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      5  025     KIM BRANDT    FAIRPORT   RMG .F. 04/04/77   36000.00 MEMO
```

نظراً لأننى استخدمت دالة UPPER.. فإننى قدمت القيمة بالحروف الكبيرة كذلك.

٧ – اسرد كل العاملين الذين يعملون فعلاً منذ أكثر من 3 أشهر، ولكنهم لم يكملوا 6 أشهر من العمل (معتبراً أن عدد أيام الشهر هى 30 فى المتوسط).

```
LIST FOR DATE() - DT_OF_HIRE >= 90  AND  DATE() - DT_OF_HIRE <= 180
```

(لايوجد أى شخص فى ملف PERSNL يحقق هذا الشرط).

٨ – اسرد كل العاملين الذين يعملون فعلاً منذ خمس سنوات، من تاريخ اليوم.

```
LIST FOR DATE() - DT_OF_HIRE  >=  365 * 5     <cr>
```

(كل العاملين فى ملف PERSNL يحققون هذا الشرط).

٩ – اسرد كل العاملين الذين يتمون خمس سنوات خدمة أو أكثر، بعد انقضاء 20 يوماً من تاريخ اليوم.

```
LIST FOR (DATE( )+20) - DT_OF_HIRE  >=  365 * 5     <cr>
```

(كل العاملين فى ملف PERSNL يحققون هذا الشرط).

١٠ - اسرد كل العاملين فى المؤسسة BSG المعينين بعد 01 يونيو عام 1970.

```
LIST FOR ORG = 'BSG'  .AND.  DT_OF_HIRE  >  CTOD('06/01/70')     <cr>

Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      1  005     NINA BHARUCHA  WEBSTER    BSG .T. 05/24/80   25000.00 MEMO
      2  010     PETE JOHNSON   brighton   BSG .T. 02/03/76   27590.00 MEMO
      9  045     MORRIS KATZ    webster    BSG .F. 09/14/80   20450.00 memo
     10  050     PAUL BHARUCHA  BRIGHTON   BSG .T. 05/23/73   29100.00 MEMO
```

يمكن كذلك ادخال العبارة السابقة على النحو التالى، مع استخدام أقواس { } لتعريف التاريخ :

```
.LIST FOR ORG = 'BSG'  .AND.  DT_OF_HIRE  >  {06/01/70}     <cr>
```

لتحصل على نفس النتيجة مثل النتيجة السابقة.

الصعوبة مع التواريخ الفارغة Difficulty with Blank Dates

يكون لدى dBASE صعوبة واضحة فى مقارنة تواريخ حقيقية مع تواريخ فارغة؛ انظر الأمثلة التالية :

```
.STORE  CTOD('  /  /  ')  TO MBLANK
.STORE  {}                 TO MBLANK
```

أو

أى من الأمرين السابقين يخزن تاريخاً فارغاً فى متغير مذكرة (مساحة الذاكرة أعدت بمفردها) اسمه KNALBM. (تناقش متغيرات المذكرة بتفصيل أكبر فى قسم البرمجة من هذا الكتاب).

```
.STORE  CTOD('01/01/86')  TO MDATE
.STORE  {01/01/86}        TO MDATE
```

أو

أى من الأمرين السابقين يخزن التاريخ 01/01/86 فى متغير مذكرة (مساحة الذاكرة أعدت بمفردها) اسمه MDATE.

لاحظ الآن نتائج المقارنات التالية :

MBLANK < MDATE ? . وهى تاريخ فارغ أقل من تاريخ حقيقى. النتيجة .F .

MDATE < MBLANK ? . وهى تاريخ حقيقى أقل من تاريخ فارغ. النتيجة .F .

MDATE > MBLANK ? . وهى تاريخ حقيقى أكبر من تاريخ فارغ. النتيجة .F .

النتيجة كانت .F . فى كل الحالات وهذا ليس واضحاً!

لاحظ المشكلة التى تحدث لنظام dBASE عند تعامله مع تاريخ فارغ. وفى عديد من المواقف (وبصفة خاصة فى بيئة البرمجة) يجب أن يكون dBASE قادراً على تمييز التاريخ الفارغ (أو الأقل) بطريقة صحيحة. ويمكن علاج اللبس السابق بتقديم تاريخ أقل من أى تاريخ آخر فى التطبيق المقترح.

```
.STORE CTOD('01/01/01') TO MBLANK
.STORE {01/01/01}       TO MBLANK
```

أى من الأمرين السابقين يخزن التاريخ 01/01/1901 فى متغير ذاكرة (مساحة الذاكرة أعدت بمفردها) اسمه MBLANK. وتصبح المقارنات دقيقة الآن! باستخدام العبارات السابقة.. تكون قيمة القرن هى القيمة التقليدية 19.

اعمل ملاحظة مهمة جداً بهذا اللبس!! فيجب عليك - كمبرمج - أن تقدم كل التواريخ الفارغة على الصورة ('01/01/01') CTOD، أو على الصورة {01/01/01} داخل منطق برامجك، إذا كان القرن التقليدى 19 كافياً.

إلا أنه فى بعض الحالات.. قد تحتاج أن تقلل التاريخ عن ذلك أيضاً. ومثال ذلك إذا كان تطبيقك يتعامل مع بيانات تاريخية أو قرون سابقة.. فإنه يجب أن تنتج تاريخاً أقل على النحو التالى :

```
.STORE CTOD('01/01/1701') TO MBLANK
STORE {01/01/1701}       TO MBLANK
```

والافتراض الآن هو أنك لا تدخل أى تاريخ أقل من 01/01/1701 فى تطبيقك.

ويمكنك دائماً – إذا كانت هناك حاجة لذلك – أن تستخدم أياً من الدالتين CTOD ('01/01/0100') أو {01/01/0100} كأقل تاريخ يمكن أن يتعامل معه dBASE دون حدوث أى لبس.

دالة الوقت (وقت النظام) THE TIME ( ) FUNCTION

تقدم الدالة ( ) TIME ? . وقت النظام فى الصورة hh:mm:ss. إذا رغبت فى تغيير وقت النظام الموجود به حالياً لأى سبب من الأسباب.. يمكنك إدخال أمر RUN المعتاد على الصورة التالية :

.RUN TIME <cr>

يقدم لك الوقت الحالى كما هو مخزن فى النظام، مع امكانية تغييرك له.

يمكن أن تجد هذه الدالة استخداماً طيباً فى حالة البرمجة من حالات dBASE: حيث يمكن إعداد الشاشة مع ظهور الوقت فى أحد أركانها، كما يمكنك كذلك إجراء اختبارات على أوامر محددة؛ لترى طول الوقت اللازم لإتمام تنفيذها، مع قواعد بيانات لها أحجام مختلفة.

دالة السلسلة THE STR ( ) FUNCTION

عادة ما تعتبر القيمة المقدمة لحقل عددى هى القيمة الجبرية، إلا أنه من الممكن جعل dBASE يرى الحقل العددى كحقل حرفى؛ بحيث يمكنك أن تأخذ سلاسل جزئية من هذا الحقل العددى.

التعبير : STR (SALARY, 8, 2) يجعل dBASE ينظر إلى حقل SALARY على أنه حقل حرفى الآن.

وبصفة عامة.. لجعل dBASE ينظر إلى حقل عددى على أنه حقل حرفى، يجب أن تحدد:

.STR (field-name, field-width)

وكأنك تقول «اعمل الحقل العددى على هيئة سلسلة مثل الحقل الحرفى».

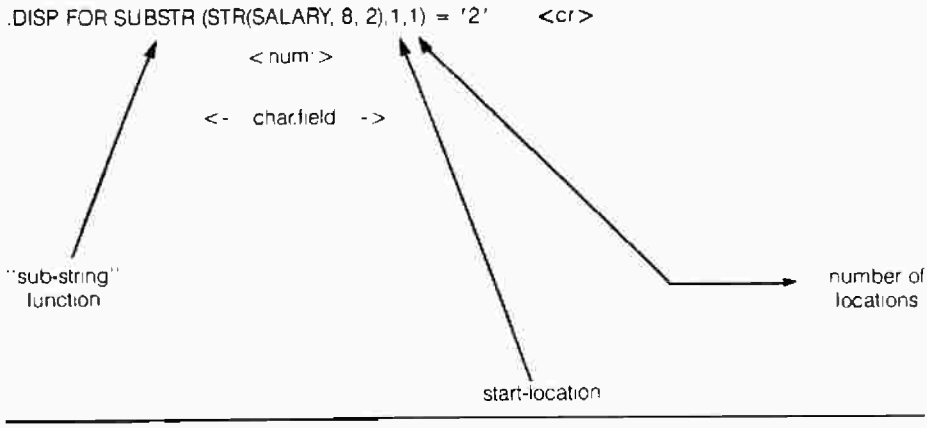
وإذا أردت الآن الحصول على قائمة بكل العاملين اللذين لهم الرمز 2، فى أول موقع من مواقع الحقل SALARY.. يمكنك أن تحدد ما يلى (انظر شكل 2-10 لتوضيح ذلك) :

.DISP FOR SUBSTR (STR (SALARY, 8, 2), 1,1) = '2' <cr>

(يقرأ هذا على النحو التالى : «اعمل سلسلة جزئية للحقل الحرفى المسمى SALARY، بدءًا من أول موقع لموقع واحد. فإذا كان هذا مساوياً القيمة '2' فاعرض السجل.»)

Record#	EMP_NUM	EMP_NAME	TOWN	ORG	EXE	DT_OF_HIRE	SALARY	NOTE
1	005	NINA BHARUCHA	WEBSTER	BSG	.T.	05/24/80	25000.00	MEMO
2	010	PETE JOHNSON	brighton	BSG	.T.	02/03/76	27590.00	MEMO
3	015	GLORIA PATEL	FAIRPORT	RMG	.T.	07/16/82	27500.00	memo

etcetera.



شكل 2-10 : توضيح مثال استخدام دالة السلسلة الجزئية.

فى المثال السابق.. وضعت القيمة المقدمة بين علامتى تنصيص؛ لأنك تشير الآن إلى حقل حرفى.

تتحقق بالفعل أنك إذا أردت قائمة العاملين السابقة الموجود لديهم '2' فى أول موقع من مواقع الحقل SALARY، أى كل الذين لهم رواتب مساوية أو أكبر من 20000 وأقل من 30000.. يمكنك أن تفعل ذلك على النحو التالى :

.DISP FOR SALARY > = 20000 .AND. SALARY < 30000 <cr>

إلا أن الغرض هنا هو توضيح دالة السلسلة الجزئية على حقل عددى محول إلى حقل حرفى. (وهذا ليس مزعجاً كما يبدو من الاسم - جربها فقط!)

بجانب ذلك.. هناك أسباب عديدة لحاجتك اعتبار حقولاً عددية كحقول حرفية، وسوف يوضح ذلك فى المواقع المناسبة من هذا الكتاب. وللحظة الحالية... دعنى أوضح أن سمات معينة مثل الفهرسة Indexing والتسلسل Concatination) تتطلب تحويل الحقول العددية إلى حقول حرفية (أو مايكافئها من الحقول الحرفية.)

تحديد وضع الفلتر (المرشح) SET FILTER TO

عند هذه النقطة.. تكون قد غطيت كما من أساس استخدام شروط FOR وشروط WHILE لاستخلاص وعرض بيانات. وبقليل من التدريب.. يجب أن تصبح قادراً على استخلاص سجلات من قاعدة بيانات جيدة ونظيفة، بعديد من الطرق.

أريد أن أذكر سمة مفيدة أخرى لاختيار سجلات من قاعدة بيانات، يمكن أن توفر كثيراً من مجهود الكتابة. قل إنه فى إحدى جلسات العمل مع dBASE.. أردت أن تعمل مع مجموعة معينة من السجلات فقط فى هذه الجلسة. إذا حققت مجموعة السجلات الشرط :

"ORG = 'BSG' .AND. SALARY > 19999.99 .AND. SALARY < 30000.01

فإنك تريد أن تحدد شرط FOR أطول كما هو مبين هنا، وذلك فى كل عبارة من العبارات التى تريد كتابتها. إذا أردت تأهيل السجلات داخل هذه المجموعة بشرط آخر من شروط FOR.. تصبح عبارة الأمر أكبر من ذلك كثيراً.

يقدم dBASE أمر SET FILTER؛ حيث يمكنك أن تحدد شرط فلتر للملف مرة واحدة، وبعد ذلك يعمل كل أمر على السجلات التى تحقق شرط الفلتر فقط، مثل :

```
USE PERSNL
SET FILTER TO ORG = 'BSG'  AND  SALARY > 19999.99
                           .AND  SALARY < 30000.01
GO TOP
```

من الآن.. يظهر الملف على أنه يحتوى على السجلات التى تحقق الشرط فقط، وهذا يوفر لك كثيراً من مجهود تكرار شرط FOR لكل عبارة تكتبها.

ملاحظة : إذا كنت تعمل مع أكثر من ملف واحد فى نفس الوقت (الشيء الموضح فى المواقع المناسبة من الكتاب).. يمكنك أن تحدد فلاتر مختلفة للملفات المختلفة.

وتكون الفلاتر مؤقتة فقط... فإذا أحضرت ملفاً آخراً للاستخدام (فى نفس منطقة العمل).. فإن الفلتر الأصلى الذى كان موجوداً للملف السابق يلغى. فإذا حددت فلتراً لشرط آخر فيسرى الشرط الجديد، وليس الشرط السابق.

ولا تنشط الفلاتر إلا إذا نقل مشير السجل، وعلى هذا... فأنصحك بكتابة عبارة GO TOP بعد عبارة SET FILTER مباشرة.

لايقاف نشاط شرط الفلتر.. عليك بإدخال الأمر : SET FILTER TO <cr> لمنطقة العمل المناسبة، وهذه العبارة بدون أى شرط توقف نشاط الفلتر الحالى.

يمكنك أن تنقل مشير السجل لأى أى سجل محدد آخر، حتى إذا كان هذا السجل لا يحقق شرط الفلتر.

GO 10. تنقل المشير إلى السجل رقم 10، حتى إذا كان هذا السجل لا يحقق شرط الفلتر التالى.

لاحظ ما يلى :

```
.1    <cr>
.DISP NEXT 5 EMP_NUM  EMP_NAME  TOWN

Record#  EMP_NUM EMP_NAME       TOWN
      1  005     NINA BHARUCHA  WEBSTER
      2  010     PETE JOHNSON   brighton
      3  015     GLORIA PATEL   FAIRPORT
      4  020     MAX LEVINSKY   HENRIETTA
      5  025     KIM BRANDT     FAIRPORT
```

هذه هى المخرجات المعتادة، مع وجود عمود فارغ بين أعمدة البيانات المعروضة.

حاول الآن عمل ما يلى

```
.1    <cr>
.DISP NEXT 5  EMP_NUM + EMP_NAME + TOWN     <cr>
```

قد يساعد إذا ترجمت اشارة الموجب السابقة على النحو التالى : EMP - NUM، يليه على الفور EMP - NAME، يليه على الفور TOWN... الخ. ويعنى ذلك عدم وجود أى فراغات بين الحقول، ويحذف الآن العمود الواحد الفارغ بين الأعمدة، وهذا هو التسلسل con-catination للحقول (وهو وصل السلاسل ببعضها واقعياً)!

```
Record#  EMP_NUM                              <---- NOTE THE HEADING !!
      1  005NINA BHARUCHAWEBSTER
      2  010PETE JOHNSON brighton
      3  015GLORIA PATEL FAIRPORT
      4  020MAX LEVINSKY HENRIETTA
      5  025KIM BRANDT   FAIRPORT
```

فى عبارة الأمر السابقة.. قمت بإدخال فراغات قبل وبعد اشارة +، وذلك لتحسين القراءة ببساطة. إلا أنه بينما يكون سرد البيانات صحيحاً.. يبدو أن العنوان الموجود فوق البيانات يهمل كل شيء بعد أول موقع.

إذا أدخلت عبارة الأمر على النحو التالى :

```
.1    <cr>
.DISP NEXT 5  EMP_NUM + EMP_NAME + TOWN     <cr>
```

يتم الحصول على نفس السرد، ويعكس عنوان المخرجات عبارة الأمر بدقة.

```
Record#   EMP_NUM+EMP_NAME+TOWN                <---- NOTE THE HEADING, NOW !!
      1   005NINA BHARUCHAWEBSTER
      2   010PETE JOHNSON brighton
      3   015GLORIA PATEL FAIRPORT
      4   020MAX LEVINSKY HENRIETTA
      5   025KIM BRANDT   FAIRPORT
```

هذا بالطبع خطأ بسيط في عرض العناوين.

ملاحظة : تطبق دالة + على حقول حرفية فقط، وعلى هذا إذا أردت استخدام نفس البديل مع حقول عددية.. فعليك أن تحدد الحقول العددية بأنها دوال سلاسل STRings، وإذا أردت استخدام دالة + مع حقول تواريخ.. فعليك أن تحدد حقول التواريخ بأنها سلاسل حرفية؛ مستخدماً في ذلك دالة من تاريخ إلى رمز DTOC.

```
.1    <cr>
LIST NEXT 5 TOWN + STR(SALARY,8,2)    <cr>

Record#   TOWN+STR(SALARY,8,2)
      1   WEBSTER  25000.00
      2   brighton 27590.00
      3   FAIRPORT 27500.00
      4   HENRIETTA27550.00
      5   FAIRPORT 36000.00

1    <cr>
LIST NEXT 5 EMP_NAME + ORG + DTOC(DT_OF_HIRE)    <cr>

Record#   EMP_NAME+ORG+DTOC(DT_OF_HIRE)
      1   NINA BHARUCHABSG05/24/80
      2   PETE JOHNSON BSG02/03/76
      3   GLORIA PATEL RMG07/16/82
      4   MAX LEVINSKY RMG04/13/69
      5   KIM BRANDT   RMG04/04/77
```

يمكنك أن تستخدم ثوابت كذلك في مؤشراتك، وذلك للحصول على أشكال خاصة للمخرجات.

```
1    <cr>
DISP NEXT 5 EMP_NUM + ' ' + EMP_NAME + ' ' + TOWN    <cr>

Record#   EMP_NUM+'   '+EMP_NAME+'   '+TOWN
      1   005    NINA BHARUCHA    WEBSTER
      2   010    PETE JOHNSON     brighton
      3   015    GLORIA PATEL     FAIRPORT
      4   020    MAX LEVINSKY     HENRIETTA
      5   025    KIM BRANDT       FAIRPORT
```

كمثال آخر :

```
.8
.? 'Employee ' + EMP_NAME + ' lives in ' + TOWN + ', and earns
  $' + STR(SALARY,8,2) + '.'   (complete statement in one line...)
```

يمكن أن تفسر "?" بـ «ماذا إذا؟» أو بـ «اسرد». ويجعل الأمران السابقان السجل رقم 8 هو السجل الحالى، ثم يضعان عبارة فى السطر التالى، على الشاشة، على النحو التالى:

```
Employee JAMES JAMESON lives in ROCHESTER, and earns $29800.00.
```

لاحظ كيفية تسلسل هذه الثوابت والحقول مع بعضها البعض! دالة التسلسل « + » تقدم مرونة كبيرة فى الحصول على مخرجات مشكلة، ويمكن أن يكون لديك أى نوع من أنواع الثوابت موضوعاً بين علامتى تنصيص، بما فى ذلك الفواصل، والعلامات العشرية، وعلامات الدولار $.

يجب ملاحظة أنه بسبب أن محتويات الحقل EMP-NAME (وهى JAMES "JAMESON") ومحتويات حقل TOWN (وهى "ROCHESTER") تحتل مواقعها فى حقولها، ومخرجات العبارة السابقة تخرج نظيفة. إلا أنه إذا كانت محتويات هذه الحقول لا تحتل الطول الكلى من هيكل الحقل.. تحتوى المخرجات على فراغات إضافية بين الكلمتين، ويمكن أن تنتهى – على سبيل المثال – على النحو التالى :

```
Employee JAMES JAMESON     lives in ROCHESTER          , and earns $29800.00.
```

وبعد القليل من دراستك فى هذا الكتاب.. فإنك تدرس دالة أخرى (اسمها دالة التهذيب TRIM)، التى يمكنها عمل أى تشكيل للمخرجات وتنتهى بما أن تريد أن تراه بالضبط، بغض النظر عن محتويات البيانات الموجودة داخل الحقل.

دوال "+" و "_" و TRIM، والتالى توضيحهما، مفيدة جداً فى مرحلة البرمجة لإنتاج مخرجات مشكلة وتقارير مشكلة، مثل عناوين المراسلة. وتستخدم دالة "+" كذلك فى الفهرسة Indexing، والموضحة بالتفصيل فيما بعد فى هذا الكتاب.

```
.1    <cr>
.DISP NEXT 5  EMP_NUM – EMP_NAME – TOWN    <cr>
```

يحذف هذا الفراغات اللاحقة بالحقول EMP-NUM و EMP-NAME منها. وقد يساعد أنك تفسر إشارة "-" السابقة على النحو التالى : EMP-NUM ناقص كل الفراغات اللاحقة، يتبعها EMP-NAME ناقص كل الفراغات اللاحقة، يتبعها TOWN و... هكذا، وهذا يعنى تسلسلاً كاملاً لبيانات الحقول المسماة.

لاحظ الاختلاف فى المخرجات بين الأمرين التاليين :

```
.1    <cr>
.LIST NEXT 5 EMP_NUM + EMP_NAME + TOWN    <cr>

Record#  EMP_NUM+EMP_NAME+TOWN
      1  005NINA BHARUCHAWEBSTER
      2  010PETE JOHNSON brighton   (The + function results in a
      3  015GLORIA PATEL FAIRPORT   concatenation of fields.)
      4  020MAX LEVINSKY HENRIETTA
      5  025KIM BRANDT   FAIRPORT

.1    <cr>
.LIST NEXT 5 EMP_NUM - EMP_NAME - TOWN    <cr>

Record#  EMP_NUM-EMP_NAME-TOWN
      1  005NINA BHARUCHAWEBSTER
      2  010PETE JOHNSONbrighton  (The - function results in a
      3  015GLORIA PATELFAIRPORT     concatenation of the data in
      4  020MAX LEVINSKYHENRIETTA    the fields.)
      5  025KIM BRANDTFAIRPORT
```

ملاحظة : تطبق دالة – على الحقول الحرفية فقط، فإذا أردت استخدام نفس البديل مع حقول عددية.. يجب أن تحدد الحقول العددية كسوال سلاسل STRing.

مثال:

```
.1    <cr>
.DISP NEXT 5  ORG – STR(SALARY,8,2)    <cr>

Record#  ORG-STR(SALARY,8,2)
      1  BSG25000.00
      2  BSG27590.00
      3  RMG27500.00
      4  RMG27550.00
      5  RMG36000.00
```

تذكر أن : دالة + ينتج عنها تسلسل للحقول، بينما ينتج عن دالة – تسلسل للبيانات الموجودة داخل الحقول.

THE TRIM ( ) FUNCTION — دالة التهذيب

```
.1     <cr>
.DISP NEXT 5     TOWN,ORG     <cr>
```

تنتج عن ذلك المخرجات المعتادة :

```
Record#   TOWN        ORG
      1   WEBSTER     BSG
      2   brighton    BSG
      3   FAIRPORT    RMG
      4   HENRIETTA   RMG
      5   FAIRPORT    RMG
```

لكن لاحظ الأوامر التالية :

```
.1     <cr>
.DISP NEXT 5     TRIM(TOWN) + ORG     <cr>
```

تقدم المخرجات :

```
Record#   TRIM(TOWN)+ORG
      1   WEBSTERBSG
      2   brightonBSG
      3   FAIRPORTRMG
      4   HENRIETTARMG
      5   FAIRPORTRMG
```

تهذب دالة TRIM من حقل TOWN بالنسبة لكل الفراغات اللاحقة به (أى تزيلها)، ويتم توصيل ORG للنتيجة. وعلى هذا.. تحصل على المخرجات السابقة، وهذا متطابق مع دالة – التى سبق رؤيتها، إلا أن هذه الخاصية لدالة TRIM مفيدة جداً، كما ترى ذلك فيما يلى :

```
.1     <cr>
.LIST NEXT 5 TRIM(TOWN) + ' ' + ORG     <cr>

Record#   TRIM(TOWN)+' '+ORG
      1   WEBSTER BSG
      2   brighton BSG
      3   FAIRPORT RMG
      4   HENRIETTA RMG
      5   FAIRPORT RMG
```

في الأمر السابق.. طلبت تهذيب TRIM حقل المدينة TOWN، وهذا يحذف كل الفراغات اللاحقة من الحقل (أو الحقول)، وقمت بتوصيل (سلسلة) فراغ واحد يتبعه حقل ORG.

الصيغة السابقة للدالة مفيدة إذا كانت البيانات في الحقل التالي مطلوب تحريكها "slide over" إلى الحقل الذي أجرى عليه التهذيب "trimmed". مثال ذلك.. في هيكل الملف الذي أعددته، قد يكون هناك حقل اسمه TITLE, C, 12، والذي يمكن أن يحتوي على بيانات استهلالية مثل : "Mr." أو "Mr. and Mrs" أو "Mrs."... الخ. في هذه الحالة.. بالرغم من أن طول الحقل TITLE يكون ثابتاً 12 خانة، إلا أن البيانات الموجودة فيه يمكن أن يتغير طولها، طبقاً للاستهلال. إن الاسم الذي يلي الاستهلال مطلوب للتحريك "slide over" المجاور تماماً للاستهلال، بغض النظر عن طول الاستهلال الفعلي، وتساعدنا دالة TRIM في هذا.

مثال ذلك :

```
.DISP ALL TRIM (TITLE) +' '+TRIM (FIRSTNAME) +' '+ LASTNAME
```

لاحظ أنه بالرغم من أنه حدث تهذيب لحقل FIRSTNAME.. إلا أن بيانات حقل LASTNAME يمكن أن تأتي مجاورة تماماً لبيانات الاسم الأول.

كمثال آخر.. لوضع المدينة City، والولاية State، والرقم البريدي Zip code في سطر واحد – بغض النظر عن البيانات الفعلية للمدينة – يمكنك أن تكتب ما يلي :

```
.DISP TRIM (CITY) + ' , ' + STATE + ' ' + ZIP + ' . '    (cr)
```

يمكن أن تترجم العبارة السابقة – على سبيل المثال – إلى ما يلي :

```
ROCHESTER, NY 14620.
```

كما يمكنك أن تقدر.. فإن دالة TRIM مفيدة عملياً في تطبيقات إعداد عناوين المراسلة بصفة خاصة، وتستخدم هذه الدالة – بصفة عامة – مع دالة + في انتاج مخرجات مشكلة تشكيلاً محدداً.

ونعود إلى مثال سابق للمخرجات المشكلة :

```
.8
.? 'Employee ' + EMP_NAME + ' lives in ' + TOWN + ', and earns $ ' + STR(SALARY,8,2) + '.'
```

إذا تذكرت.. فقد حددت أن مخرجات هذا الأمر يمكن أن تنتهي أو لا تنتهي نظيفة؛ طبقاً لما إذا كانت البيانات الفعلية التي تشغل طول الحقل مشمولة. والآن.. عندي وسيلة للتأكد من أن المخرجات تكون جيدة، بغض النظر عن البيانات الموجودة في الحقول.

```
.8
.? 'Employee ' + TRIM(EMP_NAME) + ' lives in ' + TRIM(TOWN) + ', and
   earns $' + STR(SALARY,8,2) + '.'   (complete statement in one line...)

Record#  'Employee                                <---- NOTE THE HEADING !!
     8  Employee JAMES JAMESON lives in ROCHESTER, and earns $29800.00.
```

لاحظ شمول دوال TRIM لحقلي EMP-NAME و EMP-NUM. وتظهر المخرجات المشكلة الآن نظيفة، بغض النظر عن حجم البيانات الفعلية الموجودة في الحقول المهذبة. ونظراً لأن دالة TRIM تحذف الفراغات اللاحقة.. فعليك بملاحظة استخدامي فراغاً بعد TRIM (EMP-NAME)، وآخر بعد TRIM (TOWN).

وكمثال آخر :

```
4    <cr>
LIST NEXT 5  'Employee ' + TRIM(EMP_NAME) + ' lives in'
             + TRIM(TOWN) + ', and earns $' + STR(SALARY,8,2) + '.'   (in one line.)

Record#  'Employee                                <---- NOTE THE HEADING !!

     4  Employee MAX LEVINSKY lives in HENRIETTA, and earns $27550.00.
     5  Employee KIM BRANDT lives in FAIRPORT, and earns $36000.00.
     6  Employee TIM MONTAL lives in ROCHESTER, and earns $41900.00.
     7  Employee WILLIAM PATEL lives in penfield, and earns $28900.00.
     8  Employee JAMES JAMESON lives in ROCHESTER, and earns $29800.00.
```

مثال آخر :

```
1     <cr>
.LIST NEXT 5 'Employee ' + TRIM(EMP_NAME) + ' is in ' + TRIM(ORG) + ' as of
            ' + DTOC(DT_OF_HIRE) + '.'     <cr>

Record#   'Employee                              <---- NOTE THE HEADING !!
      1   Employee NINA BHARUCHA is in BSG as of 05/24/80.
      2   Employee PETE JOHNSON is in BSG as of 02/03/76.
      3   Employee GLORIA PATEL is in RMG as of 07/16/82.
      4   Employee MAX LEVINSKY is in RMG as of 04/13/69.
      5   Employee KIM BRANDT is in RMG as of 04/04/77.
```

تطبق دالة TRIM على حقول حرفية فقط. إلا أنك إذا أردت تطبيقها على حقول عددية، فما عليك إلا أن تحول الحقول العددية إلى حقول حرفية، وعند ذلك تعمل دالة TRIM بصورة طبيعية.

يمكن استخدام دالة ( ) TRIM كخليط مع دالة ( ) LEN فى إيجاد طول البيانات الفعلية الموجودة فى حقل حرفى. مثال ذلك :

```
USE PERSNL      <cr>
? LEN(TRIM(TOWN))      <cr>
```

ويصبح السؤال الذى يسأل : ما هو (أعرض) طول الصيغة المهذبة من الحقل TOWN؟ ويقدم هذا القيمة 7، لأن أول مدينة هى Webster.

THE RTRIM ( ) ("right trim") FUNCTION دالـة التهذيب الأيمـن

تحذف هذه الدالة كل الفراغات اللاحقة من حقل حرفى، وهى متطابقة مع دالة ( ) TRIM التى سبق شرحها.

THE LTRIM ( ) ("left trim") FUNCTION دالـة التهذيب الأيسـر

تحذف هذه الدالة كل الفراغات السابقة من ناحية اليسار من سلسلة حرفية.

