تعلم برمجة
فيجوال بيسك ٦

خطوة بخطوة

الجزء الأول : بناء قواعد بيانات أكسيس ولغة الإستعلامات الهيكلية

الجزء الثاني : بناء برنامج قواعد بيانات باستخدام فيجوال بيسك

الجزء الثالث : بناء برنامج سوبر ماركت خطوة بخطوة

الجزء الرابع : بناء موقع باستخدام Asp للسبر ماركت

إعداد وتقديم

أحمد جمال خليفة

مقدمة .

بسم الله الرحمن الرحيم ، والـصلاة والـسلام على أشرف الأنبياء والمرسلين ، نبينا محمد وعلى آله وصحبه أجمعين .

اللهم لا سهل إلا ما جعلته سهلاً وأنت تجعل الحزن سهلاً إذا شئت

أما بعد ... فهذا الكتاب يحتوي على مجموعة من الدروس التي كتبت أجزاء منها في منتديات vb4arab ... وتختص بالتعامل مع الفيجوال بيسك وقواعد البيانات إضافة إلى التعامل مع برمجة المواقع من خلال Asp وهو موجه أساساً إلى المبتدئين في الفيجوال بيسك ٦ وبرمجة المواقع .

وهذا هو الجزء الأول والخاص بالتعامل مع قواعد البيانات

مايكروسوفت أكسيس Microsft Access

أسأل الله أن يكون هذا الكتاب مفيداً ، فما كان فيه من صواب فمن الله ، وما كان من خطأ فمن نفسي والشيطان ، والله الهادي إلى سواء السبيل .

أخوكم : أحمد جمال خليفة عبد العال .

Hammada2091@Hotmail.com

القسم الأول

بناء قواعد البيانات باستخدام Microsoft Access ولغة الاستعلامات الهيكلية

في هذا الفصل :

- مقدمة عامة : ما هي قواعد البيانات ومصطلحات مهمة .
- أنوع البيانات .
- إنشاء قاعدة بيانات بسيطة باستخدام MicrsoftAccess
- عمليات على قواعد البيانات
- النماذج .
- العلاقات .
- عمليات خاصة على قواعد البيانات .
- لغة الإستعلامات الهيكلية TSQL

مقدمة عامة

ما هي قاعدة البيانات ؟

قاعدة البيانات هو أحد وسائل حفظ وترتيب البيانات بصورة معينة (جداول – حقول) مع إيجاد علاقات ، وتحتوي على بعض التقنيات والأدوات التي تسهل الوصول إلى المعلومات . وتقلل من حجم التخزين.

ومن أمثلة قواعد البيانات Access ، FoxPro , Sql Server من انتاج شركة مايكروسوفت إضافة إلى منتجات من شركات أخرى مثل أوراكل .

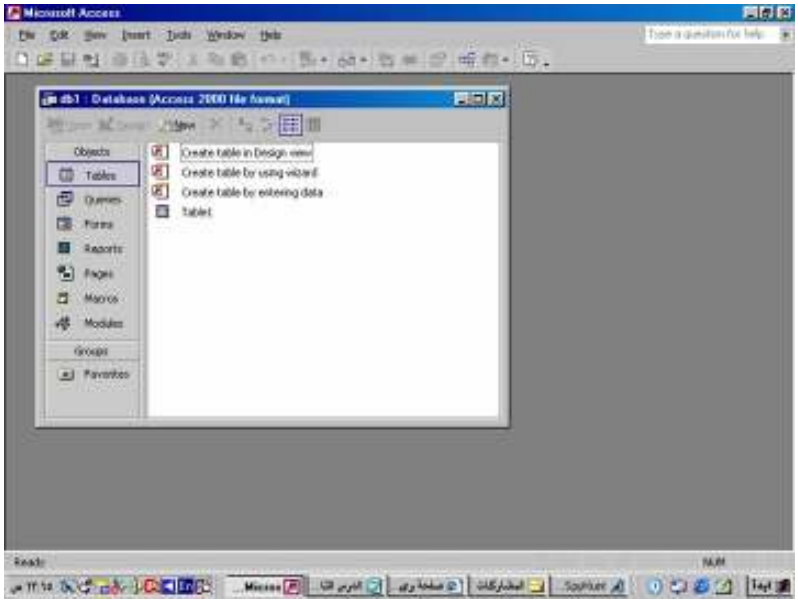
وتسمى هذه البرامج نظم ادارة قواعد البيانات DBMS وتشترك جميعها في التركيب الاساسي لقواعد البيانات ، كما انهم جميعاً يستخدمون لغة الاستعلامات المركبة SQL القياسية.

والآن سنتعرف على معنى بعض المصطلحات الأساسية في قواعد البيانات والتي سوف نعتمد عليها في شرحنا مستقبلاً إن شاء الله.

قاعدة البيانات – DataBase:

يطلق هذا الإسم على الملف – إن صح التعبير – الذي يحتوي على الجداول اضافة للاستعلامات والتقارير وغيرها من مكونات قاعدة البيانات.

وهذه صورة لما يمكن أن نطلق عليه قاعدة بيانات.



الجدول – Table :

يحتوي الجدول على المعلومات التي نود حفظها في قاعدة البيانات في ترتيب معين على شكل صفوف وأعمدة.

وهذه الصورة تمثل جدولاً ما.

	Number	Name	Age
	0	Ahmed	18
	0	Hosame	17
	0	Khaled	19
	0	Mohammed	21
✎	0	Ali	30
	0	Sami	80
*	0		0

الإستعلام – Quires :

يقوم الإستعلام بمهمة تعيين سجلات معينة أو حقول معينة والتي يمكن أن تحقق شرطاً معينا.

وهذه صورة لاستعلام يتضمن اسماء الأشخاص الذين تزيد أعمارهم عن ١٨ سنة.

	Name
▶	Khaled
	Mohammed
	Ali
	Sami
*	

في حين أن الجدول الرئيسي هو:

	Number	Name	Age
	0	Ahmed	18
	0	Hosame	17
	0	Khaled	19
	0	Mohammed	21
✎	0	Ali	30
	0	Sami	80
*	0		0

السجل – Record :

يشكل كل سجل شخصاً أو سلعة و مجموعة بيانات لوصف عنصر مستقل ، ما وتوجد لها قيم في كل حقل.

الحقل – Field :

يشكل الحقل نوعاً معيناً من البيانات سيتم إضافته لكل سجل.

الشكل التالي يمثل حقل البريد الإلكتروني.

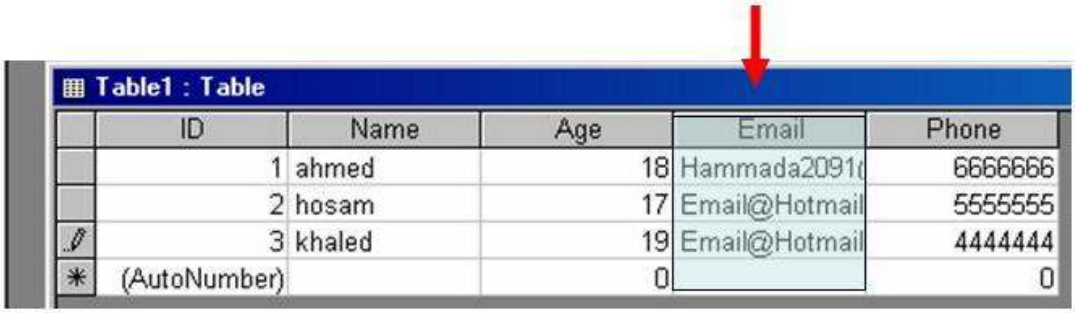


Table1 : Table

	ID	Name	Age	Email	Phone
	1	ahmed	18	Hammada2091(	6666666
	2	hosam	17	Email@Hotmail	5555555
	3	khaled	19	Email@Hotmail	4444444
*	(AutoNumber)		0		0

نوع البيانات – Data Type :

تعتبر خاصية للحقل وتحدد نوع البيانات التي ستظهر في الحقل.

والشكل التالي يوضح أنواع البيانات في جدول الأصدقاء مثلاً:

Field Name	Data Type	Description
Number	Number	
Name	Text	
Age	Number	
Email	Memo	
Phone	Number	

وسوف نتعرف بصورة سريعة على اشهر انواع البيانات لاحقاً.

المفتاح الرئيسي – Primary Key :

نوع معين من الحقول لا يُسمح بتكرار قيمته لأي سبب ... مثال ذلك الحقل الخاص بالرقم القومي للمواطن.

العلاقة – Relationship :

تشكل علاقة معينة بين حقلين في جدولين مختلفين ، وسوف نتعرض لأنواع العلاقات لاحقاً.

التقرير – Report :

هو تجميع لبعض البيانات كي تشكل معلومات مفيدة تمهيداً لطباعتها – غالباً-
مثال ذلك تقرير يقوم بحساب الصادر والوارد ليقوم بعرض الأرباح.

أنوع البيانات

البيانات النصية – Text :

حتوي هذا النوع من البيانات على جميع النصوص والأرقام بشرط أن لا يكون الرقم هو الخانة الأولى.

بعض الخصائص المهمة للحقول النصية:

Field Size : تحتوي على الحد الأقصى من الخانات التي يمكن كتابتها في الحقل.

Default Value : تحتوي على القيمة الإفتراضية للحقل.

Required : إذا احتوت على True فهذا يعني أنه لا بد من ادخال قيمة وإلا لن تنجح عملية الأدخال.

البيانات الرقمية – Number :

تحتوي على البيانات الرقمية بمختلف أنواعها والتي يمكن أن نحددها من خاصية. Field Size

الخصائص المهمة للحقول الرقمية:

Field Size : تحتوي على أحد الخصائص التالية:

Byte : تحتوي على أرقام بين ١ و ٢٥٥ بلا كسور.

Decimal : تحتوي على أرقام بين بين ١٠^٢٨ إلى –١٠^٢٨ في ملفات mdb. العادية – يمكن أن يحتوي على كسور. -

Integer : تحتوي على أرقام بين –٣٢٧٦٨ إلى ٣٢٧٦٧.

Long Integer : تحتوي على أرقام بين –٢١٤٧٤٨٣٦٤٨ إلى ٢١٤٧٤٨٣٦٤٧.

Single : يمكن تقسيم الفترة الخاصة بها إلى قسمين ، وهو يسمح بالكسور (المنازل العشرية)

في الجزء السالب : من –٣,٤٠٢٨٢٣ إلى –١,٤٠١٢٩٨ * ١٠^–٤٥.

في الجزء الموجب : ١,٤٠١٢٩٨ * ١٠^–٤٥ إلى ٣,٤٠٢٨٢٣.

Double : أيضاً يمكن تقسيم الفترة إلى قسمين وهو يسمح بالكسور.

في الجزء السالب : -١,٧٩٧٦٩٣١٣٤٨٦٢٣١ * ١٠^٣٠٨ إلى -٤,٩٤٠٦٥٦٤٥٨٤١٢٤٧ * ١٠^-٣٢٤

في الجزء الموجب : ٤,٩٤٠٦٥٦٤٥٨٤١٢٤٧ * ١٠^-٣٢٤ إلى ١,٧٩٧٦٩٣١٣٤٨٦٢٣١ * ١٠^٣٠٨.

وهذه صورة من مساعدة مايكروسوفت أكسيس XP المتعلق بمجالات الأرقام ، وفيه أيضاً في الصف الثالث عدد الخانات العشرية المسموحة ، وفي الصف الأخير كم يستهلك من الذاكرة.

Setting	Description	Decimal precision	Storage size
Byte	Stores numbers from 0 to 255 (no fractions).	None	1 byte
Decimal	Stores numbers from -10^38-1 through 10^38-1 (.adp) Stores numbers from -10^28-1 through 10^28-1 (.mdb)	28	12bytes
Integer	Stores numbers from -32,768 to 32,767 (no fractions).	None	2 bytes
Long Integer	(Default) Stores numbers from -2,147,483,648 to 2,147,483,647 (no fractions).	None	4 bytes
Single	Stores numbers from -3.402823E38 to -1.401298E-45 for negative values and from 1.401298E-45 to 3.402823E38 for positive values.	7	4 bytes
Double	Stores numbers from -1.79769313486231E308 to -4.94065645841247E-324 for negative values and from 4.94065645841247E-324 to 1.79769313486231E308 for positive values.	15	8 bytes
Replication ID	Globally unique identifier (GUID)	N/A	16 bytes

أما بقية الخصائص فهي مشابهه لخصائص النصوص.

