يأتى هذا الفصل كمقدمة لكتابة الإجراءات أو البرامج، ولأن التعليمات تكتب دائما داخل إجراءات أو وحدات نمطية،فمن الضروري أن تعرف كيف تتعامل مع نافذة الإجـــراء أو الوحـــدة النمطية ، بالإضافة إلى مفاهيم أساســـية تســـاعدك فى فهـــم لغـــة Access VBA وفى الفصل التالي نشرح كيفية كتابة تعليمـــات Access VBA بانتهاء هذا الفصل ستتعرف على:

- متى يتم استخدام كود Access VBA
- فلسفة البرمجة باستخدام Access VBA .
- الوحدات النمطية والدوال والإجراءات الفرعية .
- وصف نافذة الوحدة النمطية.
- البرامج والإجراءات والدوال.
- طلب المساعدة أثناء البرمجة.

يتكون البرنامج من مجموعة من الأوامر والتعليمات تطلب من الحاسب الوصول إلى نتيجة معينة.مثل حساب معادلة رياضية أو قراءة ملف من على الأقراص أو تغيير خاصية في أداة أو طباعة النتائج أو استخدام بعض أوامر الرسم على شاشة الكمبيوتر.

لغة البرمجة التي تستخدمها Access

تستخدم Access 2007 لغة Access VBA وهى إحدى لهجات اللغة الأم Visual Basic وليست لغة (Macros) مختزلات، والتى من الأفضل استخدامها بدلاً من المختزلات التى تقترب من الانقراض. حيث يمكنك بسهولة تامة استخدام دوال وإجراءات لغة VBA لإنشاء الكود المكافئ للمختزل. وعلى الرغم من إمكانية تنفيذ الإجراءات من داخل وحدة نمطية مفتوحة ، إلا أنك غالباً ما تقوم بتنفيذها أثناء الأحداث التى يقوم المستخدم بأدائها كنقر زر أمر مثلاً كما سنرى فيما بعد. كما يتم أيضاً استدعاء الدوال داخل أدوات التحكم الموجودة بالنماذج أو التقارير أو من خلال خاصية التحقق من الصحة Validation Rule الخاصة بالجدول أو بأحد حقوله أو حتى من داخل أحد إجراءات التطبيق.

متى يتم استخدام كود VBA ؟

تستخدم إجراءات VBA التعبير Docmd Object وهو غالباً ما يكون كافياً لفتح النماذج والتقارير أو تنفيذ الاستعلامات داخل التطبيقات البسيطة. كما يمكنك من خلال الدوال المبنية داخل VBA أداء العديد من العمليات الحسابية المركبة داخل الاستعلامات. وعلى ذلك فأنت فى حاجة لاستخدام VBA فى الحالات التالية:

- إنشاء الدوال المعرفة من قِبَل المستخدم User-defined Functions (UDFs) التى تستخدم لأداء التعبيرات المركبة المستخدمة لاختبار صحة البيانات أو حساب القيم الموجودة داخل مربعات النصوص والكثير من المهام الأخرى.
- كتابة التعبيرات التى تتكرر باستمرار داخل البرنامج ونحتاج فيها لاتخاذ قرار معين كما فى التعبير If ...Then...Else...End If.

- إجراء عمليات التبادل مع SQL Server 2005 Expression Edition عن طريق عبارات T-SQL .
- العمل مع أدوات ActiveX والكائنات الأخرى للتطبيق باستخدام الكود التلقائى.
- فتح أكثر من قاعدة بيانات داخل نفس التطبيق.
- تحسين توثيق التطبيق عن طريق طباعة كود التطبيق من خلال بيئة تطوير VBA .
- احتواء الأخطاء ومعالجتها أثناء التشغيل حيث يمكنك باستخدام VBA التحكم فى استجابة التطبيق للأخطاء مثل البيانات أو النماذج المفقودة أو القيم الغير صحيحة التى يقوم المستخدم بإدخالها والكثير من الأخطاء الأخرى. وهذا ما يتميز به VBA عن المختزلات التى لا يمكنها احتواء الأخطاء.

فلسفة البرمجة باستخدام Access VBA

قبل أن تظهر بيئة Windows الرسومية، كانت البرامج تحمل المستخدم على تنفيذ خطوات متسلسلة لا يمكن أن يحيد عنها، بحيث تنفذ كل خطوة قبل التى تليها، وعندما ظهرت بيئة Windows قدمت وجها جديدا للبرمجة سمح للمستخدم بالعمل بحرية أكثر.

هذا الوجه الجديد للبرمجة عرف بـ: *طريقة البرمجة المسيرة بالأحداث* Event-Driven. وهذا المفهوم يتطلب من البرنامج أن يستجيب للمستخدم بطريقة الاستجابة للحدث عند وقوعه.