```
.STORE '          ABC' TO ANY
ABC

.? LTRIM(ANY)
ABC
```

لهذه الدالة استخدام خاص، عندما تنتج المخرجات من حقل عددى تعبير، فى العبارة.

مثال ذلك :

```
.STORE 1 TO COUNTER
1

.? COUNTER
1

.? 'The value in counter is ' + STR(COUNTER,8)
The value in counter is        1
```

لاحظ مدى حذف القيمة العددية من الثابت الحرفى.

```
.? 'The value in counter is ' + LTRIM(STR(COUNTER,8))
The value in counter is 1
```

تساعد دالة LTRIM السابقة فى الحصول على التأثير المرغوب فيه، ويمثل COUNTER بصيغة السلسلة، لأن دالة LTRIM يمكنها أن تعمل على سلسلة حرفية فقط.

دالة الرموز اليسرى THE LEFT ( ) ("left-n-characters") FUNCTION

تحضر هذه الدالة n من الرموز الموجودة فى أقصى يسار السلسلة، وهذا هو نفس التأثير مثل دالة ( ) SUBSTR؛ حيث يكون موقع البداية هو 1.

```
.1
.DISP
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE    SALARY NOTE
      1  005     NINA BHARUCHA WEBSTER    BSG .T. 05/24/80    25000.00 Memo

.? LEFT(EMP_NAME 3)
NIN

.? LEFT(TOWN,5)
WEBST
```

تعمل هذه الدالة على حقول المذكرة كذلك، مثل : ? LEFT (NOTE, 10) . والتى تعرض 10 رموز الموجودة فى أقصى يسار حقل المذكرة.

دالة الرموز اليمنى THE RIGHT ( ) ("righr-n-characters") FUNCTION

تعيد هذه الدالة n رمزاً الموجودة فى أقصى يمين السلسلة، باستخدام نفس السجل المعروض :

```
.? RIGHT(TOWN,3)
R
```

بينما يبدو هذا غير صحيح من الوهلة الأولى، إلا أنه يجب أن تتذكر أن الحقل المسمى TOWN يكون طوله 9 رموز. وعلى هذا.. فالرمز الثالث من ناحية اليمين هو R، ويتبعه فراغان.

```
? RIGHT(EMP_NAME,4)
UCHA
```

تعمل هذه الدالة على حقول مذكرات كذلك، مثل : ? RIGHT (NOTE, 10) . يعرض 10 رموز الموجودة فى أقصى اليمين فى حقل المذكرة.

دالة عدد الفراغات THE SPACE ( ) ("how many blanks") FUNCTION

إذا أردت أن تدخل فراغات بين حقول فى عرض للمخرجات، فإحدى طرق عمل هذا الأمر هى كما يلى :

```
LIST EMP_NUM + '          ' + EMP_NAME + '          ' + TOWN      <cr>
```

لقد طلبنا ادخال 10 فراغات بين كل حقلى مخرجات، وفيما يلى طريقة شبيهة لإدخال الفراغات :

```
.LIST EMP_NUM + SPACE(10) + EMP_NAME + SPACE(10) + TOWN
```

دالة ( ) SPACE مفيدة فى مرحلة البرمجة كذلك. فإذا أردت انتاج متغير ذاكرة بـ 25 فراغاً، فمن الأسهل تحقيق ذلك على النحو التالى:

```
STORE SPACE(25) TO MBLANKS      <cr>
```

ينتج عن هذا متغير ذاكرة اسمه MBLANKS يحتوى على 25 فراغاً. (متغيرات الذاكرة مشروحة بالتفصيل فى الجزء الثانى من هذا الكتاب.)

دالة الإضافة والحذف والإحلال THE STUFF ( ) FUNCTION

تساعد هذه الدالة فى ادخال أو حذف أو احلال رموز فى سلسلة حرفية، دون الحاجة إلى اعادة تكوين محتوى السلسلة.

```
          1    <cr>
          DISP    <cr>
Record#  EMP_NUM EMP_NAME      TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      1  005     NINA BHARUCHA WEBSTER    BSG .T. 05/24/80   25000.00 Memo

          ? STUFF(EMP_NAME,5 2,'AL')
          NINAALHARUCHA
```

فى المثال السابق.. سألت : كيف يكون حال الاسم إذا استبدل الرمزان الموجودان فى الموقعين الخامس والسادس بالثابت الحرفى 'AL'? لاحظ أن خامس موقع يشغله فراغ، وهو رمز صحيح مثل أى رمز آخر.

```
.? STUFF(EMP_NAME,1,3,'ABC')
ABCA BHARUCHA
```

يسأل هذا المثال عن شكل المخرجات إذا استبدلت أول ثلاثة رموز فى EMP-NAME بالثابت الحرفى 'ABC'

وترى فى المثالين التاليين عملية التحويل :

```
. ? STUFF (EMP-NAME, 3, 3, ' X ')        يستبدل 3 رموز بـ 1)
NIXBHARUCHA
. ? STUFF (EMP-NAME, 1, 1, ' ABC ')      يستبدل 1 رمز بـ 3)
ABCINA BHARUCHA
```

لاحظ التحويل. إذا استبدلت "n" رمزاً بسلسلة أطول أو قصر، فيحدث التوسع أو الضيق المناسب فى المخرجات.

وفيما يلى.. مثال حذف رموز من سلسلة :

```
. ? STUFF (EMP-NAME, 6, 3,'  ') <cr>
```

لاحظ فى العبارة السابقة استخدام علامتى التنصيص المتتاليتين، وهذا يحدد سلسلة فارغة null string وتأثير ذلك أننى أسأل dBASE عن النتيجة إذا أصبحت الرموز رقم 6, 7, 8 من EMP-NAME فارغة.. تصبح النتيجة كما يلى : NINA RUCHA.

لاحظ السلسلة المستخدمة كسلسلة فارغة (علامتى تنصيص متتاليين، وليس علامتى تنصيص بينهما فراغ). فأنا استبدل بلاشىء، ولا استبدل بفراغات. وبالنسبة للكمبيوتر.. فلاشىء null تعنى لاشىء فعلاً، بينما يعنى الفراغ رمزاً.

THE REPL ( ) ("replicate") FUNCTION دالة التكرار

هذه الدالة مفيدة جداً فى تزيين dolling up شاشات القوائم؛ وذلك إذا أردت مثلاً أن ترسم سطراً من النجوم على الشاشة :

```
. ? REPLICATE (' * ', 50)   <cr>

**************************************************
```

كما يمكنك أن تقدم بعض رموز الرسم كذلك، على النحو التالى :

```
. ? REPL (CHR(5), 50)   <cr>
```

يمكن أن تقرأ العبارة السابقة على النحو التالى : كرر الرمز الممثل بالقيمة العشرية 5 عدد 50 مرة.

فى قسم سابق.. ذكر أنك بإمكانك أن تطلب من dBASE أن يقدم لك الرمز الممثل بقيمة عشرية محددة. القيمة العشرية 5 تمثل الرمز الشبيه بزهرة الزنبق fleur-de-lis. جرب بعض القيم العشرية الأخرى من اختيارك.

يمكنك أن تحدد أكثر من رمز واحد لتكراره، مثلما يلى :

```
. ? REPL (('ABC'), 30)   <cr>
```

القاعدة التى يجب أن تتحقق هى أن السلسلة الناتجة، يجب ألا تتعدى 254 رمزاً. فى المثال السابق.. تكون السلسلة الناتجة عبارة عن الرموز 'ABC' مكررة 30 مرة، أى الإجمالى هو 150 رمزا، الذى يجب ألا يتعدى 254 رمزاً.

سوف يستخدم هذا البديل فى مرحلة البرمجة فى الجزء الثانى هذا الكتاب، لم ساعدتك فى الحصول على عروض رسوم سريعة للأرقام.

دالة القيمة THE VAL FUNCTION

هذه هى عكس دالة STRing، نظراً لأنها تستخدم فى استخلاص البيانات العددية لحقل حرفى يحتوى على بيانات عددية.

وتوجد بعض الأمثلة المقدمة لتوضيح عمل هذه الدالة. افرض أن لديك بعض البيانات المخزنة فى حقل عددى اسمه COUNTER (انظر جدول 1-10).

جدول 1-10

محتويات الحقل الحرفى COUNTER	نتيجة التعبير : VAL (COUNTER)
1234	1234.00
12ABC	12.00
A212C	0.00
12ABC فراغات رائدة	12.00
A123C فراغات رائدة	0.00
بيانات حرفية ! 123.456	لاحظ هذه النتيجة !! 123.46
بيانات حرفية ! 123.4567	لاحظ هذه النتيجة !! 123.46

دالة الرقم الصحيح THE INT ( ) ("integer") FUNCTION

تساعد هذه الدالة فى تقديم القيمة الصحيحة لتعبير، دون الكسر العشرى. أفحص المثال التالى:

<cr> 23/3 ? . نتيجة هذا الأمر هى الاجابة 7.67 (مع التقريب التلقائى لأقرب خانتين عشريتين)

<cr> (23/3) INT ? . نتيجة هذه الدالة هى 7.

وعلى سبيل المثال... لتقريب قيمة بالدولار إلى أقرب دولار، يكتب ما يلى :

. ? INT (19.49 + 0.50) <cr>

والنتيجة هى 19، أما تقريب القيمة التالية :

. ? INT (19.51 + 0.50) <cr>

فهو 20

لتقريب قيمة بالدولار موجودة فى الحقل PAY-HIKE.. يكتب ما يلى :

. ? INT (PAY-HIKE + 0.50) <cr>

كما يتم توضيحه فى الجزء الثانى من الكتاب.. يمكن أن تستخدم هذه الدالة كذلك فيما إذا كان العدد، أى نتيجة التعبير، فردياً أو زوجياً.

دالة رقم السجل THE RECNO ( ) FUNCTION

دالة رقم السجل ( ) RECNO فى dBASE هى دالة خاصة تعنى رقم السجل.

كمثال بسيط... فى أى لحظة يوضع مشير dBASE دائماً عند أحد سجلات الملف المستخدم، وإذا ما أردت ايجاد رقم هذا السجل بسرعة.. فيمكنك دائماً أن تكتب ما يلى : DISP <cr>. فتعرض محتويات السجل بما فى ذلك رقمه. إلا أنك إذا أردت الوصول إلى رقم السجل فقط – دون الوصول إلى بقية محتويات السجل – فيمكنك استخدام هذه الدالة. وتذكر أنه ليس لديك شىء لعمله مع رقم السجل (فنظام dBASE هو الذى يقدم هذه الأرقام مع إضافتك APPENDing بيانات)، كما أنك لا تصل إلى أى حقل موجود فى هيكل السجل، الذى تحصل على رقمه مباشرة بهذه الدالة. وعلى هذا.. فالدالة: <cr> ( ) RECNO ? . لا تقدم إلا رقم السجل الحالى فقط، وفى هذا المثال تسأل السؤال التالى : ما رقم السجل الحالى؟

يكثر استخدام هذه الدالة فى برمجة الكمبيوتر. انظر الجزء الثانى من هذا الكتاب.

دالـة نوع الحقـل THE TYPE ( ) FUNCTION

فى أى وقت من أوقات تشغيل ملف dBASE، إذا أردت إيجاد تعريف نوع أى حقل (حرفى أو عددى أو عائم النقطة أو منطقى أو تاريخ أو مذكرة).. يمكنك دائماً كتابة ما يلى:

. DISP STRU <cr>

يعرض هذا الأمر محتويات السجل، وتستطيع الحصول على المعلومات التى تريدها من هيكل السجل، إلا أنك إذا أردت إيجاد نوع أى حقل. يمكنك أن تكتب ما يلى:

. ? TYPE (' EMP-NUM') <cr> قل إنك تريد التحقق من نوع حقل EMP-NUM.

النتيجة هى C أى إن الحقل حرفى. والنتيجة لأنواع الحقول الأخرى هى : N أو F أو L أو M. ملاحظة : إذا استجاب dBASE بالحرف U، الذى يعنى غير معرف undefined، فيكون قد حدث بعض أنواع الأخطاء التكوينية فى تقديمك اسم الحقل فى الأمر.

دالـة السؤال عما إذا كانت الحالة علوية
THE ISUPPER ( ) ("is it uppercase?") FUNCTION

تساعد هذه الدالة فى معرفة ما إذا كان أول رمز من السلسلة فى الحالة العليا أم ليس منها.

باستخدام نفس السجل فى العرض :

```
.? ISUPPER(ORG)
T

? ISUPPER('hI THERE')
.F
```

دالـة السؤال ما إذا كانت الحالة سفلية

THE ISLOWER ( ) ("is it lowercase?") FUNCTION

تساعد هذه الدالة فى معرفة ما إذا كان أول رمز من سلسلة حرفية فى الحالة السفلى أم ليس منها.

```
. 1
. DISP
Record#   EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE    SALARY NOTE
      1   005     NINA BHARUCHA WEBSTER    BSG .T. 05/24/80    25000.00 Memo

. ? ISLOWER(EMP_NAME)
. F

. ? ISLOWER('hI THERE')
. T
```

دالـة السؤال عما إذا كان أول رمز حرفاً أبجدياً

THE ISALPHA ( ) "is it alpha?") FUNCTION

تساعد هذه الدالة فى تعريف أول رمز من سلسلة حرفية كحرف أبجدى أو كعدد.

```
. 1
. DISP
Record#   EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE    SALARY NOTE
      1   005     NINA BHARUCHA WEBSTER    BSG .T. 05/24/80    25000.00 Memo

. ? ISALPHA(EMP_NUM)
. F
```

لاحظ ما سبق : بالرغم من أن الحقل معرف بأنه حرفى، إلا أن أول رمز من بياناته ليس حرفاً أبجدياً.

```
. ? ISALPHA(TOWN)
. T

. ? ISALPHA('ABC')
. T
```

THE ISCOLOR ( ) ("color monitor?") FUNCTION دالـة لون الموجه

يمكن استخدام هذه الدالة فى معرفة ما إذا كان الكمبيوتر يستخدم بطاقة ملونة color-card، أم بطاقة أحادية اللون monochrome.

```
. ? ISCOLOR( )
. T
```

يمكن أن تخدم هذه الدالة غرضاً جيداً جداً، إذا كتبت برامج لعملاء لأحد المشروعات، وكان المستفيدون يستخدمون خليطاً من الموجهات الملونة وأحادية اللون. وطبقاً لنتيجة هذا الأمر.. يمكنك البرمجة مع سحب الرمز المناسب للعرض الملون، أو العرض غير الملون (العرض العكسي inverse video)، أثناء تنفيذ البرامج.. تساعد هذه الصور – بصفة عامة – في اضاءة الأخطاء التي تحدث أثناء ادخال البيانات والتنقيح. ويوضح هذا المفهوم بالتفصيل، في مرحلة البرمجة من هذا الكتاب.

دالة نظام التشغيل THE OS ( ) FUNCTIO

```
.? OS( )
MS-DOS 3.00
```

إذا كان نظامك المبرمج معتمداً على استخدام مستوى معين من نظام التشغيل.. يمكن لهذه الدالة أن تساعد في التأكد من التوافقية بين النظام وبرامجك، قبل الاستمرار في سحب القائمة الرئيسية للمستفيدين.

دالة السؤال على أي قاعدة بيانات

THE DBF ( ) ("which database?") FUNCTION

تدعك هذه الدالة تعرف أي قاعدة بيانات هي المستخدمة في منطقة العمل المختارة حالياً، وسوف يوضح مفهوم قاعدة البيانات الحالية في منطقة عمل فيما بعد.

```
.? DBF( )
C:PERSNL.dbf
```

دالة السؤال على أي حقل THE FIELD ( ) ("which field?") FUNCTION

```
.? FIELD(7)
SALARY
```

يعيد هذا اسم الحقل من الهيكل، إذا قدمت موقع الحقل في الهيكل.

THE LUPDATE ( ) FUNCTION دالة آخر تجديد

هذه هي دالة تاريخ آخر تجديد date-of-last-update. ومن الواضح أن هذه المقدرة على معرفة متى حدث آخر تجديد للملف، يمكن أن يكون لها تأثير مهم على البرمجة.

```
.? LUPDATE( )
01/06/86
```

دالة المنطقة المتاحة على القرص

THE DISKSPACE ( ) ("available disk-space") FUNCTION

تحصل هذه الدالة على حجم المنطقة المتاحة للقراءة والكتابة على القرص التقليدي، ويمكن استخدامها إما بمفردها، أو كجزء من مشروع؛ حيث يلقن برنامج الكمبيوتر المستفيد ليغير الأقراص، إذا قل حجم المنطقة المتاحة عن قيمة يسبق تعريفها.

```
.? DISKSPACE( )
22802432

.? DISKSPACE( ) < 3000
.F.
```

THE RECCOUNT ( ) FUNCTION دالة عد عدد السجلات

هذه دالة لعد عدد السجلات.

```
.? RECCOUNT( )
15
```

THE RECSIZE ( ) FUNCTION دالة حجم السجل

هذه دالة لمعرفة حجم السجل.

```
? RECSIZE( )
56
```

التأكد من حجم الملف والمنطقة الموجودة على القرص

CHECKING FILE SIZE AND DISK SPACE

يمكنك استخدام خليط من الدوال ( ) DISKSPACE، و ( ) RECSIZE، و REC-COUNT ( ) فى تحديد ما إذا كان ممكناً لك أم لا، أن تعد نسخة احتياطية من أى ملف رئيسي على نفس قرص الملف الرئيسى.

إجمالى حجم أى ملف هو مجموع : حجم العناوين header-size + حجم البيانات data-size.

حجم بيانات أى ملف = عدد السجلات ( ) RECCOUNT * حجم كل سجل REC-SIZE ( )

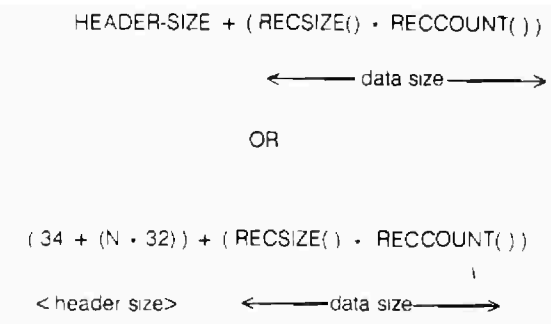
إذا كنت تستطيع حساب حجم العناوين.. فإنك تعرف حجم الملف. ودعنى أصف لك – تفصيلياً – محتويات عناوين الملف، وأبين لك كيفية حساب حجمها.

تحتوى عناوين ملف dBASE IV على معلومات عن تاريخ انتاج الملف، وتاريخ آخر تجديد للملف، وعدد سجلات الملف، وطول السجل، ومعلومات عما إذا كان الملف محتوياً على حقول مذكرات أم لا... الخ. كما يحتوى كذلك على معلومات تفصيلية عن كل حقل معرف فى هيكل الملف؛ أى إن عناوين الملف تشتمل على معلومات أساسية، ومعلومات عن الهيكل، وتقدم المعلومات الأساسية فى 34 بايت من البيانات، وتتطلب المعلومات عن الهيكل 32 بايت لكل حقل معرف فى هيكل الملف. ومما سبق.. تستطيع أن تستخلص بسهولة أن الصيغة اللازمة لحساب حجم عنوان الملف، هى :

(N * 32) + 34؛ حيث N هى عدد الحقول المعرفة فى هيكل الملف.

نظراً لأن عدد الحقول المعرفة فى هيكل الملف، يمكن أن تتغير من 1 إلى 255.. فإن المعلومات عن هيكل الملف (وبالتالى حجم عنوان ملف dBASE IV) يصبح متغيراً كذلك.

إجمالى حجم أى ملف يمكن أن يحدد الآن، كما هو مبين فى شكل 3-10.



شكل 3 - 10 : N هو عدد الحقول الموجودة فى هيكل الملف.

إذا انتجت ملفاً به 10 حقول (بغض النظر عن طول أى من هذه الحقول!).. يصبح حجم عناوين الملف 354 = (32 * 10) + 34.

إذا عرفت كلاً من الحقول البالغ عددها 10 بأنه يشغل 5 خانات.. فإن حجم السجلات 1 + 5 * 10 = ( ) RECSIZE رمزاً (الرمز المضاف خاص بمؤشر الحذف)؛ أى إن 10 حقول المعرفة بأن كل منها يشغل 5 خانات تحتاج إلى حجم سجل ( ) RECSIZE، مساوياً 51 رمزاً. (ويحقق أمر DISPLAY STRUCTURE هذا بسهولة.)

افرض أنك ادخلت 10 سجلات بيانات الآن، كل منها طوله 51، كما سبق توضيحه. يصبح ( ) RECCOUNT مساوياً 10 و ( ) RECSIZE مساوياً 51، وعلى هذا.. فإن إجمالى حجم البيانات يصبح ( ) RECCOUNT * ( ) RECSIZE أو 510 = 51 * 10 رمزاً، ويصبح إجمالى حجم الملف HEADER-SIZE + DATA-SIZE أو 510 + 354؛ أى يساوى 864 رمزاً.

حاول الآن ما يلى :

```
CREATE TRIAL      <cr>
```

والآن فى هذا الملف.. عرف 10 حقول، حجم كل منها 5 رموز.

لإغلاق الملف :

```
.USE    <cr>
.DIR    <cr>
```

ترى أن حجم الملف مبين على أنه 354. وهذا لأن الملف خالٍ الآن، ويكون حجم الملف هو نفسه مثل حجم العناوين، ويحسب حجم العناوين بواسطة :

(32 * 10) + 34 والتى ينتج عنها 254. لاحظ أن أحجام الحقول لا تؤثر فى حجم العناوين!

```
.APPEND    <cr>
```

ادخل 10 سجلات فى هذا الملف، مع ادخال رمز واحد فقط فى كل حقل من سجل.

لإغلاق الملف :

```
USE    <cr>
DIR    <cr>
```

يبين سرد الدليل الآن حجم الملف بأنه 864 رمزاً، وقد سبق أن أوضحت كيف أمكن الوصول إلى هذا الرقم.

لاحظ عبارة USE لإغلاق قاعدة البيانات، قبل التأكد من قائمة الدليل، وهذا احتياط؛ للتأكد من أن عناوين الملف تجددت بالمعلومات الجديدة، قبل أن أبدأ التأكد من الأرقام والأعداد.

إلى هنا.. تنتهى مناقشتنا للدوال الخاصة، وهناك دوال أكثر يتم تغطيتها، إلا أننى أتعرض لها فى المواقع المناسبة من هذا الكتاب، وذكرها هنا يضعها خارج المحتوى، ويجعل ادراكها صعباً.

أمر التوقيع LOCATE COMMAND

انظر فصل 20 عن الأوامر المتنوعة Miscellaneous Commands لمناقشة أمر التوقيع Locate .

الفصل الحادى عشر

عرض بيانات من قواعد بيانات متعددة

DISPLYING DATA FROM MULTIPLE DATABASES

درست بالتفصيل حتى الآن كيفية عرض البيانات فى أشكال وصور متعددة، من قاعدة بيانات واحدة. انظر إلى التوسع الآن.. عندما يمكنك استخدام بيانات من قواعد بيانات متعددة، فى نفس الوقت.

افرض أن لديك الملفات التالية، التى لها الهياكل المبينة :

```
INVEN1.dbf  PART_NUM  COST  DESC
INVEN2.dbf  PART_NUM  CGC          (Commodity-group-code)
INVEN3.dbf  PART_NUM  ONHAND
INVEN4.dbf  PART_NUM  ONORDER
```

افرض أن لديك سجلاً مناظراً لكل سجل فى الأربعة قواعد بيانات طبقاً لرقم السجل؛ أى إن لكل سجل مخزون بيانات منتشرة عبر الأربعة ملفات الفعلية، وذلك عند رقم السجل المناظر.

لتوضيح ذلك أكثر.. فإن :

السجل رقم 1 من INVEN 1 يحتوى على PART-NUM و DESC و COST لـ P1

والسجل رقم 1 من INVEN 2 يحتوى على PART-NUM و CGC لـ P1

والسجل رقم 1 من INVEN 3 يحتوى على PART-NUM و ONHAND لـ P1

والسجل رقم 1 من INVEN 4 يحتوى على PART-NUM و ONORDER لـ P1

لاحظ كيفية انتشار الحقول المختلفة لـ P1 عبر الملفات الأربعة، عند نفس مستوى رقم السجل. افرض أن هذا صحيح لكل سجل، وهذا كما لو كانت قواعد البيانات الثانية والثالثة

والرابعة امتداداً لقاعدة البيانات الأولى؛ فإذا كان هناك اختيار لوضعت الحقول على هيكل واحد، إلا أننا حالياً نجد أن الأمر بهذه الصورة.

باعتبار هذا الموقف..عليك أن تستخدم كل الملفات الأربعة الواقعية، في عرض سجل البيانات المنطقي الواحد هذا. وتبدأ بإخطار dBASE بأن تستعمل مستخدماً أكثر من ملف واقعي واحد في نفس الوقت.

```
SELECT1     <cr>
USE INVEN1     <cr>
```

يحدد هذان الأمران ملف INVEN1 كملف يفتحه dBASE في المنطقة رقم 1، ويستطيع dBASE أن يعمل مع عدد من الملفات يصل إلى 10 ملفات، موجودة في 10 مناطق عمل مختلفة، في نفس الوقت. وتعني كلمات «في نفس الوقت» أن dBASE يحفظ مؤشرات مستقلة ومنفصلة للسجل الحالي، في كل منطقة من مناطق العمل.

```
.SELECT 2     <cr>
.USE INVEN2     <cr>
```

يحدد هذان الأمران ملف INVEN2، كملف مفتوح في منطقة العمل الثانية.

```
.SELECT 3     <cr>
.USE INVEN3     <cr>
```

يحدد هذان الأمران ملف INVEN3، كملف مفتوح في منطقة العمل الثالثة.

```
SELECT 4     <cr>
USE INVEN4     <cr>
```

يحدد هذان الأمران ملف INVEN4، كملف مفتوح في منطقة العمل الرابعة.

(بنفس الطريقة... يمكنك أن تختار حتى 10 ملفات، في 10 مناطق عمل مختلفة.) نظراً لأنك تفترض أن الملفات 2, 3, 4 هي امتداد للملف الأول.. فعليك بتحديد أن الملف الأول هو الملف الأولي أو ملف التحكم، مع إشارة الملفات الأخرى له، على أساس أرقام السجلات؛ والهدف من هذه العلاقة هو التأكد أن مشيرات السجلات تتحرك كلها في انسجام؛ أي إنه إذا كان المشير في الملف الأول موجوداً عند السجل 20.. فإنك تريد أن تتأكد أن كل مشيرات

السجلات فى الملفات الأخرى عند نفس السجل رقم 20 أيضاً ويكون هذا ضرورياً بالنسبة للترتيب الطبيعى للملفات والسجلات؛ ولتناظر سجل من كل ملف مع سجل من الملف الأولى.

```
SELECT 1      <cr>
```

يخطر هذا الأمر dBASE أنك تريد فى هذه اللحظة أن يكون الملف INVEN1 نشطاً، أو أنه يكون ملف التحكم؛ أى إنه يجب تنفيذ كل الأوامر المعطاة على هذا الملف، إلا إذا حدث اختيار آخر.

```
SET RELA TO RECNO( ) INTO INVEN2  RECNO( ) INTO INVEN3
RECNO( ) INTO INVEN4
```

لقد وضعت علاقة طبقاً لأرقام السجلات بين الملف النشط INVEN1. dbf، والملفات المتصلة INVEN2، و INVEN3، و INVEN4. وهذا يعنى أنه إذا كان مشير السجل فى الملف النشط موجوداً عند السجل رقم 5.. فإن مشيرات السجلات فى الملفات المتصلة تنقل – بصورة تلقائية – إلى السجل رقم 5، وبهذا.. تكون قد نجحت الآن فى تحديد تناظر سجل لسجل، عبر قواعد البيانات هذه طبقاً لأرقام السجلات.

أسرد الآن البيانات من الملفات الأربعة.

```
a) .LIST PART_NUM,INVEN2 - >CGC,INVEN3 - >ONHAND,INVEN4 - >ONORDER

Record#   PART_NUM  INVEN2->CGC  INVEN3->ONHAND  INVEN4->ONORDER
      1   P1        C1                      100              100
      2   P2        C2                      200              200
      3   P3        C3                      300              300
      4   P4        C4                      400              400
      5   P5        C5                      500              500
```

لاحظ استخدام المؤشر : INVEN2-> CGC، أو المؤشر INVEN3-> ONHAND. وهذا يعنى الحقل CGC من الملف INVEN2، أو الحقل ONHAND من الملف INVEN3 و...هكذا.

لاحظ أن INVEN2-> أو INVEN3-> يسمى دليلاً مستعاراً alias reference للملف. وتكون فى حاجة إلى تقديم دليل مستعار، إذا أشرت إلى ملف سبق فتحه، ولكنه ليس الملف الحالى (أو الملف المختار). اصطلاح مستعار موضح بتفاصيل أكثر فى القسم التالى.

يطلب السرد السابق عرض PART-NUM من الملف النشط INVEN1، وكذلك عرض الحقول CGC و ONHAND و ONORDER من الملفات المتصلة INVEN2، و INVEN3، و INVEN4 لكل السجلات على التوالي.

من المهم أن يُفهم عند هذه النقطة أنه بعد العرض السابق... يظل INVEN1. dbf هو الملف الأولى أو ملف التحكم، وذلك لأن عبارة SELECT تشير إلى منطقة العمل رقم 1 (work-area-1)، وأنت لم تغير من الاختيار.

b) .3

تنقل العبارة السابقة مؤشر السجل إلى السجل رقم 3 في ملف التحكم، وبالتالى إلى السجل رقم 3 في كل الملفات المتصلة.

```
.LIST NEXT 3 PART_NUM,INVEN4 - > ONORDER

Record#   PART_NUM  INVEN4->ONORDER
      3   P3                    300
      4   P4                    400
      5   P5                    500
```

ويطلب هذا السرد الأخير عرض PART-NUM من الملف النشط INVEN1، وكذلك عرض ONORDER من الملف INVEN4، هذا للسجلات : رقم 3، ورقم 4، ورقم 5 فقط؛ حيث يتحكم فيها 1NEVN1.

ويظل INVEN1. dbf هو ملف التحكم.

c) .4

تنقل العبارة السابقة مشير السجل في ملف التحكم إلى السجل رقم 4، وكذلك تنقل مشير السجل إلى السجل رقم 4 في جميع الملفات الأخرى.