بيانات الوقت والتاريخ Date/Time .

تحتوي على بيانات تعرض الوقت والتاريخ ، وأهم خصائصها هي خاصية Format ، والتي تحتوي على أشكال مختلفة من تنسيقات الوقت والتاريخ.

بيانات العملة Currency .

لعرض العملة وأهم خصائصها هي Format والتي تعرض أنواع مختلفة من تنسيقات العملة.

الترقيم التلقائي AutoNumber .

تحتوي على تسلسل للأرقام يساوي عدد السجلات ، والواقع أن هذا النوع يسبب مشكلة بعد الحذف حيث أنه لا يعيد الترقيم إلا بعد عمل اصلاح وضغط لقواعد البيانات ، كما أنه يجبرك على ان يكون

نوع البيانات هو من عدد صحيح طويل Long Integer ، وهذا قد لا يكفي كما أنه على الجهة الأخرى قد يسبب هدراً كبيراً في البيانات الصغيرة.

المذكرة Memo .

يتميز هذا النوع بأنه يمكن أن يحتوي على بيانات من أي نوع بكمية ضخمة جداً . أو حتى ٢ GB كما تذكر Microsoft .

نعم/لا Yes/No .

يحتوي هذا الحقل على قيم تأخذ حالة من اثنتان ، إما صحيحة أو خاطئة.

أهم خاصية هي خاصية Defult Value والتي تحدد هل يكون في حالته الافتراضية صحيحاً أم خاطئاً.

كائن OLE Object .

يمكن أن نضع فيها كائناً مثل مستند وورد على سبيل المثال.

ارتباط تشعبي HyperLink .

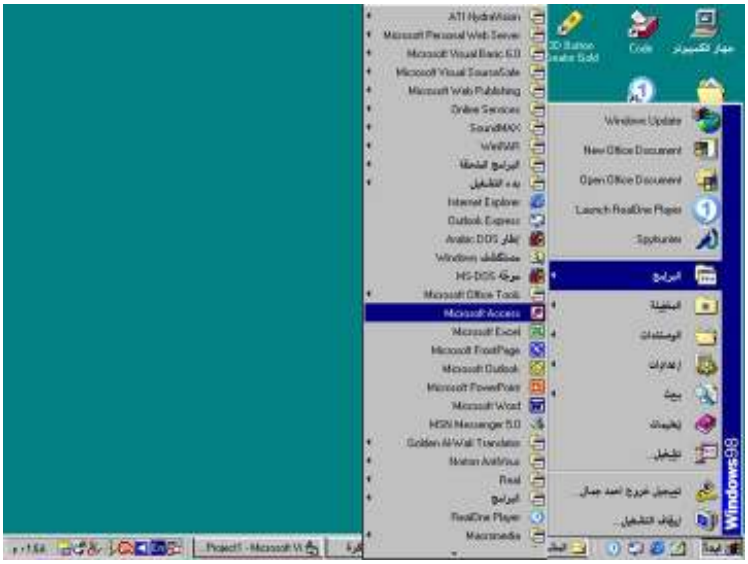
يمكن وضع رابط لصفحة أو مسار لملف على جهاز الكومبيوتر.

إنشاء قاعدة بيانات

سوف نقوم الآن بانشاء قاعدة بيانات بسيطة نعمل معها خلال المرحلة الأولى من هذه الدورة.

سنبدأ أولاً **بتشغيل برنامج مايكروسوفت أكسيس** ... Microsoft Access من قائمة البرامج

اختر البرنامج بالشكل التالي:



ونفس النظام بالنسبة لباقي اصدارات الويندوز.

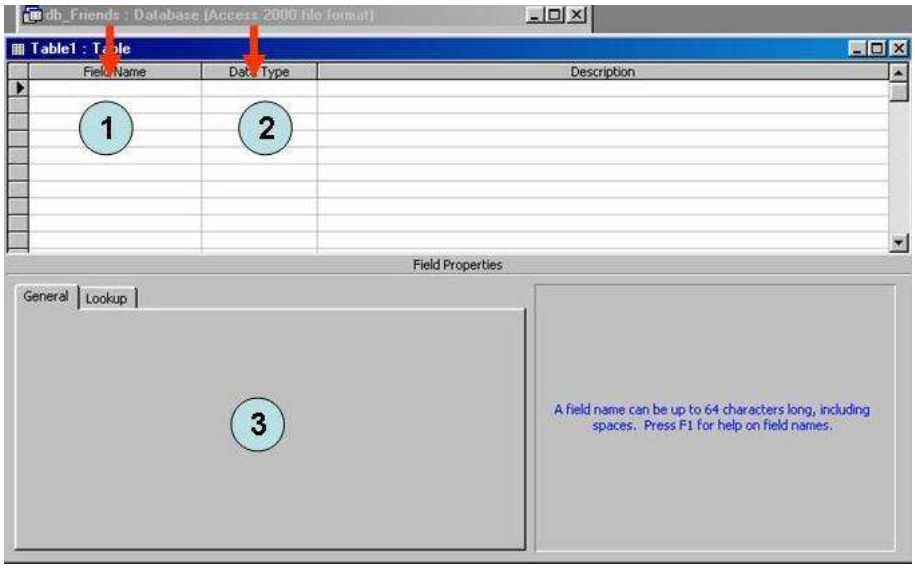
سنختار قاعدة بيانات فارغة ... Blank Database وسيطلب منا البرنامج تخزين قاعدة البيانات وسنقوم بحفظها بالاسم Db_Friends .

*** ملاحظة مهمة : هذه الصور من Microsoft Access XP والفروق طفيفة عند العمل على اصدرات أخرى.

بعد أن نضغط على عملية انشاء سوف تظهر لنا صورة قاعدة البيانات ، سنختار في البداية Tabels -جداول ، لأننا لا بد أن نصمم جدولاً قبل كل شيء.

وسنختار - Create Table in Design View انشاء جدول في طريقة عرض التصميم.

ستظهر لنا شاشة فارغة بالشكل التالي:



– هذه هي المنطقة التي سنكتب فيها اسماء الحقول Fields Name .

– هذه هي المنطقة التي سنكتب فيها نوع بيانات الحقل DataType .

– في هذه المنطقة سوف تظهر خصائص الحقول بمجرد اضافة أي حقل.

سنبدأ الآن في **تصميم نوتة الهواتف الخاصة بنا** والتي ستحتوي على المعلومات التالية:

الرقم – الإسم الأول – الإسم الأخير – رقم الهاتف – العنوان – البريد الإلكتروني – الموقع الشخصي – متزوج – ملاحظات ، وذلك بالشكل التالي:

*	خاصية ٢	خاصية ١	نوع البيانات Data Type	اسم الحقل Field Name	الحقل Field
١	مطلوب Required نعم – Yes	طول الحقل = رقم صحيح Field Size=integer	رقم Number	Number	الرقم
		مطلوب – Required نعم – Yes	نص Text	F_Name	الإسم الأول
٢			نص Text	L_Name	الإسم الأخير
٣			رقم Number	Phone	رقم الهاتف
			مذكرة Memo	Address	العنوان
			نص Text	EMail	الايميل
			ارتباط تشعبي Hyper Link	Site	الموقع
٤		الخاصية الإفتراضية = لا Default Value=No	نعم/لا Yes-No	Marry	متزوج

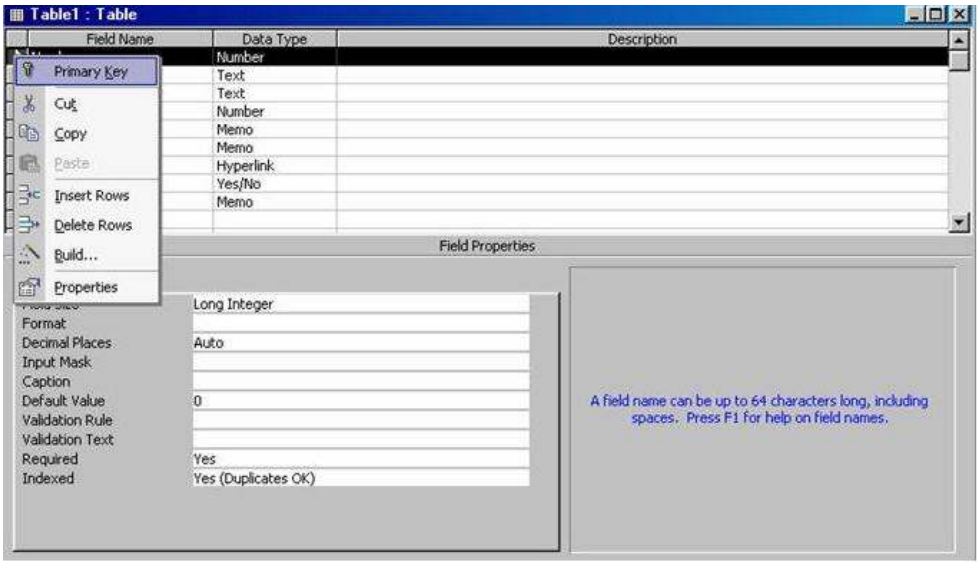
ملاحظات	Memo	مذكرة		
		Memo		
العمر	Age	رقم	نوع البيانات = بايت	٥
		Number	Field Size=Byte	

ملاحظات الجدول :

* ١ – قمنا بذلك لأنه الرقم الصحيح يوفر لنا عدداً لا بأس به – لن تضيف ضمن أصدقائك أكثر من ٣٢٧٦٧ صديق – وفي نفس الوقت تستهلك ٢ بايت من الذاكرة.
* ٢ – لن يكون بالضرورة الإسم الأخير موجوداً ... لذا فسنترك خاصية. Required
* ٣ — من المفضل ابقاء حقول الهاتف وما شابه من النوع النصي بسبب أنه اقل في اهدار الحجم اضافة إلى امكانية اضافة ٠ قبل رقم الهاتف أو المحمول .
* ٤ – حيث سنرمز للمتزوج بنعم Yes ، ولغيره بلا. No .
* ٥ – حيث ان عمر الشخص لن يتعدى – بإذن الله – ٢٥٥.

ولا تنس ان تضبط بعض الخصائص كما ترغب لاحقاً.

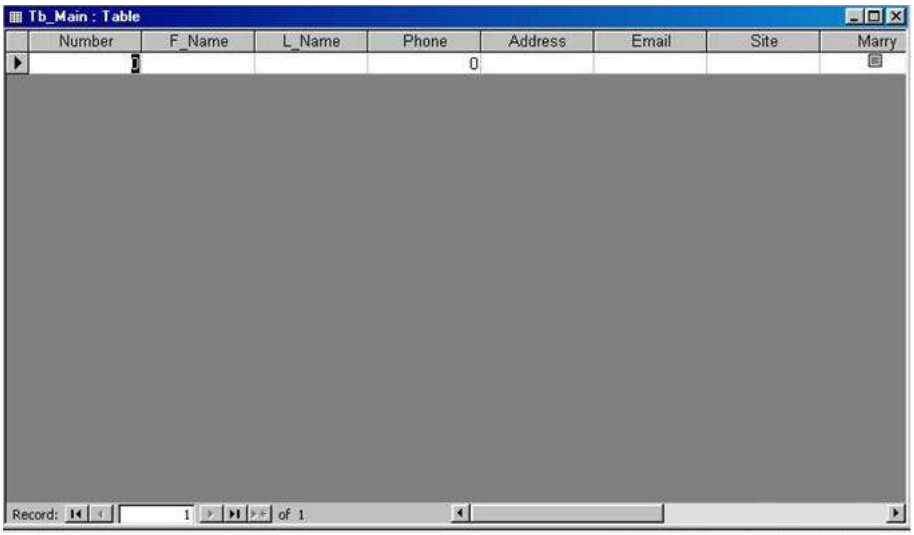
أيضاً سنعين حقل الرقم كمفتاح أساسي لأنه لا بد أن لا يتكرر في سجلين مختلفين ، وذلك بالشكل التالي:



ملف مرفق :
قم بمطالعة الملف Db1.zip **في المرفقات لمزيد من التفاصيل .**

بعد ذلك سوف نقوم بالحفظ بالضغط على رمز القرص المرن في شريط الأدوات في الأعلى ... وسنقوم بالحفظ تحت اسم Tb_Main ، ومن ثم نقوم بالخروج.