فى البرمجة المسيرة بالأحداث تترك الحرية المطلقة للمستخدم لكى ينقر أو يضغط أو يدخل بيانات مثلا فى النموذج بالترتيب الذى يحبه ويرغبه، كما يمكن للمستخدم تجاهل العديد من عناصر التحكم.

إن Access عبارة عن بيئة مسيرة بالأحداث تشمل الأدوات والقوائم ومربعات إدخال البيانات والتى تنتظر أن تستجيب لحدث يحدد بواسطة البيانات أو المستخدم.كل ما فى بيئة Windows مسير بالأحداث،وفى أى وقت تقوم فيه بانتقاء خيار من القائمة، أو تشغل برنامج، أو تنتقل إلى نافذة مختلفة فأنت بذلك تقوم بحدث ما يسير Windows بموجبه.

يمكنك اعتبار بيئة Access المسيرة بالأحداث شبيهة بأزرار التحكم على جهاز الفيديو. عندما تستخدم جهاز الفيديو الخاص بك ، هل تضغط على الأزرار الموجودة عليه بالترتيب حسب الظهور؟ لا. ولكنك بدلا من ذلك تقوم بتغير القناة، ربما تهيء ساعة التوقيت، وقد تغلق الفيديو، وقد تعدل من سرعة الشريط أو قد تفعل ما يحلو لك بالترتيب الذى تحبه أنت وليس مفروضاً عليك. وكما تشاهد فإنه تم تصميم جهاز الفيديو بحيث يمكنك إطلاق أى حدث فى أى وقت تريده.

تختلف Access VBA عن بعض بيئات ولغات البرمجة الأخرى، حيث أن فلسفة البرمجة التى تستخدمها هذه اللغة تقوم على فكرة كتابة العديد من الأقسام الصغيرة من البرامج، وهى التى تسمى دوال أو إجراءات.وهذا فى الحقيقة هو الغرض الذى من أجله صممت الدوال والإجراءات الفرعية، لكى تكون أقسام صغيرة من البرنامج تبقى منتظرة وقوع الحدث (كأن يقوم المستخدم بانتقاء شيء ما من النموذج) قبل أن تقوم بعملها. فى حين أن الدوال والإجراءات الفرعية يجب أن تعمل مستقلة عن بعضها البعض وأن يكون بالإمكان تنفيذها بالترتيب الذى يحتاج إليه المستخدم أو البيانات.

تستخدم Access VBA طريقتين للبرمجة:

- الدوال والدوال المبنية Functions and Built -In Functions
- الإجراءات Sub Procedures

وفى هذا الكتاب سنركز على هاتين الطريقتين للبرمجة وهما: الدوال function procedures والإجراءات الفرعية Sub procedures . لاستخدام هذين النوعين ينبغى معرفة Access VBA وفهمها جيداً. حيث يحتوى هذان النوعان على أوامر برمجة Access VBA التى تنفذ عندما يقع حدث معين. ويتم استخدام كل منهما من خلال نافذة الوحدة النمطية.

يوفر عليك استخدام العديد من الدوال المبنية داخل Access الكثير من الوقت ومن عناء كتابتها بنفسك مرة أخرى لإنجاز بعض الأعمال الروتينية. وفى هذا الكتاب ستجد أمثلة كثيرة على استخدام الدوال والدوال المبنية. سنشرح الدوال بالتفصيل فى الفصلين الخامس والسادس من هذا الكتاب بإذن الـــله

الوحدة النمطية (Module)

الوحدة النمطية عبارة عن وعاء يحتوى على كود VBA مثلما يحتوى النموذج على أدوات التحكم. تحتوى Access 2007 على أربعة أنواع من الوحدات النمطية والتى يمكن بيانها كما يلى:

- *وحدة نمطية للبرنامج Access Module*

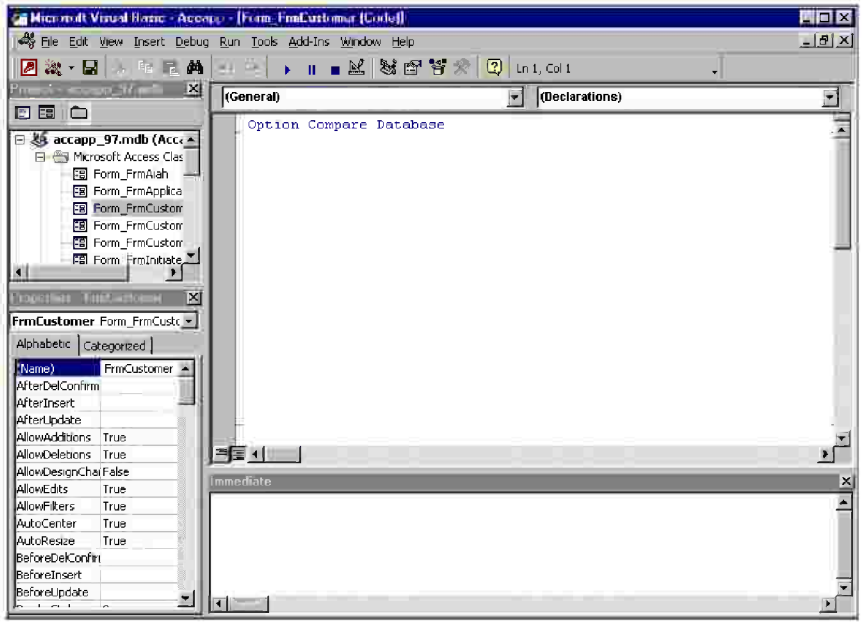
ويتم إنشاءها بنفس طريقة إنشاء كائنات Access المعتادة كالجداول والتقارير كى تحتوى على كود VBA الذى سيتم استخدامه من قِبَل جميع كائنات البرنامج.

