سنقدم لك في هذا الفصل نظرة عامة على نظم إدارة قواعد البيانات بصفة عامة وقاعدة البيانات Access بصفة خاصة، وأنواع قواعد البيانات، ثم نوضح فكرة قواعد البيانات العلائقية وأنواعها، وأخيرا خطوات إعداد نظام جيد لإدارة قواعد البيانات بصفة عامة.بانتهاء هذا الفصل ستعرف:

- الفرق بين قواعد البيانات ونظم إدارة قواعد البيانات.
- نظرة عامة على قاعدة البيانات Access 2007 .
- فكرة قواعد البيانات العلائقية وأنواعها.
- خطوات إعداد نظم إدارة قواعد البيانات.

برامج قواعد البيانات من أوسع برامج الحاسب الشخصي انتشاراً ويمكن استخدامها في مختلف المجالات التجارية والصناعية ومنها على سبيل المثال لا الحصر مجالات النظم المحاسبية والمالية ومجال شئون الأفراد ومجال التخزين والمشتريات ومجال الإنتاج ومجال التسويق ومجالات الإدارة العليا. وفيما يلي نوضح معنى كل من قاعدة البيانات ونظم إدارة قواعد البيانات والفرق بينهما، ثم نلقى نظرة عامة على قاعدة البيانات Access، ثم نشرح أنواع قواعد البيانات عموما وقواعد البيانات العلائقية بصفة خاصة، وأخيرا خطوات إعداد نظم إدارة قواعد البيانات.

المقصود بقاعدة البيانات

عبارة عن تجميع لكمية كبيرة من المعلومات أو البيانات وعرضها بطريقة أو بأكثر من طريقه تسهل الاستفادة منها فمثلاً دليل الهاتف الذي يشتمل على أسماء وعناوين وأرقام هواتف سكان مدينة القاهرة يمكن أن نعتبره قاعدة بيانات، وتتحقق الاستفادة من قاعدة البيانات هذه بإدخال رقم المشترك والحصول على اسمه وعنوانه أو إدخال اسم المشترك والحصول على رقم هاتفه وعنوانه... وهكذا.

فكرة عامة عن قاعدة البيانات Access

قاعدة البيانات Microsoft Access واحدة من أشهر وأقوى قواعد البيانات التي تعمل على الحاسب الشخصي والتي تستخدم في ترتيب قواعد البيانات واستخراج النتائج منها وعمل الاستفسارات اللازمة. تم تطوير قاعدة البيانات Access عام ١٩٩٢ بواسطة شركة "مايكروسوفت"، وتم تعريبها في عام ١٩٩٤. وطورت الشركة المنتجة منها عدة إصدارات على النحو التالي ١.٠ ثم ١.١ ثم ٢ ثم ٩٥، ثم ٩٧ ثم٢٠٠٠ ثم ٢٠٠٢ ثم ٢٠٠٣. وأخيراً ٢٠٠٧، ويشرح هذا الكتاب الإصدار الأخير وهو Access 2007 .

تنتمي قاعدة بيانات Microsoft Access2007 إلي قواعد البيانات التي تسمي Relational Database Management System وتختصر هكذا RDBMS

ومعناها "نظم إدارة قواعد البيانات العلائقية" هذا من ناحية، ومن ناحية أخري فهي تنتمي إلي مجموعة Microsoft Office 2007 ولذلك فهي تستخدم نفس الواجهة الجديدة لبرنامج Microsoft Office 2007 .

قاعدة البيانات Microsoft Access عبارة عن برنامج رسومي يعمل تحت بيئة Windows الرسومية، ولذلك فهو يستفيد من جميع الإمكانيات والقدرات التي يوفرها Windows لمستخدميه، ويسمح بالوصول المرئي إلى البيانات بأقل جهد ويوفر طرقاً بسيطة ومباشرة لعرض البيانات والتعامل معها.

تستخدم Microsoft Access أدوات تصميم تعمل بطريقة ما تراه هو ما تحصل عليه، تساعد هذه الأدوات في استخراج نماذج وتقارير متطورة تفي باحتياجاتك تماماً، حيث يمكنك استخدام الرسوم والصور، وجمع أكثر من نموذج وتقرير في وثيقة واحدة وعرض النتائج بنمط يرقى إلى مستوى النشر.

توفر Microsoft Access مجموعة متنوعة من الكائنات التي يمكنك استخدامها لعرض المعلومات وإدارتها، ويمكنك استخدام وحدات الماكرو والوحدات النمطية لمزج الكائنات الموجودة في قاعدة بيانات داخل نظام إدارة البيانات المناسب لاحتياجك، كل ذلك دون الحاجة إلى البرمجة. أما المبرمجين ومن يحتاجون لتطوير نظم إدارة قواعد بيانات شديدة التخصص، فبإمكانهم استخدام Access VBA التي تعد لغة برمجة قوية وميسرة لقواعد البيانات، وهى لغة متضمنة في Access.