نلاحظ ظهور الجدول في قائمة ... Tables جرب الضغط عليه وسيظهر لك الشكل التالي:



قم الآن بادخال عدد من البيانات تمهيداً لاجراء بعض العمليات عليها.

عمليات على قواعد البيانات

أولاً : عملية الإضافة.

أول عملية ينبغي معرفتها هي عملية الإضافة Add وفي الأكسيس كل ما نقوم به هو الكتابة في أحد السجلات RecordSet وسيبثق لك تلقائياً سجلاً آخر بعده يمكنك اضافته إن أردت.

لاحظ أنه لا بد من كتابة بيانات جميع الحقول التي ضبطت خاصية - Required) مطلوب) لها بحيث تصبح - Yes) نعم. (

لاحظ أيضاً أنه لن يسمح بتكرار قيمة في حقل Number لأنك عينته كمفتاح رئيسي Primary Key - جرب اضافة بيانات متشابهة وشاهد النتيجة. -

ثانياً : عملية التعديل.

ضع المؤشر في أي مكان وقم بتعديل البيانات كما يحلو لك.

لاحظ أنه عندما تتنقل بين السجلات والحقول سوف يحدد لك الحقل كاملاً لتعدل فيه – جرب ذلك – ، فإذا أردت أن تعدل فيه فلابد أن تضغط بالماوس في الحقل قبل الكتابة كي لا يزيل النص الموجود أساساً.

أيضاً يمكنك استخدام الكيبورد لهذا الغرض بالضغط على مفتاح. F2

ثالثاً : عملية الحذف.

حدد أي سجل بالضغط على المسافة الفارغة قبل أول حقل ، سيظلل لك السجل ، ومن هناك اضغط على زر الماوس الأيسر واختر Delete بالشكل التالي:

أيضاً يمكنك الضغط على زر Delete من الكيبورد.

ستظهر لك رسالة تحذيرية – لاحظ أيضاً بإنه لن يمكنك التراجع – في حال كنت متأكداً قم بالموافقة وسوف يتم حذف السجل.

فعلياً لا يتم حذف السجل مباشرة ، بل إن كل ما في الأمر هو وضع علامة صح أمام السجل بحيث يصبح مخفياً فقط دون الحذف.

إن الحذف يقتضي ان يتم نسخ جميع البيانات قبل السجل وتخزينها في مكان مؤقت ، ومن ثم تجاوز السجل المحذوف ، ونسخ ما بعده واضافته في المكان المؤقت ، ومن ثم افراغ الجدول الاساسي ونقل البيانات المؤقتة إليه من جديد ، وهذا ما لا يمكن عمله مع كل عملية حذف ، لذلك لا تتم هذه العملية إلا بعد الضغط على (اصلاح قواعد البيانات واصلاحها Compact and Repair - Database) حيث تستطيع الملاحظة بأن حجم قاعدة البيانات يصبح أقل.

رابعاً : عمليات النسخ والقص واللصق.

يمكنك نسخ السجل بعد تحديده وكذا قصه – لاحظ أنه ستظهر لك رسالة تحذيرية شبيه بتلك التي تظهر عند الحذف – ، لاحظ أيضاً أنك لو قمت بلصق السجل والمؤشر على سجل آخر فسوف يستبدل بياناتهما.

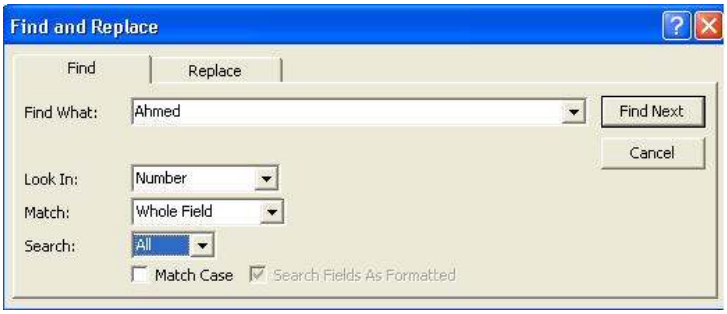
عملية البحث والإستبدال:

قبل أن تبدأ أي عملية بحث فهناك خياران.

* الخيار الأول أن تبحث في حقل معين.
* والثاني أن تبحث في كافة الحقول.

إذا كنت تريد الخيار الأول فلا بد أولاً أن تضع المؤشر في ذلك الحقل في أي سجل.
أما الثاني فيمكن اختياره تلقائياً.

سنبدء بتشغيل جزء البحث وذلك بالضغط على رمز المنظار في شريط الأدوات.
أو من قائمة (تحرير Edit) - اختر (بحث Find) -
ستظهر لنا شاشة بالشكل التالي وسوف نتعرف سوية على مكوناتها.



الجزء الأول - Find What) البحث عن .

في هذا الجزء سوف نحدد الكلمة المراد البحث عنها.

الجزء الثاني - Look In) البحث في .

في هذا الجزء نستطيع تحديد الحقل الذي نريد البحث فيه وهو الذي كان المؤشر عليه قبل عملية البحث ، أو اختيار الجدول كاملاً.

الجزء الثالث(Match)

لدينا ثلاثة خيارات في هذ الجزء وهي:

- Any part Of Field أي جزء من الحقل : يستخدم للبحث عن الكلمة في أي جزء من بيانات الحقل ، بمعنى أنك لو كتبت hm فسيظهر لك Ahmed و Mohamed

- Whole Field الحقل بالكامل : بهذه الطريقة سوف يبحث عن الحقول المطابقة لكلمة البحث ولذا لن يظهر شيئاً لو كتبنا. ah

- Start Of Field بداية الحقل : سيبحث عن الحقل الذي تطابق بدايته كلمة البحث ، فمثلاً لو كتبنا ah سيعطينا Ahmed ولكنه لن يعطينا Mohammed لأنها لم تبدأ ب. Ah

الجزء الرابع - Search) البحث .

لدينا ثلاثة خيارات هي - All) الكل) و - Up) للأعلى) و - Down) للأسفل) وتحدد شكل عملية البحث ، هل تتجه إلى الأسفل ابتداء من موضع المؤشر الحالية أم للأعلى أم تأخذ دورة.

الجزء الخامس - Match Case) مراعاة حالة الأحرف .

إذا حددنا هذا الخيار فسوف يتأكد من أن الحرف مطابق في حالته ، فمثلاً لو بحثنا عن h لن تعطينا Hosameلأنها بدأت بحرف كبير. Capital -

بعد أن نحدد جميع خياراتنا نستطيع بدء عملية البحث بالضغط على زر - Find) بحث .

الجزء الثاني هو الذي يختص بعملية الإستبدال ، وهو نفس عملية البحث ، وهناك أربعة ازرار :

*زر للبحث بدون استبدال. *زر الإغلاق.

*زر استبدال الحالي. *زر استبدال الجميع.

آخر جزء سوف نتتطرق إليه في عملية البحث هو **عملية البحث بالتشابه باستخدام علامة الإستفهام** (?) .

نستطيع كتابة علامة الإستفهام ضمن كلمة البحث للدلالة على حرف أو رقم ما ... جرب البحث عن الكلمة التالية.

?Ho

لتشكل (حرفاً ما ، H، o)

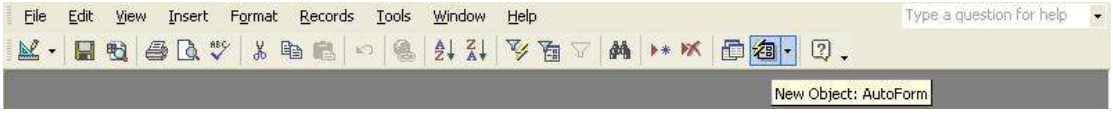
ستجد أنها لن تعطيك Hosame لأنه لا يوجد فيها ما يقوم مقام علامة الإستفهام ، في المقابل ستعطيتك كلمة Yahoo لوجود حرف a قبل المقطع. ho

تجربة أخرى ، جرب البحث عن الكلمة التالية:

No ????

ستجد أنها ستعطينا No Memo وأيضاً ... No Site ما عدا تلك الموجودة في السجل رقم ٨ لأننا كتبناها خطأ. No Mem

نستطيع عرض السجلات بصورة أخرى بالشكل التالي:



والبديل لهذه العملية هي عملية انشاء النماذج ... وسوف نتعرف عليها لاحقاً إن شاء الله.

عملية الترتيب. Sort

يمكننا الترتيب تصاعديا Ascendig وكذلك تنازليا ... Descinding حدد الحقل الذي تريد الترتيب بواسطته (وليكن الإسم الأول مثلاً) ومن ثم اضغط على رمزيهما في شريط الأدوات ، أو من قائمة - Records) سجلات) اختر - Sort) ترتيب) ثم اختر تصاعدياً أو تنازلياً.

لاحظ أنك إذا قمت بذلك فسوف يتغير ترتيب الأرقام .

النماذج – Forms

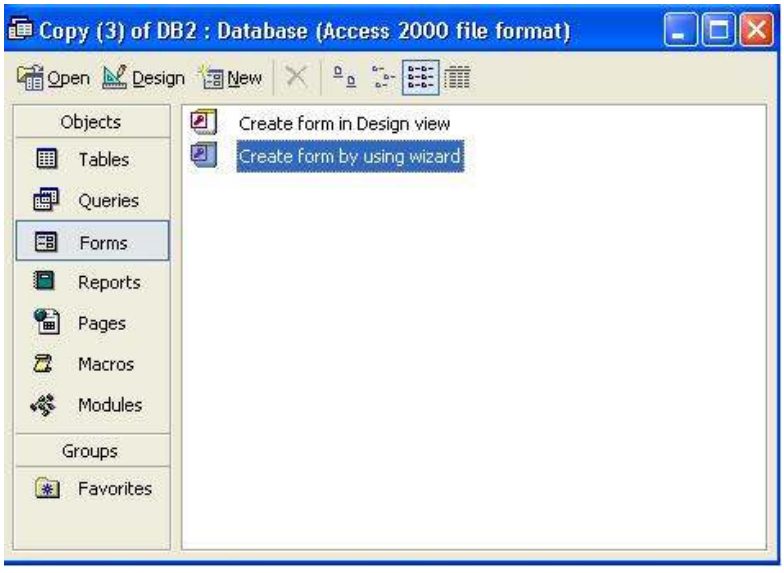
تعتبر النماذج أقرب أجزاء الأكسيس إلى الفيجوال بيسك – رغم أنها لا تستخدم في الفيجوال. - وسنقوم خلال الدروس بإنشاء نموذج قاعدة بيانات الأصدقاء التي قمنا بإنشاءها.

بعد أن تقوم بفتح قاعدة البيانات اختر نماذج Forms - وسيظهر لك أحد الإختيارين التاليين

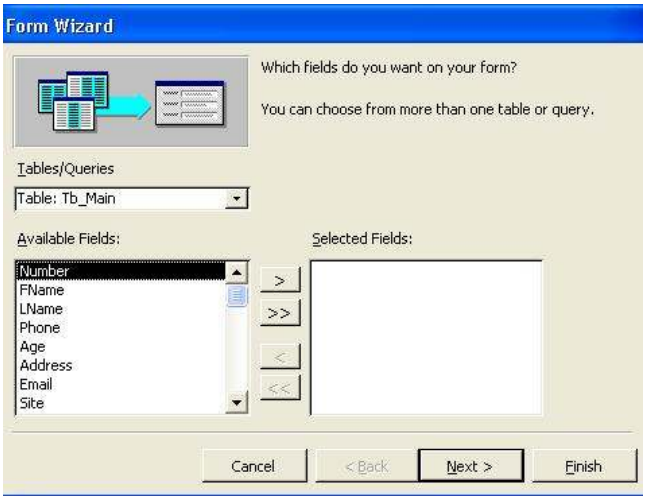
- Create form in Design view انشاء نموذج في طريقة عرض التصميم

- Create form by using wizerd انشاء نموذج باستخدام المعالج

سنقوم مبدئياً بانشاء نموذج باستخدام المعالج ... ولن يختلف الأمر في حالة القيام بذلك يدوياً ، اختر هذه الطريقة بالشكل التالي.



سيظهر لنا مربع حواري بالشكل التالي:



من الجزء الأعلى سوف نختار اسم الجدول ... سنختار جدولنا.