لذا يطلق عليها أحياناً الوحدات النمطية القياسية أو Standard Module. لإنشاء (Module) أو وحدة نمطية جديدة ، تابع معنا الخطوات الآتية:

١. تأكد أن نافذة قاعدة البيانات مفتوحة.

٢. من نافذة قاعدة البيانات نشط التبويب Create ومن مجموعة Other انقـــر زر فتح وحدات نمطية ومن القائمة التي تظهر انقر زر جديد

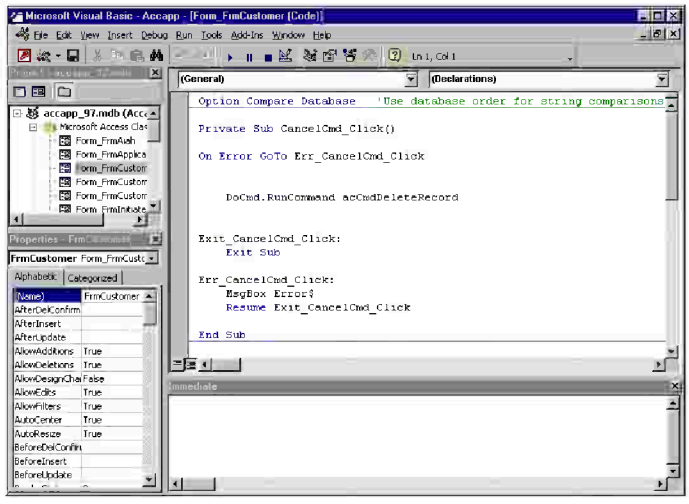
تظهر نافذة جديدة لوحدة نمطية (انظر شكل ١-٢).



شكل ١-٢ نافذة الوحدة النمطية.

- *وحدة نمطية للنموذج* **Form Module**

وتحتوى على الكود اللازم للاستجابة لأحداث النموذج أو أحداث عناصر التحكم الموجودة فى هذا النموذج. فحينما تقوم بإدخال كود أحد عناصر النموذج ، فأنت بذلك تقوم دون أن تدرى بإنشاء تصنيف جديد داخل قاعدة البيانات. لفتح الوحدة النمطية الخاصة بالنموذج من داخل نافذة التصميم ، اختر زر **Module** داخل مجموعة **Other** من داخل التبويب **Create** ومن القائمة التي تظهر انقر **Class Module** ، ستقوم **Access** بفتح وحدة نمطية داخل بيئة تطوير **VBA** تحتوى على اسم النموذج كما فى شكل ٢-٢.



شكل ٢-٢ الوحدة النمطية للنموذج.

- ***وحدة نمطية للتقرير Report Module***

وتحتوى على الكود اللازم للاستجابة لأحداث التقرير بما فى ذلك أقسام التقرير المختلفة ورأس وتذييل التقرير. يمكنك فتح الوحدة النمطية الخاصة بالتقرير بنفس الطريقة المتبعة لفتح الوحدة النمطية الخاصة بالنموذج.

- ***وحدة نمطية للتصنيف Class Module***

وهذه الوحدة النمطية لا ترتبط بنموذج أو تقرير ومن خلالها يمكنك تعريف كائناتك الخاصة من البداية وتحديد خصائصها والأحداث التى ستجرى عليها. وهذا النوع من الوحدات النمطية خارج نطاق هذا الكتاب، ويمكنك الرجوع إلى كتابنا "البرمجة المتقدمة باستخدام **Visual Basic**" لمعرفة المزيد عنه.

وصف نافذة الوحدة النمطية Module Window

يتم كتابة الدوال والإجراءات داخل نافذة الوحدة النمطية (محرر **VBA**). لفتح نافذة المحرر، تابع معنا الخطوات الآتية:

١. انتقل إلي مجموعة **Modules** من قسم التنقل داخل النافذة الرئيسية لكائنات قاعدة

البيانات.

٢. إذا أردت فتح وحدة نمطية موجودة مسبقاً، انقر اسم الوحدة النمطية نقراً مزدوجاً.

