

# Types of database

1. SQL server
2. My SQL
3. Oracle
4. Mongodb
5. Firebase
6. Sqlite

## What is data base ?

هي مكان لتخزين البيانات استطيع البحث فيها و اعدل و اضيف واحذف منها بيانات

مثال على اشكال قواعد البيانات :

(notepad-excel file-aceses file –sql server-oracle-mongodb-...etc)

## What is the different between them ?

الفروقات التي تحدد استخدام انواع من القواعد البيانات وتفضل على انواع اخرة :

1. الحجم الخاص بقاعدة البيانات
2. عدد المستخدمين بقاعدة في نفس الوقت
3. امكانية التعامل مع اموارد الجهاز
4. امكانية التوسع في قاعدة البيانات
5. مدا القدرة على العمل 24 ساعة
6. تامين البيانات
7. قوة محرك قاعدة البيانات في عملية استرجاع البيانات

## SQL Server

هناك ثلاث خدمات مهمة في sql server

1. Database engine وهو محرك قاعدة البيانات المسؤول عن ادارة القاعدة بشكل عام
2. SQL server browser هو المسؤول عن عمل connect بالاجهزة التي تريد الدخول الى قاعدة البيانات
3. SQL server agent هي الخدمة المسؤلة عن عمل اي طلب يحصل بشكل دوري

ملاحظة : توجد هذه الخدمات في قائمة services الخاصة بالوندوز

ملاحظة : في حالة وجود اكثر من نسخة SQL server تكون الخدمة SQL server browser مشتركة بين كل النسخ

توجد طريقتان لدخول الى قاعدة البيانات :

1. Windows authi. وتتم بدخول الى قاعدة البيانات باستخدام user الخاص بالوندوز دون الحاجة الى password بشرط ان يكون مسموح ل user بالدخول الى قاعدة البيانات
2. SQL authi. تتم باستخدام username , password خاص بقاعدة البيانات غير مرتبط ب users الوندوز

**الفرق بينهم** نظام ال windows authi. يصلح فقط لنظام وندوز ميعنى في حالة وجود نظام linux يلزم استخدام طريقة SQL authi –كما تستخدم طريقة windows authi. في الشبكات ال local بينما الشبكات الخارجية تستخدم SQL authi

• يوجد قواعد بيانات تأتي تلقائيا منها :

1. Master
2. Model

3. Msdb

4. Tempdb

- Master : تحفظ فيها كل معلومات المطلوبة لإدارة ال databases (اسماء ال database – بيانات ال database ومكان حفظها) يمكن عرض معلوماتها ولا يمكن التعديل عليها
- Msdb : تحتوي على كل الوظائف التي تتم بمحرك sql server ومويعيدها اذا كانت تتم بشكل دورى
- Model : هو النموذج الذى يبنى عليه جميع قواعد البيانات فيوجد فيه الصفات الاتلقائية لقاعدة البيانات
- Tempdb : يحمل الجداول والبيانات المؤقتة

## Create database

- هناك طريقتان لإنشاء قاعدة بيانات
- 1. ناتى على ملف databases فى السرفر ونضغط كليك يمين ونختار create database
  - يوجد نوعان من الملفات فى database
    - (1) Data file وتخزن فيه البيانات
    - (2) LOG file تخزن فيه كل العمليات التي تتم على ال database ويرجع اليه فى حالة اننا نريد معرفة ما تم فى ال database
      - يكون الحجم الافتراضى ل data file و log file هو 8 ميجا ويمكن تعديله
      - توجد خاصية تسمى autogrowth وهى تقوم بزيادة حجم ال data file و log file بمجرد انتهاء سعته
- 2. باستخدام امر create database database\_name

## SQL Command

- هى الاوامر المستخدمة فى لغة SQL والتي تستخدم لتحكم وإدارة قاعدة البيانات
- يوجد منها انواع مثل :
- 1 DDL (data definition language) وهى كل الاوامر التي ستنبنى قاعدة البيانات او ال object الخاص بقاعدة البيانات مثل ( – create (drop – alter
- 2 DML (data manipulation language) وهى الاوامر التي تستخدم ل التعامل ب البيانات مثل (insert-update-delete)
- 3 DCL (data control language) تستخدم لاعطاء او لالغاء صلاحيات مستخدم فى database مثل (grant-revoke)
- 4 DQL (data query language) وتستخدم لعمل الاستعلامات مثل (select)
- تتكون قاعدة البيانات فى تعرفها انها مجموعة من البيانات المترابطة وهذه البيانات تتجمع على اشكال tables واحيانا تسمى relation
- Primary key وهو column فى ال table لا تكرر العناصر الموجودة فيه
- تسمى ال column ايضا attribute

## Create and drop database

- لإنشاء قاعدة بيانات نستخدم امر create database ثم اسم ال database
- اعدادات قاعدة البيانات تكون بشكل تلقائى من النموذج الموجود فى model database
- لازالة قاعدة البيانات نستخدم امر drop database ثم اسم ال database

لاستخدام اوامر على قاعدة بيانات معينة يلزم اولاً ان نستخدم امر use ثم اسم ال database

الجدول هو البنية الاساسية فى قاعدة البيانات ويتكون من column و rows

## Create table

- لإنشاء table نستخدم امر create table ثم اسم ال table ثم ( ) وبينهم نكتب الاعمدة وانواع البيانات الموضوعه فيها مثال

Use database1

Create table table\_name (

Column1 data\_type primary key,

Column2 datatype not null,  
Column3 datatype  
)