وفي الجزء الثاني سوف نختار الحقول التي نريد لها أن تظهر في النموذج ... لذا قم باختيارها كلها ، واضغط التالي.

بعد ذلك سوف يعطينا أشكال مختلفة من عرض النموذج ، سنختار الشكل الأول ثم نضغط التالي.

في الشاشة التالية سوف يعطينا تنسيقات مختلفة ... اختر التنسيق الذي يروق لك.

آخر الشاشات سوف يطلب منك اسم النموذج . سنسميه ... Form1 الإختيارات الموجودة بالأسفل واضحة وسنتركها على ما هي عليه.

ثم نضغط انهاء.

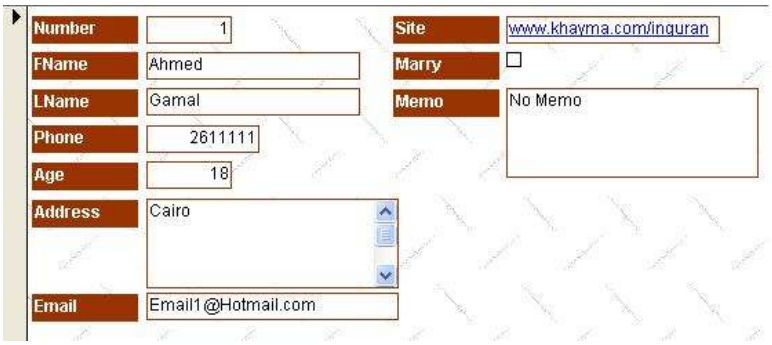
سوف يتكون لنا نموذج بالشكل التالي:

يمكنك التنقل بين السجلات واضافة وحذف سجلات.

إذا كنت تريد التعديل في شكل النموذج اضغط على طرف شريط الأدوات ، غير أماكن الأدوات كما يحلو لك ... وللعودة لوضع التنفيذ قم بالضغط على نفس المكان ثانية بالشكل التالي:

قم الآن بعرض تصميم النموذج ... وسنقوم بأول عملياتنا.

سنقوم بتحريك بعض الأدوات – مثل الفيجوال بيسك تماماً – ، وسنغير مثلاً حجم حقل البريد الإلكتروني لتعطينا الشكل التالي مثلاً.



لاحظ أنه لكي تحرك مربع نص TextBox دون تحريك مربع العنوان لا بد من أن تسحب المربع من أعلى اليسار – سيصبح المؤشر على هيئة يد تؤشر بالسبابة .

لاحظ أيضاً أن الأدوات التي على اليسار شبيهه جداً بأدوات الفيجوال بيسك في التعامل معها.

ملف مرفق :

جميع التعديلات حتى هذه اللحظة موجودة في Db2.zip

سنبدأ في اضافة أزرار ... لسهولة التعامل مع هذه الأزرار يفضل أن يكون زري السهم والمعالج مضغوطين وذلك بالشكل التالي:

سنضغط عى صورة زر الأمر . CommandButton ثم نقوم برسمها على الفورم.

ستظهر لنا شاشة تطلب منا تحديد نوع العملية Categories والحدث ... Action في البداية

سوف ننشأ زر خروج وستكون معلوماته بالشكل التالي:

نوع العملية - Application : تطبيق.

الحدث : إنهاء التطبيق. Quit Application -

ثم نضغط التالي.

سيطلب منك تحديد الصورة أو النص الذي تريده أن يظهر ، اختر ما يحلو لك ، ثم اضغط التالي.

في الشاشة الأخيرة سوف نقوم بتسمية زر الأمر ، سنسميه. Cmd_Exit

شغل الآن النموذج وقم بالضغط على الزر ... جرب تنفيذ الأمر.

الآن قم وبنفس الطريقة باضافة أزرار حتى يصبح النموذج بالشكل التالي.

لتغيير تنسيقات وخصائص مكونات النموذج بعد الضغط على خصائص أي شي في الفورم من زر الماوس الأيسر ستظهر شاشة بالشكل التالي:

غير أي خاصية تريد تغييرها ولاحظ النتيجة.

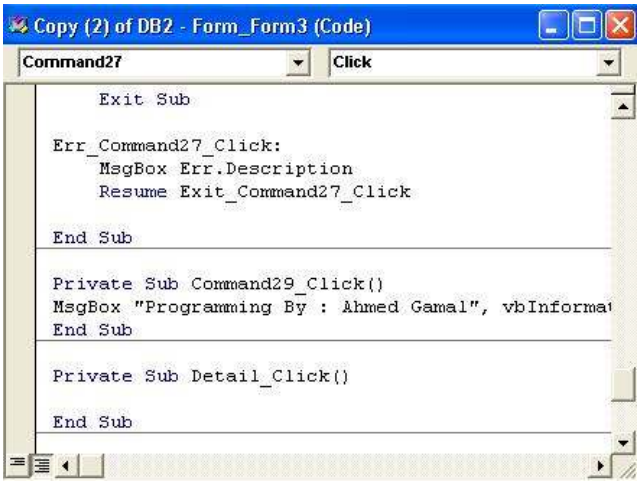
أيضاً سنقوم باضافة زر خاص يحتوي على رسالة تظهر مؤلف البرنامج ،لذا أزل التأشير من على علامة المعالج ، واختر زر أمر وقم برسمه على الفورم ... اضغط عليه واكتب نص مثلاً. About

اضغط الآن زر الماوس الأيسر واختر (بناء حدث - (build events ثم اختر من مربع الحوار (code builder) .

ستظهر لنا شاشة مثل شاشة كود الفيجوال بيسك ، وسوف نكتب فيها الأمر التالي:

```
MsgBox "Programming By : Ahmed Gamal", vbInformation, "About"
```

تماماً مثل كود الفيجوال بيسك وذلك بالشكل التالي:

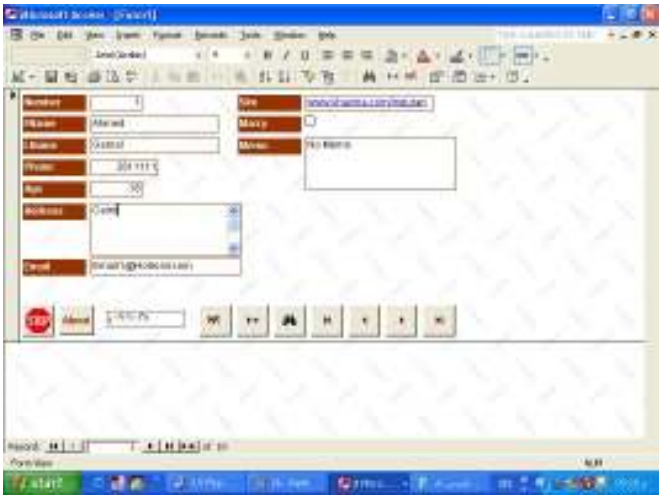


الآن اغلق محرر فيجوال بيسك وجرب زر الأمر الجديد.

سنقوم الآن باضافة مربع نص يضع الوقت فيه بمجرد الضغط عليه ... لذا سنرسم مربع نص ثم نقوم بنفس الطريقة السابقة بتشغيل محرر فيجوال بيسك ، ثم نكتب الأمر التالي في حدث. Click

```
Text30.Text = DateTime.Time
```

آمل أن يتكون لديك نموذج قريب من هذا الشكل:



لن نتوسع في النماذج فقد كان الهدف منها هو أن تعرف:

*إذا كنت تريد البرمجة بأكسيس – إن صح التعبير – : أنها شبيهه جداً بالفيجوال بيسك في موضوع قواعد البيانات.

*إذا كنت مبرمج فيجوال بيسك هذا هو النظام الذي سنحاول الوصول إليه من خلال الفيجوال بيسك في البداية ... ومن ثم سنقوم باضافة المزيد.

ملف مرفق :

قم بمطالعة الملف Db3.zip **في المرفقات لمزيد من التفاصيل .**

Relationships – العلاقات

قبل أن نبدأ نود أن نعرف أولاً لماذا نستخدم العلاقات ؟ لنعرف ذلك لا بد لنا أولاً أن نعرف ما معنى علاقة .

يقصد العلاقة هو اشتراك حقل بين جدولين بحيث تملك كل قيمة في هذا الحقل سجلين ، السجل الأول في الجدول الأول والسجل الثاني في الجدول الآخر .

نعود إلى سؤالنا السابق :نستخدم العلاقة لنضمن شيئين رئيسيين :

أولاً :تحقيق تكامل حقيقي بين جداول قاعدة البيانات .

ثانياً :منع تكرار القيم في أي من الجداول لمنع إهدار حجم قواعد البيانات .

ربما لم نبدأ بعد ، ولكن لنرى مثالاً سريعاً نضيفه سوياً لقاعدة البيانات التي كنا نعمل عليها الدروس السابقة .

لنقل أننا نريد أن نقسم الأصدقاء إلى أربع مجموعات -مثلاً -بحسب السنة الدراسية في الكلية على سبيل المثال ، ونريد أن نعرف عن كل سنة عدد الطلبة فيها وعدد المواد الدراسية .

إذا كنا نعمل بالطريقة التقليدية فسنقول : نضيف لكل صديق حقل خاص بالسنة ، وحقل آخر خاص بعدد الطلبة العام وحقل ثالث بعدد المواد .

لكن لاحظ كم مرة سنكرر عدد المواد ، وعدد الطلبة .

لاحظ أيضاً لو زادت مادة لطلبة أي سنة دراسية ، فهذا يعني أننا سنحدث بيانات جميع الحقول .

وقل نفس الأمر بالنسبة للحذف وغير ذلك .

ربما لا يتضح موضوع الحذف في مثالنا)نوتة الهواتف (ولكن سأخرج -جزئياً -عن الموضوع لأشرح مثالاً صغيراً على عملية الحذف .

لنفرض أن لدينا قاعدة بيانات تحتوي على الموظفين ويهمنا فقط اسم الموظف وفرع الشركة التي يعمل فيها ومكان الفرع .

إذا قامت الشركة مثلاً بحذف فرع ...فهذا يعني أن تقوم يدوياً -أو حتى بالكود -بحذف جميع الموظفين الذين ينتمون إلى هذا الفرع .

لكن الأمر ليس كذلك مع العلاقات ، فبمجرد حذف القسم سوف يحذف تلقائياً الموظفون التابعون له –طبعاً فقط إذا كنا نرغب في ذلك . -

لتلافي ذلك فإننا نستخدم ما يعرف بالعلاقات . RelationShip

أنواع العلاقات :

النوع الأول - One To One :علاقة واحد لواحد .

في هذه العلاقة نقول أن لكل سجل في القاعدة الرئيسية سجل واحد يقابله في القاعدة الثانية .

مثال على هذه العلاقة ضمن قاعدة بيانات الأصدقاء :

بفرض أننا نريد اضافة المعلومات الموجودة ضمن البطاقة الشخصية)رقم البطاقة -فصيلة الدم -مصدر البطاقة (

نستطيع اضافة هذه الحقول مباشرة إلى الجدول الأول ، ولكن لاحظ كم سيصبح حجم هذا الجدول ، وبالتالي إذا حاولنا استخراج اسماء الأصدقاء فقط فسيستغرق المزيد من الوقت .

ربما لن يظهر ذلك مع قاعدتنا البسيطة ولكنه يظهر مع قواعد البيانات الضخمة .

إذن سنقوم بانشاء جدول جديد يحتوي على الحقول التالية)رقم البطاقة -فصيلة الدم -مصدر البطاقة (

PassPort - BClass - From

تبقى شيء واحد وهو الحقل الذي يمثل رابطاً بين الجدولين .

وهذا الحقل الرابط له شروط هي :

*أن يكون موجوداً في الجدولين .

*أن يكون نوعه متطابقاً في الجدولين)فلا يصلح أن يكون الأول رقم والثاني نص . (

*أن يحتوي على قيم فريدة)بمعنى أن لا يتكرر (حتى لا يسبب المشاكل .

ولذا نقول أننا لا نستطيع أن نستخدم حقل الإسم لاخلاله بالشرط الثالث لاحتمال تشابه الأسماء .

لذلك فإن أنسب حقل هو حقل الرقم لأننا ذكرنا أنه سيحتوي على قيم فريدة)غير متشابهه . (

قم الآن باضافة حقل Numberإلى الجدول السابق وعينه كمفتاح رئيسي Primary Key،

وقم بحفظ الجدول تحت اسم Tb_R1أو أي اسم كما تحب .