٣. إذا أردت إنشاء وحدة نمطية جديدة ، انقر زر **Module** من مجموعة **Other** داخل التبويب **Create** ثم انقر **Module** .

٤. في كلتا الحالتين السابقتين يظهر محرر **VBA** بالخيارات الافتراضية كما في شكل ٣-٢.

شكل ٣-٢ شرح لنافذة الوحدة النمطية.

من الشكل السابق ، تلاحظ احتواء محرر **VBA** على العناصر التالية:

- شريط أدوات نافذة الوحدة النمطية **Module Toolbar** .

ويشتمل على أزرار لتشغيل إجراءات الوحدة النمطية أو إيقافها أو الحصول على تعليمات المساعدة. يمكنك التعرف على وظيفة الأزرار الموجودة بشريط الأدوات بتوجيه مؤشر الفأرة إلى الزر المطلوب ثم الإبقاء عليه ثوانى معدودة.

يوضح الجدول التالي الأزرار التى يتكون منها شريط الأدوات الذي يظهر في نافذة الوحدة النمطية ووظيفة كل منها.

الزر	اسمه	وظيفته
	إظهار Access	إظهار نافذة Access 2007
	إضافة وحدة نمطية	إضافة وحدة نمطية جديدة أو تصنيف أو إجراء إلى قاعدة البيانات الحالية
	حفظ	حفظ التعديلات التى تمت على الإجراء الحالي
	تراجع عن	التراجع عن آخر عمل تم من لوحة المفاتيح أو بواسطة الفأرة
	إلغاء التراجع	إلغاء آخر تراجع
	بحث	البحث عن نص داخل الوحدة النمطية باستخدام المربع الحوارى find
	تشغيل	استمرار تنفيذ الإجراء
	إيقاف مؤقت	إيقاف تنفيذ الإجراء مؤقتاً
	إيقاف	إيقاف تنفيذ الإجراء نهائياً
	تصميم	العمل فى طور التصميم
	مستكشف المشروع	إظهار نافذة مستكشف المشروع
	خصائص	إظهار نافذة خصائص العنصر الحالى داخل نافذة مستكشف المشروع
	مستعرض الكائنات	إظهار نافذة مستعرض الكائنات Object browser
	أدوات التحكم	إظهار مربع أدوات التحكم Toolbox
	مساعدة	إظهار تعليمات مساعدة عن استخدام نافذة الوحدة النمطية

يمكنك تنفيذ الوظائف التى تنفذ بواسطة أزرار شريط الأدوات بواسطة الأوامر الموجودة بالقوائم.

- *شريط القوائم* ***Menu Bar***

ويشتمل على مجموعة من الأوامر المستخدمة لأداء مهام معينة داخل لغة **VBA**.

ويوضح الجدول التالي القوائم الخاصة بنافذة الوحدة النمطية والأوامر التى تؤدي بعض العمليات فقط، أما القوائم الشائعة أو المشتركة (مثل قائمة **Window**) وكذلك الأوامر التى تعمل مع كائنات قاعدة البيانات الأخرى (مثل أمر **Save** في قائمة **File**) فلا يشملها الجدول، وذلك لأننا نفترض أنها مألوفة لك من دراستك لكائنات قاعدة البيانات الأخرى.

القائمة	الأمر	وظيفته
ملف File	Import file	إحضار ملف من قاعدة بيانات خارجية
	Export file	تصدير ملف لقاعدة بيانات خارجية
	Print	يظهر مربع الطباعة Print لطباعة الوحدة النمطية الحالية أو طباعة كل المشروع
تحرير Edit	Find	يظهر مربع البحث Find للبحث عن نص أو تعبير داخل الوحدة النمطية
	Find Next	يبحث عن آخر نص وجده ويضعه تحت الشريط المضاء
	Replace	يستبدل كل أو أحد النصوص المكررة التى وجدها بنص آخر
	Indent	إزاحة النص للداخل بمقدار ٨ مسافات

القائمة	الأمر	وظيفته
	Outdent	إزاحة النص للخارج بمقدار ٨ مسافات
	List Properties/Methods	إظهار قائمة بخواص ودوال العنصر الحالى داخل نافذة المحرر للاختيار منها وإلا يقوم بإظهار الخواص والدوال العامة إذا كان مؤشر الفأرة فى مكان خالى
	List Constants	إظهار قائمة بثوابت الخاصية الحالية للاختيار منها
	QuickInfo	إظهار الصيغة العامة للدالة الحالية أو المتغير الحالى
	Parameter Info	إظهار معلومات عن معاملات الدالة الحالية
	Complete word	تكملة الكلمة تلقائياً بمجرد كتابة الحروف الأولى منها
	Bookmarks	إضافة الإشارات المرجعية وحذفها والتبديل بينها من خلال قائمة فرعية
عرض View	Code	إظهار أو تنشيط نافذة الكود للعنصر الحالى داخل نافذة مستكشف المشروع
	Object	إظهار كائن الوحدة النمطية (النموذج المصاحب للوحدة النمطية مثلاً)
	Definition	إظهار مكان تعريف المتغير أو الإجراء الحالى داخل الكود
	Last position	الذهاب مباشرةً إلى المكان السابق للمؤشر داخل الكود