ومن مزايا Access تسهيل عملية البحث عن البيانات وتحليلها وصيانتها وحمايتها، حيث يتم تخزين البيانات المرتبطة في مكان واحد تقريبا. فعلى سبيل المثال، يمكنك إنشاء قاعدة بيانات لتخزين كل البيانات المرتبطة بعمل متكامل يشمل بيانات العملاء والمنتجات والمبيعات في مكان واحد.

تستخدم قاعدة البيانات Microsoft Access لإنشاء تطبيقات قوية لخدمة مستفيد واحد أو عدة مستفيدين داخل شبكة اتصالات، وهي التي تسمي تطبيقات Client/Server (خادم/عميل) وتستخدم نظام 32-bit "٣٢ بت". ويمكن أن تعمل

تحت نظام التشغيل Windows Vista/XP. من أسباب نجاح Access والتي جعلت لها قبولاً كبيراً لدي كثير من الناس أكثر من غيرها من قواعد البيانات الأخرى مثل Visual Fox Pro أو File Maker Pro، أنها تباع ضمن مجموعة برامج Microsoft Office التي تتمتع بشهرة هائلة علي مستوي العالم، ودعم قوي من أكبر شركة لإنتاج البرامج في العالم . بالإضافة إلي أنها تستطيع تطوير تطبيقات قوية وسريعة. سبب آخر لزيادة الإقبال علي Access وزيادة انتشارها حول العالم هو قدرتها علي أن تضع علي أجهزة الحاسب الشخصي PC ، إمكانيات نظم إدارة قواعد البيانات العلائقية التي تستخدم في الشبكات والتي تسمي Client/Server Relational Database Management System أو Client/Server RDBMS (وتسمي أيضاً SQL Databases) وميزة قواعد البيانات من نوع Client/Server Relational Database Management System أنها تستطيع نقل تطبيقات قواعد البيانات التي تستخدم علي أجهزة الحاسبات الكبيرة والتي تسمي Main Frames، والتي تتسم بتكلفتها الباهظة سواء في الشراء أو التشغيل، علي أجهزة الحاسبات الشخصية (PCs)، التي تستخدم داخل شبكة اتصالات، والتي تتميز برخص تكلفتها سواء في أسعار الحاسبات أو الشبكات التي تستخدمها .

ومن المعروف أن نظم إدارة قواعد البيانات التي تعمل علي الحاسبات الشخصية سواء التي تخدم مستفيد واحد أو عدة مستفيدين داخل شبكة اتصالات أسهل في تعلمها وبرمجتها حيث يستطيع أي شخص مبتدئ أن يطور هذه التطبيقات .

واحدة من أكبر أولويات شركة Microsoft بالنسبة لـ Access 2007 أن توسع قاعدة المستخدمين الجدد ، وذلك عن طريق التركيز علي التطبيقات الشائعة والتي تم تطويرها داخل حقيبة القوالب الموجودة في موقع Microsoft علي شبكة الويب ضمن حزمة قوالب Office والتي يمكن تنزيلها علي حاسبك والمعروفة باسم Office Online. وتتميز هذه القوالب أنها تستخدم أعظم المزايا الموجودة في Access 2007 في التطبيقات التي تقوم بتطويرها .

تمكن القوالب الجديدة التي توجد في Access 2007 أي شخص مبتدئ من تطوير تطبيق

بسيط لقاعدة بيانات في دقائق قليلة، وهذا يعتبر أبلغ رد علي من يجدون صعوبة في تعلم وبرمجة Access .

من أكبر الأشياء التي تساعدنا في تطوير تطبيقات قواعد البيانات لعدة مستخدمين داخل شبكة اتصالات باستخدام Access اشتمالها علي ما يسمي Access Data (ADP) Project "مشروعات بيانات Access" وهي التطبيقات التي تضع جميع كائنات التطبيق (مثل الجداول والتقارير والنماذج) في ملفات من نوع accdb .

تشتمل Access 2007 علي العديد من المعالجات والوسائل المصممة للمستخدم المبتدئ. ولذلك فإذا كنت ما تزال تستخدم إصدارات قديمة ، ننصحك بأن تجعل الترقية إلي Access 2007 واحدة من أكبر أولوياتك .