النوع الثاني - One To Many :علاقة واحد لمجموعة)والعكس صحيح . (

في هذه العلاقة نقول إن لكل سجل في الجدول الرئيسي مجموعة سجلات في الجدول الثاني .

مثال ذلك المثال الذ قمنا بشرحه في الدرس السابق .

لو لاحظت المثال ستجد أن مجموعة من الطلاب يمكن أن يكون لهم سجل واحد في جدول الفرق ، لكن لا يمكن أن يكون لطالب)سجل (واحد أكثر من سجل آخر في جدول الفرق)السنوات الدراسية . (

وهذا هو الفرق بينه وبين النوع التالي .

لذا قم بانشاء جدول ثالث تحت اسم Tb_R2 ويحتوي على الحقول التالية :

Year - Books - Students

السنة -الكتب -الطلاب

يمكننا اضافة حقل جديد في الجدول الأول Tb_Main تحت اسم Year حيث سيكون هو الحقل الرابط بين الجدولين .

قم بالضغط على الجدول الأول ، ثم اضغط على (تصميم Design -) ، ثم قم بالضغط بزر الماوس الأيسر في المكان المطلوب ثم قم بالضغط على (ادراج صف Insert Row -) .

أيضاً لا تنس أن تجعل حقل Year مفتاح رئيسي في الجدول الثاني (جدول الفرق الدراسية) .

***هذا النوع من علاقات هي الأكثر استخداماً .

النوع الثالث - Many To Many :علاقة مجموعة إلى مجموعة :

هل تتذكر العبارة التالية التي أوردتها في الدرس السابق ؟

لكن لا يمكن أن يكون لطالب (سجل واحد) أكثر من سجل آخر في جدول الفرق (السنوات الدراسية) .

إذا أزلنا (لا) فإننا نكون قد حصلنا على النوع الثالث .

مثال هذه العلاقة :

لو أردنا أن نضيف جدولاً يحتوي على اللغات فإننا سنلاحظ التالي :

*يمكن للشخص الواحد أن يكون له أكثر من لغة .

*ويمكن للغة الواحدة أن تكون موجودة عند أكثر من شخص .

للقيام بهذه النوعية من العلاقات فإننا نحتاج إلى جدول بسيط ، فمثلاً لدينا جدولين في البداية هما :

الرقم -الاسم .

- 1أحمد
- 2محمد
- 3خالد

- 4سامي

والثاني للغات :

الرقم -اللغة

- 1العربية

- 2الانجليزية

- 3الفرنسية

أما الجدول الإضافي فهو للعلاقة بين اللغات والأشخاص ، فلكي ندلل على أن أحمد يتحدث العربية والانجليزية ، فيما يتحدث محمد الفرنسية ، أما خالد فيتحدث الثلاث ، بينما لا يتحدث سامي أي لغة !فإننا سوف نشكل الجدول التالي :

رقم الشخص -رقم اللغة

1 - 1
1 - 2
2 - 3
3 - 1
3 - 2
3 - 3

سنقوم الآن ببعض التعديلات على الجداول يصبح كما في المثال المرفق :

ملف مرفق :
قم بمطالعة الملف Db4.zip **في المرفقات لمزيد من التفاصيل .**

والآن وبعد معرفة الانواع المختلفة سنبدأ الآن بانشاء العلاقات بين الجدول ، وتوجد طريقتين للقيام بذلك :

*عن طريق محرر العلاقات .

*عن طريق المعالج الموجود في نوع البيانات Data Typesالموجود في خصائص الحقل .

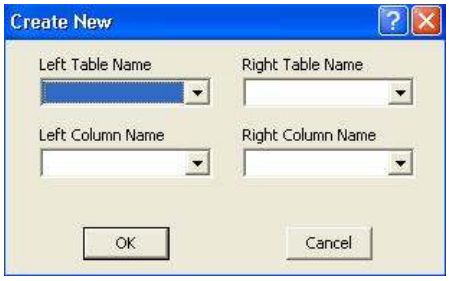
وسوف نستخدم الطريقة الاولى .

سنبدأ الآن بإنشاء العلاقات بين جدول الأصدقاء وجدول معلومات البطاقة الشخصية)تذكر أن العلاقة من نوع واحد إلى واحد . (One To One

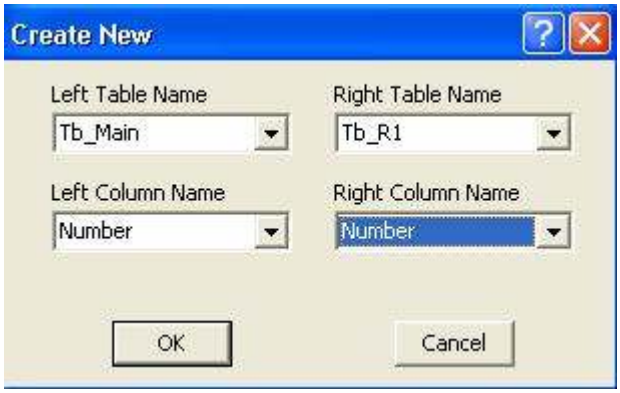
من قائمة - Tools)أدوات (اختر - Relationships)علاقات (سوف تظهر شاشة فارغة -افتراضياً ... -اضغط بزر الماوس الأيمن واختر - Show Tablesإظهار جداول ، وذلك بالشكل التالي :

والآن قم باختيار الجداول واحداً تلو الآخر واضغط - Addاضافة ، وبعد الانتهاء اختر - Close إغلاق ، وسيظهر لك شكل كالتالي :

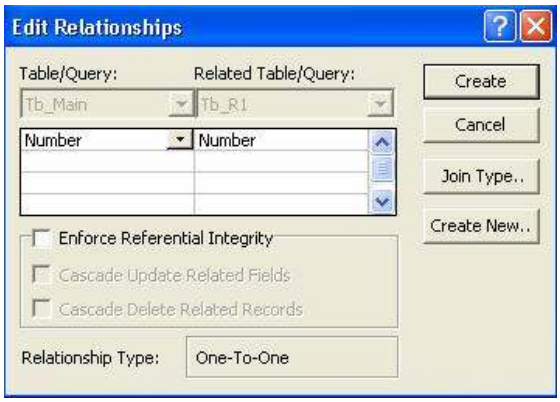
الآن من قائمة Relationships - العلاقات ، اختر تحرير علاقة . Edit Relationship -
سوف تظهر لك شاشة اختر منها . Create New
سوف تظهر لك شاشة جديدة بالشكل التالي :



في خاصية Left Table Name اختر الجدول الذي سيكون مصدراً للبيانات وهو عندنا
Tb_Main
وفي الجزء الأيمن اختر Tb_R1
وبالنسبة لخاصية Left Column Name اختر الحقل الذي سيكون رابطاً من الجدول الرئيسي
، ولذا سنختار Number
وفي الجزء الأيمن سيكون الحقل أيضاً هو . Number
وذلك بالشكل التالي :



قم الآن بالضغط على زر Ok وستظهر لك شاشة بالشكل التالي :



لاحظ ظهور نوع العلاقة في الجزء الأسفل من الشاشة .

أما بالنسبة لأزرار الاختيار :

- 1مربع الإختيار الأول :في هذا الخيار تحدد إذا ما كنت تريد منع المستخدم على ادخال حقل في الجدول الثاني دون أن يكون له نظير في الجدول الأول .

- 2مربع الإختيار الثاني :في هذا الخيار تحدد إذا ما كان تحديث بيانات الجدول الثاني عند تغيير قيم حقل المفتاح الرئيسي .

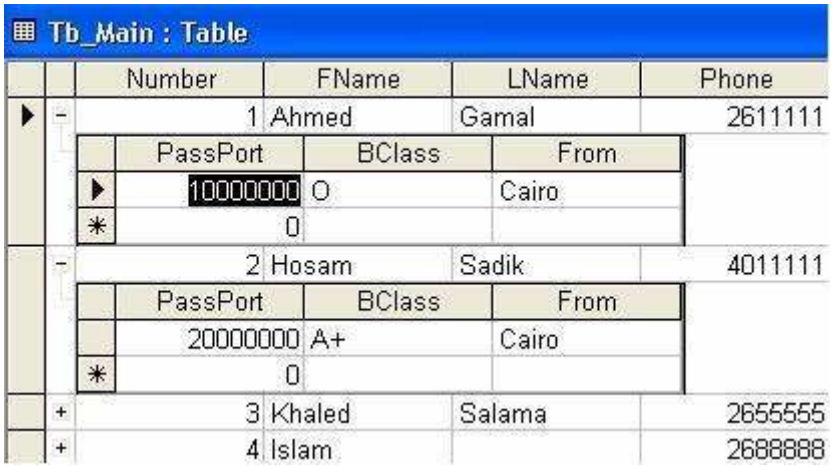
- 3مربع الإختيار الثالث :في هذا الخيار تحدد إذا ما كنت تريد حذف بيانات الجدول الثاني عند حذف نفس السجل في الجدول الرئيسي .

هناك زر آخر تحت اسم - Join Type)نوع الدمج (وبها ثلاث خيارات لأنواع مختلفة من الدمج .

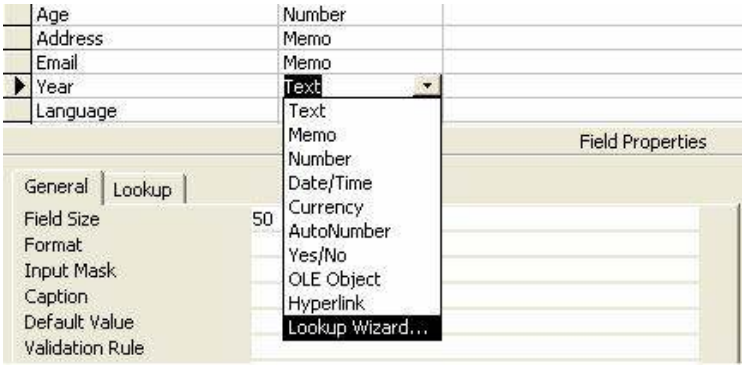
والآن قم بالضغط على زر - Create)إنشاء . (ستلاحظ ظهور رسم لعلاقة بين الجدولين وذلك بالشكل التالي :

وبهذا نكون قد أنشأنا العلاقة الأولى من نوع ... One To Oneفي العلاقة التالية سوف نستخدم المعالج .

للتأكد من عمل هذه العلاقة اذهب إلى Tb_Mainوانقر علامة +الموجودة على يساره ...أتمنى أن يظهر لديك الشكل التالي :



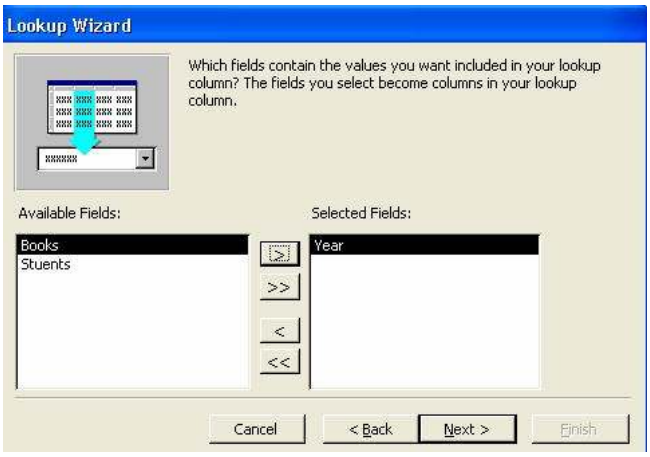
سوف نقوم الآن بإنشاء العلاقة الخاصة بالسنة الدراسية ...لذا سوف نتجه إلى جدول TB_Main ونضغط Designومن ثم نذهب إلى الحقل Yearونختار Lookup Wizard، وذلك بالشكل التالي :



أول شاشة تظهر لنا تخيرنا فيما إذا ما كنا نريد أن نختار جدولاً لننشأ معه علاقة ، أم نريد أن نضع قيم ...سوف نترك الخيار الافتراضي وهو الأول ونضغط على التالي .

الشاشة التالية سوف تطلب منا اختيار الجدول الذي نريد إنشاء علاقة معه ، وسنختار . Tb_R2

في الشاشة الثالثة سوف يطلب اسماء الحقول التي نريد لها أن تظهر في الجدول ...سوف نختار Yearوذلك بالشكل التالي :



ومن ثم نضغط على التالي .