القائمة	الأمر	وظيفته
	Object browser	إظهار نافذة مستعرض الكائنات
	Immediate window	إظهار النافذة الفورية
	Locals window	إظهار نافذة Locals أو تنشيطها إذا كانت ظاهرة
	Watch window	إظهار نافذة المراقبة Watch أو تنشيطها إذا كانت ظاهرة
	Call stack	إظهار المربع الحوارى Call stack الذى يحتوى على الإجراءات التى بدأت لكنها لم تنته بعد
	Project window	إظهار أو تنشيط نافذة مستكشف المشروع
	Properties window	إظهار نافذة خصائص العنصر الحالى أو تنشيطها
	Toolbox	التبديل بين إظهار وإخفاء مربع أدوات التحكم
	Tab Order	إظهار المربع الحوارى Tab order للنموذج الحالى
	Toolbars	التحكم فى عرض شرائط الأدوات
	Microsoft Access	إظهار نافذة Access
إدراج Insert	Procedure	إضافة إجراء جديد للوحدة النمطية الحالية
	UserForm	إنشاء نموذج جديد وإضافته إلى المشـــروع الحالى
	Module	إضافة وحدة نمطية للبرنامج للمشروع الحالى

القائمة	الأمر	وظيفته
	Class module	إضافة وحدة تصنيف نمطية للمشروع الحالى
	File	إضافة ملف خارجى للمشروع الحالى
ترجمة Debug	Compile	ترجمة المشروع الحالى
	Step into	تنفيذ الكود بواقع عبارة واحدة فى كل مرة
	Step over	تنفيذ الكود بواقع عبارة واحدة فى كل مرة وكذلك الإجراء يتم تنفيذه كوحدة واحدة.
	Step Out	تنفيذ الكود بواقع عبارة واحدة فى كل مرة وإذا وُجد إجراء يتم تنفيذ كـــوده بواقـــع عبارة واحدة فى كل مرة أيضا ثم الذهاب إلى العبارة التى تم من عندها استدعاء الإجراء
	Run to cursor	تنفيذ البرنامج إلى مكان تواجد المؤشر
	Add watch	إظهار المربع الحوارى Add watch والذى يمكنك من خلاله إدخال تعبير للمراقبة
	Edit watch	إظهار المربع الحوارى Edit watch الذى يمكنك من خلاله تعديل تعبير مراقبة موجود مسبقاً
	Quick watch	إظهار المربـــع الحـــوارى Quick watch محتوياً على قيمة التعبير الحالى
	Toggle breakpoints	التبديل بين إظهار وإخفاء نقاط التوقـــف فى السطر الحالى
	Clear all breakpoints	حذف جميع نقاط التوقف الموجودة بالمشروع الحالى

القائمة	الأمر	وظيفته
	Set next statement	تعيين بداية التنفيذ لسطر الكود الذى تقوم بتحديده
	Show next statement	وضع شريط ملون على العبارة التالية التى سيتم تنفيذها
تشغيل Run	Run Sub/UserForm	تشغيل الإجراء الحالى إذا كان المؤشر داخل إجراء أو تشغيل النموذج الحالى إذا كان نشطاً
	Break	إيقاف تشغيل البرنامج مؤقتاً
	Reset	إيقاف البرنامج نهائياً
	Design mode	التبديل بين وضع المشروع فى طور التصميم أو طور التشغيل
أدوات Tools	References	إظهار المربع الحوارى References الذى يحتوى على المراجع المستخدمة من قبل VBA
	Macros	إظهار المربع الحوارى Macros الذى يمكنك من خلاله إنشاء مختزل جديد أو حذفه أو التعديل فى مختزل موجود مسبقاً
	Options	إظهار المربع الحوارى Options الذى يمكنك من خلاله التحكم فى عناصر بيئة محرر VBA
	Properties	إظهار المربع الحوارى Project Properties الذى يمكنك من خلاله التحكم فى خصائص المشروع الحالى

- *نافذة الكود* **Code Window**

وتشتمل على محرر نصوص لكتابة الكود المطلوب تنفيذه من دوال وإجراءات ، لذا فهى النافذة الأساسية الموجودة ببيئة التطوير المتكاملة.

- *نافذة مستكشف المشروع* **Project Explorer Window**

وتحتوى على قائمة هرمية لجميع الوحدات النمطية الموجودة بقاعدة البيانات الحالية.

- *نافذة الخصائص* **Properties Window**

وتحتوى على خصائص الوحدة النمطية الحالية أو العنصر الحالى.