تستطيع Access 2007 التعامل مع عدد كبير من أنواع الملفات مثل جداول HTML وملفات XML وقائمة البريد الالكتروني الموجودة في برنامج عملاء Outlook أو جداول ورقة عمل Excel أو أي بيانات أخري علي شكل جدول حتى ولو كانت ملفات نصية.

إذا كنت تنوي استخدام Access 2007 في نظام لإدارة قواعد البيانات داخل شبكة اتصالات Client/Server RDBMS يجب أن تستخدم SQL Server 2005 Express Edition (وتختصر هكذا SSX) أو SQL Server 2005. تسمح لك Access 2007 أن تكرر نسخة من البيانات بين SSX أو SQL Server 2005 داخل شبكة اتصالات.

نظم إدارة قواعد البيانات Database Management Systems

لما كانت قاعدة البيانات تساعد في تحقيق مجموعة من الأهداف المؤثرة على الأنشطة الرئيسية في مجالات تطبيقات التجهيز الآلي للبيانات، فإنه يلزم وجود نظم معينه لتنظيم وإدارة البيانات المخزنة، وهو ما يطلق عليه عبارة "نظم إدارة قواعد البيانات" أو Database Management Systems (وتختصر هكذا DBMS). وعلى ذلك يمكن تعريف نظم إدارة قواعد البيانات على النحو التالى :

هى مجموعة من البرامج الجاهزة التى تقوم بتنفيذ جميع الوظائف المطلوبة من قاعدة البيانات، فمثلاً بعد إضافة عملاء جدد لدليل الهاتف في مدينة القاهرة فإنك قد تحتاج لإعادة ترتيب أسماء المشتركين أبجدياً أو لترتيب عناوينهم، مثل هذا العمل يطلق عليه إدارة قاعدة البيانات.

يتكون نظام إدارة قاعدة البيانات من مجموعة من الملفات بالإضافة إلى البرنامج أو مجموعة البرامج التى تتضافر لحل مشكله أو لتحويل نظام يدوى إلى نظام يعمل بالحاسب مثل تحويل نظام حسابات العملاء أو حسابات المخازن من نظام ودفاتر يدوية إلى نظام وملفات تستخدم بواسطة الحاسب. ويمكن أن يشتمل النظام على برنامج واحد وفي هذه الحالة فإن هذا البرنامج مع ملفات النظام يطلق عليه نظام إدارة قاعدة البيانات أو قد يشتمل على مجموعة من البرامج – بالإضافة إلى ملفات النظام – وفي هذه الحالة فإن هذه البرامج مجتمعه يطلق عليها نظام إدارة قاعدة البيانات.

قواعد البيانات العلائقية (Relational Databases)

أبسط صورة لقواعد البيانات العلائقية هى تلك التى تستخدم ملفا واحدا لتوضع بداخله البيانات. يسمى هذا الملف جدول أو **Table**، وهذا الجدول عبارة عن أعمدة وصفوف تشبه أعمدة وصفوف ورقة العمل التى تستخدمها برامج الجداول الحسابية مثل **Excel** أو **Lotus 1,2,3**. وفي هذا النوع من قواعد البيانات لا توجد علاقة أو رابط بين جدول البيانات وأى جداول أخرى. وعادة تستخدم هذه الجداول في أغراض بسيطة ومحددة مثل ملف العملاء الذى يشتمل على اسم العميل وعنوانه وبياناته ورقم هاتفه. (انظر شكل ١-١)

العملاء : جدول

رقم العميل	اسم العميل	العنوان	رقم الهاتف	الرصيد
١	ابنى محمد عبد العزيز	٣ش عمان الاسكندرية	٣٤١٤٠٢٠	٢٠٨٢
٢	يوسف مصيلحى جابر	٣ ش سعد زغلول الاسكندر	٥٤١٨٤١٥	١٠١٠
٣	رياض ماهر حنا	٤٣ ش الدقى	٢١٥٥٤٤٨	٤٠٠٠
٤	زكي عبد الودود عبد الفتاح	٤ ش الشهيد أحمد حمدى	٧٠٣٦٥٤٨	٥٠٠
٥	عوض محمود حمدى	١١ ش قناة السويس	٧٠٣٥٩٧٤	٢٠٠
٦	حلمي فهيم أحمد	٢١ ش الثورة بالمهندسين	٢١٥٤٤٩٤	٦٠٠
٧	حسين فريد على	٨ ش التحرير	٢١٥٣٥٢٧	٨٠٠
٨	ثروت محمود حمدى	٤٦ ش بورسعيد	٦٥٤٥٦٤٥	٢٥٤٠
٩	شريف راشد عبد الحميد	٥٦ ش الملكة زبيدة	٢١٥٦٥٤٨	٥٠٠
١٠	نادية مرسي نصار	٢٦ ش الجزيرة الوسطى	٥٤٦٤٦٤٥	٥٩٨٧
١١	هالة عبد الرحمن طه	٣٥ ش وزارة الزراعة	٥٤٤٥٤٦٤	٢٣٦٥
١٢	هارون مصطفى خليل	٢٤ ش مرقص حنا		٢١٤٥