في الشاشة التالية سوف يظهر لنا بيانات الحقل المختار الموجودة حالياً في الجدول الذي اخترناه ، لذا سوف نختار التالي .

أخيراً سوف نختار اسم الحقل الذي نريد له أن يظهر العلاقة ، وسنتركه على ما هو عليه ، ونختار Finish - إنهاء .

سيخبرك أن الجدول لا بد أن يحفظ أولاً قبل القيام بضبط العلاقة ...قم بالحفظ ، ومن ثم أغلق شاشة التصميم .

الآن قم بتشغيل الجدول واختر الحقل Yearولاحظ ظهور قيم الجدول الثاني في قائمة وبإمكانك الاختيار منها مباشرة ...وذلك بالشكل التالي :

Email	Year	Language
Email1@Hotma	First Year	
Email1@Yahoo	First Year	
Email1@Hotma	Forth Year	
Email1@Yahoo	Second Year	
Email1@Hotma	Therd Year	
Email1@Yahoo	First Year	
Email1@Hotma	Second Yea	
Email1@Hotma	Second Yea	

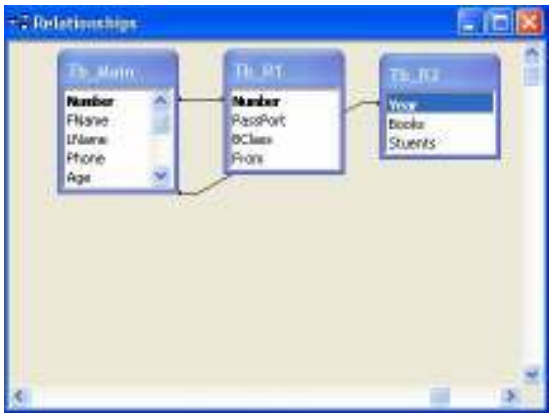
أيضاً اختر الجدول Tb_R2 وانقر علامة + سيظهر لك جميع الطلبة الذين ينتمون إلى هذه السنة وذلك بالشكل التالي :

	Year	Books	Stuents
▶ −	First Year	5	120

	Number	FName	LName	Phone
▶ +	1	Ahmed	Gamal	2611111
+	3	Khaled	Salama	2655555
+	6	Sami	Ali	1111111
*	0			0

+	Forth Year	4	50
+	Second Year	6	100
+	Therd Year	7	80

أخيراً سوف نلقي نظرة على شاشة العلاقات التي قمنا بتشغيلها في الدرس السابق ...سوف يظهر لك شكل قريب من التالي :



الملف المرفق يحتوي على جميع العلاقات اضافة إلى جميع محتويات الدروس السابقة

ملف مرفق :

قم بمطالعة الملف Db5.zip **في المرفقات لمزيد من التفاصيل .**

عمليات خاصة بقواعد البيانات

سنتعرف في هذا الدرس على بعض العمليات الخاصة بقواعد البيانات .

أولاً :عمليات الإستيراد والتصدير . (Import & Export)

الإستيراد :

نقصد بعملية الإستيراد جلب معلومات إلى قاعدة البيانات من خارجها ...

من قائمة ملف Fileاختر)جلب معلومات خارجية (Get External Data -ومنها اختر (استيراد (Import -

اختر الملف الذ تريد استيراد جزء منه اختر أي ملف وليكن حتى نسخة من قاعدة البيانات الحالية .

ستظهر لك شاشة قم بالضغط على)خيارات (Option -لتوسيع الشاشة لتصبح بالشكل التالي :

في الجزء العلوي نختار نوع محتوى قاعدة البيانات الذي نريد استيراده ...وليكن)جداول - (Tablesواختر أي جدول من القائمة .

أما الخيارات التي في الأسفل ففي الأول نحدد الاختيارات التي نريد أن نستوردها)نجلبها (مع قاعدة البيانات .

وفي المنتصف نحدد إذا ما كنا نريد أن نستورد الجدول كبناء تصميمي فقط أم الجدول اضافة إلى البيانات بداخله .

وأخيراً في اليمين نحدد إذا ما كنا نريد الاستيراد كجدول أو كاستعلام .

اضغط موافق ولاحظ ظهور الجدول ضمن قائمة الجداول .

هناك نوعية ثانية من الاستيراد وهو الاستيراد المرتبط ، وفي هذه الحالة فإن أي تعديل على بيانات الجدول المضاف ينعكس على قاعدة البيانات المضاف منها الجدول .

ولاختيار هذا النوع من قائمة ملف اختر)استيراد (Import -ثم)جدول مرتبط Link - Table) .

وبنفس الطريقة اختر الجدول ...عدل في بياناته ولاحظ التغيير في نفس الجدول من قاعدة البيانات الثانية .

التصدير :

هو العملية العكسية -نسبياً -للعملية السابقة ...الآن قم بتحديد جدول من قاعدة البيانات واضغط على قائمة ملف واختر)تصدير . (Export -

ستظهر شاشة تحدد فيها قاعدة البيانات التي نريد التصدير إليها ...

وأخيراً ستظهر شاشة تطلب منك ادخال الإسم الذ تريد أن يحمله الجدول المصدر في قاعدة البيانات تلك ...اختر أي اسم ، كما يمكنك الإختيار بين تصدير بيانات الجدول من عدمه ، ثم انقر موافق .

افتح قاعدة البيانات الثانية ولاحظ ظهور الجدول الجديد فيها .

قائمة الأدوات . Tools Bar

تحتوي هذه القائمة على العديد من الخيارات المهمة لقواعد البيانات منها :

1- أدوات مساعدة لقواعد البيانات . Database Utilites

يهمنا في هذا الجزء عمليتان ...الجزء الأول هو تحويل قاعدة البيانات Convert Database يتيح لك هذا الخيار تحويل قاعدة بياناتك إلى اصدرات سابقة من أكسيس ...لكن لاحظ أنه لن يمكنك التغيير في بناء النسخة المحولة .

نحتاج مثل هذه الخاصية مثلاً عند التعامل مع Data Controlفي الفيجوال بيسك والتي لا تتعامل إلى مع Access 97 .

والجزء الثاني هو ضغط وإصلاح قواعد البيانات . Compact and Repair Database

نحتاج بين كل فترة وأخرى إلى هذا الإختيار والذي يقوم بثلاث مهمات رئيسية :

*ضغط بيانات قاعدة البيانات بحيث يصبح حجمها أصغر ، وذلك عن طريق تحديث طريقة حفظ المعلومات إلى طريقة قياسية .

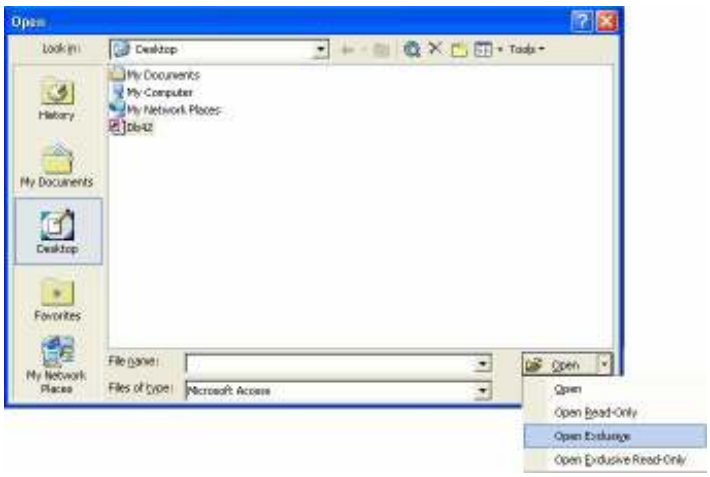
*زيادة سرعة الوصول إلى بيانات قاعدة البيانات عن طريق اعادة ترتيب المحتويات .

*اصلاح الأخطاء التي قد تحدث في قاعدة البيانات والتي قد تنشأ عن عدد من المشاكل منها مثلاً اغلاق الجهاز دون اغلاق قاعدة البيانات ...وهكذا .

*ولا تنس انه يقوم بحذف السجلات التي حذفناها -ظاهرياً -فيما مضى .

2- أمان . Securty

نحتاج إلى فتح قاعدة البيانات بطريقة ... Exuclusive وذلك من خلال أمر فتح Open اضغط على المؤشر الصغير بجانب زر الفتح واختر Exuclusive وذلك بالشكل التالي :



بعد ذلك يمكنك وضع كلمة سر للقاعدة وتغييرها .

لغة الاستعلامات الهيكلية TSQL

هذا هو الجزء الأكثر اهمية ، حيث ان جمل الاستعلام تعمل مع جميع قواعد البيانات بصورة موحدة مع اختلافات طفيفة جداً ، كما سنستخدمها في فيجوال بيسك ومن خلال Asp أيضاً وتستخدم مع غيرهما بنفس الطريقة تقريباً ، وسوف نتطرق هنا لأشهر هذه الجمل مع وجود غيرها ايضاً.

ما هي SQL ؟

هذه الكلمة اختصار ل. Structured Query Language

وتصلح تقنية الاستعلام للعمل مع جميع أنواع قواعد البيانات MS - Access - Oracle :
sql Serverوغيرها.

تقوم هذه الجملة بالبحث – كما هو شائع – إلا أنها تستخدم أيضاً في الاضافة والحذف والتعديل وانشاء الجداول وحتى قواعد البيانات . والتحكم بها والادارة وغير ذلك.

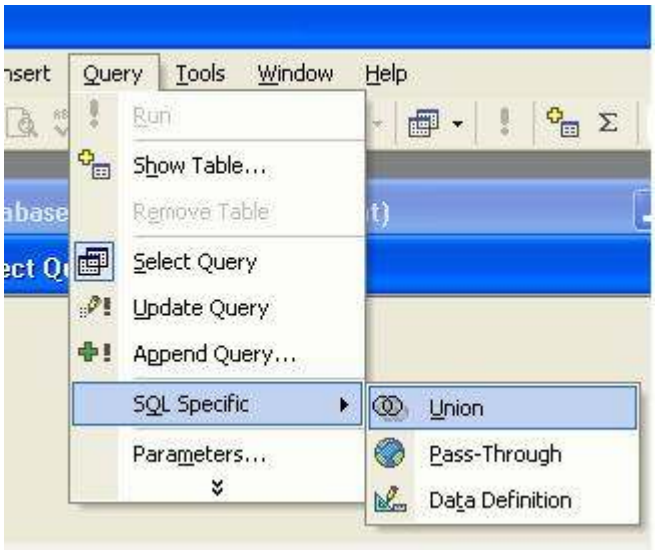
أين ستكتب جمل الإستعلام في أكسيس ؟

اختر استعلامات Quires من القائمة التي كنا تختار منها Table أو. Forms

اختر. Ceate Quriry In Deign Viwe

ستظهر لك شاشة ... اختر اغلاق. Close -

والآن من قائمة Query اختر SQL Specficومن ثم اختر Union وذلك بالشكل التالي:



ستظهر لك شاشة فارغة ... هذه هي الشاشة التي سنكتب فيها أوامر الإستعلام.

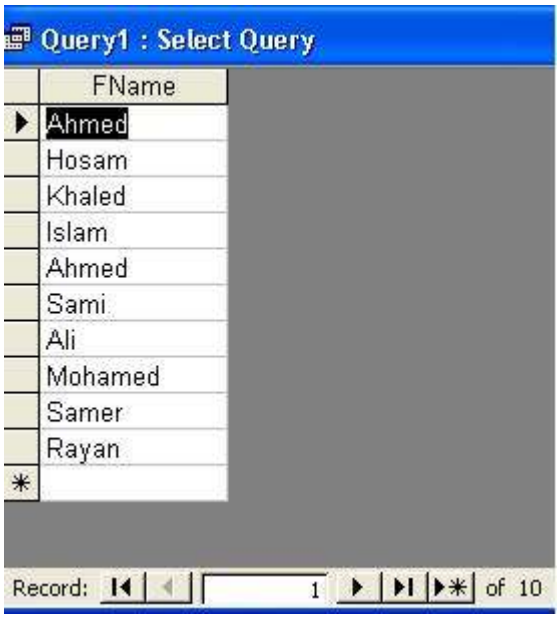
في أعلى الشاشة نجد رمزاً للقرص المرن وهذا الرمز يرمز إلى حفظ الإستعلام ... أيضاً علامة التعجب تعني (نفذ الإستعلام)

سوف نقوم بكتابة الإستعلام التالي :

```
Select FName From Tb_Main;
```

لا تقلق فلسوف نشرح ماذا يعني الأمر السابق!!!