```
.LIST NEXT 2 PART_NUM,INVEN2 - > CGC,INVEN3 - > ONHAND

Record#   PART_NUM  INVEN2->CGC  INVEN3->ONHAND
      4   P4        C4                     400
      5   P5        C5                     500
```

يطلب السرد السابق عرض PART-NUM من الملف النشط INVEN1، وكذلك عرض CGC من الملف INVEN2 و ONHAND من الملف INVEN3، وذلك للسجلين : رقم 4 و 5 فقط؛ حيث يتحكم فيها INVEN1.

أثناء التواجد فى هذه الترتيبات المتصلة.. يمكنك أن تعرض أو تنقح (والمشروح فيما بعد) سجلاً معيناً من أى ملف من هذه الملفات. ومثال ذلك إذا أردت أن ترى السجل رقم 5 من الملف INVEN4:

```
.5      <cr>
```

هذا ينقل مشير السجل إلى السجل رقم 5.

```
SELE 4      <cr>
```

ومن الآن.. يصبح الملف INVEN4 هو ملف التحكم؛ لأننى أخترت منطقة العمل رقم 4، ويكون مشير السجل عند السجل رقم 5 كما سبق إعداد ذلك.

```
DISP      <cr>
```

يعرض هذا الأمر السجل الحالى فقط، وذلك من منطقة العمل الحالية الرابعة. وبدلاً من العرض.. يمكننى أن أنقح (والمشروح فيما بعد) السجل. واستطيع - بهذه الطريقة - اختيار أى منطقة عمل، ويحضر العرض أو التنقيح السجل رقم 5 من هذا الملف.

ملاحظة : مع عرض أو تنقيح أى سجل متصل.. تأكد أنك حددت INVEN1 مرة أخرى كملف رئيسى (SELE.). وفى كلمات أخرى.. لا يمكنك الانتقال إلى رقم سجل آخر، إلا عندما تختار الملف الرئيسى!، وعدم مقدرة تتبع هذه القاعدة يؤدى إلى نقل مشيرات السجلات إلى أى مكان.

وفيما يلى.. مثال كامل لاتصال 4 ملفات واستخلاص سرد من هذه الملفات، المرتبطة ببعضها البعض.

```
SELE 1
USE INVEN1
SELE 2
USE INVEN2
.SELE 3
.USE INVEN3
.SELE 4
.USE INVEN4
```

تحدد الأوامر السابقة 4 ملفات مختلفة لـ 4 مناطق عمل مختلفة، ويمكنك أن تحدد حتى 10 ملفات مختلفة لـ 10 مناطق عمل مختلفة.

هذا يعرف ملف التحكم

```
.SELE 1
.SET RELA TO RECNO( ) INTO INVEN2, RECNO( ) INTO INVEN3,
            RECNO( ) INTO INVEN4
```

يضع الأمر السابق العلاقات بين الملفات، وهذه العلاقات مبنية على أرقام السجلات. ملف INVEN1 فى منطقة العمل 1 هو ملف التحكم. وتستطيع الآن عمل السرد، كما سبق توضيحه. مثال ذلك:

```
.LIST PART_NUM,INVEN2 – >CGC,INVEN3 – >ONHAND,INVEN4 – >ONORDER
```

SAVING THE LINKED ARRANGEMENT حفظ الترتيب المتصل

عندما تكون لديك مجموعة من الملفات المرتبطة ببعضها البعض كما هو الحال الآن.. يمكنك أن تحفظ هذه اللقطة السريعة من البيئة snapshot of the environment كما هى فى dBASE؛ أى إنك تستطيع إنتاج ملف خاص، يحفظ شكل ملفات قاعدة البيانات المفتوحة والموجودة، وعلاقاتها كما هى.

إنك تطلب من dBASE أن ينتج ملف رؤية (منظر) view-file اسمه INVVIEW1. vue من البيئة الموجودة حالياً. يمكن أن يكون الاسم الأولى لملف الرؤية أى اسم تريد تسميته، إلا أن المؤهل VUE. هو الاسم الثانوى للملف.

```
.CREATE VIEW INVVIEW1 FROM ENVI     <cr>
```

يمكنك الآن أن تغلق كل الملفات المتصلة؛ لأنك لست فى حاجة للترتيب المتصل.

```
.CLOSE ALL
```

يغلق الأمر السابق يغلق كل الملفات المشمولة فى عملية الاتصال؛ إلا أنه لايؤثر على متغيرات الذاكرة الموجودة حالياً، التى توضح بالتفصيل فى المواقع المناسبة من الكتاب. والآن.. أقبل من فضلك العبارة القائلة بأنك تستطيع غلق الملفات، دون أن تفقد متغيرات الذاكرة كما هى علماً بأن العبارة CLOSE ALL، وهى التى تؤدى ذلك.

إذا لم تغلق كل الملفات المشمولة فى عملية الاتصال.. فإنها تظل مفتوحة، ولن تنجح محاولتك استخدام USE ملف INVEN1 أو ملف INVEN2 فى عملية أخرى بمفردهما؛ لأن الملفات مازالت تحت الاستخدام USE فى مناطق عمل مختلفة. وعلى هذا.. اغلق من فضلك كل الملفات بعد أن ينتهى عمل ترتيب الاتصال.

إعادة تنشيط ترتيب الاتصال

REACTIVATING THE LINKED ARRANGEMENT

بسبب أنك أنتجت ملف رؤية للترتيب فعلاً.. يمكنك الآن فى فرص متتابعة ادخال أمر واحد فقط؛ لإعادة تنشيط كل ملفات قاعدة البيانات وعلاقاتها مع بعضها البعض، كما كانت تماماً عندما انتجت ملف الرؤية.

```
.SET VIEW TO INVVIEW1      <cr>
```

تفتح هذه العبارة الوحيدة كل ملفات المخزون الأربعة، وتضع العلاقات بينها تلقائياً، كما كانت عليه عندما أعد ملف الرؤية، كما أنها تعرف الملف INVEN1 فى منطقة العمل رقم 1، بأنه هو ملف التحكم. وتستطيع الآن أن تستخدم ترتيب الاتصال، وعندما تنتهى من عملك مع ترتيب الاتصال.. لاتنس أن تغلق الملفات.

الفصل الثانى عشر

تحديد مناطق العمل والأسماء المستعارة

WORK-AREA ASSIGNMENTS AND ALIAS NAMES

هناك طريقتان مختلفتان لتعريف مناطق العمل والاشارة إليها، وللملفات المفتوحة فى مناطق العمل.

يمكن الاشارة إلى مناطق العمل على أنها: 1, 2, 3, 4, ..., 10، أو على أنها A, B, C, ..., J

ويمكننى إدخال SELE 1. لمنطقة العمل رقم 1

أو إدخال SELE A. لمنطقة العمل رقم 1.

افرض أن لديك ملفين فقط، هما : INVEN1 و INVEN2، وذلك باعتبار المثال السابق الذى يربط الملفات طبقاً لأرقام السجلات؛ إذ يمكنك أن تكتب العبارات كما يلى :

```
.SELE A
.USE INVEN1
.SELE B
.USE INVEN2
```

عند تعريف أحد الملفات بأنه مفتوح فى منطقة عمل.. يمكنك أن تحدد أسماً مستعاراً ali-as name للملف إذا أردت ذلك. والأسماء المستعارة ليست إجبارية، إلا أنها يمكن أن تساعد فى تقليل مجهود الطباعة اللازم لإتمام بعض العبارات، فإذا لم تستخدم اسماً مستعاراً يصبح اسم الملف نفسه اسماً مستعاراً alias. مثال ذلك:

- .SELE 1 (أو SELE A)
- .USE INVEN1 (اسم الملف هو الاسم المستعار)
- .SELE 2 (أو SELE B)
- .USE INVEN2 ALIAS I2 (مقدم اسم مستعار محدد)

لاحظ العبارة السابقة... لقد فتحت INVEN2 فى منطقة العمل رقم 2، وأعطيت الملف الاسم الاستعارى 12.

لربط هذين الملفين طبقاً لأرقام السجلات :

```
.SELE 1   (or SELE A)
.SET RELA TO RECNO( ) INTO I2, or
.SET RELA TO RECNO( ) INTO B, or
.SET RELA TO RECNO( ) INTO 2
```

أى من العبارات السابقة تعمل بطريقة صحيحة.

إلا أن العبارة :

```
.SET RELA TO RECNO( ) INTO INVEN2
```

لا تعمل فى هذه المرة لأن INVEN2 تم تعريفه بالأسم المستعار 12، وعلى هذا.. فلايسمح باستخدام الاسم الفعلى للملف فى عمل علاقات.

ولاستخلاص قائمة من هذين الملفين :

```
.LIST PART_NUM, I2 - > CGC, or
.LIST PART_NUM, B - > CGC
```

أو

لاحظ مرة أخرى أن الاسم الفعلى للملف لا يمكن استخدامه؛ لأن الاسم المستعار 12 سبق تقديمه، ولاحظ كذلك أن dBASE يأتى بخطأ إذا حاولت استخدام رقم منطقة العمل كدليل مستعار، بالرغم من أن الدليل التقنى لنظام dBASE يؤكد أنك تستطيع عمل ذلك.

فيما يلى... مثال كامل لاتصال أربعة ملفات واستخلاص قوائم من هذه الملفات المرتبطة ببعضها البعض، وقد أوضحت فى هذه المرة الامكانيات المختلفة بتحديدى حروفاً وأرقاماً وأسماء مستعارة.

```
.SELE A
.USE INVEN1
.SELE B
.USE INVEN2 ALIAS I2
.SELE C
.USE INVEN3 ALIAS I3
.SELE D
.USE INVEN4 ALIAS I4

.SELE A
SET RELA TO RECNO( ) INTO I2, RECNO( ) INTO I3, RECNO( ) INTO I4
                                    or
SET RELA TO RECNO( ) INTO B, RECNO( ) INTO C, RECNO( ) INTO D
                                    or
SET RELA TO RECNO( ) INTO 2, RECNO( ) INTO 3, RECNO( ) INTO 4

.LIST PART_NUM, I2 - >CGC, I3 - >ONHAND, I4 - >ONORDER
                                    or
.LIST PART_NUM, B->CGC, C - >ONHAND, D - >ONORDER
```

لاحظ كيف أن استخدام الحرف، أو الرقم، أو الاسم المستعار الدال على منطقة عمل جعل سطر الأمر أقصر !

العلاقة المبنية على أرقام السجلات لا تمثل موقفاً واقعياً صحيحاً. فبعد كل شيء... ليس من المفضل أن يكون ملفان أو أكثر لهم سجلات فردية فى علاقة سجل واحد لسجل واحد بالضبط، وذلك بالنسبة لأرقام السجلات.

الموقف الأكثر واقعية هو الذى تريد أن تضع العلاقات بالنسبة إلى أحد الحقول الرئيسية، وليس طبقاً لأرقام السجلات ببساطة. مثال ذلك قد يكون لديك ملف مخزون رئيسى، يحتوى على PART-NUMBERS، و DESCRIPTIONS، و UNIT-COSTS، ويمكن أن تحفظ فى ملف آخر المعلومات الخاصة ببيانات العمليات الجارية لنفس سجلات المخزون، وتقع المشاركة بين هذين الملفين فى الحقل PART-NUMBER.

وفى كل الاحتمالات.. لا تتبع سجلات العمليات الجارية نفس التسلسل للسجلات المناظرة لها فى الملف الرئيسى. وفى هذه الحالة.. يمكنك أن تعد نوعاً شبيهاً من العلاقة كذلك، ولكنها تكون معتمدة على الحقل PART-NUMBER، دون أرقام السجلات. وسوف يشار إلى تفاصيل هذا الترتيب فى قسم لاحق؛ لأنها تتطلب أن تكون الملفات المتصلة مفهرسة indexed، وهذا الموضوع نتناوله مرة أخرى فى قسم الفهرسة من هذا الكتاب.

الفصل الثالث عشر

أوامر المجموع والمتوسط والحساب

SUM, AVERAGE AND CALCULATE

بعد أن أصبحت لديك المقدرة على سرد LIST (أو عرض DISPLAY) البيانات بعدة طرق.. فإننى أدعم هذا القسم بثلاثة أوامر، يمكن أن تجدها مفيدة.

يقدم أمر المجموع SUM إجمالى القيم فى حقول (عددية) معينة، ويقدم أمر AVERAGE قيمة متوسطة للسجل، ويسمح أمر CALCULATE بأخذ قراءات إحصائية من قاعدة البيانات.

المجموع SUM

```
USE PERSNL
.SUM
```

لاحظ الأمر السابق... إنك لم تحدد لنظام dBASE أى الحقول التى تريد حساب المجموع لها. وتقليدياً.. يقدم dBASE مجموعاً واحداً لكل حقل عددى موجود فى هيكل الملف – بعد تشغيل كل سجلات الملف، وتعرف المخرجات على الشاشة عدد السجلات التى حُسب لها المجموع، واسم كل حقل (عددى)، وقيمة مجموعه.

```
SUM SALARY
```

يقدم هذا مجموع SALARY، لجميع السجلات.

```
SUM SALARY FOR ORG = 'BSG'
```

يقدم هذا مجموع SALARY للسجلات التى تحقق الشرط فقط.

تظهر الأوامر السابقة هذه المجاميع على الشاشة، بالنسبة إلى المعلومات المرئية فقط. إلا أنك تستطيع أن تحدد لنظام dBASE أنك تريد الأعداد مخزنة، فى مواقع تخزين مؤقتة فى

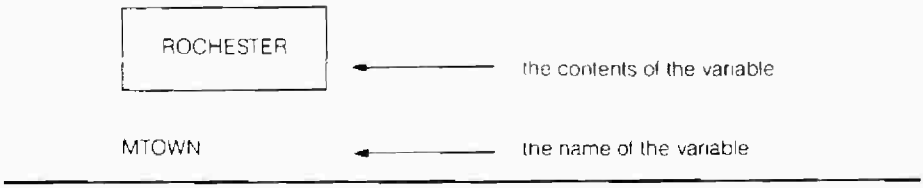
الذاكرة، وذلك للاستخدام المستقبلى أثناء جلسة عملك مع dBASE، وتسمى مواقع التخزين المؤقتة هذه بمتغيرات ذاكرة memory variables.

وأصف الآن اصطلاح متغير ذاكرة memory variable بإيجاز، وهذا المفهوم مشروح بتفاصيل أكثر فى الجزء الثانى من هذا الكتاب.

إذا استطعت أن تخصص جزءاً صغيراً من ذاكرة الكمبيوتر، وتعطيه اسماً، وتخزن بعض البيانات فيه.. فإنك تكون قد عرفت متغير ذاكرة.

مثال ذلك : STORE 'ROCHESTER' TO MTOWN

يخزن هذا الأمر القيمة الثابتة ROCHESTER فى قسم من الذاكرة، خصص له الاسم MTOWN. تخيل ذلك كما فى شكل 1-13.



شكل 1-13 : تخزين القيمة ROCHESTER فى موقع الذاكرة MTOWN.

يمكن لمتغيرات الذاكرة أن تكون حرفية أو عددية أو متحركة النقطة أو منطقية أو تاريخاً. وبعد إنتاجها.. يمكنك الاتصال بمحتوياتها. وتساعد أوامر SUM و AVERAGE فى إنتاج متغيرات، تحتوى على المجموع والمتوسط على التوالى لعناصر عددية. وفيما يلى بعض الأمثلة:

SUM TO N1

تعرف المخرجات على الشاشة عدد السجلات التى أخذ المجموع لها، واسم كل حقل (عددى)، وقيمة مجموعه. كما يضع هذا الأمر المجموع (لأول حقل عددى موجود فى هيكل الملف) فى متغير اسمه N1.

SUM TO N1, N2.

ينتج هذا الأمر مخرجات الشاشة كما سبق ذكره، ويضع كذلك مجموعى أول حقلين عدديين (موجودين فى هيكل الملف) فى المتغيرين N1 و N2.

```
SUM SALARY TO MSAL
```

يجمع الأمر السابق كل بيانات SALARY، مبيناً النتيجة على الشاشة، ومخزناً إياها فى متغير ذاكرة اسمه MSAL.

```
4
.SUM NEXT 5 SALARY TO MSAL
```

يؤدى هذا الأمر نفس الشيء، ولكن ذلك على السجلات التى تبدأ من السجل رقم 4 – حتى السجل الثامن فى الملف.

```
SUM SALARY TO MSAL FOR TOWN = 'R'
```

يمكنك أن تجد مجموع السجلات التى تحقق الشرط، ويمكن للشرط أن يكون بسيطاً أو مركباً.

لاحظ أنه إذا كانت لديك حقول عددية متعددة معرفة فى الملف – ولتكن مسماة F1, F2, F3, F4 – يمكنك أن تحسب المجموع لها كلها، مستخدماً عبارة واحدة، وذلك كما يلى :

```
.SUM F1,F2,F3,F4
```

يعرف الأمر السابق كل حقل، ويقدم مجموعه على الشاشة.

```
.SUM F1,F2 F3,F4 TO M1,M2,M3,M4
```

يعرف الأمر السابق كل حقل، ويقدم مجموعه على الشاشة، كما أنه ينتج كذلك متغيرات ذاكرة، مناظرة للأربعة حقول العددية؛ أى إن، مجموع F1 يكون فى متغير الذاكرة M1، ويكون مجموع F2 فى متغير الذاكرة M2 و...هكذا.

المتوسط AVERAGE

يأخذ شكل أمر AVERAGE شكلاً متطابقاً مع شكل أمر SUM. ادخل AVER ببساطة بدلاً من SUM فى الأمثلة السابقة.

الحســـاب CALCULATE

يسمح أمر الحساب بإعداد بعض التقارير المالية والإحصائية لبيانات، خلال المرور فى قاعدة البيانات.

```
.CALCULATE  [scope]  <OPTIONS>
                     [FOR    <condition>]
                     [WHILE  <condition>]
                     [TO     <mem-vars>]
```

لقد سبق أن رأيت تأثير مؤشر المدى [ALL, RECO <n>, NEXT <n>, - REST] العديد من المرات فى الصفحات السابقة. بالمثل... فأنت تعرف بالضبط كيفية عمل مؤشرى FOR و WHILE.

إذا لم يقيد أمر CALCULATE بمؤشر مدى أو بأى من المؤشرين FOR أو WHILE.. فإنه يأخذ فى الاعتبار كل السجلات الموجودة فى الملف، والبديل الذى يمكن أن تطلبه هو :

AVG	متوسط	CNT( )	عد
MIN	الأقل	STD	الانحراف المعيارى
SUM	المجموع	VAR	التباين
NPV( )	صافى القيمة الحالية	MAX	الأكبر

١ – للحصول على قيمة المتوسط، وأكبر قيمة وأصغر قيمة للحقل SALARY، لكل السجلات– أثناء مرور مرة واحدة فى قاعدة البيانات – اكتب ما يلى :

```
.CALCULATE AVG(SALARY), MAX(SALARY), MIN(SALARY)      <cr>
```

٢ – للحصول على عدد سجلات الملف، وعلى المجموع، وعلى الانحراف المعيارى، والتباين للحقل SALARY، لكل السجلات – أثناء المرور مرة واحدة فى قاعدة البيانات – اكتب ما يلى :

```
.CALCULATE CNT( ), SUM(SALARY), STD(SALARY), VAR(SALARY)
                   FOR SALARY > 20000
```

الفصل الرابع عشر

عمليـة التنقيـح

EDITING PROCESS

تنتقل الآن إلى عالم التنقيح EDITING، وتدرس طرقاً عديدة، تستطيع أن تغير بها محتويات سجلات قاعدة البيانات أو الهيكل؛ لكى تحفظ البيانات الأصلية لتقارير لاحقة، نفذ من فضلك الأوامر التالية :

```
USE PERSNL      <cr>
COPY TO PRESERVE      <cr>
```

(يشرح أمر COPY فيما بعد)

الملف المستخدم هو ملف PERSNL مع تغطية سمات التنقيح هنا، وبعد ذلك.. يمكنك أن تستعيد ملف PERSNL الأصلى.

أمر التنقيح (أو التغيير) EDIT (Or CHANGE) Command

يسمح أمر التنقيح EDIT (أو التغيير CHANGE) بإجراء تغييرات على قاعدة البيانات، سجلاً سجلاً.

والشكل الشامل لأمر التنقيح هو كما يلى :

```
.EDIT  [<record-number>]  [<scope>]  [FIELDS <field-list>]
                                     [FOR <condition>]
                                     [WHILE <condition>]
                                     [NOINIT]
                                     [NOAPPEND]
                                     [NOMENU]
                                     [NOEDIT]
                                     [NODELETE]
                                     [NOCLEAR]
                                     [NOFOLLOW]
```

عليك أن تقرر رقم السجل الذى تريد أن تبدأ عملية التنقيح عنده، وتساعدك أوامر DI-SPLAY السابقة فى الحصول على أرقام السجلات المراد تنقيحها.

```
.USE PERSNL     <cr>
.EDIT 5     <cr>
```

يحضر هذا السجل رقم 5 على الشاشة – فى حالة تنقيح الشاشة الكاملة – تحت تحكم حركة نقطة البداية فى الشاشة كلها، كما سبق. تظهر البيانات فى صورة رأسية؛ حيث تستطيع أن تجرى التنقيح على السجلات سجلاً سجلاً.

```
                                                    Caps
Record No       5
EMP_NUM         025
EMP_NAME        KIM BRANDT
TOWN            FAIRPORT
ORG             RMG
EXE             F
DT_OF_HIRE      04/04/77
SALARY          36000.00
NOTE            MEMO
```

ملاحظة ، فى حالة التنقيح.. لايسرى الاستثناء الذى سبق أن أجريته (أثناء مناقشة أمر APPEND)؛ أى إنه بالرغم من أن نقطة البداية تكون فى أول موقع من أول حقل من حقول الهيكل.. فإن الضغط على مفتاح الادخال <cr> ينقل نقطة البداية حقلاً لأسفل، ولا يخرجك من حالة التنقيح EDIT.

تذكر أن الحقل EXEMPT هو حقل منطقى، ولا يمكن أن يحتوى إلا على أى حرف من الحروف T/F/Y/N من البيانات.

لتغيير محتويات حقل مذكرة.. يجب أن تنقل نقطة البداية إلى هذا الحقل، وتضغط على Ctrl-Home، وتجرى التغييرات على المذكرة ثم تضغط على Ctrl-End لحفظ المذكرة.

بعد اجراء التغييرات على السجل الحالى – على الشاشة – يكون أمامك أن تختار أيًا من الخيارين التاليين :

١ – قد تريد أن تحفظ السجل وتجرى تغييراً شبيهاً على السجل التالى، وفى هذه الحالة اضغط على PgDn.

ملاحظة : إذا كان السجل الحالى على الشاشة هو آخر سجل فى قاعدة البيانات.. فإن الضغط على PgDn يحفظ السجل، ويسألك عما إذا كنت تريد ادخال سجلات أخرى؛ أى إن التنقيح يقدم لك طريقة أخرى لإضافة سجلات لقاعدة البيانات.

٢ – قد تريد أن تحفظ هذا السجل وتظهر السجل السابق؛ لإجراء تغييرات شبيهة عليه. اضغط على PgUp.

ملاحظة : إذا كان السجل الحالى على الشاشة هو أول سجل فى قاعدة البيانات.. فإن الضغط على PgUp يحفظ هذا السجل، ويترك السجل الحالى على الشاشة. وإما أن تضغط على Esc، أو على Ctrl-End للوصول إلى ملقن النقطة.

٣ – قد تريد حفظ هذا السجل والخروج من حالة التنقيح. اضغط على Ctrl-W، أو على Ctrl-End.

٤ – قد تريد العدول عن التغييرات التى أجريت (على السجل الحالى فقط!)، والخروج من حالة التنقيح. اضغط على Ctrl-Q، أو على Esc.

٥ – قد تريد حذف السجل الحالى. اضغط على Ctrl-U.

ملاحظة : فى إجراء التغييرات على السجل : إذا تغير آخر رمز من آخر حقل، أو إذا استمر الضغط على مفتاح الادخال بعد آخر حقل فسوف يؤدى هذا إلى نفس التأثير مثل الضغط على PgDn.؛ أى إن السجل التالى يظهر للتنقيح.

كما أنه إذا كان فى السجل كثير من الحقول، ولا تستطيع رؤيتها كلها عند ظهور السجل لأول مرة للتنقيح عند ذلك.. فإن الضغط على PgDn يتسبب فى اظهار الجزء التالى من السجل، واعادة نفس الإجراءات تتسبب فى التأثير الفعلى للضغط على PgDn

يقدم تحريك مفتاح حركة السهم لأعلى، بعد الحقل الأول يقدم كذلك السجل السابق للتنقيح، ويقدم تحريك مفتاح حركة السهم لأسفل بعد الحقل الأخير كذلك السجل التالى للتنقيح.

أثناء التواجد فى حالة التنقيح إذا ضغطت على Ctrl-U.. فإنك تشير إلى سجلات يراد حذفها. لاحظ أن مؤشر الحذف يظهر فى قمة الشاشة. يلغى الضغط مرة أخرى على Ctrl-U هذه الاشارة؛ أى إن الضغط على Ctrl-U ينقل من وإلى إظهار مؤشر الحذف أثناء حالة التنقيح.

```
5
.EDIT NEXT 5      <cr>
```

الآن يقيد إجراء التنقيح على مؤشر المدى <scope>. إنه يبدو كما لو كانت محتويات قاعدة البيانات هى من السجل الخامس، حتى السجل التاسع.

```
EDIT FIELDS TOWN,ORG,SALARY      <cr>
```

والآن... يقيد إجراء التنقيح على الحقول المذكورة فى قائمة الحقل.

```
.EDIT FOR    <condition>
.EDIT WHILE    <condition>
```

يمكنك أن تحدد أى شرط تريده، وعند ذلك يصبح إجراء التنقيح مقيداً للسجلات التى تحقق هذا الشرط. ويجبر شرط FOR بدء البحث عند قمة الملف والاستمرار من هناك. ولا تختر إلا السجلات التى تحقق الشرط فقط للتنقيح. ويبدأ شرط WHILE البحث عند السجل الحالى، ولا ينقح إلا السجلات المتتالية التى تقدم نفس الشرط فقط.

يجد الضغط على PgUp السجل السابق فى الاتجاه العكسى، الذى يحقق الشرط، اما الضغط على PgDn.. فإنه يجد السجل التالى، الذى يحقق الشرط، وذلك فى شروط كل من FOR وWHILE.

سبق لك أن رأيت أمثلة عديدة لشروط FOR وWHILE فى أمر DISPLAY.

```
.EDIT NOINIT      <cr>
```

يقول مؤشر noinit : لا تضع قيماً ابتدائية لمؤشرات التنقيح الحالية؛ أى إن أمر التنقيح السابق له بعض التأثير:

```
.EDIT FIELDS TOWN, ORG, SALARY      <cr>
```

فيقيد تأثير التنقيح على الحقول المذكورة فقط؛ فإذا أدخلت الآن الأمر : EDIT <cr> فإنك فى واقع الأمر تنفى قائمة الحقل field-listing السابقة، وتصبح كل الحقول متاحة للتنقيح، إلا أنك إذا حددت :

.EDIT NOINIT <cr>

.. فإنك تؤكد فى الواقع أن المؤشرات التى سبق تعريفها لا تتأثر.

ملاحظة : يطبق مؤشر NOINIT على قائمة حقول، وعلى مؤشرات أخرى (تناقش فيما بعد) ولايطبق على شروط FOR و WHILE.

ويعمل مؤشر NOINIT إذا كانت أوامر EDIT المتتالية لها مؤشر NOINIT فقط، وأى أمر EDIT يظهر دون أن يحتوى على أى مؤشرات.. فإنه ينقح كل السجل، وكذلك.. فإن EDIT NOINIT التالى ينقح EDIT كل السجل كذلك.

.EDIT NOAPPEND <cr>

عادة عندما تضغط على PgDn بعد آخر سجل فى الملف.. يسألك dBASE إذا كنت تريد إدخال سجلات أكثر؛ أى إن التنقيح يقدم لك طريقة للاضافة بعد آخر سجل، وينفى مؤشر NOAPPEND هذا البديل.

.EDIT NOMENU <cr>

هذا ينفى تأثير مفتاح F10، ويمنعك على ذلك من الاتصال بقضيب قائمة menu-bar التنقيح. وتأثير قضيب قائمة التنقيح وبدائله مشروح بتفاصيل أكبر- فى ملحق ج - عن مركز التحكم The Control Center.

.EDIT NOEDIT <cr>

يمكنك هذا من رؤية سجل على الشاشة فى حالة التنقيح، إلا أنه لا يسمح لك بعمل أى تغييرات على السجل.ويمكن أن يكون هذا مفيداً فى مرحلة تطوير وإعداد التطبيقات؛ حيث يمكنك أن تسمح للمستفيد بأن يرى سجلاً، دون أن يستطيع إجراء أية تغييرات عليه.

وبهذا البديل.. يمكنك أن تضيف سجلات بعد السجل الأخير (بافتراض أنك غير مستخدم NOAPPEND) ويمكنك الضغط على Ctrl-U؛ للإشارة إلى السجلات المراد حذفها (بافتراض انك لست مستخدماً NODELETE – انظر العنصر التالى).

```
.EDIT NODELETE    <cr>
```

وباستخدام NODELETE كمؤشر.. فإن الضغط على Ctrl-U لا تأثير له على السجل الحالى.

```
.EDIT NOCLEAR    <cr>
```

عادة.. عندما تخرج من أمر EDIT.. فإن الشاشة تخلو ويظهر ملقن النقطة. وباستخدام NOCLEAR.. يمكنك أن تطلب من dBASE أن يحفظ صورة آخر سجل على الشاشة.