السجل: ١٢ من ٥٤

شكل ١-١جدول "العملاء"

إلا أن مثل هذا الجدول لا يصلح للتطبيق والاستخدام العملى بدون ربطه مع جداول أخرى، فمثلا لا يمكنك استخدام عنوان العميل الموجود في دليل الهاتف مع طلبات الشراء التى ترد من العميل نفسه.

في هذه الحالة ستضطر لإدخال اسم العميل وعنوانه في كل مرة تستلم فيها طلباً جديداً للشراء وتحتاج لإضافة طلب الشراء إلى قاعدة البيانات.

والمشكلة هنا أنك تضطر لتكرار إدخال نفس البيانات مرتين. الأولى في دليل الهاتف والثانية في طلبات الشراء. ليس ذلك فقط، وإنما قد تضطر لتعديل بيانات العميل أو عنوانه إذا تغير في كل الجداول التى ورد فيها، ولاشك أن هذه المشكلة تسبب ضياع كثير من الوقت والجهد ووسائط التخزين التى نحتاج دائما لتوفيرها.

والحل الأمثل لهذه المشكلة هو إنشاء جدولين. جدول للعملاء وجدول لأوامر الشراء ثم تربط الجدولين. بعبارة أخرى تنشئ علاقة بينهما. والجداول التى يتم ربطها معاً في قاعدة بيانات واحدة تسمى قاعدة بيانات علائقية.

في قواعد البيانات العلائقية يتم تخزين البيانات في عدة جداول ثم يتم تعريف علاقة بين البيانات الموجودة بهذه الجداول ويتم ربط الجداول بحقول مشتركة في كل منها، ويوضح شكل ٢-١ قاعدة بيانات علائقية تشتمل على ٤ جداول تم ربطهم على النحو التالى:

تم ربط جدول "العملاء" بجدول "الاعتمادات" بناء على "رقم العميل" المشترك بينهما، وربط جدول "الاعتمادات" بجدول "بيانات الشركة" بناء على رقم الاعتماد المشترك، بينما تم ربط جدول "بيانات الشركة" بجدول "بيانات المنتج" بناء على "رقم المنتج"....وهكذا .

شكل ٢-١ نموذج قاعدة بيانات علائقية

أنواع علاقات الارتباط

تسمح Access بإنشاء ٣ أنواع من العلاقات بين الجداول في قواعد البيانات العلائقية: علاقة "واحد مقابل واحد"، وعلاقة "واحد مقابل مجموعة"، وعلاقة "مجموعة مقابل مجموعة". ونوضح فيما يلى الفرق بين هذه الأنواع من العلاقات

علاقة واحد مقابل واحد one-to-one

تسمى هذه العلاقة حسب الترجمة الواردة بكتاب الشركة المنتجة لـ ACCESS "علاقة ارتباط رأس برأس". وفيه كل سجل في الجدول الرئيسى يقابله سجل واحد في الجدول المرتبط به. ومن الأمثلة التى تستخدم فيها علاقة واحد مقابل واحد، عندما ترغب في فصل معلومات العميل إلى بيانات عامة وبيانات خاصة، فمثلا يمكن أن تضع معلومات عامة عن العميل مثل الاسم والعنوان في الجدول الرئيسى وتضع معلومات خاصة عن العميل مثل الرصيد في الجدول الثانى.

استخدام الجدول الثانى يسهل لك الوصول إلى البيانات التى تريدها. ولأن كل سجل في الجدول الأول يقابله سجل في الجدول الثانى فان العلاقة بين هذه الجداول تسمى واحد مقابل واحد.

يشتمل شكل ٣-١ على مثال لعلاقة واحد مقابل واحد وفيه يقابل كل سجل من جدول "الموظفين" سجلاً من جدول "بيانات شخصية" ، ومن شكل ٣-١ نلاحظ أن كلا الجدولين تم ربطهما بحقل مشترك ، ألا وهو "رقم الموظف" .