اضغط على حفظ واحفظه تحت الاسم الذي يناسبك ... وجرب تنفيذ الإستعلام بالضغط على علامة التعجب ... أيضاً أغلق الشاشة واضغط على اسم الإستعلام ... أتمنى أن تظهر لك في الحالتين شاشة شبيه بذلك:



أيضاً للعودة لشاشة التصميم ... اضغط على الاسم بزر الماوس الأيمن واختر (عرض التصميم – Design View) وستظهر شاشة كتابة الكود.

ملف مرفق :
يحتوي الملف db6.zip على كافة التعديلات حتى الآن .

والآن سنبدأ في تفصيل جمل الاستعلام ، **وسوف نتعرف أولاً على جمل البحث والتي تبدأ بـ Select .** وهذه هي الصيغة العامة :

```
Select Colums_Name From Table_Name;
```

نلاحظ تكون جملة الإستعلام من أربعة أجزاء:

Select : وهي التي تميز جملة البحث عن غيرها من جمل TSQL.

Colum_Name : اسم الحقل أو العمود الذي تريد له أن يظهر.

From : وهي أيضاً صيغة خاصة بجملة البحث وتتغير في باقي جمل TSQL.

Table_Name : هنا نكتب اسم الجدول.

ولنلق نظرة قبل البدء على ما يمكن تسميته (القواعد النحوية: Syntex)

*لا يوجد أي فرق بين الأحرف الكبيرة والصغيرة.

*مسافة = مسافتان = عشرة أسطر : لا يعترف بأكثر من مسافة.

*تنتهي جميع جمل الإستعلام بفاصلة منقوطة ؛ إلا أننا في الفيجوال بيسك وفي بعض البرامج الأخرى لا نحتاج إليها.

لنلق نظرة على المثال السابق

: SELECT FName FROM Tb_Main;

لاحظ أننا استخدمنا جملة Select لاستخراج الاسم الأول من الجدول. Tb_Main

وبهذه الصورة يمكننا استخراج مواقع الأصدقاء وذلك بالشكل التالي

SELECT Site FROM Tb_Main;

وهكذا مع الباقي.

وإذا كنا نريد استخراج أكثر من عمود فيمكننا وضعهم بالتوالي وتفصل بينهم الفاصلة ، فمثلاً لاستخراج الإسم الأول والإسم الأخير :

SELECT FName,LName FROM Tb_Main;

ومثلاً لاستخراج الإسم الأول والعمر والبريد الإلكتروني :

SELECT FName, Age,EMail FROM Tb_Main;

ولاستخراج جميع البيانات فإننا نضع * محل اسماء الحقول وذلك بالشكل التالي :

SELECT* FROM Tb_Main;

ولاستخراج البيانات ولكن بدون تكرار نستخدم ... DISTINCT فمثلاً لاستخراج الإسم الأول مع حذف التكرار :

SELECT DISTINCT FNAme FROM Tb_Main;

أيضاً لترتيب البيانات المخرجة تصاعدياً أو تنازلياً نستخدم الأمر:

ORDER BY colum

أو للتصاعدي

ORDER BY colum DESC

فمثلاً لاستخراج جمع البيانات مع الترتيب التصاعدي حسب الإسم الأول :

```
SELECT * FROM Tb_Main ORDER BY FName DESC;
```

ويمكننا عمل مستويين للترتيب – كما هو شائع – وذلك باستخدام الفاصلة ، فمثلاً للترتيب حسب الإسم الأول ومن ثم الإسم الأخير ولكن تنازلياً هذه المرة :

```
SELECT * FROM Tb_Main ORDER BY FName,Lname ;
```

أيضاً لاستخراج الإسم الأول تحت مسمى MyName فإننا نستخدم As وذلك بالشكل التالي :

```
Select Fname as MyName From Tb_Main;
```

وسنستفيد من As لاحقاً في عمليات الاستعلام باستخدام الدوال.

يمكننا أيضاً اجراء العمليات البسيطة على جمل البحث ، فمثلاً لاستخراج أعمار الأصدقاء بزيادة عشرة سنوات:

```
SELECT Fname,Age+10 as Age2 FROM Tb_Main;
```

لاحظ أنك تضطر إلى اعادة تسمية الحقل الناتج وإلا فإن Access سوف يضع له اسماً من عنده.

ومثلاً لدمج الإسم الأول مع الإسم الأخير :

```
SELECT Fname+ " " + LName As CName FROM Tb_Main;
```

استخدام الجمل الشرطية مع أوامر الإستعلام

استخدام معامل المساواة والمعاملات الرياضية.

تأخذ صيغة استخدام المساواة كشرط في عملية الإستعلام الصيغة التالية:

```
Select * From Tb_Main Where Field = Value;
```

ويمكننا استخدام > أو < أو => أو =< أو <> وطبعاً. =

وأخيراً معاملين جدد سنشرحهم بإذن الله لاحقاً وهما Like و. Between

فمثلاً للاستعلام عن الأصدقاء الذي تساوي أعمارهم ١٨ سنة :

```
SELECT * FROM tb_Main WHERE age = 18;
```

ولعرض أسماء من هم أصغر من ٥٠ سنة:

```
SELECT FName,LName FROM tb_Main where age < 50;
```

وسنضع Value بين علامتي تنصيص مفردة في حالة كانت نصوصاً وذلك بالشكل التالي – للبحث عن الأشخاص والذين لهم الإسم (أحمد : (

```
SELECT * FROM tb_Main WHERE Fname = 'ahmed';
```

كما سنضع علامتي # عند البحث عن تواريخ ، ولا توجد أمثلة لتاريخ في قاعدة البيانات ، لكنها تأخذ صيغة شبيهه بالتالي :

```
Select * Form Table1 Where Date > #12/03/04#;
```

والآن لندمج بعض ما تعملناه في جملة واحدة:

لاستخراج الإسم الأول والأخير من الجدول بترتيب تنازلياً ومدمجة سوياً تحت اسم Name والتي تحقق شرط أن العمر أكبر من ١٧ :

```
SELECT Fname & " " & Lname AS MyName FROM tb_Main
WHERE age > 17 ORDER BY FNAME, LName DESC;
```

والآن : ماذا لو أردنا البحث بتحقيق مجموعة شروط أو أحدها أو تحقيق شرط مع انتفاء آخر ؟

من أجل هذا الغرض نستخدم المعاملات المنطقية البسيطة.

لدينا نوعين من المعاملات المنطقية البسيطة الأكثر استخداماً:

And : تعطي قيمة صحيحة True في حالة كان الطرفين صحيحاً والباقي. false

Or : تعطي قيمة صحيحة True في حالة كان أحد الطرفين صحيحاً وإلا فإنه يعطي. false

والآن إلى مثال سريع ، سنبحث عن الأشخاص الذين يكبر عمرهم عن ١٧ بشرط ألا يكونوا متزوجين :

```
Select Fname from tb_main where age > 17 and marry = false;
```

لنفترض أيضاً أننا نريد معرفة الأشخاص الذين تزيد أعمارهم عن ١٧ سنة وليس لديهم بريد الكتروني:

قبل أن تجرب المثال قم بتشغيل قاعدة البيانات وقم بحذف البريد الإلكتروني لبعض الأشخاص ، واكتب للبعض كلمة No Mail وللآخرين No.

الآن انظر الأمر التالي :

```
SELECT * FROM Tb_Main WHERE age>17 And (Email='No'
or Email='No Mail');
```

سنفصل الآن هذه الجملة:

*الشرط الأول : Age > 17 : العمر أكبر من ١٧.

*الشرط الثاني : Email = 'No' : البريد الإلكتروني يحتوي على كلمة. No

*الشرط الثالث : Email = 'No Mail' : البريد الإلكتروني يحتوي على كلمة. No Mail

واستخدمنا نوعين من المعاملات المنطقية And و. Or

متى تحقق الشرط الأول (العمر أكبر من ١٧) فإنه ينتظر تحقيق أحد الشرطين الذين تدمج بينهم Or .

وعندها يعرض السجل الذي يحقق قيمة True لهذا التعبير.

قبل النهاية **كن حذراً عند استخدام And مع Or في جملة واحدة ، وذلك عن طريق ترتيب الأقواس بطريقة صحيحة.**

استخدام المعامل. Like

نستخدم المعامل Like للبحث عن الكلمات المشابهه لتعبير معين ...

ونستخدم للمعامل Like الصيغة التالية :

```
Select * From Table Where Field Like '*Name*';
```

نستخدم * للدلالة على وجود أحرف ما ... وقد نستخدمها في البداية ، أو النهاية ، أو كليهما .

وسيتضح الأمر في الأمثلة:

لعرض الأشخاص الذين قد تحتوي أسماءهم الأولى على ... hm لذا سنستخدم جملة استعلام بهذا الشكل:

```
SELECT * FROM Tb_Main WHERE FName like '*hm*';
```

أما لو أردنا البحث عن الأشخاص الذين تبدأ أسماءهم بحرف A لذا لن نضع * قبل كلمة البحث :

```
SELECT * FROM Tb_Main WHERE FName like 'A*';
```

ولو أردنا البحث عن الأشخاص الذين ينتهي اسمهم بحرف معين سنضع * في البداية دون النهاية.

*في MS SQL Server نستخدم ٪ بدلاً من. *

مما سبق نستنتج أن * تعني أي عدد من الحروف ، لكن ماذا لو أردنا تحديد عدد الحروف ؟
في هذه الحالو نستخدم "؟" والتي ذكرناها في درس البحث في البداية.

إذن : ماذا لو أردنا أن نبحث عن الأشخاص الذين تبدأ أسماءهم بحرف A بشرط أن يكون عدد حروفهم ٣ فقط.

```
SELECT * FROM Tb_Main WHERE FName like 'A??';
```

والآن إلى مثال واقعي نسبياً . جرب تعديل حقل العنوان Address للأشخاص بحيث يحتوي على أكثر من معلومة اضافة للمدينة.
في هذه الحالة نستطيع استخراج الأشخاص الذين يسكنون مدينة واحدة عن طريق الشرط . Like هل تعرف كيف ؟؟؟

استخدام المعامل. Between

نستخدم هذا المعامل للبحث ضمن نطاق معين ، وأشهر استعمال لهذا المعامل هو استخدامه في البحث ضمن التواريخ.
يأخذ البحث باستخدام between الصيغة التالية:

```
SELECT colum FROM table WHERE field BETWEEN v1
AND v2;
```

فمثلاً لاستخراج الأشخاص الذين تتروح أعمارهم بين ٢٠ و ٦٠:

```
SELECT *
FROM tb_main
WHERE age between 20 and 60;
```

ويمكن استخدام نفس الطريقة مع الأسماء ، والتواريخ بطبيعة الحال.

سنبتعد الآن موؤقتاً عن جو البحث وسنتحدث عن اضافة وحذف الجداول في قواعد البيانات .

سنبدأ أولاً بصيغة **انشاء جدول** وذلك بالشكل التالي :

```
CREATE TABLE tb1  ( colum type, colum type,.....);
```

وكمثال على ذلك ... لننشئ جدولاً تحت اسم Tb2 يحتوي على حقلين : الإسم Name من نوع (نص – String) والآخر Age من نوع Number :

```
CREATE TABLE tb1  (name text,age number );
```

إذن ماذا لو أردنا أن نحدد حجم حقل العمر Name بخمس خانات فقط ... إذن ضع عدد الخانات بين قوسين مع تعيين نوع بيانات الاسم ك Text وذلك بالشكل التالي :

```
CREATE TABLE tb1  (name text(6),age number );
```

آخر نقطة سأشرحها في انشاء الجداول هي كيف نجعل الحقل لا يقبل فراغ – Null ... ولنجرب هذا المثال مع الإسم أيضاً :

```
CREATE TABLE tb1  (name text(6) Not Null,age number
);
```

ولكن ماذا عن **الحذف** ... في الواقع هذا الكود يقوم بحذف الجدول الذي قمنا بإنشاءه تواً :

```
DROP Table tb1;
```

ولو كان هذا الجدول مرتبطاً بعلاقات فلا بد من تحديثها باضافة CASCADE إلى آخر الكود وذلك بالشكل التالي :

```
DROP Table tb1 CASCADE;
```

لا تنسى أن تنشأ الجدول بعد حذفه لأننا سوف نعمل عليه مجدداً .