```
EDIT NOFOLLOW    <cr>
```

يطبق هذا البديل إذا كان الملف مفهرساً فقط. وإيجازاً.. يخطر هذا المشير نظام dBASE الا يتبع السجلات المعاد وضعها فى ترتيب منطقى للفهرس، وإنما يختار السجل التالى فقط فى الترتيب الواقعى للفهرس. وتفهم هذا البديل أفضل بعد أن تدرس تفاصيل الفهرسة.

استخدام مفتاحى Shift-F8 Using Shift-F8 (Ditto)

تستطيع أثناء وجودك فى حالة التنقيح أن تستخدم Shift-F8 فى أى حقل، وذلك للكتابة فوق البيانات من الحقل المناظر، من السجل السابق، فى حقل السجل الحالى.

استخدام شكل خاص للشاشة Using a Special Screen Format

سبق أن رأيت فى أمر APPEND أنك تستطيع أن تظل مستخدماً الحالة التقليدية لتخطيط الشاشة، التى تأتى مع أمر APPEND، أو تكون حراً لإختيار تشكيل خاص للشاشة. وميزة الشكل الخاص للشاشة سبق ذكرها فى القسم الخاص بـ APPEND.

ومع أمر التنقيح EDIT.. يمكنك كذلك اختيار شاشة تنقيح تقليدية، أو إعداد شكل خاص بالشاشة.

أنظر ملحق ب لتوضيح كيفية انتاج ملفات أشكال خاصة للشاشة؛ لاستخدامها مع ملف بيانات.

افرض الآن أن لديك ملف تخطيط خاصاً بالشاشة، اسمه LAYOUTA. FMT لدعم مجهوداتك فى ادخال البيانات وتنقيحها فى ملف PERSNL. dbf.

```
USE PERSNL
SET FORMAT TO LAYOUTA
EDIT <with/without options>
```

نظراً لأنه تم تنشيط ملف تشكيل.. فإن تخطيط شاشة التنقيح يصبح معرفاً فى ملف تشكيل، ولايسرى تخطيط الشاشة التقليدى حالياً.

وعندما تكون فى حالة التنقيح.. فلا تنس إيقاف تنشيط (إغلاق) ملف التشكيل.

```
SET FORMAT TO    <cr>
```

أمر SET FORMAT TO – دون أى مؤشرات – يلغى نشاط ملف التشكيل.

التنقيح من عدة قواعد بيانات فى نفس الوقت

Editing from Multiple Databases Simultineously

فى قسم العرض display... تعلمت كيفية فتح ملفات متعددة فى مناطق عمل مختلفة، مع ربطك هذا على أساس رقم السجل، وتستخدم نفس الاجراءات هنا في تنقيح ملفات متعددة فى مناطق عمل متعددة.

يمكنك استخدام نفس ملفات المخزون، التى سبق استخدامها والمسماة INVEN1, IN-VEN2, INVEN3, INVEN4. وتستطيع عند هذه النقطة أن تفترض أن هذه الملفات لها سجل واحد فقط، من كل ملف مناظر لرقم السجل، كما سبق ذكر ذلك.

```
.SELE 1
.USE INVEN1
.SELE 2
.USE INVEN2
.SELE 3
.USE INVEN3
.SELE 4
.USE INVEN4
```

تفتح العبارات السابقة الملفات فى أربع مناطق عمل مختلفة، وتربط الآن هذه الملفات طبقاً لأرقام السجلات.

```
.SELE 1
.SET RELATION TO RECNO( ) INTO INVEN2, RECNO( ) INTO INVEN3,
                                       RECNO( ) INTO INVEN4
```

عند هذه النقطة.. ترتبط الملفات مع بعضها البعض؛ طبقاً لأرقام السجلات. ولايزال IN-VEN1 هو ملف التحكم، ويمكنك الآن أن تقدم أمر التنقيح على النحو التالى :

```
.EDIT FIELDS PART_NUM, B - > CGC, C - > ONHAND, D - > ONORDER
```

لاحظ أنك استخدمت أسماء مستعارة للإشارة إلى INVEN2, INVEN3, INVEN4.

يحضر أمر التنقيح السابق حقل PART _ NUM من INVEN1 إلى الشاشة، ويحضر حقل CGC من INVEN2 إلى الشاشة، ويحضر حقل ONHAND من INVEN3 إلى الشاشة، ويحضر حقل ONORDER من INVEN4 إلى الشاشة.

وتطبق كل مؤشرات التنقيح المعتادة، والاختلاف الوحيد هو فى طريقة استدعائك الحقول المختلفة من ملفاتها المختلفة لتنقيحها.

بينما تكون فى حالة التنقيح.. لا تنس أن تغلق ترتيب الاتصال.

```
.CLOSE ALL
```

الفصل الخامس عشر

أمـر التصفـح

BROWSE COMMAND

على عكس ما يدل اسم هذا الأمر.. فهو صيغة قوية جداً من أمر التنقيح.

يمكنك بهذا الأمر أن تنقح حتى من 17 إلى 19 سجلاً فى نفس الوقت، وذلك طبقاً لاختيارك، ويظهر كل سجل أفقياً عبر الشاشة؛ فإذا كان طول السجل أكبر من عرض الشاشة.. فلن تظهر بقيته، ودائماً ما يقدم سطر بقضيب الحالة.

ملاحظة : عند إصدار أمر BROWSE يظهر dBASE حتى 17 سجلاً على الشاشة بدأ بالسجل الحالى المشار إليه؛ أى إنك إذا أردت تصفح قاعدة البيانات من القمة إلى القاعدة.. فيجب أن تكتب مايلى :

```
.GO TOP
BROWSE
```

والصيغة الشاملة لأمر BROWSE هى كما يلى :

```
.BROWSE  [FIELDS      <field-list> [/R] [/<width>,]
                      <Calculated-field-name = expression>,]
         [WIDTH       <expression-N>]
         [WINDOW      <window-name>]
         [COMPRESS]
         [FORMAT]
         [LOCK]
         [FREEZE]
         [NOINIT]
         [NOAPPEND]
         [NOMENU]
         [NOEDIT]
         [NODELETE]
         [NOCLEAR]
         [NOFOLLOW]
```

وفيما يلى بعض الأمثلة :

```
. 1    <cr>
. BROWSE    <cr>

  Records      Fields      Go To      Exit                          11:28:56 a

EMP_NUMEMP_NAME      TOWN      ORGEXEDT_OF_HIRESALARY   NOTE

005     NINA BHARUCHAWEBSTER  BSGT  05/24/80   25000.00MEMO
010     PETE JOHNSON brighton BSGT  02/03/76   27590.00MEMO
015     GLORIA PATEL FAIRPORT RMGT  07/16/82   27500.00memo
020     MAX LEVINSKY HENRIETTARMGF  04/13/69   27550.00MEMO
025     KIM BRANDT   FAIRPORT RMGF  04/04/77   36000.00MEMO
030     TIM MONTAL   ROCHESTERRBGF  07/07/81   41900.00memo
035     WILLIAM PATELpenfield GSDF  08/17/71   28900.00memo
040     JAMES JAMESONROCHESTERGSDT  10/21/77   29800.00memo
045     MORRIS KATZ  webster  BSGF  09/14/80   23450.00memo
050     PAUL BHARUCHABRIGHTON BSGT  05/23/73   29100.00MEMO
055     PHIL MARTIN  WEBSTER  RMGF  07/19/80   31000.00memo
060     JOHN PETERSONBRIGHTON RBGT  04/17/79   31480.00MEMO
065     JOY HARDY    fairport RBGF  01/19/79   34200.00MEMO
070     JAN MOREY    ROXY     GSDT  04/23/67   18190.00memo
075     JOHN JONES   rochesterGSDT  04/04/70   25100.00memo

Browse  C:\...db4\data\PERSNL   Rec 1/15        File                Caps
                        View and edit fields
```

دائماً ما تبدأ نقطة البداية عند أول موقع لأول حقل، كما هو الحال تماماً مع حالة التنقيح IT

يحدد قضيب المسافات أى السجلات الموجود أنت عندها الآن، ويعرف موقع نقطة البداية فى أى مكان داخل السجل، أن السجل هو السجل الحالى.

نظراً لأنه لم تذكر أى قائمة حقل فى الأمر.. فسوف تكون كل السجلات الموجودة فى الهيكل تحت التصفح.

يمكن استخدام مجموعة التحكم التالية مع أمر التصفح BROWSE :

١ – استخدم مفتاح الجدول Tab (أو مفتاح الادخال) للتقدم للأمام حقلاً بحقل داخل السجل. استخدم Shift-Tab للعودة إلى الخلف، من حقل إلى حقل داخل السجل.

٢ – يمكن استخدام البديل Ctrl-V لسمة الادخال داخل الحقل كذلك.

٣ – يمكن استخدام البديل Ctrl-G داخل الحقل كذلك فى حذف الرمز الموجودة نقطة البداية تحته.

٤ – يمكن استخدام مفتاح العودة للخلف داخل الحقل؛ لحذف الرمز الموجود على يسار نقطة البداية.

٥ – عندما تنتهى من إجراء التغييرات على سجل، وليكن السجل رقم 5.. فإن الضغط على مفتاح حركة السهم لأسفل ينقل نقطة البداية عمودياً لأسفل إلى السجل التالى، والضغط على مفتاح حركة السهم لأعلى ينقل نقطة البداية عمودياً لأعلى إلى السجل السابق. وأى من هذين الاجرائين يؤدى إلى حفظ السجل رقم 5.

٦ – عندما تنتهى من إجراء التغييرات على السجل، وليكن السجل رقم 5.. فإن الضغط على مفتاح حركة السهم لأعلى ينقل نقطة البداية عمودياً للسجل السابق، وهذا يحفظ السجل رقم 5 (مثل حالة التنقيح تماماً.

٧ – إذا قررت أن تحذف السجل أثناء وجودك فى السجل.. يمكنك أن تضغط بسهولة على Ctrl-U؛ للاشارة إلى السجل المراد حذفه. لاحظ ظهور مؤشر الحذف فى سطر الحالة. والآن.. عندما تترك السجل.. فإنك تحفظه كسجل يراد حذفه. وفيما بعد إذا وجدت نقطة البداية نفسها فى أى مكان من هذا السجل (المحذوف) مرة أخرى.. يظهر مؤشر الحذف فى سطر الحالة مرة أخرى كذلك؛ مذكراً اياك أن هذا السجل مشار إليه ليحذف. وعند هذه النقطة.. يمكنك بالطبع أن تعيد الضغط على Ctrl-U مرة أخرى؛ للابقاء على السجل دون حذفه.

٨ – بعد انتهائك من إجراء آخر تغيير تريد اجراؤه على السجل.. يمكنك الآن الضغط على Ctrl-W أو على Ctrl-End للخروج من حالة التصفح BROWSE، دون حفظ التغييرات التى أجريت على السجل.

٩ – إذا كان أحد الحقول خارج نطاق الرؤية على الشاشة، وأردت الوصول إليه لتنقيحه.. يمكنك استخدام تأثير الدوران scroll للوصول إلى الحقل.

أ – اضغط على Ctrl-B أو على Ctrl مع مفتاح حركة السهم لليمين أو على F4، ثم أضغط على مفتاح الادخال فيحدث دوران لليمين حقلاً بحقل.

ب - اضغط على Ctrl-Z أو على Ctrl مع مفتاح حركة السهم لليسار أو على F3، ثم اضغط على مفتاح الادخال فيحدث دوران لليسار حقلاً بحقل.

```
BROWSE FIELDS TOWN,ORG,SALARY    <cr>
```

هذا يحد من تأثير BROWSE ليؤثر على الحقول المذكورة فى قائمة الحقل فقط.

```
BROWSE FIELDS TOWN /R, ORG, SALARY    <cr>
```

لاحظ اشارة R/ عند سجل TOWN؛ إذ تعنى هذه الاشارة القراءة فقط read-only؛ أى إن كل الحقول المسرودة فى قائمة الحقل يتم احضارها إلى الشاشة، إلا أن حقل TOWN يكون حقل قراءة فقط، ولايمكنك أن تجرى أى تعديلات عليه.

```
BROWSE FIELDS TOWN /R, ORG /R, SALARY    <cr>
```

والآن كل من TOWN و ORG للقراءة فقط، ولا تستطيع إجراء تغييرات إلا على بيانات الحقل SALARY فقط.

```
BROWSE FIELDS TOWN /5, ORG, SALARY, DT_OF_HIRE /R    <cr>
```

لاحظ المؤشر 5/ عند حقل TOWN، وهذا يحد من عرض بيانات المدينة لتشغل 5 خانات فقط.

يجب ألا يقل مؤشر عرض العمود عن 4 رموز بالنسبة إلى الحقول الحرفية، ولا يقل عن 8 رموز بالنسبة إلى الحقول العددية وحقول التاريخ، وأقصى حجم هو 100؛ فإذا لم يتحدد عرض للعمود.. فإنه يسرى العرض الذى سبق تعريفه فى الهيكل.

وإذا تحدد هذا المشير لحقل عددى أو لحقل تاريخ.. فيهمل إذا كانت مواصفة العرض أقل من الحجم الفعلى للحقل العددى أو لحقل التاريخ.

```
BROWSE FIELDS TOWN,ORG,SALARY,NEWSAL = SALARY*1.1    <cr>
```

لاحظ مقدمة المخرجات المسماة NEWSAL؛ فهذا مثال لحقل محسوب calculated field. إنك تريد توضيح بيانات المدينة، والمؤسسة، والراتب، وكذلك الراتب بعد زيادته 10%، ومثل هذه الحقول تسمى بالحقول المحسوبة، وكما يمكن أن تتوقع.. فكل الحقول المحسوبة تكون من نوع القراءة فقط.

.BROWSE WIDTH 5 <cr>

لاحظ عدم ذكر أى قائمة حقل، وقد وضعت كل الحقول الموجودة فى الهيكل تحت التصفح.

لاحظ مؤشر WIDTH؛ إذ إنه يسرى على الحقول العددية فقط. وإذا ذكرت قائمة حقل.. فسوف تسرى مواصفة WIDTH على كل حقل حرفى مذكور فى قائمة الحقل، ويكون عرض الحقل الحرفى فى المخرجات عبارة عن أقل قيمة من هذه القيم الثلاث : العرض المذكور فى الهيكل، أو عرض العمود المحدد بمؤشر N/ أو بمؤشر WIDTH الموجود فى الأمر.

مثال ذلك، فى الأمر التالى :

.BROWSE FIELDS TOWN /5, ORG, SALARY WIDTH 7 <cr>

يكون العرض الذى يظهر لحقل TOWN عبارة عن 5 خانات؛ حيث إن الرقم 5 هو أقل قيمة محددة من : العرض المحدد فى الهيكل (9)، أو عرض العمود فى المؤشر 5/ وهو (5)، أو مؤشر WIDTH فى الأمر وهو (7).

ويكون العرض الظاهر للحقل ORG هو 3؛ حيث إن الرقم 3 هو أقل قيمة من القيم الثلاثة : القيمة الموجودة فى عرض الهيكل (3)، وقيمة مؤشر WIDTH وهى (7). لاحظ أنك لم تحدد عرضاً خاصاً لعمود ORG فى الأمر السابق.

تأخذ الحقول العددية وحقول التاريخ أقل عرض عمود، يتطلبه الهيكل الفعلى للحقول، ولا تتأثر الحقول المنطقية أو حقول المذكرات بمؤشر WIDTH.

يجب أن تكون مواصفة WIDTH أربعة رموز على الأقل، ولا تزيد عن 100.

.BROWSE FIELDS TOWN.ORG.SALARY WINDOW TINY <cr>

العمل باستخدام نوافذ windows مشروح بالتفصيل فى فصل 38، فى الجزء الثانى من الكتاب ونعرف النافذة حالياً بأنها جزء من الشاشة، ويمكن بالطبع أن تكون النافذة شاغلة الحجم الكلى للشاشة.

افرض أن لديك نافذة اسمها TINY معرفة كجزء من الشاشة، الذى له الإحداثيات 2,10 و7,50، وأنك تريد تحديد شاشة التصفح لهذه النافذة. ينشط الأمر السابق التصفح فى النافذة المعرفة، ويكمن تأثير ذلك فى مقدرة المستفيد على رؤية 5 سجلات فقط فى حالة تصفح، وذلك بسبب أن الاحداثيات تقع من السطر الثانى حتى السطر السابع.

.BROWSE COMPRESS <cr>

عادة ما يجعلك BROWSE ترى حتى 17 سجلاً على الشاشة فى نفس الوقت. وباستخدام المؤشر COMPRESS.. يمكنك أن ترى حتى 19 سجلاً على الشاشة فى نفس الوقت.

BROWSE FORMAT <cr>

سبق أن رأيت كيفية تنشيط ملف تشكيل للشاشة واستخدامه فى حالتى APPEND و EDIT؛ بحيث يمكنك الحصول على شاشة أفضل وأكثر فائدة، عن الشاشات التقليدية التى يقدمها dBASE.

فى أمر BROWSE إذا استخدمت ملف تشكيل شاشة نشطاً.. فلن تحصل على نفس التأثير تماماً. مثال ذلك :

```
SET FORMAT TO LAYOUTA <cr>
BROWSE <cr>
```

لا تعطيك شاشة BROWSE طريقاً لتخطيط الشاشة كما هو معرف فى ملف التشكيل، وبدلاً من ذلك.. تنشط بدائل GET فقط، ويظل مظهر شاشة BROWSE كما هو دون تغيير. والآن أثناء تنقيح البيانات الفعلية.. تسرى قواعد التنقيح كما هى معرفة فى محتويات GET؛ أى إنك قد تكون حددت فى ملف تخطيط الشاشة أن بعض الحقول يجب أن تحتوى على حالة عليا فقط، ويمكنك كذلك أن تحدد أن بعض الحقول يجب أن تكون عددية فى مواقع محددة فقط من الحقل.. الخ، وهذه هى الخواص التى تنشط أثناء BROWSE.

.BROWSE LOCK 3 <cr>

يحدد مؤشر BLOCK عدد الحقول التى يجب إغلاقها أثناء عملية الدوران scroll، وعادة ماتضغط على F4 للدوران لليمين حقلاً بحقل، وتضغط على F3 للدوران لليسار حقلاً بحقل.

يحدد المثال السابق أن أول ثلاثة حقول من الهيكل يجب أن تغلق أثناء عمليات الدوران؛ أى إن هذه الحقول تظهر دائماً على الشاشة أثناء عمليات الدوران، ولاتدور إلا الحقول الأخرى فقط.

```
BROWSE FIELDS EMP_NAME,ORG,SALARY,TOWN,EXEMPT LOCK 2      <cr>
```

فى المثال السابق... نظراً لذكر قائمة الحقل.. فإن أول حقلين سبق ذكرهما فى قائمة الحقل، هما الحقلان اللذان يغلقان الآن عن أى دوران فعلى.

يجب ملاحظة أن الدوران يحدث إذا كانت هناك بعض الحقول، التى تقع خارج نطاق الرؤية على الشاشة أو فى النافذة المعرفة فقط... فإذا كانت الشاشة أو النافذة الحالية كافية؛ لتحتوى على الهيكل كله، أو على قائمة الحقل.. فلا يمكن أن يحدث أى دوران.

```
BROWSE FIELDS EMP_NAME,TOWN ORG,SALARY FREEZE SALARY      <cr>
```

يحدد مؤشر FREEZE إلى أى حقل يتقيد إجراء التصفح. وفى المثال السابق.. تظهر كل الحقول المحددة فى قائمة الحقل على الشاشة، إلا أن إجراء التنقيح الفعلى يتقيد بحقل SALARY. وهذه الصورة مفيدة، عندما تريد تنقيح حقلاً واحداً فقط، كما تريد فى - نفس الوقت - ظهور حقول أخرى على الشاشة للرجوع إليها.

لاحظ أن النتيجة السابقة يمكن الوصول إليها بواسطة الأمر التالى كذلك :

```
BROWSE FIELDS EMP_NAME /R, TOWN /R, ORG /R, SALARY      <cr>
```

لقد جعلت كل الحقول مقروءة فقط - فيما عدا حقل SALARY - وذلك للحصول على نفس التأثير. وتقدم الصيغة السابقة أمراً أصغر فى حجمه.

المؤشرات الأخرى : NOINIT و NOAPPEND و NOMENU و NOEDIT و NODELETE و NOCLEAR و NOFOLLOW سبق توضيحها فى مؤشرات أمر EDIT.

استخدام Shift-F8 Using Shift-F8 (Ditto)

يمكنك – وأنت فى حالة التصفح – أن تستخدم الضغط على Shift-F8 فى أى حقل، وذلك لنسخ البيانات الموجودة فى الحقل المناظر له فى السجل السابق فيه، أى فى الحقل الحالى.

الإنتقال من التنقيح إلى التصفح والعكس

Switching from Edit to Browse

يمكنك الضغط على مفتاح F2 للانتقال المتكرر بين شاشات التنقيح والتصفح، وذلك بغرض تنقيح البيانات؛ أى إنك عندما تكون فى حالة التنقيح.. تضغط على F2 للانتقال إلى حالة التصفح، وعندما تكون فى حالة التصفح.. فإنك تضغط على F2 للعودة إلى حالة التنقيح وهكذا.

كلمة عن الدوران A Word on Scrolling

يشير الدوران فى بيئة تشغيل البيانات إلى الحركة الظاهرية (رأسياً أو أفقياً) للبيانات عبر شاشة الموجه.

وعليك أن تتخيل فى عملية الدوران أن البيانات ثابتة، وأن الشاشة تتحرك عبر البيانات؛ فإذا انتقلت الشاشة إلى اليسار.. يكون قد حدث دوران لليسار. وإذا انتقلت الشاشة لأعلى.. يكون قد حدث دوران لأعلى و... هكذا. فى الواقع لا تنتقل الشاشة بالطبع إلى أى مكان، وعندما يحدث دوران لليسار – أى عندما تنتقل الشاشة لليسار – تبدو البيانات كما لو كانت تحركت لليمين! وبالمثل... يحدث دوران لأعلى إذا تحركت البيانات لأسفل فقط و...هكذا. قد تكون تعريفات الحركة لليسار واليمين ولأعلى ولأسفل مثاراً للخلط.

إذا التصقت باصطلاح ثبات البيانات وحركة الشاشة عبرها.. يصبح للدوران لليسار أو لأعلى أو لغيرها معنى محدد، كما سبق توضيحه تماماً. وفى هذا الاصطلاح... يعنى الدوران لليسار أن البيانات تتحرك لليمين، والعكس صحيح.

التصفح خلال عديد من قواعد البيانات فى نفس الوقت
Browsing from Multiple Databases Simultaneously

كما سبق رؤيته لأوامر العرض والتنقيح.. يمكنك أن تفتح عديداً من الملفات فى مناطق عمل مختلفة وتربطها مع بعضها البعض طبقاً لأرقام السجلات. وتستخدم نفس العملية هنا فى تصفح عديد من الملفات، فى عديد من مناطق العمل.

تستخدم نفس ملفات المخزون، التى سبق استخدامها من قبل، والمسماة INVEN1, IN-VEN2, INVEN3, INVEN4. ويمكنك عند هذه النقطة... أن تفترض أن هذه الملفات بكل منها سجل مناظر لسجل فى بقية الملفات؛ طبقاً لأرقام السجلات، كما سبق توضيحه.

```
.SELE 1
.USE INVEN1
.SELE 2
.USE INVEN2
.SELE 3
.USE INVEN3
.SELE 4
.USE INVEN4
```

تفتح العبارات السابقة الملفات فى أربع مناطق عمل مختلفة، ويمكنك الآن أن تربط هذه الملفات مع بعضها البعض؛ طبقاً لأرقام السجلات.

```
SELE 1  SET RELATION TO RECNO( ) INTO INVEN2, RECNO( ) INTO INVEN3, RECNO( ) INTO INVEN4
```

عند هذه النقطة.. تكون الملفات مرتبطة مع بعضها البعض طبقاً لأرقام السجلات، وما يزال الملف INVEN1 هو ملف التحكم، وتستطيع أن تقدم أمر التنقيح الآن على النحو التالى:

```
BROWSE FIELDS PART_NUM, B->CGC, C->ONHAND, D->ONORDER
```

لاحظ أنك استخدمت اسماء مستعارة للاشارة إلى INVEN2, INVEN3, INVEN4.

يحضر أمر التصفح السابق إلى الشاشة حقل PART_NUM من ملف INVEN1، وحقل CGC من ملف INVEN2، وحقل ONHAND من INVEN3، وحقل ONORDER من INVEN4.

الفصل السادس عشر

أمـر الحـذف

DELETE COMMAND

يستخدم هذا الأمر للاشارة إلى سجل واحد أو أكثر بأنه محذوف منطقياً. والاشارة هى نجمة توضع فى آخر خانة، يقدمها dBASE (خانة أو حقل مؤشر الحذف) عند انتاجك الهيكل لأول مرة؛ مستخدماً نظام dBASE. وهذا الحذف المنطقى – على عكس الحذف الواقعى – يوفر لك ميزة أنك تستطيع تغيير رأيك، فى حالة ما إذا قررت أن توقف تنشيط بعض أو كل السجلات المراد حذفها.

والصيغة الشاملة لأمر الحذف هى ما يلى :

.DELETE [scope] [FOR <condition>] [WHILE <condition>] <cr>

المؤشرات اختيارية. وفى حالة وجود أمر حذف دون مؤشرات... يحذف السجل الحالى المشار إليه بواسطة dBASE.

ملاحظة : فى مجموعة الأمثلة التى تلى ذلك، تكون كل مجموعة عبارات مستقلة عن غيرها من المجموعات؛ أى إنك تستطيع أن تفترض أن النسخة الأخيرة من سجلات قاعدة البيانات تكون متاحة لكل مجموعة من التعليمات.

حاول عمل ما يلى :

```
.5    <cr>
.DELETE    <cr>
.LIST
```

```
Record#  EMP_NUM EMP_NAME       TOWN       ORG EXE DT_OF_HIRE   SALARY NOTE
      1  005     NINA BHARUCHA  WEBSTER    BSG .T. 05/24/80   25000.00 MEMO
      2  010     PETE JOHNSON   brighton   BSG .T. 02/03/76   27590.00 MEMO
      3  015     GLORIA PATEL   FAIRPORT   RMG .T. 07/16/82   27500.00 memo
      4  020     MAX LEVINSKY   HENRIETTA  RMG .F. 04/13/69   27550.00 MEMO
      5 *025     KIM BRANDT     FAIRPORT   RMG .F. 04/04/77   36000.00 MEMO
      6  030     TIM MONTAL     ROCHESTER  RBG .F. 07/07/81   41900.00 memo
      7  035     WILLIAM PATEL  penfield   GSD .F. 08/17/71   28900.00 memo
      8  040     JAMES JAMESON  ROCHESTER  GSD .T. 10/21/77   29800.00 memo
      9  045     MORRIS KATZ    webster    BSG .F. 09/14/80   23450.00 memo
     10  050     PAUL BHARUCHA  BRIGHTON   BSG .T. 05/23/73   29100.00 MEMO
     11  055     PHIL MARTIN    WEBSTER    RMG .F. 07/19/80   31000.00 memo
     12  060     JOHN PETERSON  BRIGHTON   RBG .T. 04/17/79   31480.00 MEMO
     13  065     JOY HARDY      fairport   RBG .F. 01/19/79   34200.00 MEMO
     14  070     JAN MOREY      ROXY       GSD .T. 04/23/67   18190.00 memo
     15  075     JOHN JONES     rochester  GSD .T. 04/04/70   25100.00 memo
```

إذا تذكرت.. فإن مؤشر المدى يمكن أن يكون واحداً من :

ALL RECOn NEXTn

حاول عمل مايلى :

```
.5
.DELE NEXT 5
.LIST
```

Record#	EMP_NUM	EMP_NAME	TOWN	ORG	EXE	DT_OF_HIRE	SALARY	NOTE
1	005	NINA BHARUCHA	WEBSTER	BSG	.T.	05/24/80	25000.00	MEMO
2	010	PETE JOHNSON	brighton	BSG	.T.	02/03/76	27590.00	MEMO
3	015	GLORIA PATEL	FAIRPORT	RMG	.T.	07/16/82	27500.00	memo
4	020	MAX LEVINSKY	HENRIETTA	RMG	.F.	04/13/69	27550.00	MEMO
5	*025	KIM BRANDT	FAIRPORT	RMG	.F.	04/04/77	36000.00	MEMO
6	*030	TIM MONTAL	ROCHESTER	RBG	.F.	07/07/81	41900.00	memo
7	*035	WILLIAM PATEL	penfield	GSD	.F.	08/17/71	28900.00	memo
8	*040	JAMES JAMESON	ROCHESTER	GSD	.T.	10/21/77	29800.00	memo
9	*045	MORRIS KATZ	webster	BSG	.F.	09/14/80	23450.00	memo
10	050	PAUL BHARUCHA	BRIGHTON	BSG	.T.	05/23/73	29100.00	MEMO
11	055	PHIL MARTIN	WEBSTER	RMG	.F.	07/19/80	31000.00	memo
12	060	JOHN PETERSON	BRIGHTON	RBG	.T.	04/17/79	31480.00	MEMO
13	065	JOY HARDY	fairport	RBG	.F.	01/19/79	34200.00	MEMO
14	070	JAN MOREY	ROXY	GSD	.T.	04/23/67	18190.00	memo
15	075	JOHN JONES	rochester	GSD	.T.	04/04/70	25100.00	memo

لاحظ أن السجلات من رقم 5 إلى رقم 9 مشار إليها بأنها محذوفة.

```
.DELE FOR TOWN = 'ROCH'
```

كما هو الحال دائماً.. إذا لم يتحدد مؤشر المدى مع وجود شرط FOR تكون ALL هى الحالة التقليدية للمدى.