الجداول ذات العلاقة "واحد مقابل واحد" نادرة الاستخدام وذلك لأن الجدولين يمكن دمجهما معاً في جدول واحد إلا أن فصلهما قد يحقق بعض الأغراض مثل السرية في بعض البيانات

شكل ٣-١ مثال لعلاقة ارتباط واحد مقابل واحد (رأس برأس)

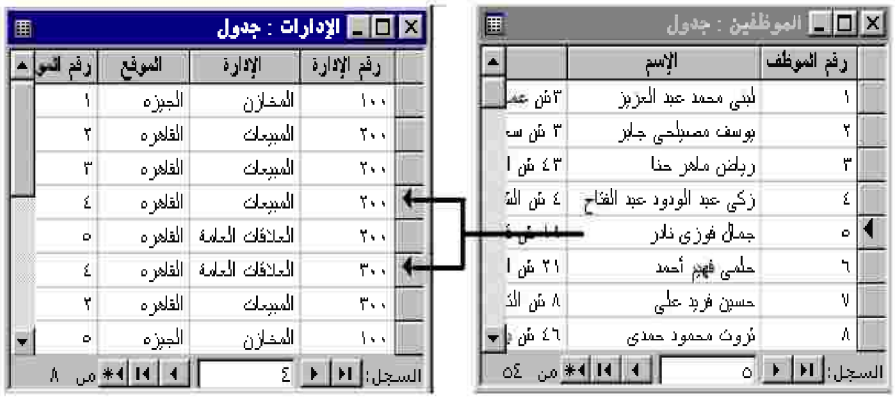
علاقة واحد مقابل مجموعة one-to-many

تسمى هذه العلاقة حسب الترجمة الواردة بكتاب الشركة المنتجة لـ ACCESS "علاقة ارتباط رأس بأطراف" ومعناها أيضا علاقة واحد مقابل مجموعة. وهي الأكثر استخداما. وتعنى أن السجل الواحد في جدول البيانات (يسمى الجدول الرئيسي أو Primary Table) يقابله أكثر من سجل في جدول آخر(يسمى الجدول المرتبط أو Related Table). يشتمل شكل ٤-١ على مثال لعلاقة واحد مقابل مجموعة وفيه يقابل كل سجل في جدول "المحافظات" وهو الجدول الرئيسى اكثر من سجل في جدول "العملاء" وهو الجدول المرتبط، وقد تم ربط الجدولين بناء على بيانات حقل "رقم المحافظة"

شكل ٤-١ مثال لعلاقة ارتباط واحد مقابل مجموعة (رأس بأطراف)

علاقة مجموعة مقابل مجموعة Many-to-many

تسمى هذه العلاقة حسب الترجمة الواردة بكتاب الشركة المنتجة لـ ACCESS "علاقة ارتباط أطراف بأطراف". وهذا النوع من العلاقة أيضا نادر الاستخدام وفيه يقابل كل سجل من الجدول الرئيسى عدة سجلات في الجدول المرتبط ، ويقابل السجل الواحد في الجدول المرتبط عدة سجلات في الجدول الرئيسى. ومن الأمثلة على ذلك في قاعدة البيانات التى تشتمل على جدول للموظفين وجدول للإدارات (انظر شكل ٥-١)، يمكن أن يقابل السجل الواحد في جدول "الموظفين" أكثر من سجل في جدول "الإدارات"، ومن الناحية الأخرى، من الممكن أن تجد لكل سجل في جدول "الإدارات" أكثر من سجل في جدول "الموظفين". فمثلا تلاحظ من شكل ٥-١ أن الموظف رقم ٥ (جمال فوزى) يعمل في كل من إدارتي العلاقات العامة والمبيعات ولذلك يخصه سجلين في جدول "الإدارات".



شكل ٥-١مثال لعلاقة ارتباط مجموعة مقابل مجموعة (أطراف بأطراف)
الموظف رقم ٥ في جدول "الموظفين" يظهر في إدارتين في جدول "الإدارات"

هذا النوع من العلاقات معقد ويحتاج لدراية كافية بـــ Access لأنه من الممكن أن يسبب مشكلة ما لـــم تتدخل لربط الجدولين بأسلوب غير مباشر يتلخص في إنشاء جدول ثالث يعمل على تجزئة علاقة ارتباط "مجموعة مقابل مجموعة" إلى علاقتين من نوع "واحد مقابل مجموعة"، وفي هذه الحالة تضع المفتاحين الأساسيين لكلا الجدولين في الجدول الثالث، ويحقق هذا الأسلوب ميزة التخلص من البيانات غير الضرورية ومنع التكرارات في البيانات.