والآن سنبدأ في **اضافة حقول** إليه وذلك بعد عملية الانشاء – أعتقد أن الفرق بين الجدول وقاعدة البيانات والحقول أصبح واضحاً – .

نستطيع اضافة الحقول بصيغة عامة بالطريقة التالية :

```
ALTER TABLE tb1 ADD colum type;
```

فلنضف حقلين جديدين : الأول هو تاريخ الميلاد ، والثاني النوع (ذكر – أنثى) .

```
ALTER TABLE tb1 ADD birth_Date date,Gender Text;
```

والآن **لنحذف أحد هذه الحقول** وليكن حقل Gender ... لاحظ أننا دائماً نضيف كلمة CASCADE لتحديث العلاقات – في حال وجودها – ، وذلك بالشكل التالي :

```
ALTER TABLE tb1 DROP gender CASCADE;
```

والآن سوف نتعلم **اضافة السجلات** ، وسوف نستخدم Tb_Main كجدول لنجرب أوامرنا عليه ، والآن إلى الصيغة العامة لأمر الإضافة :

```
INSERT INTO table VALUES (v1, v2,.....);
```

لاحظ أننا سنمرر القيم واحداً وراء الثاني وتفصل بينهم فاصلة ، مع مراعاة وضع النصوص بين علامتي تنصيص "" والتواريخ بين علامتي ## .

سنمرر القيم التالية : الرقم – الإسم الأول – الإسم الأخير – رقم الهاتف – العمر – العنوان – البريد الإلكتروني – السنة – اللغة – الموقع – متزوج/لا – ملاحظات .

وذلك بالشكل التالي :

```
Insert into tb_main values(11,'ahmed','gamal',6666666,18,'Cairo','Hammada2091','First Year','','',false,'');
```

تواجهنا مشكلة وهو أننا جعلنا الرقم حقلاً وحيداً – لا تتكرر قيمه – لذا وجب علينا في كل مرة أن نغير الرقم ... برمجياً يمكننا ذلك ولكن من خلال الأكسيس جرب ازالة رقم ١١ ووضع كلمة MyNumber بالشكل التالي :

```
Insert into tb_main values(MyNumber,'ahmed','gamal',6666666,18,'Cairo','Hammada2091','First Year','','',false,'');
```

جرب تنفيذ جملة الإستعلام وستظهر لك رسالة تطلب منك ادخال قيمة MyNumber . ضع قيمة بشرط ألا تكون مكررة وسوف يتم اضافتها في الجدول .

* ملاحظة :

إن هذا الجزء في الأكسيس سيقابله متغير في الفيجوال بيسك .. كذا الامر بالنسبة للبحث .

هل تتذكر جملة البحث التالية :

```
;'Select * From Tb_Main where name='ahmed
```

جرب أن تزيل Ahmed وضع متغيراً MyName بالشكل التالي :

```
Select * From Tb_Main where name = myname;
```

وهذا يعطيك تحكماً فيما ترغب في البحث عنه ...

التعديل :

الصيغة العامة لتعديل بيانات سجل بالشكل التالي :

```
UPDATE table_name SET colum1 = v1 , colum2 = v2
WHERE colum = v;
```

سنجرب الآن مثالاً لجعل العمر = ٢٨ والموقع = www.vb4arab.com للشخص الذي اسمه الأول (Ahmed) والاخير (Gamal) :

```
Update tb_main set age=25 , site='www.vb4arab.com'
;'where fname='ahmed' and lname='Gamal
```

ولكن ماذا لو كان الأمر يشمل أكثر من سجل . في الواقع هذا الأمر شائع ويشبهه المثال التالي :

```
Update tb_main set fname='Ahmad' where
;'fname='ahmed
```

قد يستخدم مثل هذا المثال لتوحيد الأسماء خصوصاً مع اللغة العربية ، كما يمكن أن يستخدم أيضاً لتجاهل تأثير الهمزة في عمليات البحث ، فمثلاً يصبح أحمد = احمد .

ولكن ماذا لو أردنا تعميم التغييرات ... في هذه الحالة لن نضع الشرط ، ويمكن وضع أمر شبيه بالأمر التالي لزيادة الأعمار ٥ سنوات :

```
Update tb_main set age=age+5;
```

يعتبر هذا الأمر من أكثر الأوامر شيوعاً في قواعد بيانات الشركات ... فمثلاً يستخدم لزيادة رواتب الموظفين بنسبة معينة (يمكن أن تكون زيادة ثابته أو نسبيه) .

ولكن ماذا عن **الحذف** ... في الواقع تتخذ جملة البحث صيغة عامة كالتالي :

```
DELETE FROM table WHERE colum = value;
```

ويمكن أن يكون الحذف تحت تأثير تحقق أكثر من شرط .. في المثال التالي سوف نحذف جميع الأشخاص الذين تزيد أعمارهم عن الأربعين :

```
Delete from tb_main where age>40;
```

ولكن ماذا عن **حذف كامل محتويات جدول ما** ... هذه هي الطريقة :

```
DELETE * FROM table;
```

الدوال في الاستعلامات .

تستخدم العديد من الدوال ضمن طيات جمل الاستعلام ، وهي شائعة الاستخدام ، ومريحة ، وتعيد قيمة وحيدة – لا تعيد جدول – سنتعرف على بعض الدوال مع بعض الأمثلة خلال هذا الدرس .
ولننظر نظرة سريعة إلى الصيغة العامة لاستخدام الدوال والتي تأخذ الشكل التالي :

```
SELECT func(colum) FROM table WHERE condition;
```

الدالة AVG :

تعطينا هذه الدالة متوسط حقل ما ، ولحساب متوسط الأعمار في قاعدة البيانات مثلاً نكتب أمراً كالتالي :

```
Select AVG(Age) from tb_main;
```

ولك ان تتخيل ماذا سنفعل لو لم نستخدم هذه الدالة ، كنا سندور على جميع السجلات حيث نجمع ارقام كل سجل ثم ننتقل إلى التالي وهكذا – برمجياً – .

أيضاً يمكننا استخدام الدالة بشرط ... فمثلاً لحساب متوسط أعمار الأشخاص الذين لا تزيد أعمارهم عن ٢٥ :

```
SELECT AVG(Age) FROM tb_main where age < 25;
```

لو لاحظت لوجدت أن ناتج الدالة يظهر في حقل تحت اسم Expr100 أو ما شابه ... لذا قم – إذا كنت تريد – باعادة تسمية حقل الناتج عن طريق As كما تعلمنا سابقاً وبالشكل التالي :

```
SELECT AVG(Age) as AVGAGE FROM tb_main WHERE
age < 25;
```

لاحظ أن الدالة سوف تتجاهل السجلات الفارغة ...

الدوال Sum,Max,Min :

تعطي هذه الدوال المجموع – الأكبر – الأصغر على التوالي ، وهذا مثال على أكبر عمر في قاعدة البيانات :

```
SELECT max(Age) AS mxAGE FROM tb_main;
```

لا تنس أن بإمكاننا وضع شرط لعملية الإستعلام .

الدالة Count :

وتعيد هذه الدالة عدد السجلات ولها صورتان تعمل مع أكسيس ... أما الصورة الثالثة فهي لا تعمل على الاكسيس :

الصيغة الأولى : Count(colum) ... وتعيد عدد السجلات التي لم يكن هذا الحقل فارغاً فيها .

الصيغة الثانية : Count(*) ... وتعيد عدد السجلات دون النظر إلى إذا ما كانت فارغة أم لا .

أما الصيغة الثالثة : COUNT(DISTINCT Colum) حيث تقوم بعد السجلات مع تجاهل المكرر منها .

ولنر مثالاً على الصيغة الثانية :

```
Select count(*) From Tb_Main;
```

لا تنس أن بإمكانك وضع شرط لعملية العد .

كانت هذه بعض الدوال البسيطة والاكثر شهرة في TSql والآن ، ماذا لو كانت **لدينا علاقات ونرغب في العمل عليها** .

في مثل هذه الحالة ... نريد الحصول على معلومات البطاقة الشخصية الخاصة باسم ما وذلك من Tb_R2 – لاحظ أنه لا يوجد حقل للاسم في الجدول الثاني – .

```
SELECT tb_main.Fname, tb_main.lname,
Tb_R1.passport, tb_r1.bclass, tb_r1.from
FROM tb_main, tb_r1
WHERE tb_main.Number = tb_r1.Number;
```

دعنا الآن نلاحظ الفروق بين هذه الطريقة والطريقة السابقة :

أولاً : نقوم بكتابة اسم الجدول.اسم الحقل وذلك مهم لأننا نتعامل مع أكثر من جدول .

ثانياً : نقوم بكتابة اكثر من جدول بعد عبارة From لأننا نريد النواتج من أكثر من جدول .

ثالثاً : السطر الأخير من جملة الاستعلام هو لكي يعرض المعلومات التي تتشابه أرقامها في الجدولين سوية .

دعنا الآن نجرب مثالاً آخراً ... وذلك بالاستعلام عن كافة المعلومات التي تتعلق بالأصدقاء الذين تزيد أعمارهم عن ٢٠ سنة .

```
SELECT tb_main.Fname, tb_main.lname,
Tb_R1.passport, tb_r1.bclass, tb_r1.from
FROM tb_main, tb_r1
WHERE tb_main.Number = tb_r1.Number and
tb_main.age >25؛
```

والآن نريد تطبيق امثلة جديدة ، ولكي نبدأ في تطبيق أمثلة هذا الدرس لا بد أن يكون بعض الأصدقاء لديهم سجلات في الجدول الثاني والبعض الآخر لا .

ومازال حديثنا حول العلاقات وما يتعلق بها ... وسنتحدث عن الدالة Join .

النوع الأول : Inner Join :

هي الطريقة العادية والتي تحدثنا عنها في الدرس السابق تعطينا النواتج في حال كان شرط العلاقة متحققاً في كلا الجدولين .

بمعنى : جرب أن تحذف بعض سجلات الأصدقاء من الجدول الثاني ، وستجد أن معلوماتهم – حتى الأساسية – لن تظهر .

هذا النوع من الربط يطلق عليه Inner Join وهو الافتراضي ... ويمكننا كتابته بالشكل التالي للتفريق بين الأنواع المختلفة :

```
SELECT tb_main.Fname, tb_main.lname,
Tb_R1.passport, tb_r1.bclass, tb_r1.from
FROM tb_main inner join tb_r1
on tb_main.Number=tb_r1.Number;
```

لاحظ أننا استبدلنا Where ب on كما وضعنا inner join بين اسمي الجدولين .

النوع الثاني : Left Join :

في هذا النوع يعرض جميع بيانات الجدول الأول (على اليسار) ومن ثم يعرض البيانات المقابلة في الجدول الثاني .

لاحظ في هذا النوع أمرين :

* إذا كانت هناك سجلات (على اليسار) لا تقابلها سجلات على اليمين فإن اليسار يعرض ويعرض اليمين فارغاً .
* وإذا كانت هناك سجلات (على اليمين) ليس لها مقابل على اليسار فإنها لا تُعرض أبداً .

مثال هذه النوعية :

```
SELECT tb_main.Fname, tb_main.lname,
Tb_R1.passport, tb_r1.bclass, tb_r1.from
FROM tb_main Left JOIN tb_r1 ON
tb_main.Number=tb_r1.Number;
```

النوع الثالث : Right Join :

هو النوع العكسي تماماً للعملية السابقة ، وهذا مثال عليه :

```
SELECT tb_main.Fname, tb_main.lname,
Tb_R1.passport, tb_r1.bclass, tb_r1.from
FROM tb_main Right JOIN tb_r1 ON
tb_main.Number=tb_r1.Number;
```

في النهاية لا تنس ان SQL لا زالت تحتوي على الكثير من الأوامر والتي يمكنك البحث في المزيد منها على الانترنت .

بهذا نكون بحمد الله قد اتممنا الجزء الأول من الدورة ، وإلى اللقاء في الجزء القادم .

تم نشره في منتديات فيجوال بيسك : ٢٠٠٣-٢٠٠٤ .