- *النافذة الفورية* **Immediate Window**

ويمكنك من خلالها تنفيذ أمر من الأوامر أو استدعاء دالة من الدوال لترى نتيجتها مباشرةً.

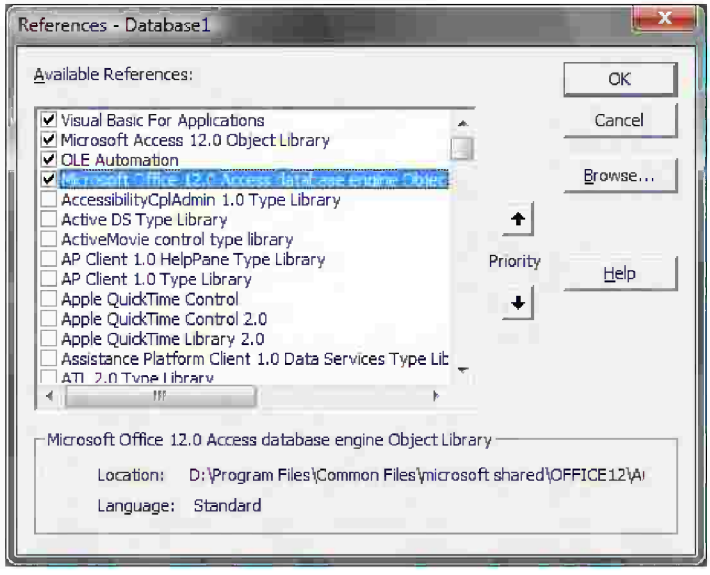
مراجع الكود والوحدات النمطية

تحتوى **Access 2007** على العديد من المراجع المستخدمة لتضمين الكائنات داخل الوحدات النمطية. للتعرف على المراجع الافتراضية داخل **Access 2007** ، تابع معنا الخطوات الآتية:

١. افتح أى وحدة نمطية داخل قاعدة البيانات.

٢. اختر **References** من قائمة **Tools** يظهر المربع الحوارى **References** كما فى شكل ٤-٢ .

٣. يمكنك من الشكل تنشيط أو تعطيل أيٍ من المراجع الموجودة لتضمينها داخل محرر **VBA** .



شكل ٤-٢ تضمين المراجع داخل محرر VBA .

البرامج والإجراءات والدوال

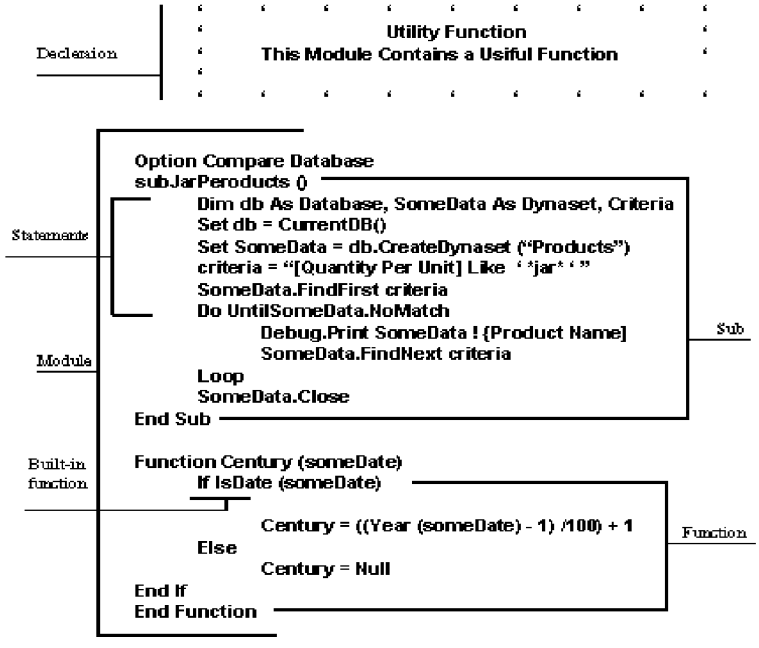
سنستخدم في هذا الكتاب مصطلحات هامة نرى من المفيد توضيحها قبل الخوض في الفصول التالية،مثل "البرنامج" **Module**، "الإجراء" **Procedure**، "الإجراء الفرعي" **Sub Procedure**، "الدالة" **Function**. ونوضح فيما يلي المقصود بالبرنامج والدوال والإجراءات الفرعية.

البرنامج Module

يتكون البرنامج كما يتضح من شكل ٥-٢ من جزئين . الجـزء الأول تعريـف البرنامج (**Declaration**)، ويظهر دائما في بداية الوحدة النمطية ، والجزء الثانى يمكن أن يتكون من واحد أو أكثر من الإجراءات الفرعيـة **Sub procedures** و/أو الـدوال **Functions** و/أو الدوال المبنية (**Built-in Function**) .