فمثلا أسماء الإدارات وأماكنها تظهر أكثر من مرة في جدول "الإدارات". والتصميم الجيد لقاعدة البيانات يحل هذه المشكلة بإنشاء جدول ثالث يعمل على تجزئة علاقة ارتباط "مجموعة مقابل مجموعة" إلى علاقتين من نوع "واحد مقابل مجموعة". يوضح شكل ٦-١ كيف يمكن تجزئة علاقة "مجموعة مقابل مجموعة" إلى علاقتين من نوع "واحد مقابل مجموعة" عن طريق إضافة جدول ثالث هو جدول "موظفين وإدارات"

شكل ٦-١ تجزئة علاقة ارتباط مجموعة مقابل مجموعة (أطراف بأطراف) إلى علاقتين من نوع "واحد مقابل مجموعة" عن طريق جدول ثالث

مزايا قواعد البيانات العلائقية

من الشرح السابق يمكن إيجاز المزايا التى تحققها قواعد البيانات العلائقية فيما يلى:

- *المرونة*

ذلك لأن الجدول الواحد ضمن قاعدة البيانات العلائقية يكون صغيراً لأنه يشتمل عادة على حقول قليلة تنتمى إلى فصيلة أو مجموعة واحدة من المعلومات لأنك تستطيع أن تربط أى عدد من الجداول حسب ما تراه ضروريا ومناسباً. والتغييرات التى تتم على جدول لاتوثر على بقية الجداول في قاعدة البيانات.

- *زيادة كفاءة تنظيم البيانات*

تسمح قواعد البيانات العلائقية بإنشاء علاقة بين بيانات الجداول المختلفة لإظهار البيانات بأشكال متعددة .

- *زيادة كفاءة إدارة البيانات*

تسهل قواعد البيانات العلائقية التعامل مع البيانات لأن الجداول تكون حقولها قليلة وبالتالى يسهل التعامل معها بعكس قاعدة البيانات التى توضع في جدول واحد لأنه

يكون كبيراً ومعقداً

- *اختصار البيانات*

ولتوضيح ذلك سنوضح لك الفرق بين حالتين. الحالة الأولى تستخدم جدول واحد لقاعدة البيانات لتسجبل بيانات الفاتورة وتفصيلاتها ، والحالة الثانية قاعدة بيانـــات تستخدم جدولين الأول يشتمل على بيانات الفاتورة و الثانى على تفصيلات فاتورة البيع مع ملاحظة ربطهما معاً طبقاً لبيانات حقل رقم الفاتورة .

في الحالة الأولى (حالة استخدام جدول واحد لقاعدة البيانات)، سيتم تسجيل سطر لكل صنف موجود بالفاتورة ولأن الفاتورة الواحدة يمكن أن تشتمل على عشرات الأصناف. فلابد من إدخال سجل مستقل لكل صنف موجود بالفاتورة، وفي هذه الحالة سيتم تكرار بيانات العميل مع كل صنف يتم تسجيله بالجدول وهذا معناه استخدام كم هائل من البيانات لا داعي له، أو يمكن اتباع طريقة أخرى وهى إضافة حقل لكل صنف من الأصناف التى يمكن أن ترد بالفاتورة وفي هذه الحالة إذا اشتملت الفاتورة على صنفين فقط في حين أن حقول الأصناف عددها ١٠ مثلاً فان ثمانية حقول ستبقى خالية من البيانات ، ولاشك أن هذا المفهوم يسبب ضياع مساحة كبيرة على القرص المغناطيسى.

وفي الحالة الثانية (حالة استخدام قاعدة بيانات علائقية) يمكنك إنشاء جدولين. جدول للفواتير وجدول لتفصيلات الفواتير ثم ربط سطر واحد لكل فاتورة بسطر أو أكثر من أصناف الفاتورة. وهذا المفهوم يسمح بتسجيل أى عدد من الأصناف، لكل صنف سطر مستقل ثم ربط كل أصناف الفاتورة الموجودة بجدول التفصيلات بالسطر الخاص بالفاتورة في جدول الفواتير

والتوفير هنا يتضح من أن بيانات الفاتورة تسجل مرة واحدة في سطر واحد في جدول الفواتير، وليس في كل سطر من سطور جدول "تفصيلات الفاتورة"

يطلق على هذا المفهوم من العلاقة بين الجداول "واحد مقابل مجموعة" لأن كل فاتورة يمكن أن يخصها أكثر من صنف

خطوات إعداد نظام إدارة قاعدة البيانات

يجب أن يسبق كتابة برامج نظام إدارة قاعدة البيانات تصميم جيد لأن التصميم الجيد لا يقل أهمية عن كتابة البرامج أو تنفيذها. وفي النظم الصغيرة التى تقتصر مهمتها على وظيفة واحدة مثل إعداد شاشات الإدخال للنظام أو استخراج تقرير يمكنك كتابة البرنامج اللازم للنظام مباشرة بدون حاجه لإعداد تصميم سابق لهذا النظام. أما في النظم الكبيرة والمعقدة فلابد أن نضع خريطة للنظام توضح البرامج التى يشتمل عليها هذا النظام وعلاقة كل برنامج بالآخر. وفيما يلى سنوضح خطوات التصميم الجيد لنظام إدارة قواعد البيانات.