```
.LIST
```

Record#	EMP_NUM	EMP_NAME	TOWN	ORG	EXE	DT_OF_HIRE	SALARY	NOTE
1	005	NINA BHARUCHA	WEBSTER	BSG	.T.	05/24/80	25000.00	MEMO
2	010	PETE JOHNSON	brighton	BSG	.T.	02/03/76	27590.00	MEMO
3	015	GLORIA PATEL	FAIRPORT	RMG	.T.	07/16/82	27500.00	memo
4	020	MAX LEVINSKY	HENRIETTA	RMG	.F.	04/13/69	27550.00	MEMO
5	025	KIM BRANDT	FAIRPORT	RMG	.F.	04/04/77	36000.00	MEMO
6	*030	TIM MONTAL	ROCHESTER	RBG	.F.	07/07/81	41900.00	memo
7	035	WILLIAM PATEL	penfield	GSD	.F.	08/17/71	28900.00	memo
8	*040	JAMES JAMESON	ROCHESTER	GSD	.T.	10/21/77	29800.00	memo
9	045	MORRIS KATZ	webster	BSG	.F.	09/14/80	23450.00	memo
10	050	PAUL BHARUCHA	BRIGHTON	BSG	.T.	05/23/73	29100.00	MEMO
11	055	PHIL MARTIN	WEBSTER	RMG	.F.	07/19/80	31000.00	memo
12	060	JOHN PETERSON	BRIGHTON	RBG	.T.	04/17/79	31480.00	MEMO
13	065	JOY HARDY	fairport	RBG	.F.	01/19/79	34200.00	MEMO
14	070	JAN MOREY	ROXY	GSD	.T.	04/23/67	18190.00	memo
15	075	JOHN JONES	rochester	GSD	.T.	04/04/70	25100.00	memo

```
.DELE FOR SUBSTR(TOWN,2,3) = 'OCH'
.LIST
```

Record#	EMP_NUM	EMP_NAME	TOWN	ORG	EXE	DT_OF_HIRE	SALARY	NOTE
1	005	NINA BHARUCHA	WEBSTER	BSG	.T.	05/24/80	25000.00	MEMO
2	010	PETE JOHNSON	brighton	BSG	.T.	02/03/76	27590.00	MEMO
3	015	GLORIA PATEL	FAIRPORT	RMG	.T.	07/16/82	27500.00	memo
4	020	MAX LEVINSKY	HENRIETTA	RMG	.F.	04/13/69	27550.00	MEMO
5	025	KIM BRANDT	FAIRPORT	RMG	.F.	04/04/77	36000.00	MEMO
6	*030	TIM MONTAL	ROCHESTER	RBG	.F.	07/07/81	41900.00	memo
7	035	WILLIAM PATEL	penfield	GSD	.F.	08/17/71	28900.00	memo
8	*040	JAMES JAMESON	ROCHESTER	GSD	.T.	10/21/77	29800.00	memo
9	045	MORRIS KATZ	webster	BSG	.F.	09/14/80	23450.00	memo
10	050	PAUL BHARUCHA	BRIGHTON	BSG	.T.	05/23/73	29100.00	MEMO
11	055	PHIL MARTIN	WEBSTER	RMG	.F.	07/19/80	31000.00	memo
12	060	JOHN PETERSON	BRIGHTON	RBG	.T.	04/17/79	31480.00	MEMO
13	065	JOY HARDY	fairport	RBG	.F.	01/19/79	34200.00	MEMO
14	070	JAN MOREY	ROXY	GSD	.T.	04/23/67	18190.00	memo
15	075	JOHN JONES	rochester	GSD	.T.	04/04/70	25100.00	memo

```
.DELE FOR ORG = 'BSG' .AND. DT_OF_HIRE > = CTOD('09/14/80')
.LIST
```

Record#	EMP_NUM	EMP_NAME	TOWN	ORG	EXE	DT_OF_HIRE	SALARY	NOTE
1	005	NINA BHARUCHA	WEBSTER	BSG	.T.	05/24/80	25000.00	MEMO
2	010	PETE JOHNSON	brighton	BSG	.T.	02/03/76	27590.00	MEMO
3	015	GLORIA PATEL	FAIRPORT	RMG	.T.	07/16/82	27500.00	memo
4	020	MAX LEVINSKY	HENRIETTA	RMG	.F.	04/13/69	27550.00	MEMO
5	025	KIM BRANDT	FAIRPORT	RMG	.F.	04/04/77	36000.00	MEMO
6	030	TIM MONTAL	ROCHESTER	RBG	.F.	07/07/81	41900.00	memo
7	035	WILLIAM PATEL	penfield	GSD	.F.	08/17/71	28900.00	memo
8	040	JAMES JAMESON	ROCHESTER	GSD	.T.	10/21/77	29800.00	memo
9	*045	MORRIS KATZ	webster	BSG	.F.	09/14/80	23450.00	memo
10	050	PAUL BHARUCHA	BRIGHTON	BSG	.T.	05/23/73	29100.00	MEMO
11	055	PHIL MARTIN	WEBSTER	RMG	.F.	07/19/80	31000.00	memo
12	060	JOHN PETERSON	BRIGHTON	RBG	.T.	04/17/79	31480.00	MEMO
13	065	JOY HARDY	fairport	RBG	.F.	01/19/79	34200.00	MEMO
14	070	JAN MOREY	ROXY	GSD	.T.	04/23/67	18190.00	memo
15	075	JOHN JONES	rochester	GSD	.T.	04/04/70	25100.00	memo

كما هو الحال دائماً... يمكن أن يكون الشرط بسيطاً أو مركباً كما تحب، وتذكر أن الدالة CTOD ('09/14/80') يمكن أن تحدد – كذلك – على النحو التالي : (09/14/80).

```
.1      <cr>
.DELE WHILE TOWN = 'WEB'      <cr>
```

يحذف الأمر السابق السجل رقم 1 فقط؛ لأن السجل رقم 2 ينفى حدوث الشرط.

لحذف كل السجلات.. ابدأ بسجل معين :

```
.11     <cr>
.DELE REST     <cr>
```

يحذف هذا بقية الملف بدءًا من السجل الحالى، مع حذف السجل الحالى كذلك.

سرد السجلات المحذوفة List Deleted Records

```
.5
.DELE NEXT 5
```

كما ترى... يحذف الأمر السابق السجلات من السجل الخامس، حتى السجل التاسع.

يعطيك الأمر التالى قائمة على الشاشة بكل السجلات المحذوفة فقط، وتستخدم دالة ( ) DELETED هنا.

```
.LIST FOR DELETED( )
```

Record#	EMP_NUM	EMP_NAME	TOWN	ORG	EXE	DT_OF_HIRE	SALARY	NOTE
5	*025	KIM BRANDT	FAIRPORT	RMG	.F.	04/04/77	36000.00	MEMO
6	*030	TIM MONTAL	ROCHESTER	RBG	.F.	07/07/81	41900.00	memo
7	*035	WILLIAM PATEL	penfield	GSD	.F.	08/17/71	28900.00	memo
8	*040	JAMES JAMESON	ROCHESTER	GSD	.T.	10/21/77	29800.00	memo
9	*045	MORRIS KATZ	webster	BSG	.F.	09/14/80	23450.00	memo

سرد السجلات النشطة List Active Records

تسرد هذه الصيغة من أمر LIST السجلات النشطة فقط.

```
LIST FOR .NOT. DELETED( )     <cr>
```

Record#	EMP_NUM	EMP_NAME	TOWN	ORG	EXE	DT_OF_HIRE	SALARY	NOTE
1	005	NINA BHARUCHA	WEBSTER	BSG	.T.	05/24/80	25000.00	MEMO
2	010	PETE JOHNSON	brighton	BSG	.T.	02/03/76	27590.00	MEMO
3	015	GLORIA PATEL	FAIRPORT	RMG	.T.	07/16/82	27500.00	memo
4	020	MAX LEVINSKY	HENRIETTA	RMG	.F.	04/13/69	27550.00	MEMO
10	050	PAUL BHARUCHA	BRIGHTON	BSG	.T.	05/23/73	29100.00	MEMO
11	055	PHIL MARTIN	WEBSTER	RMG	.F.	07/19/80	31000.00	memo
12	060	JOHN PETERSON	BRIGHTON	RBG	.T.	04/17/79	31480.00	MEMO
13	065	JOY HARDY	fairport	RBG	.F.	01/19/79	34200.00	MEMO
14	070	JAN MOREY	ROXY	GSD	.T.	04/23/67	18190.00	memo
15	075	JOHN JONES	rochester	GSD	.T.	04/04/70	25100.00	memo

ملاحظة : يبين الأمر DISP STRU، أو الدالة ( ) RECCOUNT عدد السجلات الموجودة حالياً في الملف. ويشمل هذا العد عدد السجلات المحذوفة !

اشارة وضع الحذف على The "Set Dele On" Flag

ملاحظة : هناك أمر قوى جداً يطلب من dBASE أن يهمل كل السجلات المحذوفة وهو أمر SET DELE TO.

```
.SET DELE ON
.LIST
```

يضع أول أمر مؤشر يخطر dBASE بأن يهمل كل السجلات المحذوفة، ويطلب الأمر الثاني سرداً LIST للبيانات. وتأثير ذلك هو أنه من الضرورى تحديد : FOR. LEST. ( ) DELE. NOT. اعمل ملاحظة تذكيرية بأمر SET DELE ON، ومن المهم جداً أن تكون قادراً على إهمال كل السجلات المحذوفة من القوائم والتقارير.

أوامر تذكر واضغط واحذف وادخل

RECALL, PACK, ZAP, INSERT COMMANDS

انظر فصل 20 عن الأوامر المتنوعة Miscellaneous Commands لمناقشة هذه الأوامر.

الفصل السابع عشر

أمـر الاضافـة

APPEND COMMAND

يمكن أن يأخذ أمر الاضافة إحدى الصيغ التالية :

```
.APPE
.APPE BLANK
APPE FROM <file-name> <file-type> FOR <condition>
.APPE FROM ARRAY <array-name> FOR <condition>
APPE MEMO <memo-field-name> FROM <filename> [OVERWRITE]
```

سبق أن درست بالفعل أول صيغة من صيغ APPEND فى الأقسام السابقة.

انظر الآن إلى السمات المتسعة لهذا الأمر؛ فأمر الاضافة APPEND أمر قوى جداً، ويجب أن تدرس تفاصيله.

اضافة فراغ — APPEND BLANK

إحدى صيغ أمر APPEND هى ما يلى :

```
.USE PERSNL      <cr>
APPEND BLANK     <cr>
```

يضيف هذا الأمر سجلاً فارغاً إلى نهاية قاعدة البيانات الموجودة.

```
.LIST     <cr>   [Note record #16]
```

Record#	EMP_NUM	EMP_NAME	TOWN	ORG	EXE	DT_OF_HIRE	SALARY	NOTE
1	005	NINA BHARUCHA	WEBSTER	BSG	.T.	05/24/80	25000.00	MEMO
2	010	PETE JOHNSON	brighton	BSG	.T.	02/03/76	27590.00	MEMO
3	015	GLORIA PATEL	FAIRPORT	RMG	.T.	07/16/82	27500.00	memo
4	020	MAX LEVINSKY	HENRIETTA	RMG	.F.	04/13/69	27550.00	MEMO
5	025	KIM BRANDT	FAIRPORT	RMG	.F.	04/04/77	36000.00	MEMO
6	030	TIM MONTAL	ROCHESTER	RBG	.F.	07/07/81	41900.00	memo
7	035	WILLIAM PATEL	penfield	GSD	.F.	08/17/71	28900.00	memo
8	040	JAMES JAMESON	ROCHESTER	GSD	.T.	10/21/77	29800.00	memo
9	045	MORRIS KATZ	webster	BSG	.F.	09/14/80	23450.00	memo
10	050	PAUL BHARUCHA	BRIGHTON	BSG	.T.	05/23/73	29100.00	MEMO
11	055	PHIL MARTIN	WEBSTER	RMG	.F.	07/19/80	31000.00	memo
12	060	JOHN PETERSON	BRIGHTON	RBG	.T.	04/17/79	31480.00	MEMO
13	065	JOY HARDY	fairport	RBG	.F.	01/19/79	34200.00	MEMO
14	070	JAN MOREY	ROXY	GSD	.T.	04/23/67	18190.00	memo
15	075	JOHN JONES	rochester	GSD	.T.	04/04/70	25100.00	memo
16					.F.	/ /		memo

هذه الصيغة من APPEND ليست مسهبة، كما يمكن أن تبدو، ويمكن إظهار أهمية هذه الصيغة على النحو التالى :

افرض أنك تكتب برنامج كمبيوتر مستخدماً dBASE (الشىء الذى يمكنك عمله من خلال دراستك لهذا الكتاب)، الذى يظهر عند استدعائه شاشة مشكلة، تطلب من مشغل الكمبيوتر إدخال أحد سجلات المعلومات. ويطلب التشكيل الموجود على الشاشة – من مشغل الكمبيوتر– أن يدخل رقم تعريف العامل، واسمه، واسم المنظمة، والراتب على سبيل المثال.

عندما ينتهى ادخال قطع المعلومات المطلوبة... يبدأ البرنامج انتاج سجل بيانات جديد من المعلومات التى ادخلها مشغل الكمبيوتر؛ أى إن البرنامج يضيف سجلاً فارغاً APPEND BLANK إلى قاعدة البيانات، والأكثر أهمية من ذلك... هى وضع البرنامج مشير السجل؛ ليشير إلى السجل الجديد المضاف. ويستطيع البرنامج الآن أن يستبدل الحقول الفارغة المناسبة الموجودة فى السجل الجديد، بمعلومات سبق أن أدخلها مشغل الكمبيوتر، وسوف ترى أمثلة لهذا فيما بعد فى الكتاب.

عند ملقن النقطة.. لا ينفذ استخدام APPEND BLANK نشاطاً مفيداً.

.APPE FROM <file-name> <file-type> FOR <condition>

وبهذا الأمر تستطيع – أساساً – أخذ سجلات من ملف، ونسخها فى ملف آخر.

يمكن أن يكون الملف المنقول منه أحد ملفات dBASE IV، أو أحد ملفات dBASE +III، أو أحد ملفات dBASE II، أو أى ملف من أى من VisiCalc، أو Framework، أو RapidFile أو Lotus، كما يمكن أن يكون كذلك ملفاً، سبق إنتاجه باستخدام أى منقح كلمات نمطى، مثل : الملفات الناتجة عن طريق تشغيل الكلمات.

ملاحظة : يعتبر الملف المنقول منه أنه أحد ملفات dBASE IV، ما لم يذكر خلاف ذلك.

الإضافة من ملفات dBASE IV

APPENDING FROM dBASE IV FILES

```
.USE MASTER      <cr>
.APPEND FROM TRANS      <cr>
```

(لديك ملف اسمه MASTER. DBF، توجد أو لا توجد به سجلات، ولديك ملف آخر اسمه TRANS. DBF به سجلات).

الأمران السابقان يأخذان كل السجلات من ملف TRANS، ويضيفانها APPEND (ينسخانها) إلى نهاية الملف المسمى MASTER

لا تتغير محتويات الملف المنقول منه.

إذا كانت هناك سجلات محذوفة منطقياً فى الملف المنقول منه.. فإنها تنسخ كذلك، ويعاد تنشيطها فى ملف المقصد.

إذا كانت هناك ملفات مذكرات فى الملف المنقول منه.. تنسخ محتويات المذكرات كذلك !

من الواضح أن هذا يشمل أن هيكلى الملفين يجب أن يكونا متطابقين. ماذا يحدث إذا كانت الهياكل متشابهة، ولكنها غير متطابقة؟.

مثال ذلك :

MASTER	TRANS
EMP_NUM,C,6	ORG,C,3
TOWN,C,12	DT_OF_HIRE,N,6
ORG,C,3	EMP_NUM,C,6
SALARY,N,8,2	SALARY,N,8,2

للهياكل هنا بعض التشابه، ولكنها ليست متطابقة. فماذا يحدث إذا كتبت ما يلى :

```
USE MASTER      <cr>
APPEND FROM TRANS      <cr>
```

اسماء الحقول المتطابقة – فقط – هى التى تؤخذ فى الاعتبار فى عملية APPEND.. وتهمل الحقول الأخرى، كما أنه ليس من الضرورى أن تكون للحقول التى لها نفس الأسماء – فى كل من الملفين – نفس الموقع كذلك، وتحدث إعادة تشكيل كاملة إذا كان هناك حاجة لذلك.

ومع نفس القاعدة سالفة الذكر... تنقل محتويات حقول المذكرات، إذا كانت أسمائها متطابقة فقط.

فى هذا المثال.. نظراً لتوافق أسماء الحقول EMP _ NUM، و ORG، و SALARY تنقل بيانات هذه الحقول فقط إلى هيكل الملف MASTER، وذلك لكل سجل من السجلات. ويظل العمود المعرف أنه حقل TOWN فى ملف MASTER فارغاً لكل سجل من السجلات الجديدة، التى تضاف من ملف TRANS. ولا تنسخ أى بيانات من حقل DT _ OF _ HIRE الموجود فى ملف العمليات الجارية إلى الملف الرئيسى، ومن الواضح أن السجلات الموجودة فى الملف الرئيسى – قبل بدء عملية الاضافة – لاتتأثر.

انتج قاعدة بيانات أخرى، تسمى STUDENTS على النحو التالى :

ادخل أى بيانات تريدها فى هذا الملف.

انتج 4 سجلات بيانات على الأقل.

```
EMP_NUM,C,3
EMP_NAME,C,13
SALARY,N,8,2
```

```
USE PERSNL      <cr>
APPEND FROM STUDENTS      <cr>
LIST      <cr>
```

Record#	EMP_NUM	EMP_NAME	TOWN	ORG	EXE	DT_OF_HIRE	SALARY	NOTE
1	005	NINA BHARUCHA	WEBSTER	BSG	.T.	05/24/80	25000.00	MEMO
2	010	PETE JOHNSON	brighton	BSG	.T.	02/03/76	27590.00	MEMO
3	015	GLORIA PATEL	FAIRPORT	RMG	.T.	07/16/82	27500.00	memo
4	020	MAX LEVINSKY	HENRIETTA	RMG	.F.	04/13/69	27550.00	MEMO
5	025	KIM BRANDT	FAIRPORT	RMG	.F.	04/04/77	36000.00	MEMO
6	030	TIM MONTAL	ROCHESTER	RBG	.F.	07/07/81	41900.00	memo
7	035	WILLIAM PATEL	penfield	GSD	.F.	08/17/71	28900.00	memo
8	040	JAMES JAMESON	ROCHESTER	GSD	.T.	10/21/77	29800.00	memo
9	045	MORRIS KATZ	webster	BSG	.F.	09/14/80	23450.00	memo
10	050	PAUL BHARUCHA	BRIGHTON	BSG	.T.	05/23/73	29100.00	MEMO
11	055	PHIL MARTIN	WEBSTER	RMG	.F.	07/19/80	31000.00	memo
12	060	JOHN PETERSON	BRIGHTON	RBG	.T.	04/17/79	31480.00	MEMO
13	065	JOY HARDY	fairport	RBG	.F.	01/19/79	34200.00	MEMO
14	070	JAN MOREY	ROXY	GSD	.T.	04/23/67	18190.00	memo
15	075	JOHN JONES	rochester	GSD	.T.	04/04/70	25100.00	memo
16	001	A. ADAMS			.F.	/ /	10000.00	memo
17	002	B. BROWN			.F.	/ /	20000.00	memo
18	003	C. COLBY			.F.	/ /	30000.00	memo
19	004	D. DAVIDSON			.F.	/ /	40000.00	memo

سترى فجوات فى قائمة الملف PERSNL للسجلات المضافة حديثاً، بينما تترك أسماء الحقول غير المتطابقة (بين ملف PERSNL وملف STUDENTS)، دون أن تدخلها بيانات من ملف STUDENTS.

ويظهر المقطع السابق استخداماً مهماً جداً لأمر APPEND.

قل - على سبيل المثال - إنك تعرف أنه يوجد ملف رئيسى فى قاعدة البيانات، وإنك تريد استخدام جزءاً من معلوماته فى أحد استخداماتك الخاصة. كل ما أنت فى حاجة إليه، هو انتاج هيكل الملف الذى تريده؛ متأكداً أنك تستخدم الأسماء المتطابقة، وكذلك نفس أنواع وأحجام الحقول، كما هى معرفة فى هيكل الملف الرئيسى.

افرض أن الملف الذى انتجته اسمه MYFILE.

تنقل بيانات كل اسماء الحقول المتطابقة، وبهذا تكون قد نسخت جزءا من المعلومات من ملف MASTER إلى ملفك. (استخدم أمر LIST لتتأكد من صحة ذلك.)

فيما بعد... عندما تدرس أمر COPY.. فإنك ترى طريقة أكثر سهولة لعمل نفس الشىء : انتاج جزء من المعلومات من أحد الملفات الرئيسية.

يظهر السؤال : ماذا يحدث إذا توافقت أسماء الحقول، مع اختلاف فى النوع والحجم لها؟

مثال ذلك :

ماذا يحدث إذا حاولت اضافة APPEND بيانات من حقل معرف، على أنه EMP-NUM, C, 6 إلى حقل معرف على أنه EMP_NUM, N, 5؟

الاجابة هى الحصول على نتائج غير متوقعة، وذلك لعدم تشابه النوع والحجم.

فى نهاية هذا القسم عن APPEND.. يمكنك أن تغطى هذا النوع من أسئلة «ماذا... إذا؟»، وذلك لأن أمر APPEND من أوجه dBASE المهمة، وتحتاج إلى تقدير التأثير الكلى انتاج الاضافى APPEND بحقول غير متطابقة.

وأركز الآن على الحقيقة بأنه يجب أن تتوافق أسماء الحقول لأى حركة بيانات، تحدث أثناء عملية APPEND. (والحالة المثالية هى أن أحجام وأنواع الحقول تتشابه كذلك.)

الإضافة طبقاً لشرط APPENDING FOR A CONDITION

قد يكون لديك شرط موجود لتعزيز الفائدة من APPEND، مثال ذلك :

```
USE PERSNL      <cr>
APPEND FROM STUDENTS FOR SALARY > 20000      <cr>
```

والآن السجلات التى تحقق الشرط فقط، هى التى تنقل من ملف STUDENTS، ومن هذه السجلات لا تنسخ إلا البيانات من الحقول المتوافقة فقط. ومن الواضح أنه يجب أن يحتوى هيكل ملف STUDENTS على حقل اسمه SALARY.

ملاحظة :

يجب أن تتواجد الحقول المستخدمة فى تعبير FOR فى هيكل كل من قاعدتى البيانات! ونظراً لأن حقل SALARY استخدم فى شرط FOR.. يجب أن تحتوى هياكل كل من قاعدتى البيانات المنقول منها والمنقول إليها على تعريف لحقل SALARY، ومن الأفضل أن يكون تعريف هذا الحقل متشابهاً تماماً فى كل من قاعدتى البيانات.

والآن.. يجب تعريف كل من SALARY و ORG فى هيكلى قاعدتى البيانات، وهذا صحيح حتى إذا كان لديك مؤثر .OR. كمؤثر منطقى !

إذا حاولت أن تضيف APPEND من هيكلين مختلفين تماماً... فلا تنقل أى سجلات على الإطلاق.

الإضافة من ملف آخر غير ملفات dBASE IV

APPENDING FROM A NON-dBASE IV FILE

```
USE PERSNL
APPE FROM REGISTER DBASEII
```

ينقل هذا البيانات من قاعدة بيانات dBASE II اسمها REGISTER. dbf، وتأكد أن أسماء الحقول متوافقة، كما أن أحجامها وأنواعها يجب أن تكون متشابهة تماماً.

APPE FROM REGISTER DIF

يشير هذا إلى تشكيل ملف VisiCalc، وتتحول صفوف VisiCalc إلى سجلات، وتتحول أعمدتها إلى حقول، وذلك في ملف dBASE IV.

APPE FROM REGISTER FW2

يشير هذا إلى REGISTER كقاعدة بيانات في Framework II، أو في إحدى صفحات الانتشار.

.APPE FROM REGISTER RPD

يشير هذا إلى تشكيل ملف بيانات من RapidFile.

APPE FROM REGISTER SYLK

يشير هذا إلى تشكيل صفحة انتشار MultiPlan في الترتيب الرئيسي للصفوف، وتتحول صفوف MultiPlan إلى سجلات، وتتحول الأعمدة إلى حقول، ولا يعمل هذا إذا كان ملف MultiPlan في الترتيب الرئيسي للأعمدة.

.APPE FROM REGISTER WKS

يشير هذا إلى تشكيل الصيغة IA من صفحة انتشار Lotus 123، وتتحول صفوف Lotus إلى سجلات، وتتحول الأعمدة إلى حقول. ويبدأ الملف بالخلية الموجودة بالركن العلوي الأيسر من صفحة الانتشار، ويجب أن تكون صفحة الانتشار مرتبة بالترتيب الرئيسي للصفوف، وليس بالترتيب الرئيسي للأعمدة.

ملاحظة :

بالنسبة لملفات صفحات الانتشار المستخدمة مع dBASE (صفحات الانتشار المستوردة importing spreadsheets)... يجب أن تكون البيانات في الترتيب الرئيسي

للصفوف، وليس فى الترتيب الرئيسى للأعمدة، كما يجب حذف عناوين الأعمدة. كما أنه فى حالة ملفات Lotus 123.. يجب ألا توجد فى الملفات صفوف فارغة رائدة، أو أعمدة فارغة رائدة؛ أى إنه إذا كانت هناك مثل هذه الصفوف أو الأعمدة الفارغة.. يجب أن تضبط بياناتك فى الركن العلوى الأيسر، قبل استخدام APPEND.

ملفات نصية (أو ملفات ASCII أو SDF)

TEXT / ASCII / SDF FILES

يسمح dBASE باستيراد ملفات فى شكل نص text إلى تشكيل dBASE، ومن الضرورى قبل أن أوضح عمل ذلك أن يكون مفهوماً لديك معنى نص text بوضوح؛ لأن هناك أشكالاً متعددة للملفات النصية.

الملف النصى هو الملف الموجود به بيانات فقط وليس هيكلاً، كما يسمى الملف النصى كذلك بأنه ملف ASCII، أو ملف تشكيل نمطى للبيانات Standard Data Format (SDF). وإحدى طرق انتاج ملف ASCII، هى عن طريق استخدام حالة عدم التوثيق non-document mode لأحد مشغلات الكلمات التجارية.

وهذا يعنى أن TEXT = ASCII = SDF؛ أى إن هذه الاصطلاحات مرادفات لبعضها البعض.

يمكن كذلك تعريف ملف ASCII بأنه الملف الموجود فى مجموعة الرموز المستخدمة، فيه رموز تقع فى المدى السادس عشرى من 00 إلى 7F، وإذا لم تكن هذه الجملة مفهومة.. فلا تضيع وقتاً فى فهمها. وبطريقة تقريبية... اعتبر ملف ASCII أو الملف النصى، أنه ملف قابل للعرض على الشاشة أو للطباعة بواسطة الطابع، وإذا نتج عن محاولتك إظهار محتويات ملف من ملفن DOS، أو من dBASE ظهور أشياء لا معنى لها بها أجراس وصفارات وأوجه ضاحكة على الشاشة.. فلن يكون هذا الملف ملف ANSII. ويطبق نفس النمط إذا لم يظهر الطابع كتابة مقروءة، وطبع بدلاً من ذلك أشكالاً مثل التى سبق وصفها، مثال ذلك :

TYPE <filename> <cr>

إذا ظهرت على الشاشة أشياء مثل التى سبق وصفها؛ فالملف المسمى فى العبارة لا يكون ملفاً نصياً (ملف ANSII).

إذا كان مطلوباً انتاج أول ثلاثة سجلات من ملف PERSNL. DBF – على هيئة ملف SDF معتاد – فسوف تظهر هذه السجلات على النحو التالى :

```
005NINA BHARUCHAWEBSTER  BSGT1980052425000.00
010PETE JOHNSON brighton BSGT1976020327590.00
015GLORIA PATEL FAIRPORT RMGT1982071627500.00
```

لاحظ عدم وجود هيكل واقعى لهذا الملف فى حد ذاته، إلا ما يمكن للمستفيد استدلاله عن الملف. وافتراضياً... فإنك تستطيع أن ترى هيكلاً داخلياً لهذا الملف، وذلك لأنك تستطيع تمييز رقم العامل من اسم العامل... الخ. إلا أنه – دون بعض الاعتياد على البيانات – فلن تعرف أن BSGT هى بيانات المؤسسة BSG، تليها بيانات الاستثناء أو عدم الاستثناء (T فى هذه الحالة) فى الواقع، وقد تعتبر BSGT بسهولة كأحد الرموز المستخدمة.

يشبه ملف SDF المعتاد الملف المبين أعلاه، ويمكن أن تكون لبعض أنواع ملفات SDF محددات؛ أى إن هذه الملفات تكون بها علامات تنصيص، أو فواصل تفصل حقول معلوماتها عن بعضها البعض. يقدم dBASE تسهيلات عديدة لاستيراد بيانات من أى من تشكيلات الملفات النصية (النمطية أو التى لها محددات) إلى تشكيل dBASE.

وفيما يلى.. أمثلة لأنواع ملفات نمطية، معتادة أو بها محددات :

ملف SDF معتاد :

```
Regular SDF.
005NINA BHARUCHAWEBSTER  BSGT1980052425000.00
010PETE JOHNSON brighton BSGT1976020327590.00
015GLORIA PATEL FAIRPORT RMGT1982071627500.00
```

محدد بفواصل، وعلامتى تنصيص مزدوجتين :

```
Delimited with Commas and Double-Quotes.
"005","NINA BHARUCHA","WEBSTER","BSG",T,19800524,25000.00
"010","PETE JOHNSON","brighton","BSG",T,19760203,27590.00
"015","GLORIA PATEL","FAIRPORT","RMG",T,19820716,27500.00
```

محدد بفواصل، وعلامتى تنصيص فرديتين :

```
Delimited with Commas and Single-Quotes.
'005','NINA BHARUCHA','WEBSTER','BSG',T,19800524,25000.00
'010','PETE JOHNSON','brighton','BSG',T,19760203,27590.00
'015','GLORIA PATEL','FAIRPORT','RMG',T,19820716,27500.00
```

محدد بفواصل فقط :

```
Delimited with Commas, only.
,005,,,NINA BHARUCHA,,,WEBSTER,,,BSG,,T,19800524,25000.00
,010,,,PETE JOHNSON,,,brighton,,,BSG,,T,19760203,27590.00
,015,,,GLORIA PATEL,,,FAIRPORT,,,RMG,,T,19820716,27500.00
```

محدد بمواقع فارغة :

```
Delimited with a Blank Space.
005 NINA BHARUCHA WEBSTER BSG T 19800524 25000.00
010 PETE JOHNSON brighton BSG T 19760203 27590.00
015 GLORIA PATEL FAIRPORT RMG T 19820716 27500.00
```

محدد برمز خاص :

```
Delimited with a Special Character.
$005$,$NINA BHARUCHA$,$WEBSTER$,$BSG$,T,19800524,25000.00
$010$,$PETE JOHNSON$,$brighton$,$BSG$,T,19760203,27590.00
$015$,$GLORIA PATEL$,$FAIRPORT$,$RMG$,T,19820716,27500.00
```

يسمح dBASE باستيراد معظم تشكيلات الملفات النصية إلى هيكل قواعد بيانات dBASE.

الإضافة من ملفات SDF معتادة

APPENDING FROM REGULAR SDF FILES