ويتم تقسيم البرنامج بهذه الصورة لتسهيل كتابته وسهولة صيانته واكتشاف الأخطاء به. يمكن استخدام أكثر من وحدة نمطية داخل قاعدة البيانات الواحدة لكتابة الإجراءات

أو الوظائف التى تحتاجها، كما يمكن كتابة كل الإجراءات والوظائف التى تحتاجها قاعدة البيانات داخل وحدة نمطية واحدة.



شكل ٥-٢ مكونات برنامج Access VBA

الإجراءات/الإجراءات الفرعية Procedures

عبارة عن برامج فرعية تتم الإشارة إليها من برنامج آخر. وعندما يصل البرنامج إلى اسم الإجراء (البرنامج الفرعي) ، يتم استدعائه وتنفيذه. وبعد الانتهاء من تنفيذ الإجراء يرجع التنفيذ مرة أخرى إلى البرنامج الذى استدعاه . ويمكن أن يستدعى الإجراء (البرنامج الفرعي) إجراءً (برنامجا فرعياً) آخر... وهكذا. يبدأ الإجراء بعبارة **Sub** وينتهي بعبارة **End Sub** والشكل العام التالي يوضح التركيب العام للإجراء:

```
Sub  procName                                    اسم الإجراء
        [Start of procedure code]          أول تعليمة في الإجراء
        ..........
        Statements                                 تعليمات الإجراء
        ............
        [End of procedure code]            آخر تعليمة في الإجراء
End Sub                                          نهاية الإجراء
```

الدوال Functions

هي نوع من الإجراءات ، ولكنها تعيد دائما قيمة وهي إما دوال موجودة داخل لغة **Access VBA** وتسمى دوال مبنية ، أو دوال يقوم المستخدم بإنشائها وتسمى دوال المستخدم أو **UDF** (**User Defined Function**) . تنشأ الدوال بتركيب عام مشابه لتركيب الإجراءات كما يلي:

```
Function  FuncName([Arguments])          اسم الدالة(المعاملات)

        [Start of function code]           أول تعليمة في الدالة
        .......
        Statements                                 تعليمات الدالة
        .......
        [End of function code]             آخر تعليمة في الدالة
End Function                                     نهاية الدالة
```

ولاستدعائها يجب استخدام الوظيفة في تعبير كالآتي:

```
inReturnValue=nilFunctionName([Arguments])
```

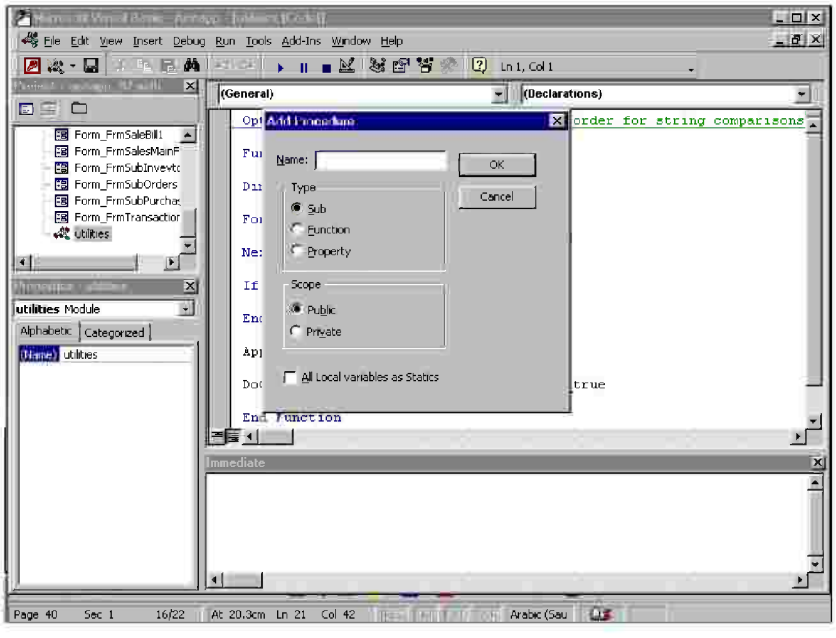
حتى لو كانت الدالة لا تعيد أي قيمة.

إضافة إجراءات أو وظائف جديدة للوحدة النمطية

لإضافة إجراء جديد لوحدة نمطية موجودة من قبل أو مازالت مفتوحة اتبع الآتي:

١. افتح نافذة الوحدة النمطية أو تأكد أنها مفتوحة.

٢. اختر **Procedure** من قائمة **Insert** ، سيظهر أمامك المربع الحوارى **Add Procedure** (كما فى شكل ٦-٢). سيسألك عن نوع الإجراء الجديد الذي تريد إضافته ، وهو إما أن يكون إجراء فرعي **Sub** أو دالة **Function** أو خاصية **Property** .