نتناول فيما يلى خطوات التصميم الجيد لنظام إدارة قواعد البيانات ثم نورد اقتراحات هامة لتحسين أداء النظم التى يتم تطويرها باستخدام قاعدة البيانات **Access**.

تحديد الهدف من النظام

يجب أن يكون الهدف من النظام أو وظيفة النظام واضحة تماماً في ذهن الشخص الذى سيقوم بإعداد هذا النظام وتتطلب هذه المرحلة وجود شخص له دراية فنية بالنظام أو الجلوس مع شخص ذو دراية فنية بالنظام. فمثلاً لإعداد نظام محاسبة يجب أن يعده محاسب أو أن يجلس أخصائي الحاسب أو المبرمج مع المحاسب ليشرح له طريقه العمل اليدوية والمستندات المستخدمة والدورة المستندية التى تمر بها إلى مستندات مماثلة باستخدام قاعدة البيانات فإذا أصبحت أهداف النظام واضحة فيجب الانتقال إلى المرحلة التالية.

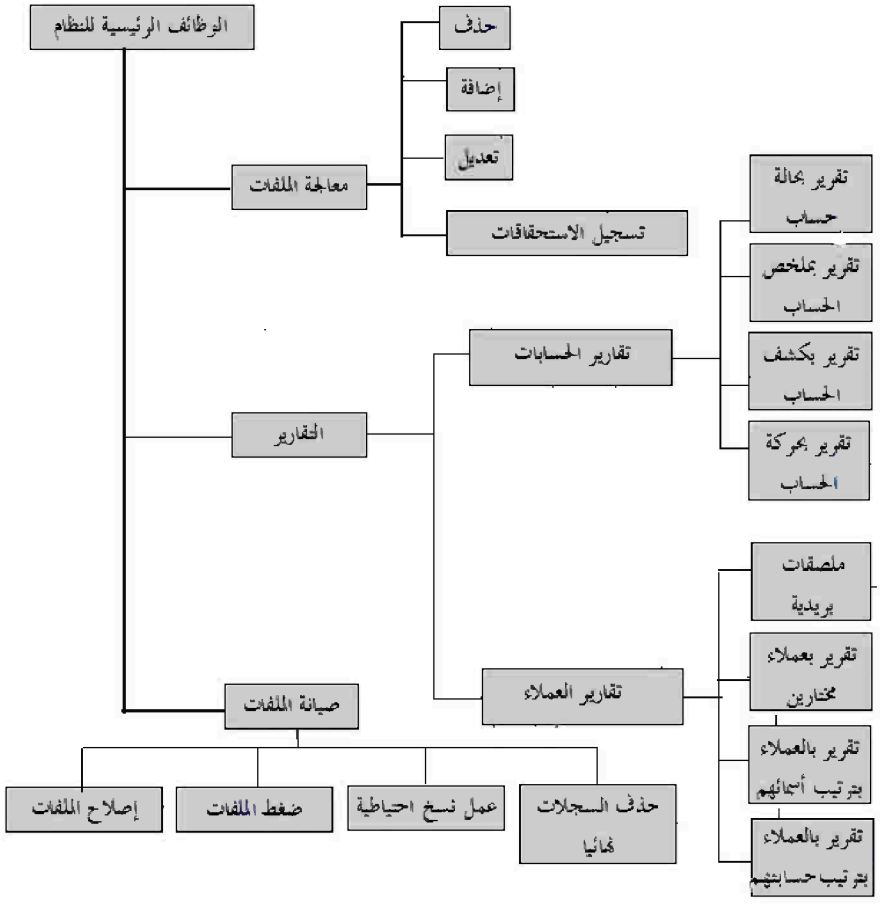
تحديد متطلبات النظام

وفي هذه الخطوة يتم تحديد الملفات المستخدمة في النظام بدقه ووضوح مثل أسماء الملفات والحقول التى تشتمل عليها وطرق ترتيبها والبحث فيها والتقارير المستخدمة وأشكالها وشكل شاشات الإدخال ومحتوياتها... الخ كما يجب تحديد طريقه الحصول على الخدمات التى يتطلبها مثل نسخ الملفات على أقراص تحفظاً للتلف أو الفقدان أو استرجاع الملفات عند الحاجة إليها ويتم أيضاً اختيار شكل القوائم الرئيسية التى يشتمل عليها النظام.