```
.USE PERSNL      <cr>
.APPEND FROM MYFILE SDF      <cr>
```

فى الأمر السابق...SDF عبارة عن مؤشر يعنى تشكيلاً نمطيًا للبيانات STANDARD DATA FORMAT، وهو المؤشر الذى يذكر لنظام dBASE أن الملف المسمى MYFILE،

هو ملف نصى معتاد وليس ملف dBASE؛ أى إن مؤشر SDF يساعد فى تعريف الملف المسمى فى الأمر.

يبحث dBASE الآن عن ملف اسمه MYFILE. TXT (المؤهل TXT هو مؤهل تقليدى للملفات النصية، وإذا وجده.. فإنه يأخذ سجلات منه، وذلك رمزاً رمزاً لكل سجل فى نفس الوقت، ويضيف ببساطة البيانات تحت هيكل قاعدة البيانات المسماة PERSNL، وتأثير ذلك الآن هو أن هيكل قاعدة البيانات يعرف ما تشير إليه حقول البيانات المضافة حديثاً.

من الواضح أنه يتبقى تفسير: أنه إذا ما أريد استيراد ملف نصى إلى تشكيل قواعد بيانات dBASE.. فيجب أن تتأكد من أن البيانات الموجودة فى الملف النصى تناظر الشكل المتوقع التالى لملف dBASE تماماً؛ فإذا لم تستطع اتباع هذه القاعدة.. فلن يعرف dBASE الاختلاف الموجود، ويمكن أن تنهى عملك بالحصول على نفايا كنتائج لهذا العمل.

مثال ذلك :

إذا كان الملف النصى المستخدم به السجل التالى :

My dear so-and-so. How are you?

وإذا حاولت أن تضيف هذا السجل إلى هيكل ملف اسمه PERSNL.. فسوف يأخذ dBASE هذا السجل، ويدخله تحت هيكل PERSNL رمزاً برمز، حتى إنه – بعد الانتهاء من الاضافة – يظهر هذا السجل على النحو التالى : (الرمز يعنى رمزاً فارغاً.)

My^	تصبح EMP_NUM لأنها معرفة بأنها C, 3
dear so-and-s so	تصبح EMP_NAME لأنها معرفة بأنها C, 13
o. How ar	تصبح TOWN لأنها معرفة بأنها C, 9

كما يمكنك أن ترى : يفترض dBASE أن تشكيل الملف النصى المعتاد يتفق تماماً مع هيكل قاعدة البيانات، التى يضاف إليها السجلات النصية، التى تكون كافية منطقياً بالطبع.

ملاحظة : إذا حددت ما يلى :

```
.USE PERSNL
.APPEND FROM MYFILE ABC SDF
```

فإن dBASE يبحث عن ملف نصى له نفس الاسم الذى قدمته بالضبط، ولا تكون بحاجة لأن تظل مع TXT، كمؤهل للاسم الثانوى.

الإضافة من ملفات SDF لها محددات

APPENDING FROM DELIMITED SDF FILES

يذكر وجود مؤشر وجود الحدود delimited parameter فى الأمر لنظام dBASE أن الملف المنقول منه، هو ملف نصى به حدود، وليس ملف dBASE، أو ليس ملفاً نصياً معتاداً؛ أى إن مؤشر DELIM يساعد فى تعريف الملف المسمى فى الأمر.

والملف الذى به حدود، هو ملف نصى يحتوى على علامات تنصيص (فردية أو زوجية)، أو على فواصل لتمييز حقول البيانات عن بعضها البعض فى هذا الملف، ويمكنك أن تجرى عديداً من الأمثلة لأوامر APPEND، تأخذ بيانات من ملفات بها محددات إلى تشكيلات ملفات dBASE.

أ – إذا كان الملف النصى المستخدم اسمه MYFILE. TXT، وله الشكل التالى :

```
"-----C-----","-----C-----",-----N-----,"-----C-----",---L---,---D---
```

ما يلى هو كيفية كتابتك أمر dBASE لأخذ بيانات منه :

```
APPEND FROM MYFILE DELIM      <cr>
```

فى هذا الملف المنقول منه... استخدمت علامات التنصيص المزدوجة والفواصل كمحددات فيه، ولاحظ أن كل حقل حرفى محصور (محدداً) بين علامتى تنصيص مزدوجتين، وكل حقل مفصول (محدد) بواسطة فاصلة، وأن الحقول العددية والمنطقية وحقول التاريخ ليست محصورة بين علامات تنصيص، ولكنها محددة بواسطة فواصل فقط.

ب – إذا كان الملف النصى المستخدم اسمه MYFILE. TXT، وله الشكل التالى :

```
'---C-----','-----C-----',-----N-----,'-----C-----',---L---,---D---
```

فما يلى... كيفية كتابتك أمر dBASE لأخذ بيانات منه :

```
.APPEND FROM MYFILE DELIM WITH '    <cr>
```

فى هذا الملف المنقول من... يحدد كل حقل حرفى بعلامتى تنصيص فرديتين، ويحدد كل حقل بفاصلة.

جـ – إذا كان الملف النصى المستخدم اسمه MYFILE. TXT، وله الشكل التالى :

```
"-----C-----","-----C-----",-----N-----,"-----C-----",---L---,---D---
```

فما يلى هو كيفية كتابتك أمر dBASE لأخذ بيانات منه :

```
APPEND FROM MYFILE DELIM WITH "     <cr>
```

لاحظ أن هذا الملف المنقول منه متطابق مع الملف المبين فى (أ).

د – افرض أن الملف النصى المستخدم اسمه MYFILE. TXT، وله الشكل التالى :

```
----C----,,,----C----,,-----N-----,,----C-----,,---L---,---D---
```

الملف النصى محدد هنا بفواصل فقط، دون استخدام أى علامات تنصيص من أى نوع. ولتوضيح الرقم العشوائى للفواصل التى تظهر بين الحقول.. تذكر أنه فى هذه الحالة تستخدم الفواصل نفسها كمحددات، ولا تكون المحددات إلا حول الحقول الحرفية فقط، كما يحدد كل حقل بفاصلة كذلك.

والآن إذا حاولت نقل بيانات على النحو التالى :

```
.APPEND FROM MYFILE DELIM WITH ,    <cr>
```

فإنك لا تنتهى إلا بنفايا نتيجة لذلك! ولا يستطيع dBASE أن يعالج استيراد ملف نصى محدد بفواصل فقط.

هـ – افرض أن الملف النصى المستخدم اسمه MYFILE. TXT، وله الشكل التالى :

```
-----C----- -----C----- -----N----- -----C----- ---L--- ---D---
```

والآن إذا حاولت أن تأخذ البيانات منه على النحو التالى :

```
.APPEND FROM MYFILE DELIM WITH BLANK      <cr>
```

فقد تنتهى أو لا تنتهى بنتائج دقيقة؛ طبقاً للبيانات التى تحصل عليها. وفى مثالنا هذا نظراً لوجود فراغ فى حقل اسم العامل بين الاسم الأول والاسم الأخير.. يقذف الفراغ حقل اسم العامل، وبقية الحقول التى تليه فى كتابة لا معنى لها. ونظراً لأن كل سجل به حقل اسم العامل، مع وجود فراغ فى مكان ما فى الحقل.. فإنك تتوقع أن عملية الاضافة تنتهى بنفايات كنتائج لها.

و – إذا كان الملف النصى المستخدم اسمه MYFILE. TXT، وله الشكل التالى :

```
$----C----$,$----C----$,-----N-----,$----C----$,---L---,---D---
```

ففيما يلى كيفية كتابتك أمر dBASE لنقل بيانات منه :

```
.APPEND FROM MYFILE DELIM WITH $      <cr>
```

كما يمكنك أن ترى.. يستطيع dBASE أن يشمل رموزاً خاصة كمحددات فى الملف المستورد، ويمكن استخدام أى من الرموز الموجودة فى أعلى صف من صفوف لوحة المفاتيح كمحدد فى الملف النصى المنقول منه، ويقبله نظام dBASE.

مؤشر SDF مع شرط FOR

THE SDF PARAMETR, WITH THE FOR CONDITION

لاحظ ما يلى بعناية... يسمح dBASE بأن تكون مخيراً فى اضافة البيانات من ملف نصى معتاد، إلى تشكيل ملفات dBASE.

```
.USE PERSNL
APPE FROM MYFILE FOR TOWN = 'R' SDF      <cr>
```

عند وقت الاضافة... يرشح dBASE السجلات التى لا تحقق الشرط. ويمكن أن يكون الشرط بسيطاً أو معقداً كما تريده. لاحظ أنه يمكنك إدخال ما يلى كذلك :

مؤشر DELIM مع شرط FOR

THE DELIM PARAMETER, WITH THE FOR CONDITION

كما فى الحالة السابقة... يجعلك dBASE تختار السجلات المراد إضافتها من ملف نصى، يستخدم محددات إلى تشكيل ملف dBASE.

```
.USE PERSNL
APPE FROM MYFILE FOR TOWN = 'R' DELIM      <cr>
```

عند عمل الاضافة.. يرشح dBASE السجلات التى لا تحقق الشرط، ويمكن أن يكون الشرط بسيطاً أو مركباً كما تريد له أن يكون، ولاحظ أنه يمكنك أن تكتب العبارة كما يلى كذلك :

```
.APPE FROM MYFILE DELIM FOR TOWN = 'R'
```

لاحظ فى الحالة السابقة... أن كل الحقول المذكورة فى شرط FOR، يجب أن توجد فى هيكل ملف dBASE.

لاحظ – من فضلك – أن وضع الفواصل وعلامات التنصيص يعتمد على المواقع فى طبيعته؛ أى إنه إذا فقدت بعض البيانات من الملف النصى المنقول منه.. فإنك تظل فى حاجة إلى الفواصل وعلامات التنصيص، ومثال ذلك.. افرض أن السجل النصى التالى قد أضيف إلى الملف المسمى PERSNL.

```
"005","AL ADAMS","","BSG","",19800524,25000.00
```

وفيما يلى كيفية عمل الإضافة :

005	تدخل الحقل المسمى EMP_NUM
Al Adams	تدخل الحقل المسمى EMP_NAME
The "null-field"	تتسبب فى حذف الحقل المسمى TOWN
BSG	تدخل الحقل المسمى ORG
The "null-field"	تتسبب فى حذف الحقل المسمى EXE
19800524	تدخل الحقل المسمى DT-OF-HIRE
25000.00	تدخل الحقل المسمى SALARY

أى إنه فى نهاية عملية الاضافة.. تلغى الفواصل وعلامات التنصيص تلقائياً، وتحدد كل فاصلة نهاية حقل ولا تضاف إلا البيانات – الفعلية والفراغات – فقط فى هيكل قاعدة البيانات المنقول إليها.

لاحظ أنه إذا أريد اضافة تاريخ من ملف نصى إلى تشكيل dBASE.. فيجب أن يكون شكل البيانات فى الملف النصى على النحو التالى : CCYYMMDD؛ حيث تشير CC إلى القرن، وتشير YY إلى السنة، وتشير MM إلى الشهر، وتشير DD إلى اليوم. وعند انتهاء الإضافة.. يكون التاريخ فى ملف dBASE على الشكل التالى : MMDDYY.

سبق توضيح فهم أهمية مؤشر التحديد delimited parameter فى أوامر APPEND وماذا تتوقعه أن يحدث إذا أردت اضافة بيانات من ملف نصى محدد، مع إخطارك dBASE أن الملف هو ملف SDF؛ ناسياً استخدام مؤشر DELIM فى الأمر؟

تنقل كل فاصلة وكل علامة تنصيص إلى داخل هيكل قاعدة البيانات، كما لو كانت بيانات معتادة! ولا يستطيع dBASE أن يميز بين الفواصل وعلامات التنصيص كبيانات من ناحية، والفواصل وعلامات التنصيص لمحددات من ناحية أخرى. فإذا كانت الفواصل وعلامات التنصيص مستخدمة كمحددات.. فيجب أن تخطر dBASE بهذه الحقيقة!

يمكن أن يكون والخطأ العكسى على نفس مستوى درجة السوء، وافرض أن لديك الملف النصى غير المستخدم للمحددات التالى، الذى به السجلات المبينات فقط. إن هذا الملف يحتوى على بيانات رقم العامل واسمه فقط.

001JOHN JONES
002PAUL PONES

أقوم الآن بإضافة هذا الملف النصى إلى تشكيل dBASE، مع الوقوع فى خطأ استخدام مؤشر DELIM.

- USE PERSNL (قل إن هذا الملف يحتوى حالياً على 3 سجلات فقط).
- APPE FROM MYFILE DELIM
- LIST

Record#	EMP_NUM	EMP_NAME	TOWN	ORG	EXE	DT_OF_HIRE	SALARY	NOTE
1	005	NINA BHARUCHA	WEBSTER	BSG	.T.	05/24/80	25000.00	MEMO
2	010	PETE JOHNSON	brighton	BSG	.T.	02/03/76	27590.00	MEMO
3	015	GLORIA PATEL	FAIRPORT	RMG	.T.	07/16/82	27500.00	memo
4	001				.F.	/ /		Memo
5	002				.F.	/ /		Memo

لاحظ أن رقم العامل فقط هو الذى نقل، والسبب هو أن dBASE بحث عن محدد فى الملف النصى ولم يجد شيئاً، فافترض على ذلك أن السجل لا يحتوى إلا على حقل واحد من البيانات. وعلى هذا... فقد أضاف هذا الحقل الواحد من البيانات فى أول حقل من حقول هيكل قاعدة البيانات، وحيث إن هذا الحقل الأول هو حقل EMP_NUM، الذى لا يزيد طوله عن 3 رموز.. فلن تنقل إلا الرموز الثلاثة الأولى فقط.

وعلى هذا.. فمن المهم فهم شكل بياناتك تماماً، قبل البدء فى اضافة بيانات من ملف إلى آخر.

أصل الملفات ذات الشكل النصى

THE ORIGIN OF THE TEXT-FORMAT FILES

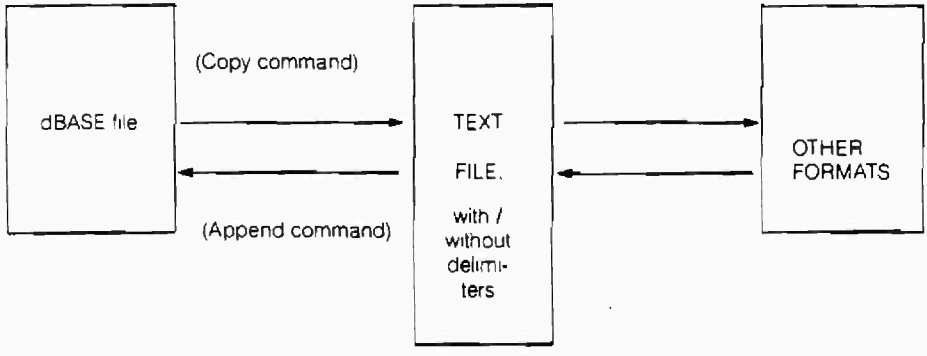
افهم من فضلك أنك لن تنتج ملفات نصية فعلية بنفسك. فبعد كل شىء... يمكنك انتاج ملفات بيانات فى dBASE مباشرة، باستخدام أمر CREATE أولا، ثم باستخدام أمر APPEND-.

ويمكن أن تأتى هذه الملفات النصية كنتيجة للتحويل من نظم برامج تجارية أخرى، ومثال ذلك... إذا أردت استخدام ملفات نصية انتجت باستخدام MBASIC، أو SUPERCALC، أو LOTUS وغيرها.. فمثل نظم البرامج هذه لديها المقدرة على انتاج ملفات نصية من ملفاتها الخاصة، ويمكن أن تستخدم هذه الملفات النصية أو لا تستخدم محددات؛ طبقاً للبدائل المحددة فى نظم برامج المصدر لها، وتستطيع الآن نقل هذه الملفات النصية إلى تشكيل dBASE المطلوب.

هذا يعنى أن التشكيل النصى (باستخدام أو بعدم استخدام محددات) هو الأرضية المشتركة بين مجموعات نظم البرامج المختلفة، إذا أريد اقتسام البيانات بينها. وبالرغم من أن dBASE يقدم لك المقدرة على نقل بيانات، من مجموعات نظم برامج مختلفة مباشرة – سبق أن رأيت أمثلة لهذا فى هذا القسم – إلا أنه يمكنك كذلك استخدام الشكل النصى لتحقيق نفس الشىء، وهذا مفيد؛ خاصة فى المواقف التى لايدعم dBASE استيراد تشكيل معين لملف مباشرة.

كما تستطيع أن تقلل كذلك من الحمل على بيئة أجهزة الكمبيوتر الكبيرة Mainframe إلى تشكيل dBASE، طالما أنه يمكن نقل البيانات على هيئة ملفات نصية.

يبين شكل 1-17 مسار البيانات إلى ومن هيكل قاعدة بيانات dBASE، ويمكنك أن تستخدم أمر APPEND من dBASE فى نقل بيانات من ملفات نصية إلى هيكل dBASE، وتستطيع استخدام أمر COPY (المغطى فيما بعد) فى نسخ بيانات dBASE فى تشكيل نصى.



شكل 1-17 : مسار بيانات نصية إلى ومن هيكل قاعدة بيانات dBASE.

دعنى أعيد التركيز على نقطة هنا : تجعل مؤشرات SDF و DELIM من الممكن استيراد بيانات من ملف نصى نمطى (معتاد أو محدد) إلى تشكيل dBASE، إلا أنك لا تستطيع

الخروج من طريقك الخاص بإنتاج الملف فى تشكيل نصى؛ بحيث يمكنك أن تنقله إلى تشكيل dBASE بعد ذلك. وتأتى الملفات النصية هذه من الحاجة إلى التحويل من نظم برامج أخرى، أو لتقليل الحمل من بيئة أجهزة الكمبيوتر الكبيرة.

وننصح بالتأكد من أن لديك علامات تنصيص مزدوجة وفواصل كمحددات فى الملفات (المستوردة)؛ حيث إنه من الممكن أن تشتمل البيانات المحولة فى بعض الحقول الحرفية تشتمل على فواصل محتواة، مثل ذلك : افرض أن البيانات التى حصلت عليها من مصدر خارجى بها اسم الشركة كما يلى: Joseph Armstrong & Sons, Inc. والآن توجد فاصلة محتواة فى الحقل؛ فإذا كان الملف النصى يستخدم الفواصل فقط كمحددات.. فعندما تريد نقل هذه البيانات إلى تشكيل dBASE – أثناء عملية APPEND – يمكنك أن تتخيل أن وجود الفاصلة فى اسم الحقل تجعل dBASE يضع .Inc فى الحقل التالى؛ معطياً نفايا على ذلك كنتيجة لهذا السجل.

لنفس السبب.. إذا كان لديك ملف نصى، يستخدم فراغات BLANK كمحددات.. فإن الاضافة التالية من هذا الملف النصى إلى ملف PERSNL – مثلاً – تتسبب فى أن كل سجل يظهر فى صورة غير مفهومة، وذلك بسبب الفراغ الزائد فى حقل الاسم (بين الاسم الأول والاسم الأخير)، ويجب أن تحدد مثل هذه البيانات فى الشكل النصى بفواصل وفراغات.

من الواضح أن هذا يقود إلى سؤال منطقى : هل ينتج نفس النوع من المشاكل إذا احتوت بيانات الملف النصى على علامات تنصيص مزدوجة، عندما تستخدم علامات التنصيص المزدوجة كمحددات؟

الاجابة هى : نعم ولكن هذا يحدث لهذا الحقل المعين فقط. وعلى سبيل المثال... افرض أن السجل النصى المحدد التالى موجود. لاحظ علامتى التنصيص المزدوجتين الاضافيتين داخل اسم الحقل.

"005","NINA"BHARUCHA","WEBSTER","BSG",T,19800524,25000.00

إذا أردت الآن أن أحدد ما يلى :

```
.USE PERSNL      <cr>
.APPE FROM TEXTFILE DELIM WITH "      <cr>
```

005 تدخل الحقل المسمى EMP_NUM

NINA تدخل الحقل المسمى EMP_NAME، وتهمل بقية الاسم إلى الفاصلة، التى تحدد نهاية الحقل.

WEBSTER تدخل الحقل المسمى TOWN، و...هكذا.

بالرغم من فقدان جزء من الاسم - إذا لم تكن النقطة مستخدمة كمحدد - فإن كل قطعة بيانات تلى الاسم NINA، تظهر فى صورة غير مقروءة بالنسبة لهذا السجل.

إذا طلب منى تلخيص المناقشة السابقة.. فإننى أركز على أنه يجب :

(١) أن تعرف البيانات التى تستخدمها.

(٢) تتأكد - إذا كان هذا ممكناً - أن الملف النصى المنقول منه يكون محدداً بعلامات تنصيص مزدوجة وفواصل، إذا كان مطلوباً نقل البيانات إلى هيكل dBASE. وهذا بالطبع ليس إلا إجراء احتياطى للحماية، فى حالة احتواء البيانات الفعلية المنقولة على علامات تنصيص أو فواصل أو فراغات.

حقول المذكرات والملفات النصية MEMO FIELDS AND TEXT FILES

عند الاضافة من ملف نصى (سواء كان ملفاً نمطياً معتاداً أو يستخدم محددات) لا يحدث نقل إلى حقل المذكرة، إذا كان حقل مذكرة فى ملف المصدر. وهذا صحيح حتى إذا كانت هناك رموز سريان زائد فى ملف المصدر.

مثال ذلك... هيكل قاعدة البيانات المسماة MASTER هو على النحو التالى :

```
NAME    C  10
NOTE    M  10          (هذا هو حقل المذاكرة)
```

وملف المصدر النصى به سجل البيانات التالى :

```
JOHN JONES YOUR ACCOUNT IS PAST DUE. PLEASE PAY NOW.
```

لاحظ أن الاسم JOHN JONES يشغل 10 رموز بالضبط، والآن... إذا أردت أن تضيف هذه البيانات النصية فى قاعدة البيانات مستخدماً الأوامر التالية :

```
.USE MASTER      <cr>
.APPE FROM MYFILE SDF      <cr>
```

يدخل الاسم JOHN JONES فى حقل NAME، إلا أن الرسالة : YOUR AC-COUNT.. لن تسجل فى ملف DBT. المصاحب لملف dBASE، وهذا يكون صحيحاً حتى إذا كان ملف المصدر النصى مستخدماً محددات.

الإضافة من منظومة طبقاً لشرط FOR

APPEND FROM ARRAY array-name FOR condition

المنظومات موضحة بالتفصيل فى القسم الخاص بمتغيرات الذاكرة Memory Variables، وهذا البديل من أمر APPEND موضح فى ذلك القسم كذلك.

اضافة حقل مذكرة من ملف نصى

APPEND MEMO memo-field-name FROM text-file (OVERWRITE)

يمكننى استخدام هذا البديل فى قراءة محتويات ملف نصى، داخل حقل مذكرة لسجل معين. افرض اننى استخدم مشغل كلمات فى انتاج الملف المسمى TEXTFILE. txt، الذى يحتوى على مذكرة، وأريد أن أضع هذه المذكرة فى حقل المذكرة الموجود، فى أول سجل من سجلات الملف PERSNL.dbf.

```
.USE PERSNL
.APPE MEMO NOTE FROM TEXTFILE
```

تضع عبارة USE المشير عند السجل رقم1، وينسخ أمر APPE محتويات الملف TEXTFILE.txt؛ كإضافة إلى محتويات المذكرة الحالية، إذا كانت بها محتويات، للسجل الحالى.

يمكننى استخدام مؤشر OVERWRITE لاستبدال المحتويات الحالية للمذكرة بالمذكرة المنقولة وليس اضافتها. وبدون مؤشر OVERWRITE.. يضاف الحقل إلى المحتويات الموجودة لحقل المذكرة إذا وجدت به محتويات.

يمكن استخدام هذا البديل فى نسخ ملف داخل حقل مذكرة، كما يمكن كذلك أن تقدم اتساعاً خاصاً بك لاسم الملف، والاتساع التقليدى هو TXT..

إذا لم يكن السجل به مذكرة سابقة، ونسخت ملفاً كمذكرة للسجل... يبدأ مؤشر المذكرة فى الظهور بالحروف الكبيرة.

مخرجات حركة البيانات DATA MOVEMENT OUTCOMES

يمكن أن يكون الوقت الحالى هو الوقت المناسب لتغطية أنواع أسئلة «ماذا... إذا؟»، التى تتعامل مع حركة البيانات. وفى كلمات أخرى : ما المخرجات المتوقعة إذا حاولت اضافة -AP PEND بيانات بين حقل معرف أنه EMP_NUM, C, 6 مثلاً، وحقل آخر معرف أنه EMP_NUM, N, 5؟

يمكن محاولة عمل كل الخليط من الحالات كما فى جدول 1-17، وتستطيع أن ترى من هذا الجدول مايلى :

جدول 1-17 : التأكد من هياكل البيانات قبل الإضافة.

ملاحظة : الرمز ^ فى الجدول يعنى رمز الفراغ.

	"FROM" field and data		"TO" field and result	
a)	C,6	ABCDEF	C,5	ABCDE
b)	C,6	ABCDEF	C,7	ABCDEF^
c)	N,4	1972	C,3	(blanks)
d)	N,4	1972	C,4	1972
e)	N,4	1972	C,5	^1972
f)	N,4	1972	N,3	(blanks)
g)	N,4	1972	N,5	^1972

```
h)  C.3          N.3
                 12A . . . . . 12
                 1A2 . . . . . 1
                 A21 . . . . . 0
                 123 . . . . . 123
i)  C.3          N.4
                 12A          12
                 1A2          1
                 A21          0
                 123          123
j)  C.3          N.2
                 12A          12
                 1A2          1
                 A21          0
                 123          (blanks)
```

١ – لحركة البيانات العددية من أى من الحقول الحرفية أو العددية... إذا كان الحقل المستقبل ليس واسعاً بدرجة كافية لاستقبال كل الرموز العددية من الحقل الراسل، فإما أن ينقل إليه صفر (بالنسبة إلى الحقل المستقبل العددى)، أو فراغ (بالنسبة إلى الحقل المستقبل الحرفى).

٢ – إذا أتت بيانات عددية من حقل حرفى.. فإن أول رمز غير عددى يوقف الحركة.

٣ – كل البيانات التى تدخل حقل عددى دائماً ما تضبط من ناحية اليمين.

مغزى القصة هو :

من الأفضل أن تكون آمناً بدلاً من الأسف لعدم الأمان، الذى يؤدى لأخطاء. اختبر هياكل ملفاتك بالنسبة إلى التوافقية، قبل أن تبدأ الاضافة.

افرض انك بدأت عملية اضافة.. فبعد اتمام الاضافة، إذا رأيت فراغات فى أحد الحقول المستقبلة أو فى أكثر من حقل منها، وكان يجب ألا تتواجد هذه الفراغات.. فلتعرف أن هذا يشمل عدم توافق هياكل البيانات كما سبق توضيحه. إلا أنه أثناء الاضافة.. فإن dBASE لا يقدم أسماء الحقول أو أرقام السجلات للبيانات المشمولة فى العملية. وعند هذه النقطة تحتاج العودة إلى كل إضافاتك؛ لتعيدها مرة أخرى، بعد أن تعدل من الهياكل المناسبة.

افرض أنك اضفت بيانات فى FILE1 من FILE2، وعادة عندما يتم تنفيذ أمر -AP PEND.. يومض dBASE برسالة على الشاشة، محدداً عدد السجلات التى اضيفت إلى FILE1 فى نهاية البيانات الموجودة فيه. وهذه السجلات هى السجلات التى يراد حذفها من FILE1، ويمكن عند ذلك ضغط PACK الملف FILE1، وذلك لحذف كل السجلات التى أضيفت للملف FILE1 حذفاً واقعياً.

إلا أنه فى بعض المواقف.. تكون قد أصدرت أمراً سابقاً للتأثير : SET TALK OFF (يوضح هذا الأمر فيما بعد، فى الجزء الثانى من هذا الكتاب). تحت تأثير أمر SET السابق.. أنه لا تتواجد أى رسالة من dBASE خاصة بعدد السجلات، التى أضيفت من الملف FILE2.

وعلى هذا.. يجب أن تحل هذه المشكلة كما يلى : إذا كانت كل سجلات الملف FILE2 قد أضيفت فيمكنك أن تفعل ما يلى :

```
.USE FILE2      <cr>
GO BOTT         <cr>
DISP            <cr>
```

وهذا يعرض آخر سجل مع رقمه؛ بحيث تعرف عدد السجلات، التى تريد إزالتها بالضبط من الملف FILE1.

أما إذا ما أضيفت سجلات مختارة فقط من FILE2 إلى الملف FILE1.. فيجب أن تكتب مايلى:

```
USE FILE2       <cr>
SET TALK ON
.COUNT FOR <the same condition used for the APPEND>      <cr>
```

والآن يقدم لك dBASE على الشاشة رقماً يحدد عدد السجلات، التى حققت الشرط، وبالتالى.. عدد السجلات، التى يجب حذفها من الملف FILE1.

باستخدام إحدى الطريقتين السابقتين.. يمكنك أن تعرف بسهولة عدد السجلات، المراد حذفها من الملف FILE1.

والآن بالنسبة إلى عملية الحذف الفعلية لهذه السجلات :

افرض انك تريد حذف 100 سجل مثلاً. يجب أن تكتب على ذلك ما يلى :

```
USE FILE1      <cr>
GO BOTT        <cr>
DISP           <cr>
```

يقدم لنا رقم آخر سجل إجمالى عدد السجلات الموجودة فى الملف FILE1، وافرض أنه كان هناك 300 سجل فى الملف FILE1.

من الواضح أنك إذا حذفت آخر 100 سجل من إجمالى 300 سجل... فمن المنطقى جداً أن تحذف السجلات من السجل رقم 201، حتى السجل رقم 300؛ أى إن رقم أول سجل للحذف هو 201 ويحسب كما يلى : (1 + (300-100)).

```
201      <cr>
```

وهذا ينقل مشير السجل إلى السجل رقم 201، والملف المستخدم بالطبع هو الملف FILE1.