شكل ٦-٢ إضافة دالة/إجراء جديد للوحدة النمطية.

٣. اكتب اسم الإجراء أو الدالة في خانة **Name**، ثم انقر زر **Ok**.

تظهر أمامك نافذة تحرير جديدة وستكتب **Access** تلقائياً السطر الأول والأخير الذي تحتاج إليه دائما في كل دالة (في حالة اختيار دالة **Function**) أو أول وآخر سطر تحتاج إليه في الإجراءات الفرعية (في حالة اختيار **Sub**).

٤. اكتب الدالة/الإجراء الفرعي في الجزء المخصص، وفيما يلى بعض النصائح التى تساعدك في تسهيل كتابة الإجراءات أو الدوال أو تعديلها باستخدام نافذة المحرر:

- استخدم مفاتيح **Del, Backspace ,Ins** ومفاتيح الأسهم للحذف والإدراج والانتقال بين السطور والكلمات بنفس الطريقة التى تستخدمها لتحرير النصوص مع برامج معالجة النصوص.
- استخدم مفاهيم القص والنسخ واللصق بنفس الطريقة التى تستخدمها مع بقية برامج **Windows** سواء من شريط الأدوات أو بالأوامر
- استخدم بصفة خاصة المفاتيح التالية لتسهيل التعامل مع نافذة المحرر: **Ctrl+N** لفتح سطر جديد، **Ctrl+Y** لحذف السطر الحالي ، **Ctrl+z** للتراجع عن آخر عملية تحرير
- استخدم أوامر البحث والاستبدال الموجودة في قائمة تحرير (تجدها في جدول القوائم والأوامر) كما تستخدمها مع برنامج **Word**.

٥. عندما تنتهي من كتابة الإجراء/الدالة الجديدة، اضغط مفتاح **Ctrl+F4**. إذا سألك **Access** هل تريد حفظ التغييرات التى قمت بها، اختر **No** لتعود إلى نافذة قاعدة البيانات.

طلب المساعدة أثناء البرمجة

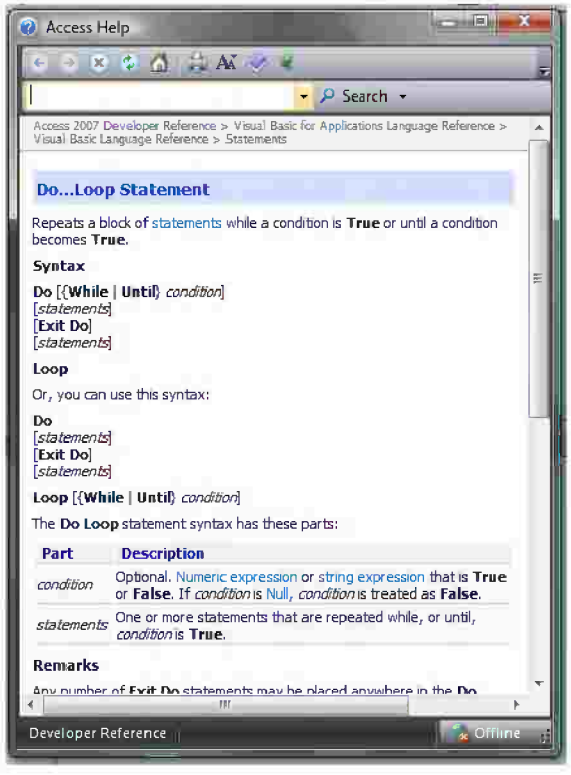
لاشك أن تعليمات المساعدة المتوفرة في **Access** توفر وقتك وجهدك في الرجوع إلى الكتب والمراجع. بمجرد أن تعرف كيف تستخدم تعليمات **Access** ستستخدمها بديلا للكتب والمراجع، لن نتعرض لتفاصيل الحصول على تعليمات مساعدة عن كائنات **Access** فقد تناولناها في كتابنا *"المرجع الأساسي لقاعدة البيانات* **Access 2007***"* . سنركز هنا على نظام تعليمات المساعدة الذى تحتاج إليه أثناء استخدام لغة **Access VBA** .

تشتمل تعليمات المساعدة على العديد من الشاشات والشاشات المتفرعة عنها

التى تزودك بالتعليمات التى تحتاجها لتعلم واستخدام لغة Access VBA . تابع الخطوات التالية:

١. عندما تريد تعليمات مساعدة عن أمر معين أو كلمة أساسية (Keyword)، انقل المؤشر فوق الأمر أو الكلمة الأساسية ثم اضغط مفتاح F1.

٢. تظهر نافذة تعليمات المساعدة للكلمة الأساسية (Keyword) . عن كيفية استخدام هذه الكلمة أو الأمر داخل Access VBA (انظر شكل ٧-٢).



شكل ٧-٢ نافذة Access Help للحصول علي مساعدة عن كلمة أساسية