وضع خريطة للنظام

بعد ذلك يجب إعداد خريطة للنظام تشتمل على الوظائف الأساسية المطلوبة من النظام. يشتمل شكل ٧-١ على خريطة نظام حسابات العملاء ومنه تلاحظ أننا قسمنا النظام إلى وظائف والوظائف إلى وظائف فرعية ويتم تقسيم النظام إلى وظائف أساسية والوظائف إلى وظائف فرعيه تبعاً للتحليل الأولى الذى أعددته للنظام. وتعتمد هذه الخريطة على مفهوم يسمى التصميم من أعلى لأسفل

نظام حسابات العملاء



شكل ٧-١ خريطة نظام حسابات العملاء

إعداد النظام

تشتمل هذه المرحلة على عدة خطوات نوجزها فيما يلى:

- تصميم الكائنات (أو المصادر) التي يتكون منها نظام قاعدة البيانات (مثل الجداول والنماذج والتقارير والاستفسارات ...الخ)
- تحديد الروابط بين الكائنات التى يتكون منها النظام.
- تحديد خصائص الكائنات التى يتكون منها النظام،وكيفية ربط خصائص الكائنات مع بعضها.
- إنشاء قاموس لتعريف محتويات الجداول التى يتكون منها النظام.
- تصميم العلاقة بين الجداول التى يتكون منها النظام (راجع بند قواعد البيانات العلائقية في هذا الفصل) بناء على الروابط الموجودة في الجداول.
- تحديد أنواع التعديلات والتحديثات التى من شأنها إنشاء أو تعديل بيانات الجداول.
- تحديد كيفية استخدام الفهارس ومتى تستخدمها لتسريع الاستعلامات.
- تحديد الشخص الذى له صلاحيات التعامل مع البيانات وتعديلها عند الضرورة (تأمين البيانات).

اختبار النظام

بعد الانتهاء من إعداد النظام واختبار كل مرحلة من مراحله على انفراد يجب اختبار النظام كله. وفي هذه المرحلة يجب إدخال بيانات اختباريه يتم تجربة النظام عليها. ويجب أن يستخدم النظام بجميع الاحتمالات الواردة لاستخدامه من قبل المستخدم النهائى. ولذلك فيجب أن تجلس أمام النظام وتقضى وقتاً طويلاً في اختباره حتى تتأكد أن النظام خالياً من الأخطاء ومن المفيد أن تدخل بيانات خاطئة لتتأكد أن النظام يكتشفها. وبعد تجربة كل حالة من حالات استخدام النظام راجع بيانات الملفات لتتأكد أن السجل المطلوب إدخاله أو تعديله أو حذفه تم إدخاله أو تعديله أو حذفه فعلاً، بمعنى آخر ضع نفسك مكان المستخدم النهائي للنظام الذي ليست لديه دراية عن النظام وفكر بنفس الطريقة التي يفكر

بها هذا الشخص، أو اطلب من شخص آخر تجربة النظام لأنه سيحاول تنفيذ الأشياء التي لم ترد بحساباتك وبالتالي فإنه يكتشف الأخطاء التي قد لا تكتشفها أنت.

توثيق النظام

والتوثيق المقصود ذو شقين :

الشق الأول: توثيق خطوات الحل السابقة وتشمل : أهداف النظام وخريطة النظام ومحتويات البرامج التى يشتمل عليها النظام ويجب أن توضع كل هذه المستندات داخل ملف خاص بالنظام ليمكن الرجوع إليه في المستقبل عندما تحتاج لتعديل النظام لأنك إذا حاولت الرجوع إلى نظام أعددته منذ عام مثلاً فستجد أنك نسيت كثيراً من التفاصيل الضرورية ومن مزايا هذا التوثيق تسهيل عمل أى شخص آخر غير الشخص الذى قام بإعداد النظام إذا أراد إجراء تعديل أو الرجوع إلى أحد البرامج.

الشق الثانى : إعداد كتيب صغير على أهداف النظام بالإضافة إلى التعليمات الضرورية لتشغيل النظام ويجب أن يكون الكتيب مختصراً وواضحاً بقدر الإمكان. كما يفضل أن يشتمل على أمثلة إرشادية للمستخدم في الحالات التى تحتاج لذلك.وتتوقف طريقة إعداد كتيب التشغيل على مستوى المستخدم النهائى الذى سيتولى تشغيله فكلما كان المستخدم بعيداً عن النظام كلما احتاج لتوضيح أكثر.