```
.DELE REST      <cr>
.PACK           <cr>
```

تحذف هذه العملية كل السجلات المضافة وغير المطلوبة.

الفصل الثامن عشر

أوامر النسخ والإحلال

COPY AND REPLACE COMMANDS

أمر COPY من أقوى الأوامر فى dBASE، ويمكنك باستخدام هذا الأمر أن تؤدى أياً مما يلى:

- عمل نسخة احتياطية لأى قاعدة بيانات أو هيكل أو بيانات.
- نسخ الهيكل فقط.
- نسخ البيانات فقط على هيئة ملف نصى.
- نسخ سجلات من اختيارك.
- نسخ هيكل محدود فقط دون بيانات أو مع البيانات.
- انتاج ملف يستخدم التحديد لاستخدامه مع نظم برامج أخرى.

أمر النسخ THE COPY COMMAND

الصيغة الشاملة لأمر النسخ هى كما يلى :

```
.COPY TO <file> [scope] [FIELD <list>] [FOR <condition>]
                                       [WHILE <condition>]
                                       [TYPE <filetype>]
.COPY FILE <file1> TO <file2>
.COPY INDEXES
.COPY MEMO
.COPY TO <file> [STRU] [FIELD <field-list>]
.COPY TAG
.COPY TO ARRAY
```

كما ينسخ أمر COPY سجلات محذوفة منطقياً كذلك، إلا إذا استخدمت SET DELE ON.

إذا كان الملف مشتملاً على حقول مذكرات.. ينسخ ملف المصدر DBT. كذلك، وفى كلمات أخرى.. فإنك تحصل على نسخة كاملة وجيدة من الملف الأصلى.

انسخ إلى ملف آخر COPY TO <file>

```
.USE PERSNL     <cr>
.COPY TO BACKUP     <cr>
```

ينتج هذا ملفاً مميزاً ومنفصلاً آخر اسمه BACKUP. DBF، يكون متطابقاً فى هيكله وبياناته مع الملف المسمى PERSNL. DBF، وتنتج هذه العملية كذلك ملفاً اسمه BACK-UP. DBT، وهو ملف المذكرة، الذى يدعم قاعدة البيانات BACKUP.

وهذه هى طريقة سريعة لعمل نسخة احتياطية كاملة لأى ملف رئيسى لقاعدة بيانات dBASE. وفى الواقع... قبل بدء عمل التنقيح EDITING، عليك باستخدام هذا الأمر لحفظ ملف PERSNL فى قاعدة بيانات PRESERVE. DBF.

إذا وجد ملف يسمى BACKUP. DBF، قبل بدء هذه العملية.. يحذرك dBASE بأن الاستمرار فى تنفيذ هذا الأمر يعنى الكتابة على بيانات ملف موجود فعلاً، وتستطيع – عند هذه النقطة – أن تخرج بأمان من هذا الموقف، إذا لم ترد إزالة محتويات الملف الموجود فعلاً.

ملاحظة :

إذا كنت تكتب بسرعة كبيرة، ولمست مفتاح الادخال أكثر من مرة واحدة فى عملك السريع.. فإنك بذلك تأخذ فى الاعتبار بديل حذف بيانات الملف الموجود، والكتابة عليها ! سبق أن ذكرت نفس الاحتياط فى أمر CREATE إذا تذكرت ذلك.

مؤشر المدى Scope Parameter

يمكن أن يكون مؤشر المدى أياً مما يلى :

ALL RECORD n NEXT n REST

```
.USE PERSNL     <cr>
.COPY TO BACKUP NEXT 10     <cr>
```

يضع أمر USE تحكم dBASE عند أول سجل من ملف PERSNL، ويمكن – عند ذلك– أن ينسخ أمر COPY محتويات الهيكل، مع نسخ أول عشرة سجلات بيانات من الملف فى قاعدة البيانات BACKUP. وبالطبع العشرة سجلات التالية هى مدى العملية، وتنسخ السجلات المحذوفة منطقياً، التى تقع داخل المدى كذلك.

مؤشر الحقل Field Parameter

```
.USE PERSNL     <cr>
.COPY TO BACKUP FIELDS EMP_NUM,EMP_NAME,ORG     <cr>
```

ينتج هذا ملفاً اسمه BACKUP. DBF، يشمل كل سجلات البيانات من ملف PE-RSNL، إلا أن هيكل هذه السجلات لا تكون به إلا الحقول EMP_NUM، و EMP_NAME، و ORG فقط، كما سبق تعريفها تماماً فى هيكل الملف PERSNL.

يظهر المقطع السابق أحد الاستخدامات الأكثر أهمية لأمر COPY، وهى إمكانية إنتاج مجموعة من المعلومات من ملف رئيسى.

وبشمول مؤشر FIELDS... تستطيع ببساطة تحديد أى الحقول، التى تريد نسخها من الملف الرئيسى، وفى أى ترتيب كذلك.

شرط من اجل The FOR Condition

يمكنك أن تحدد شرطاً، تريد أن تستمر عملية النسخ عند تحققه.

```
USE PERSNL     <cr>
COPY TO BACKUP FOR TOWN = 'ROCHESTER'     <cr>
```

والآن... كل السجلات التى تحقق شرط FOR فقط هى التى تنسخ، ويجب أن تعرف الحقول المستخدمة فى شرط FOR بالطبع فى هيكل الملف المنقول منه.

فى المثال السابق... يكون لملف BACKUP نفس هيكل الملف PERSNL، إلا أن سجلات البيانات تكون قاصرة على السجلات، التى تحقق الشرط TOWN = 'ROCHESTER'.

وكما هو الحال دائماً.. يمكن للشرط أن يكون بسيطاً أو مركباً كما تريده، ويمكن استخدام الدوال الخاصة التى سبق ذكرها فى الشرط.

```
USE PERSNL
COPY TO BACKUP FOR ORG = 'BSG' .AND.
DT_OF_HIRE < {01/01/77} .OR. DT_OF_HIRE > CTOD('12/31/80')
```

(بينت كلاً من صيغتى دالة من «حرفى إلى تاريخ» كتذكرة بسيطة لك.)

تذكر أن حقل DT_OF_HIRE معرف بأنه حقل تاريخ، وعلى هذا... يجب ألا يقارن إلا بحقل تاريخ آخر فقط. والتواريخ المقدمة بين علامات تنصيص، هى سلاسل حرفية، يجب أن تحول إلى تواريخ حقول، باستخدام دالة من حروف إلى تاريخ. ويريد الأمر السابق نسخ كل العاملين فى المؤسسة BSG الذين عينوا اما قبل 01/01/77، أو بعد 12/31/80. (أكتب كل الأمر فى سطر واحد عند ملقن النقطة.)

يمكنك – بالطبع – تحديد أكثر من مؤشر واحد فى الأمر، مثال ذلك :

```
COPY TO BACKUP FOR TOWN = 'R' FIELDS EMP_NUM,EMP_NAME,ORG
```

ينتج الأمر السابق ملفاً اسمه BACKUP. DBF، يوجد فى تكوينه ثلاثة حقول فقط، ويحتوى على بيانات منسوخة من الملف الرئيسى، تحقق الشرط 'R' = TOWN فقط.

شرط بينما (أو أثناء) The WHILE Condition

كما سبق توضيحه من قبل.. فإن سلوك مؤشر WHILE متشابه أيضاً، بالرغم من أنه ليس متطابقاً مع مؤشر FOR.

```
COPY TO BACKUP WHILE TOWN = 'R'     <cr>
```

ويستمر هذا فى نسخ سجلات، طالما أن السجل التالى يحقق الشرط المذكور أعلاه. وعندما لايحقق السجل التالى الشرط.. يتوقف تنفيذ الأمر، بالرغم من إمكانية وجود سجلات أخرى فى الملف – فى مواقع أخرى منه – تحقق نفس الشرط.

Saving Deleted Records — حفظ سجلات محذوفة

أثناء تغطيتى لمجموعة الأوامر DELETE، و RECALL، و PACK... ذكرت أنه قبل أن تضغط PACK قاعدة بياناتك فعلاً، فقد تريد أن تحفظ كل السجلات المحذوفة فى قاعدة بيانات أخرى، لها نفس هيكل الملف الرئيسى. وأنصح بذلك على حساب الحقيقة، بأنك قد تقرر- بعد إجراء الضغط فى قاعدة بياناتك – أن تحتفظ ببعض أو كل السجلات المضغوطة، وأن قرارك بالضغط كان متسرعاً.

(بافتراض وجود بعض السجلات المحذوفة)

```
.USE PERSNL      <cr>
.COPY TO KEEPEM FOR DELETED()      <cr>
```

وهذا ينسخ السجلات المحذوفة فقط فى قاعدة بيانات اسمها KEEPEM. DBF، لها نفس هيكل الملف PERSNL. وتحفظ السجلات فى ملف KEEPEM. DBF فى نفس حالة الحذف، مع وجود مؤشر النجمة فى كل السجلات.

وبالتالى إذا غيرت رأيك وأردت هذه السجلات.. يمكنك كتابة ما يلى :

```
.USE PERSNL
.APPEND FROM KEEPEM
```

ويحضر هذا السجلات إلى قاعدة البيانات PERSNL كاملة التنشيط مرة أخرى.

(TYPE) SDF Parameter — مؤشر SDF

```
.USE PERSNL      <cr>
.COPY TO BACKUP NEXT 5 SDF      <cr>
```

تعمدت استخدام نفس الاسم BACKUP؛ لتوضيح ما يفعله الأمر بالضبط.

يفيد المؤشر SDF نظام dBASE أن الملف المسمى BACKUP غير مطلوب انتاجه كملف dBASE، ولكن كملف نصى (إذا تذكرت.. فإنه اختصار تشكيل بيانات نمطى Stan-dard Data Format). يؤدى انتاج ملف نصى بواسطة dBASE إلى وجود TXT، كاسم ثانوى يعطى للملف.

ويمكنك أن تؤدى ما يلى كذلك :

```
.COPY TO BACKUP.FIL NEXT 5 SDF      <cr>
```

هذا ينتج ملفاً نصياً اسمه BACKUP. FILE. وفى غياب الاسم الثانوى يعطى dBASE المؤهل TXT. للملف النصى.

استخدام مؤشر SDF هو إحدى الطرق للانتقال من تشكيل dBASE إلى التشكيل النصى، وتمرير بيانات dBASE لتشغيلها مع نظم برامج أخرى. وهذا مكمل لأمر AP-PEND، الذى يجعلك تنتج ملف dBASE من ملف SDF (أنظر شكل 1-17).

تذكر أن الملف النصى ليس له هيكل محدد، فيما عدا ما يمكن أن يستبدله المستفيد الذى يستخدمه، كما أن التحويل إلى تشكيل نصى، يسمح لك بادخال بيانات من ملف dBASE كجزء من وثيقة معدة بواسطة مشغل كلمات.

وهذا هو السبب فى أن الملف المسمى BACKUP. TXT يشبه ما يلى :

```
005NINA BHARUCHAWEBSTER  BSGT1980052425000.00
010PETE JOHNSON brighton BSGT1976020327590.00
015GLORIA PATEL FAIRPORT RMGT1982071627500.00
020MAX LEVINSKY HENRIETTARMGF1969041327550.00
025KIM BRANDT   FAIRPORT RMGF1977040436000.00
```

ملاحظة :

لا تنسخ المذكرات من ملف dBASE إلى تشكيل SDF.

مؤشر التحديد (TYPE) DELIMITED Parameter

يساعدك هذا المؤشر على انتاج ملفات نصية فى تشكيل نمطى، تستخدم التحديد كذلك.

```
.USE PERSNL      <cr>
.COPY TO MYFILE NEXT 5 DELIM      <cr>
```

كما سبق ذكره... ينتج هذا الأمر ملفاً اسمه MYFILE. TXT، الذى يكون ملفاً نصياً مستخدماً محددات.

وفيما يلى... كيف يبدو ملف MYFILE. TXT :

```
"005","NINA BHARUCHA","WEBSTER","BSG",T,19800524,25000.00
"010","PETE JOHNSON","brighton","BSG",T,19760203,27590.00
"015","GLORIA PATEL","FAIRPORT","RMG",T,19820716,27500.00
"020","MAX LEVINSKY","HENRIETTA","RMG",F,19690413,27550.00
"025","KIM BRANDT","FAIRPORT","RMG",F,19770404,36000.00
```

لاحظ أن كل الحقول مفصولة عن بعضها البعض بواسطة فواصل، مع تحديد الحقول الحرفية بعلامات تنصيص مزدوجة (مثل الإجراء التقليدى).

```
.1    <cr>
.COPY TO MYFILE NEXT 5 DELIM WITH '    <cr>
```

وفيما يلى.. كيف يبدو ملف MYFILE. TXT الآن :

```
'005','NINA BHARUCHA','WEBSTER','BSG',T,19800524,25000.00
'010','PETE JOHNSON','brighton','BSG',T,19760203,27590.00
'015','GLORIA PATEL','FAIRPORT','RMG',T,19820716,27500.00
'020','MAX LEVINSKY','HENRIETTA','RMG',F,19690413,27550.00
'025','KIM BRANDT','FAIRPORT','RMG',F,19770404,36000.00
.1    <cr>
.COPY TO MYFILE SDF DELIM WITH "    <cr>
```

وفيما يلى... كيف يبدو ملف MYFILE. TXT الآن (سبق أن رأيت هذا من قبل)

```
"005","NINA BHARUCHA","WEBSTER","BSG",T,19800524,25000.00
"010","PETE JOHNSON","brighton","BSG",T,19760203,27590.00
"015","GLORIA PATEL","FAIRPORT","RMG",T,19820716,27500.00
"020","MAX LEVINSKY","HENRIETTA","RMG",F,19690413,27550.00
"025","KIM BRANDT","FAIRPORT","RMG",F,19770404,36000.00

.1    <cr>
.COPY TO MYFILE NEXT 5 DELIM WITH .    <cr>
```

وفيما يلى.. كيف يبدو ملف MYFILE. TXT الآن :

```
,005,,,NINA BHARUCHA,,,WEBSTER,,,BSG,,T,19800524,25000.00
,010,,,PETE JOHNSON,,,brighton,,,BSG,,T,19760203,27590.00
,015,,,GLORIA PATEL,,,FAIRPORT,,,RMG,,T,19820716,27500.00
,020,,,MAX LEVINSKY,,,HENRIETTA,,,RMG,,F,19690413,27550.00
,025,,,KIM BRANDT,,,FAIRPORT,,,RMG,,F,19770404,36000.00
```

```
.1    <cr>
.COPY TO MYFILE NEXT 5 DELIMITED WITH BLANK    <cr>
```

فى هذه المرة، المحدد هو موقع فراغ، ويشبه الملف MYFILE. TXT ما يلى :

```
005 NINA BHARUCHA WEBSTER BSG T 19800524 25000.00
010 PETE JOHNSON brighton BSG T 19760203 27590.00
015 GLORIA PATEL FAIRPORT RMG T 19820716 27500.00
020 MAX LEVINSKY HENRIETTA RMG F 19690413 27550.00
025 KIM BRANDT FAIRPORT RMG F 19770404 36000.00
```

يمكنك أن تقدم محدداً خاصاً بك :

```
.COPY TO MYFILE NEXT 5 DELIM WITH $      <cr>

$005$,$NINA BHARUCHA$,$WEBSTER$,$BSG$,T,19800524,25000.00
$010$,$PETE JOHNSON$,$brighton$,$BSG$,T,19760203,27590.00
$015$,$GLORIA PATEL$,$FAIRPORT$,$RMG$,T,19820716,27500.00
$020$,$MAX LEVINSKY$,$HENRIETTA$,$RMG$,F,19690413,27550.00
$025$,$KIM BRANDT$,$FAIRPORT$,$RMG$,F,19770404,36000.00
```

- لا تنسخ المذكرات من ملف dBASE إلى التشكيل المحدد DELIMITED.

- ملف DELIM هو ملف نصى دائماً، ويمكن أن يكون أو لا يكون الملف النصى ملف DE-LIM.

- لا يعتبر انتاج ملف نصى بفواصل فقط أو بفراغات فقط فكرة طيبة؛ لأن أى فاصلة أو فراغ فى بيانات الملف النصى تجعل التشكيل غير قابل للاستخدام. مثال ذلك.. أمر AP-PEND متتالٍ على نفس الملف، قد لا يعيد كل البيانات إلى تشكيل قاعدة البيانات بطريقة دقيقة، وذلك طبقاً لوضع الفاصلة أو الفراغ فى الملف النصى.

تشكيل (صيغة) نسخ الملف The COPY FILE format

يسمح هذا التشكيل بعمل نسخة من أى نوع من أنواع الملفات، فى أى تشكيل :

```
.COPY FILE <filename> TO <filename>      <cr>
```

ينتج عن هذا زوج من أى نوع من الملفات :

```
.COPY FILE PROGRAM1.PRG TO PROGRAM2.PRG
```

نظراً لأن هذا التشكيل يمكن أن ينسخ أى نوع من الملفات.. يجب أن تقدم المؤهل المستخدم للاسم الثانوى للملف.

إذا استخدمت هذا التشكيل فى نسخ ملف dBASE به حقول مذكرات.. يجب عند ذلك نسخ ملف DBT. المصاحب منفصلاً.

اقتراح :

لنسخ ملفات dBASE صغيرة نسبياً، استخدم إحدى الصيغ (التشكيلات) الأخرى التى سبق تقديمها :

```
.USE MASTER
.COPY TO BACKUP
```

لنسخ ملفات كبيرة... يمكن أن تثبت صيغة ...COPY FILE سرعتها.

تشكيل (صيغة) نسخ فهارس The COPY INDEXES format

توضح هذه الصيغة من أمر COPY فى القسم الخاص بالفهرسة.

تشكيل (صيغة) نسخ مذكرات The COPY MEMO format

تسمح هذه الصيغة بنسخ محتويات مذكرة من حقل مذكرة إلى ملف آخر، والصيغة الشاملة لها هى ما يلى :

```
.COPY MEMO <memo-field-name> TO <filename> [ADDITIVE]
```

مثال ذلك :

```
.USE PERSNL
.COPY MEMO NOTE TO NOTEOUT
```

ينسخ الأمران السابقان محتويات المذكرة من أول سجل إلى الملف المسمى NOTE-OUT. TXT، و TXT. هو المؤهل التقليدى، ويمكنك أن تقدم المؤهل الذى تريده كاسم ثانوى.

إذا وجد فعلاً ملف بنفس هذا الاسم NOTEOUT. TXT.. تظهر لك رسالة تحذيرية، وبدون مؤشر ADDITIVE. فإن محتويات المذكرة تكتب مع إزالة محتويات الملف الحالي المسمى NOTEOUT. TXT، ومع وجود المؤشر ADDITIVE.. تضاف محتويات المذكرة إلى الملف الموجود المسمى NOTEOUT. TXT.

مؤشر الهيكل Structure Parameter

```
.USE PERSNL     <cr>
COPY TO BACKUP STRU     <cr>
```

تفيد هذه الصيغة نظام dBASE بأنك تريد نسخ هيكل الملف المسمى PERSNL، دون نسخ أى بيانات. ينتج عن ذلك ملفاً اسمه BACKUP. DBF له الهيكل فقط، دون أى بيانات.

فاذا كان الملف الأصلى له ملف مذكرة مصاحب.. ينتج هيكل ملف مذكرة BACKUP. DBF، وكذلك بالرغم من أنه لا توجد بالطبع بيانات مذكرة منقولة، وهذا مفهوم لأنك تستطيع- فيما بعد - إضافة APPENDing بيانات فى الملف BACKUP. DBF محتويات المذكرات كذلك.

لماذا يمكنك أن تنسخ هيكل أى ملف فقط؟.

من الممكن أنك تريد انتاج قاعدة بيانات أخرى، لها هيكل شبيه، ولكنه ليس متطابقاً مع هيكل ملف PERSNL. ومن الأسهل جداً تعديل هيكل MODIFY STRU قاعدة -بيانات موجودة فعلاً- بإجراء بعض التغييرات عن إعادة إعداد هيكل جديد مرة أخرى؛ خاصة إذا كانت لديك حقول عديدة، تريد تعريفها فى الهيكل الجديد.

```
.COPY TO BACKUP STRU FIELDS EMP_NUM,EMP_NAME,ORG,SALARY
```

هذا ينسخ هيكل STRU الحقول المعرفة فقط فى الاسم BACKUP. DBF، ولا تنسخ أى بيانات بالطبع.

تشكيل (صيغة) نسخ الإشارة COPY TAG format

هذا البديل موضح فى القسم الخاص بالفهرسة.

هذا البديل موضح فى قسم متغيرات الذاكرة.

أمـــر الإحـــلال THE REPLACE COMMAND

أمر الاحلال هو سمة قوية جداً للتنقيح البعيد remote-edit feature؛ أى إنه باستخدام هذا الأمر فقط.. يمكنك عمل عديد من التغييرات فى قاعدة البيانات. وحتى الآن سبق أن أدخلت سجلاً واحداً أو أكثر (من خلال EDIT و BROWSE)، وأدخلت تغييرات فردية بنفسك. ويسمح أمر REPLACE بأن تكون بعيداً عن قاعدة البيانات stand-aloof، وتأمر dBASE أن يجرى تغييرات على أى عدد من السجلات موجودة فى الملف، وهذا يجعل من أمر REPLACE أمراً قوياً جداً، وفى نفس الوقت يجعله خطراً.

الصيغة الشاملة لأمر REPLACE، هى كما يلى :

```
.REPLACE  [scope]  <field-name> WITH <expression> [ADDITIVE]:
                   [,<field-name> WITH <expression> [ADDI]]:,
                   [ FOR <condition> ]
                   [WHILE <condition> ]
```

- يغير أمر REPLACE من محتويات الحقول من الملف المستخدم فقط.
- يعمل أمر REPLACE على حقول محنوفة كذلك.
- يسرى مؤشر الاضافة ADDI على حقول MEMO فقط.

كما هى العادة دائماً إذا لم يقدم مدى... يعمل الأمر على السجل الحالى، الذى يشير إليه المشير. أما إذا لم يقدم مدى مع وجود شرط FOR.. تكون القيمة التقليدية للمدى هى ALL.

أمثلة : (افترض لكل مثال أنك تبدأ بنسخ جديد لقاعدة البيانات.) :

```
.USE PERSNL     <cr>
```

يكون مشير السجل عند أول سجل :

```
.REPLACE ORG WITH 'XYZ'
```

نظراً لعدم ذكر مدى للعملية.. يعمل الإحلال على السجل الحالى فقط.

```
.REPLACE TOWN WITH 'PERINTON' FOR ORG = 'BSG'     <cr>
```

نظراً لعدم وجود مدى مع وجود شرط FOR.. فإن السجلات التى تحقق الشرط تتغير.

توصيـة :

نظراً لأنه من الممكن لك أن تبدل محتويات قاعدة البيانات – بسرعة كبيرة من خلال استخدام هذا الأمر – ننصحك باستخدام الاحتياط التالى، قبل إصدارك هذا الأمر :

```
.USE PERSNL     <cr>
.COPY TO BACKUP     <cr>
.REPLACE ALL..      <cr>
```

بعد اتمام الاحلال REPLACE.. اختبر إحلالاتك؛ فإذا كانت جيدة.. يمكنك أن تحذف ملف BACKUP دائماً على النحو التالى :

```
DELETE FILE BACKUP.DBF     <cr>
```

إلا أنه فى حالة حدوث أى أخطاء فى قاعدة بيانات PERSNL.. يمكن أن يكون ملف BACKUP هو طريقك للتصحيح، ويتم ذلك على النحو التالى :

```
.USE BACKUP     <cr>
.COPY TO PERSNL     <cr>
 USE PERSNL     <cr>
```

يمكن أن يعيد هذا كلاً من الملفين PERSNL و BACKUP إليك. وتأكد من اتخاذك هذا الاحتياط بانتاج ملف احتياطى، أو فكّر أكثر من مرة، قبل أن تلمس مفتاح الإدخال لتنفيذ أمر REPLACE.

لتوضيح قوة الأمر أكثر.. افرض أن لديك ملفاً مخزوناً INVENTRY رئيسياً، وقررت أنه آن الأوان لزيادة تكلفة الوحدة بمقدار 10%، فإن أمر REPLACE التالى يجعل أداء هذا النشاط كشىء بسيط جداً :

```
.USE INVNTRY      <cr>
.COPY TO BACKUP      <cr>   Keep a backup!!
.REPLACE ALL UNIT_COST WITH UNIT:COST * 1.1      <cr>
```

(**ملاحظة** : رمز النجمة يعنى عملية ضرب)

تستبدل كل قيم تكلفة الوحدة الآن بقيم أعلى 10% عن القيم السابقة لها، وبعد أن تتأكد من صحة الإحلال.. يمكنك أن تحذف الملف الاحتياطى.

لتوضيح هذا الأمر فى مثال ملف PERSNL.. فقد تقرر أن تزيد رواتب العاملين فى إحدى المؤسسات بمقدار 10%.

```
.USE PERSNL      <cr>
.COPY TO BACKUP      <cr>   (as a precaution!)
.REPLACE ALL SALARY WITH SALARY * 1.1 FOR ORG = 'BSG'      <cr>
.LIST      <cr>
```

السجلات التى تحقق شرط ORG = 'BSG' فقط، هى السجلات التى يتغير حقل الراتب فيها.

تأكد من التغييرات، ثم احذف ملف BACKUP لإخلاء المكان.

افرض أنك تريد تنظيف الملف من كل الرموز المكتوبة بالحالة السفلى، على أنك تريد تحويل كل الحروف الصغيرة إلى حروف كبيرة فعلاً فى الملف.

```
.USE PERSNL      <cr>
.REPL ALL TOWN WITH UPPER(TOWN), ORG WITH UPPER(ORG)      <cr>
```

ينفذ الأمر السابق الإحلال على الحقلين فى الملف كله، ويمكنك أن تجرى احلالاً لحقول متعددة، مستخدماً أمراً واحداً.

```
.USE PERSNL      <cr>
.REPL EXE WITH .T., ORG WITH 'GSD' FOR SALARY > 25000      <cr>
```

هذا الأمر يحل .T. محل حقل الاستثناء، ويحل GSD محل حقل المؤسسة لكل السجلات، التى يزيد الراتب فيها عن 25000.

ملاحظة :

فى المثالين السابقين.. اجريت احلالاً على أكثر من حقل واحد - فى نفس الوقت - وفى مثل هذه الحالات.. تأكد من وجود فاصلة بين كل حقلين، كما هو موضح فى الأمثلة. دون هذه الفواصل بين الحقول.. لن يحدث احلال إلا لآخر حقل فقط، مذكور فى قائمة الحقل !!.

كيف يمكنك احلال حدوث محدد من الرموز فى حقل بحقل آخر فى كل سجلات الملف؟

مثال ذلك... افرض أن الرمز الثالث فى الرمز البريدى zip code تم ادخاله بطريقة خاطئة، وتكرر الخطأ خلال الملف (نظراً لأن ادخال البيانات حدث مع استخدام SET CAR-RY ON). وفى هذا المثال... لا تكون هناك حاجة إلا إلى تغيير رمز واحد فقط.

افرض أن لديك حقلاً اسمه ZIPCODE

```
.USE PERSNL
.REPL ALL ZIPCODE WITH STUFF(ZIPCODE,3,1,'2')
```

لاحظ العبارة السابقة. إنك تجرى احلالاً لكل حقول الرمز البريدى فى كل السجلات، وتستخدم دالة STUFF فى إحلال الرمز الثالث فى حقول الرمز البريدى بالرمز 2..

إلا أنه فى مواقف أكثر أهمية.. يمكن أن يكون لديك نفس الرمز فى مواقع متعددة، داخل نفس الحقل، وتحتاج كل الحدوثات إلى تغيير فى كل السجلات - بغض النظر عن المكان الذى يوجد فيه الرمز داخل الحقل - وسوف تقدم طريقة تحقيق ذلك فى قسم البرمجة المطورة فى الجزء الثانى من هذا الكتاب.

يمكنك كذلك استبدال محتويات حقل مذكرة.

```
.USE PERSNL
.REPL NOTE WITH 'Please keep this appointment!'
```

فى هذا المثال... تستبدل محتويات المذكرة، إذا كانت هناك محتويات كلية بالسلسلة الحرفية الموجودة فى الأمر.

```
.USE PERSNL
.REPL NOTE WITH 'Please keep this appointment!' ADDITIVE
```

يحدد مؤشر الاضافة ADDITIVE أن السلسلة الحرفية يجب إضافتها إلى محتويات المذكرة الموجودة، إذا كانت هناك مذكرة موجودة.

كمثال آخر.. إذا كان لديك ملف عملاء به حقل اسمه CUSTID (من النوع العددى ويشغل 5 مواقع)، وأردت أن تقدم رقم تعريف فريد ID للعملاء الموجودين.. يمكنك عمل ما يلى :

```
.USE <filename>
.REPL ALL CUSTID WITH RECNO( ) + 1000      <cr>
```

لاحظ أنه لكل سجل.. تقوم باحلال حقل CUSTID (عددى النوع) بجمع 1000 على رقم السجل، وعلى هذا.. فأول عميل يصبح له رقم التعريف 1001، ويصبح للعميل الثانى رقم تعريف 1002... وهكذا.

وبمجرد تحديد أرقام تعريف فريدة للسجلات الموجودة.. تستمر البيانات التى يتم ادخالها فيما بعد فى نفس التسلسل، ولاحظ أنك إذا حذفت أو ضغطت بعض سجلات الملف.. فعليك بإعادة أمر الإحلال السابق؛ لتحدد أرقام تعريف عملاء ID جديدة للسجلات، ويمكن أن يكون لهذا تأثير كبير على بقية النظام، استخدم أمر الإحلال السابق بحذر.

الفصل التاسع عشر

أمـر الوصـل
JOIN COMMAND

افرض أن لديك ملفاً رئيسياً اسمه INVENTRY، له الهيكل التالى :

```
PART_NUM        DESC        UNIT_COST        ONHAND
```

افرض كذلك أن لديك ملفاً آخر، يحتوى على أوامر ORDERS لنفس أرقام الأجزا، وهيكله يحتوى على ما يلى :

```
PART_NUM        CUST_NAME        ONORDER
```

PART_NUM هو بالطبع العامل المشترك فى الهيكلين، وأنك تريد استخدام هذا العامل المشترك فى انتاج ملف آخر، له الهيكل التالى :

```
PART_NUM        CUST_NAME        ONORDER        UNIT_COST
```

أى إنك تريد أن تكون قادراً على انتقاء مجموعة من الحقول؛ لتجميعها لكل الأوامر الفردية للعملاء، وذلك لأرقام العناصر المختلفة.

يصل أمـر JOIN البيانات من قاعدتين بيانات؛ طبقاً لمعايير اختيار معينة، تحددها بنفسك، وذلك لانتاج قاعدة بيانات ثالثة. وتظل كل من قاعدتى بيانات المدخلات مفتوحتين فى نفس الوقت، وهذا فقط يعنى أن dBASE يحفظ مؤشرات سجلات منفصلة للملفين. وأكثر من ذلك.. فإنك تحدد أحد الملفين (ملف INVENTRY) كملف نشط أو ملف تحكم، مع اعتبار الملف الآخر (ملف ORDERS) ملفاً غير نشط.

قبل أن تقدم التعليمات الفعلية لأمر JOIN.. عليك أن تفتح قواعد البيانات :

```
.SELE 1
.USE INVENTRY
```

الأمران السابقان يعرفان ملف INVENTRY، كملف مفتوح فى منطقة العمل رقم 1.

```
.SELE 2
.USE ORDERS
```

الأمران السابقان يعرفان ملف ORDERS، كملف مفتوح فى منطقة العمل رقم 2.

```
.SELE 1
```

عليك بالانتهاء بهذا الأمر لتحديد أن ملف INVENTRY هو ملف التحكم. وبالانتهاء بهذا الأمر.. فإنك اخترت 1 كملف نشط؛ أى إنك حددت ملف INVENTRY كملف نشط، ويشمل هذا تلقائياً أن ملف ORDERS هو ملف غير نشط.

ملاحظة :

يمكن عكس ترتيب الأوامر السابقة؛ ليأخذ الشكل التالى :

```
.SELE 2
.USE ORDERS
.SELE 1
.USE INVENTRY
```

بهذه الطريقة... تحصل على نفس النتيجة؛ مستخدماً أربعة أوامر. ونظراً لأنك انتهيت باختيار 1.. فإنك جعلت ملف INVENTRY هو الملف النشط.

يمكنك الحصول على نفس النتيجة بطريقة أخرى :

وهذا يشمل منطقة العمل 1

```
.USE INVENTRY
.USE ORDERS IN 2
```

هذا يفتح الملف الثانى فى منطقة العمل رقم 2.. إلا أن الملف المسمى INVENTRY ما يزال هو الملف النشط !

بعد تحديد الملف النشط والملف غير النشط... يمكنك أن تصدر الآن أمر JOIN .

```
.JOIN  WITH ORDERS    TO NEWFILE    FOR
PART_NUM = ORDERS->PART_NUM    <cr>
```

إنك تحدد لنظام dBASE، وتريد وصل JOIN محتويات الملف النشط INVENTRY مع محتويات الملف غير النشط ORDERS؛ لإنتاج ملف ثالث، اسمه NEWFILE. DBF فى حالتنا هذه.

كما أنك حددت كذلك أن شرط الوصل، يجب أن يكون عبارة عن تساوى رقم الجزء PART_NUM من الملف النشط مع رقم الجزء من الملف غير النشط. والعبارة ORDERS-> PART_NUM< ماهى إلا طريقة لقول «رقم الجزء من ملف ORDERS». ويمكنك أن تكتب ما يلى بدلاً منها: B->PART-NUM< حيث B هو الاسم المستعار التقليدى لمنطقة العمل الثانية.

فإذا كان موجوداً ملف له الاسم NEWFILE. DBF، قبل بدء الأمر.. يقدم dBASE الرسالة التحذيرية المعتادة، ويمكنك أن تلغى الأمر إذا أردت ذلك.

وطريقة عمل أمر الوصل، هى كما يلى :

يقع dBASE على أول سجل فى الملف النشط، ويقارنه بدوره مع كل من السجلات الموجودة فى الملف غير النشط، محاولاً أن يجد توافقاً مبنياً على المعايير المختارة (التى تحدد أنه يجب تساوى PART_NUM فى كل من الملفين). وفى كل مرة يوجد توافق... ينتج dBASE سجل مخرجات (فى الملف NEWFILE. DBF)، يكون هيكله معرفاً بما هو مقدم، أو غير مقدم، فى أمر JOIN. تستمر هذه العملية، حتى تتم مقارنة كل السجلات الموجودة فى الملف غير النشط، وعند ذلك يقع dBASE على السجل الثانى فى الملف النشط، ويكرر نفس المقارنة مع كل سجل من سجلات الملف غير النشط؛ محاولاً إيجاد توافق، مبنياً على المعايير المستخدمة. وكما سبق ذكره.. ففى كل مرة يوجد توافق يكتب سجل فى ملف المخرجات، المسمى NEWFILE. DBF فى حالتنا هذه، وتستمر هذه العملية، حتى يتوافق كل سجل نشط مع سجل من السجلات غير النشطة.

الهيكل التقليدى لملف المخرجات Default Output File Structure

إذا لم يكن واضحاً لك الهيكل الذى تريده لملف المخرجات الجديد (وهو عبارة عن كيفية اصدارك الأمر السابق).. فإن dBASE يقدم لك هيكلاً تقليدياً، عبارة عن تسلسل (وصل

سلاسل مع بعضها وصلاً واقعياً) لهيكلى قاعدتى البيانات النشطة وغير النشطة بنفس هذا الترتيب. فإذا وجدت أسماء حقول متطابقة فى كل من الهيكلين... تختار الحقول الموجودة فى الملف النشط فقط؛ لتكون ضمن هيكل ملف المخرجات، مع إهمال الحقول المتطابقة معها، والموجودة فى الملف غير النشط.

مثال ذلك :

فى الهياكل السابقة لملف INVENTRY النشط وملف ORDERS غير النشط... نظراً لعدم وجود مؤشرات صريحة لهيكل ملف المخرجات الجديد؛ إذ يصبح هيكله على النحو التالى:

```
PART_NUM     DESC     UNIT_COST     ONHAND     CUST_NAME     ONORDER
```

هذا الهيكل عبارة عن تسلسل لهيكل الملف الأولى، يليه هيكل الملف الثانوى. ولاحظ أنه فى هذه الحالة.. يكون لديك حقل واحد اسمه PART_NUM فى هيكل ملف المخرجات، وهذا الحقل هو الذى أتى من الملف النشط، إلا أنه ليس لديك إلا هذا الهيكل التقليدى؛ فباستطاعتك أن تختار الهيكل الذى تريد أن تراه فى ملف المخرجات.

تحديد هيكل ملف المخرجات Specifying Output File Structure

يمكنك – بالطبع – أن تحدد التشكيل الدقيق لملف المخرجات، الذى تريد أن تراه.

```
JOIN WITH ORDERS TO NEWFILE FOR PART_NUM = ORDERS->PART_NUM
  FIELDS PART_NUM,ORDERS->CUST_NAME,ORDERS->ONORDER,UNIT_COST
```

حددت التشكيل الدقيق لملف مخرجات اسمه NEWFILE. DBF، وتريد أن ترى حقل PART_NUM من الملف النشط INVENTRY، ثم حقل CUST_NAME من الملف غير النشط ORDERS، ثم حقل ONORDER من ملف ORDERS، ثم حقل UNIT_COST من ملف INVENTRY. (لاحظ أن كل حقل تريده من ملف غير نشط، يجب أن يسبقه اسم الملف غير النشط، الذى يمكن اختصاره – فى حالتنا هذه – ببساطة إلى الحرف B !)

كمثال آخر لاستخدام أمر JOIN.. لنقل إنك تريد تتبع أى عناصر المخزون غير الكافية لمقابلة أوامر الطلب عليها. وبافتراض أن لديك أمر طلب واحداً لكل سجل رئيسى من سجلات المخزون.. فإن العبارات التالية تؤدى العمل المطلوب.

```
SELE 1
.USE INVENTRY
.SELE 2
.USE ORDERS
.SELE 1
.JOIN WITH ORDERS TO BACKLOG
    FOR PART_NUM = ORDERS->PART_NUM .AND. ONHAND < ORDERS->ONORDER
    FIELDS PART_NUM,ORDERS->CUST_NAME,ORDERS->ONORDER,UNIT_COST
```

تقول العبارة السابقة :

صل JOIN الملفات (أى انتج سجلات مخرجات)، إذا كان رقم الجزء من ملف INVEN-TRY، مساوياً لرقم الجزء من ملف ORDERS، وكانت الكمية الموجودة onhand فى ملف INVENTRY أقل من الكمية المطلوبة on-order فى ملف ORDERS.

تحذير على امر الاتصال Caution on the JOIN Command

إذا كانت قاعدة البيانات النشطة، وقاعدة البيانات غير النشطة طويلة جداً.. فإن هذه العملية تستغرق وقتاً طويلاً، وقد لا تنتهى فى النهاية. مثال ذلك.. إذا كان لديك 100 سجل فى الملف الابتدائى، و1000 سجل فى الملف الثانوى.. فعلى dBASE أن يمر خلال 100,000؛ مقارنة قبل أن ينتهى تنفيذ الأمر. كما أنه إذا كان تعريف معايير الاتصال غير دقيق (أو خطأ)؛ بحيث يصعب وجود توافق.. فإن dBASE يحاول أن ينتج 100,000 فى قاعدة بيانات المخرجات. وقد يكون لهذا تأثير محدود على المخرجات؛ خاصة إذا كنت تنتج بياناتك على قرص مرن، وبعيد عن العمل وقتاً لا نهائياً.. فقد يحدث توقف، إذا لم يكن ممكناً وضع ملف المخرجات على القرص.

فى عملية إنتاج هيكل تقليدى لملف المخرجات.. يحاول dBASE أن يأخذ الحقول من الملف النشط أولا، ثم ذلك الحقول من الملف غير النشط. إلا أنه إذا كان التسلسل يميل إلى

انتاج أكثر من 255 حقلاً أو أكثر من 4000 رمزاً فى هيكل ملف المخرجات.. فإن dBASE يحدد ببساطة هيكل ملف المخرجات بالحد الأعلى، وهو 255 حقلاً أو 4000 رمز.

السبب الوحيد الفعلى لوصل JOIN ملفين، هو انتاج ملف ثالث، يمكن استخدامه فى استخلاص بيانات أو انتاج تقارير. إلا أنه فى التحليل النهائى.. لا يكون ضرورياً وصل JOIN ملفين ببساطة؛ لإنتاج تقرير من قاعدة البيانات، التى تنتج من الوصل؛ فيمكنك حفظ الملفين منفصلين ومميزين، مع استطاعتك كتابة برنامج كمبيوتر لمعالجة الملفين؛ بغرض انتاج التقرير المطلوب. ولسوء الحظ.. فإنه حتى هذا الوقت الذى تتعلم فيه كتابة برامج كمبيوتر، ليس أمامك إلا استخدام أمر JOIN.

فى عملية انتاج هيكل تقليدى لملف المخرجات... يهمل dBASE كل حقول المذكرات، وإذا حددت حقل مذكرة كجزء من هيكل ملف مخرجات.. فإنك تحصل على رسالة خطأ.

بعد أن يؤدى أمر JOIN عمله.. يجب أن تغلق الملفين المستخدمين فى عملية الوصل JOIN، وذلك على النحو التالى :

```
.CLOSE ALL    <cr>
```

الفصل العشرون

أوامر متنوعة
MISCELLANEOUS COMMANDS

سبق ذكر أن أمر DISP ALL يعرض شاشات من السجلات فقط، بغض النظر عن المؤشرات الأخرى.

أمر السرد LIST COMMAND

يوجد أمر اسمه أمر السرد LIST، وهو مثل أمر DISP ALL؛ باستثناء أن أمر LIST يتميز بعدم توقف كل شاشة، وهذا هو الفارق الوحيد بين DISP ALL وLIST، وتتطابق مؤشرات أمر LIST مع مؤشرات أمر DISP ALL.

يسرد أمر LIST محتويات قاعدة البيانات على الشاشة، فى حالة دوران scroll سريع.

ملاحظة :

عندما يعمل الكمبيوتر بنظام تشغيل CP/M، او نظام تشغيل MS-DOS.. يمكنك إيقاف حركة الدوران السريع على الشاشة أثناء تنفيذ أمر LIST، أو أمر DISP ALL بالضغط على Ctrl-S. وتعيد إعادة الضغط على Ctrl-S حركة الدوران السريع؛ أى إن الضغط على Ctrl-S ينقل من وإلى الدوران السريع إلى ومن توقفه.

يسرى هذا التأثير للضغط على Ctrl-S إذا ترك SET ESCAPE ON فى حالة on التقليدية فقط. إذا ما استخدم SET ESCAPE OFF.. فلن يعمل الضغط على Ctrl-S.

• USE PERSNL	<cr>	
• LIST	<cr>	يسرد البيانات على الشاشة
• LIST TO PRINT	<cr>	يرسل السرد إلى الطابع

امر وضع الفلتر (المرشح) على SET FILTER TO

حتى الآن.. فقد رأيت عديداً من الأمثلة لاستخدام شرط FOR فى استخلاص بيانات، وتأتى أمثلة أكثر فيما بعد. والآن... أريد أن آخذ مماساً صغيراً لتوضيح استخدام أمر SET FILTER.

افرض أنك أردت لإحدى جلسات العمل مع dBASE، العمل مع مجموعة من السجلات من أحد الملفات : سجلات مدينة Rochester من حقل TOWN. يمكنك أن تجعل dBASE يتظاهر بأن الملف يحتوى على سجلات مدينة Rochester من حقل TOWN فقط، وذلك كما يلى :

```
.USE PERSNL
SET FILTER TO   UPPER(TOWN) = 'ROCHESTER'
```

ومن الآن.. فإن أمر LIST بسيط يسرد سجلات Rochester فقط، ويقوم أمر RECORD (الذى يوضح فيما بعد) بعمل تقارير من سجلات Rochester فقط... إلخ.

إذا استخدمت SET FILTER TO (another condition).. فان ذلك ينفى الشرط الأول تلقائياً، وإذا أردت أن تنفى شرط الفلتر الحالى، دون تنشيط شرط فلتر آخر.. فلابد أن تكتب مايلى :

```
.SET FILTER TO      <cr>
```

امر التوقيع LOCATE COMMAND

نظراً لأنك الآن معتاداً على أمر DISPLAY.. فإننى أقدم هنا أمراً أسهل من أمر DISPLAY، إلا أن له وضعه الخاص فى مجموعة أوامر dBASE.

الصيغة الشاملة هى كما يلى :

```
.LOCATE <scope> FOR <condition>
                WHILE <condition>
```

مثال ذلك :

```
.LOCATE FOR TOWN = 'ROCH' .AND. ORG = 'BSG'     <cr>
```

يبدأ dBASE عند قمة قاعدة البيانات، ويحرك المشير إلى أول سجل، يحقق الشرط فى قاعدة البيانات، إلا أنه لا يعرض السجل. ويمكنك أن تعرض DISPLAY أو تنقح EDIT (مغطى فيما بعد) السجل. وعندما تريد من dBASE أن ينتقل إلى السجل التالى - الذى يحقق نفس الشرط - الذى سبق تحديده.. فيمكنك أن تكتب ما يلى :

.CONT <cr> [for "CONTINUE"]

هذا ينقل المشير إلى السجل التالى، الذى يحقق نفس الشرط مثل السجل السابق له. وبهذه الطريقة.. يمكنك أن تقفز خلال قاعدة البيانات إلى السجلات المختارة فقط، ويمكنك عرضها DISPLAY، أو تنقيحها EDIT أثناء التقاطك لها.

كما سبق رؤيته.. يمكنك أن تستخدم مؤشرات FOR أو WHILE، التى تناسب النشاط الذى تعمل به، ويمكن أن يكون الشرط بسيطاً أو مركباً كما تريده، ويمكنك - كذلك - أن تستخدم كذلك الدوال الخاصة، إذا كانت هناك حاجة لذلك فى توسيع الشرط.

.LOCATE FOR SUBSTR(TOWN,2,3) = 'OCH' <cr>

إذا استخدمت مؤشر CONT بصورة معتادة.. فإنك تصل رسالة إلى نهاية مدى التوقيع End of LOCATE scope.

LOCATE <scope> WHILE <condition> <cr>

فى هذه الحالة.. نظراً لاستخدام مؤشر WHILE.. فإن dBASE يبدأ التوقيع LOCATE عند موقع السجل الحالى، وليس عند قمة الملف. ويوقع LOCATE شرط WHILE السجل التالى للشرط، أثناء تحقيق السجلات الشرط فقط، كما يتطابق تفسير مؤشر WHILE مع ما سبقت مناقشته مع أمر DISPLAY.

RECALL COMMAND أمر تذكر

هذا الأمر عكس أمر DELETE فى أنه يزيل الاشارة المنطقية (النجمة) من السجلات المحذوفة، وهو متطابق فى شكله مع أمر DELETE.

مثال ذلك : .RECALL [scope] [FOR <condition>] [WHILE <condition>] <cr>

.DELETE ALL <cr>

هذا يحذف كل السجلات.

والآن.. تذكر بعض السجلات :

```
.RECALL FOR TOWN = 'ROCH' .AND. ORG = 'GSD'      <cr>
```

كل السجلات التى تحقق الشرط، تحذف منها الاشارة إلى حذفها، أى إنها تنشط مرة أخرى.

```
.LIST     <cr>
```

Record#	EMP_NUM	EMP_NAME	TOWN	ORG	EXE	DT_OF_HIRE	SALARY	NOTE
1	*005	NINA BHARUCHA	WEBSTER	BSG	.T.	05/24/80	25000.00	MEMO
2	*010	PETE JOHNSON	brighton	BSG	.T.	02/03/76	27590.00	MEMO
3	*015	GLORIA PATEL	FAIRPORT	RMG	.T.	07/16/82	27500.00	memo
4	*020	MAX LEVINSKY	HENRIETTA	RMG	.F.	04/13/69	27550.00	MEMO
5	*025	KIM BRANDT	FAIRPORT	RMG	.F.	04/04/77	36000.00	MEMO
6	*030	TIM MONTAL	ROCHESTER	RBG	.F.	07/07/81	41900.00	memo
7	*035	WILLIAM PATEL	penfield	GSD	.F.	08/17/71	28900.00	memo
8	040	JAMES JAMESON	ROCHESTER	GSD	.T.	10/21/77	29800.00	memo
9	*045	MORRIS KATZ	webster	BSG	.F.	09/14/80	23450.00	memo
10	*050	PAUL BHARUCHA	BRIGHTON	BSG	.T.	05/23/73	29100.00	MEMO
11	*055	PHIL MARTIN	WEBSTER	RMG	.F.	07/19/80	31000.00	memo
12	*060	JOHN PETERSON	BRIGHTON	RBG	.T.	04/17/79	31480.00	MEMO
13	*065	JOY HARDY	fairport	RBG	.F.	01/19/79	34200.00	MEMO
14	*070	JAN MOREY	ROXY	GSD	.T.	04/23/67	18190.00	memo
15	*075	JOHN JONES	rochester	GSD	.T.	04/04/70	25100.00	memo

كما هو الحال دائماً.. يمكن أن يكون الشرط بسيطاً أو مركباً كما تريده، ويتذكر مؤشر WHILE السجلات أثناء تحقيقها الشرط فقط.

ملاحظة :

إذا كنت مستخدماً SET DELE ON، وأدى ذلك إلى طلبك من dBASE أن يهمل كل السجلات المحذوفة.. فإن RECALL يكون عديم التأثير على قاعدة البيانات! لتذكر سجلاً معيناً باستخدام SET DELE ON.. فعليك أن تنقل المشير إلى السجل وتتذكر السجل.

• SET DELE ON من الآن... اهمل السجلات المحذوفة

- DELE ALL • يحذف هذا كل السجلات
- RECA ALL • لا تأثير لهذا الأمر على الملف !
- RECA RECO 5 • يتذكر هذا الأمر السجل رقم 5

كما يمكنك أن ترى ضرورة أن تحدد أى السجلات المراد تذكره، إذا أردت تذكر سجل، عندما تستخدم SET DELE ON.

أمـر الضغـط PACK COMMAND

هذا الأمر مسئول عن الحذف الواقعى للسجلات، التى أشير إليها حتى الآن بالحذف المنطقى.

.PACK <cr>

لاحظ أن أمر PACK ليس له أى مؤشرات، وهذا يعنى أنك لا تستطيع اختيار السجلات المحذوفة المراد ضغطها؛ فهو حرف جر للكل أو للاشيء، وطبقاً لتنفيذه تزال كل السجلات المحذوفة.

من فضلك.. افهم تماماً أنه بمجرد إجراء الضغط.. فلن توجد أى طريقة يمكنها استعادة السجلات؛ لتأتى تحت تحكمك مرة أخرى. وعلى هذا.. فعليك بالتفكير لمعرفة الحكمة من وراء قرارك، قبل عمل الضغط.

بعد عملية الضغط.. يقل حجم قاعدة البيانات بعدد السجلات المحذوفة التى تزال، ويعاد ضبط أرقام سجلات قاعدة البيانات؛ لتبدأ بالسجل رقم واحد مع استمرار التسلسل.

لاحظ أنك إذا استخدمت DISP STRU <cr>، بعد عملية PACK.. فإن العنوان الذى تراه على الشاشة إما أنه يعكس أو لا يعكس العدد الصحيح الحالى للسجلات الموجودة فى الملف. ولتجديد العنوان.. عليك بإغلاق الملف من الاستخدام، كما أن المكان الفارغ من القرص بسبب السجلات التى ازيلت، لا تستطيع استخدامه كذلك إلا بعد إغلاق الملف.

.USE <cr>

يغلق أمر USE دون أى مؤشرات الملف المستخدم حالياً، وتستطيع الآن إعادة الملف للاستخدام، مع تجديد العنوان ليحتوى على البيانات الصحيحة.

```
.USE PERSNL      <cr>
.DISP STRU       <cr>   will give the true record count.
```

قبل أن تستمر فى الضغط PACK – عادة – قد تريد عمل نسخة احتياطية للسجلات المحذوفة، حتى إذا اكتشفت أن هناك خطأ معيناً؛ إذ يمكنك أن تتصل دائماً بالبيانات الأصلية.

فيما بعد... عندما تغطى الأوامر اللازمة، سأوضح لك طريقة لحفظ السجلات المحذوفة فى قاعدة بيانات أخرى، لها نفس الهيكل مثل قاعدة البيانات الرئيسية، وذلك قبل إصدار أمر PACK، وحتى إذا احتجت فى أى حالة إحضار هذه السجلات المحذوفة مرة أخرى.. فلن تحتاج إلا إلى إصدار أمرين من أوامر dBASE فقط.

كما يمكنك أن تقدر.. فإن أوامر : DELETE، و RECALL، و PACK تمثل مجموعة منطقية من أوامر dBASE.

أمـــر الإزالـــة ZAP COMMAND

إذا أردت حذف كل السجلات من الملف.. فان إحدى طرق عمل ذلك، هى :

```
.DELE ALL      <cr>
.PACK          <cr>
```

هذا يحفظ هيكل الملف إلا أنه يكون خالياً الآن من كل سجلات البيانات. ولكن إذا كان لديك عديد من الآلاف من السجلات فى الملف.. فان عملية الحذف والضغط هذه تستغرق وقتاً طويلاً.

ويمكن إجراء الحذف الفورى لكل السجلات يمكن إجراء من خلال استخدام أمر ZAP.

```
.ZAP      <cr>
```

يقدم dBASE رسالة تحذيرية.. فإذا أردت الاستمرار.. فإنه يحذف كل السجلات الموجودة فى الملف على الفور، ويمكن أن تكون هناك عشرة آلاف من السجلات فى الملف، وتزال كلها على الفور!

ملاحظة :

إنك تزيل ZAP ملفاً فقط، ولست بحاجة إلى تحديد أى السجلات يراد إزالتها أولاً، وتتطابق النتيجة النهائية مع خليط DELE ALL/PACK.

لماذا يستغرق خليط DELETE ALL/PACK وقتاً طويلاً، وكيف يلغى أمر ZAP السجلات على الفور؟

الاجابة هى أن خليط DELETE ALL/PACK عبارة عن عملية واقعية حقيقية؛ فعندما تحدد DELE ALL.. يجب الاشارة إلى كل سجل من سجلات الملف بنجمة، وعندما تحدد PACK.. يحدث اختبار لكل سجل من السجلات الملف، لما إذا كان مشاراً إليه بالنجمة أم لا؛ وذلك بغرض إزالته. وهذا هو السبب فى الوقت الطويل المستغرق عند استخدام خليط DE-LETE ALL/PACK.

عندما تستخدم ZAP (وتؤكد اختيارك)... يؤدى dBASE نشاطين :

١ - يضع صفراً فى حقل عداد السجلات record-count فى عنوان الملف.
٢ - يضع علامة نهاية الملف end-of-file، كأول رمز فى الملف.

تأثير ذلك أنه بالرغم من وجود عشرة آلاف من السجلات، وجوداً فعلياً على القرص.. فلن يهتم dBASE بوجودها ولا تستطيع الاتصال بأى سجل من هذه السجلات، وهذا هو الحذف الفورى أو إزالة ZAP أى ملف! وبالنسبة إلى النظام.. فإن أى مواقع فعلية تشغلها السجلات، تكون متاحة الآن لاستخدامها فى أى غرض آخر.

يجب ذكر كلمة تحذيرية هنا :

تكون هناك أوقات يراد عمل فهرسة فيها للملف الرئيسى. (الفهرسة موضحة بتفاصيل أكثر فيما بعد.) وهذا يعنى أنه يراد فتح ملف واحد أو أكثر من الملفات المفهرسة، أثناء فتح الملف الرئيسى. وفى مثل هذه الحالات.. إذا استخدمت خليط DELE ALL/PACK، أو ZAP مع الملف الرئيسى.. تأكد من أنك أعدت بناء كل الملفات المفهرسة المصاحبة لهذا الملف الرئيسى، وقد تهتم -بوجه خاص- بأن تبنى الفهرس مرة أخرى، بدلاً من الاعتماد على خليط

DELE ALL/PACK أو ZAP فى إعادة بناء الملفات المفهرسة، وتبنى الفهارس من خلال استخدام أمر INDEX الموضح بالتفصيل؛ فى قسم خاص به.

أمـــر الإدخـــال (الإقحـــام) INSERT COMMAND

هذا الأمر- بالرغم من - وجوده، إلا أنه ليست له استخدامات مباشرة كثيرة فى أى تطبيق.

وتستطيع باستخدام هذا الأمر أن تحدد لنظام dBASE، أنه مطلوب إدخال INSERT سجل جديد بين سجلين محددين تماماً، وليكونا السجل السابع والسجل الثامن.

افهم أنه لا توجد حاجة حقيقية لتحديد أن السجل الجديد؛ إذ يجب أن يوضع بعد أحد السجلات الموجودة فعلاً بالضبط؛ فيمكن ببساطة إضافة APPEND السجل الجديد فى نهاية قاعدة البيانات بعد ذلك أن تضعه فى الترتيب المناسب له بإجراء عملية ترتيب sorting، التى يمكنها أو بإجراء عملية فهرسة indexing. (عملية الترتيب وعملية الفهرسة مغطاة بالتفصيل فيما بعد.)

إلا أنك إذا أردت إدخال INSERT سجلاً جديداً فى موقع محدد من قاعدة البيانات.. يمكنك أن:

١ - تختار السجل فى قاعدة البيانات، وليكن السجل رقم 7 المراد ادخال السجل الجديد بعده.

```
7    <cr>

.INSERT    <cr>
```

٢ - هذا يدخل هيكلاً فارغاً للملف لاستخدامه (مثل حالة APPEND)؛ فإذا كنت مستخدماً SET CARRY ON.. فإن القناع يحتوى على محتويات السجل رقم 7.

٣ – بعد ادخال البيانات.. يمكنك أن تؤدى أحد شيئين اثنين فقط، هما :

أ – حفظ السجل الجديد والخروج من حالة الادخال (يحدث ذلك بالضغط على Ctrl-W أو الضغط على Ctrl-End، أو عند إدخالك آخر رمز من آخر حقل.)

ب – تغيير رأيك بالنسبة إلى هذا السجل، وتخرج من حالة الإدخال INSERT (وذلك بالضغط على Ctrl-Q، أو على مفتاح الهروب.)

بغض النظر عن الإجراء المتخذ.. فأنك تخرج دائماً من حالة الادخال INSERT، ويمكنك أن تدخل سجلاً واحداً فقط فى نفس الوقت، فيما عدا حالتين، عندما يتصرف أمر INSERT مثل أمر APPEND:

١ – عندما تدخل سجلات بعد آخر سجل فى الملف.
٢ – إذا كان الملف مفهرساً.

فى هذا المثال.. ادخل السجل الجديد بعد السجل الموجود المؤشر عنده، ويأخذ السجل الجديد هذا يأخذ – فى هذه الحالة – الرقم 8 كرقم له، مع تغيير أرقام السجلات التالية بأخذ وجود هذا السجل فى الاعتبار.

يعمل INSERT الآن مثل APPEND

```
.GO BOTT      <cr>
.INSERT       <cr>
```

يمكنك كذلك ادخال السجل الجديد، قبل السجل الحالى.

```
.7      <cr>
.INSERT BEFORE      <cr>
```

كما هو الحال دائماً.. فإنك تقدم قناع الهيكل، وبعد حفظ السجل الجديد.. تتغير أرقام السجلات؛ لتأخذ وجوده فى الاعتبار. وفى هذه الحالة.. يأخذ السجل الجديد الرقم 7 كرقم له.

```
.INSERT BLANK      <cr>
```

هذا يدخل سجلاً فارغاً بعد السجل الحالى، ويمكن بعد ذلك تنقيح هذا السجل الفارغ.

```
.INSERT BEFORE BLANK
```

هذا يدخل سجلاً فارغاً قبل السجل الحالى، ويمكن بعد ذلك تنقيح هذا السجل الفارغ.

ملاحظة :

إذا كانت لديك - على سبيل المثال - عدة آلاف من السجلات فى الملف، وقمت بإدخال سجل عند الموقع 15 مثلاً فى الملف.. فعليك أن تلاحظ أن dBASE يضيف كل السجلات الموجودة بعد السجل الذى ادخلته حديثاً. وطبقاً لعدد السجلات.. يمكن أن يستغرق هذا وقتاً كبيراً! أريد أن أركز مرة أخرى على أنه لا داعٍ لاستخدامك أمر INSERT فى اضافة أحد السجلات؛ إذ يجب اضافة APPEND السجلات الجديدة باستخدام أمر الاضافة APPEND، ويجب استخدام معالم الترتيب sorting أو الفهرسة indexing؛ للحصول على تأثير التسلسل الذى تريده. وكما ترى - فيما بعد - تستطيع أن تجعل dBASE يحفظ تسلسل الملف تلقائياً، أثناء إضافتك سجلات أكثر. وإيجازاً... إننى لا أرى - فى الواقع - حاجة حقيقية لاستخدام أمر INSERT.