- يمكن كتابة ميزة للعمود مثل primary key او not null بعد كتابة نوع بياناته
- يفصل بين كل column والاخر ب ,
- خاصية not null تجعل من الازم ادخال قيمة لهذه ال column

## Data type

للارقام الصحيحة <--- نستخدم numeric او int

للارقام العشرية <---- نستخدم float

الاحرف والكلمات <--- نستخدم char او text

الوقت والتاريخ <----- نستخدم data و time

الصور <----- نستخدم image

- عند تحديد نوع عمود في الجدول يوجد انواع لاتبدأ ب var وهي الانواع التي تستخدم مساحة محددة في الذاكرة حتى لو لم تستغل كلها بينما الانواع التي تبدأ ب var في حالة وجود مكان فارغ يمكن استغلاله في الذاكرة مثال :  
varchar(50) Char(50)
- اقصى طول يمكن اضافته داخل ( ) هو 8000
- لتخطي اقصى طول وهو 8000 نستخدم كلمة max وتتمكن من اخذ مايصل الى 2GB
- عند تخزين ارقام يمكن تخزينها كحروف وهذا في حالة عدم استخدام العمليات الرياضية عليها ولاكن في حالة استخدام عمليات رياضية عليها يلزم استخدامها كـ INT
- يوجد انواع من constraints او خصائص العمود منها:

1. Not null
2. Unique
3. Check
4. Foreign key
5. Primary key

- يمكن كتابة ال constrain بعد نوع العمود او بعد كتابة جميع الاعمدة حيث تكتب نوع الconstrain ثم ( ) وبينهم يوجد الاعمدة التي سيوضع عليهم الconstrain
- Primary key هو constrain يستخدم في الاعمدة لجعل كل قيمة من قيمه غير قابلة لتكرار وترمز الى سطر معين
- Primary key يكون دائما not null
- Unique وهي خاصية تلزم عدم التكرار
- Primary key خاصية تلزم عدم التكرار وعدم تركها فارغة اي انها تحمل الخاصية not null و unique معا
- عند استخدام نوع numeric يتم ادخال له قيمتان الاولى تعبر عن عدد الاعداد الصحيحة والثانية تعبر عن عدد الاعداد ما بعد العلامة العشرية ويفصل بينهم ب ,
- Check وهي constrain يلزم بتحقق شرط معين في العمود مثل وضع شرط ان تكون القيمة المضافة في صف بين قيمة والاخرة وتكتب كالتالي :  
Salary int(10) check()
- يكتب الشرط داخل ( )
- يلزم ان يكون الشرط للعمود الموضوع فيه check
- Foreign key هو عمود من جدول يضاف الى جدول اخر لعمل علاقة بين الجدولين
- غالبا ما يكون الforeign key مأخوذ من عمود به خاصية الprimary key
- Reference وهي تشير الى مكان الcolumn الاصلى في foreign key
- يلزم عند نقل عمود من جدول الى اخر باستخدام foreign key ان تكون موصفات العمود(نوعه - عدد قيمته) ثابتة في الجدولين

- يلزم ان تكون القيم الموضوعة في الجدول الثاني موجود في الجدول الاصلى
- يكتب ال foreign key بعد كتابة columns كتالى

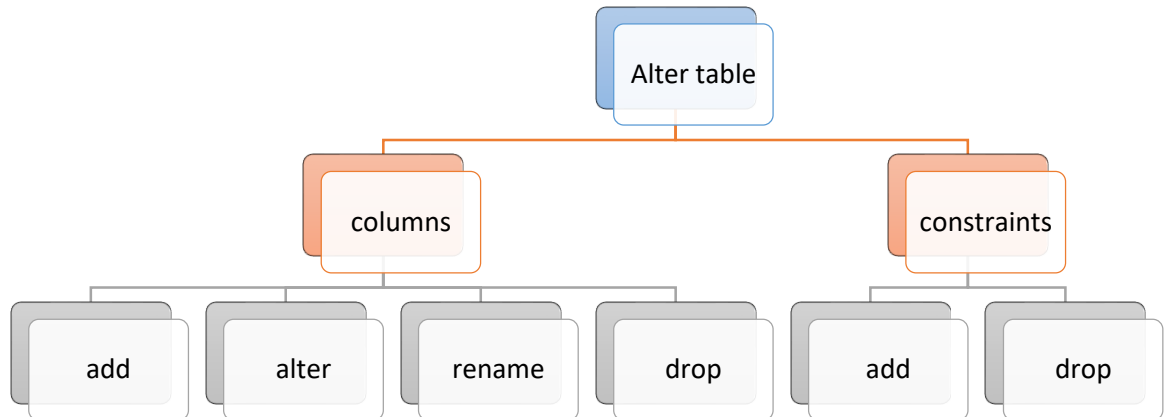
Foreign key (name of column in the current table) reference table name of main column (the name of main column)

Like:

Foreign key (staff\_id) reference store (staff\_id)

## Alter table

- تستخدم لتغيير الهيكل الخاص بالجدول وليس البيانات



1. Add : تستخدم لاضافة column كتالى :

Alter table table\_name

Add column\_name data\_type constraint

- لاضافة اكثر من column نضع , بين كل column والآخر

2. Alter column : تستخدم لتعديل على column من حيث (data type-size of data-null or not null value) كتالى :

Alter table table\_name

Alter column column\_name new\_data\_type(size)

- لتغيير نوع الcolumn وكان النوع الجديد مختلف تماما عن القديم يلزم ان يكون الcolumn فارغ

3. Drop column : يستخدم لازالة column من الجدول ويكتب كتالى:

Alter table table\_name

Drop column column\_name

4. Add constraint : يستخدم لاضافة constraint الى الcolumn كتالى :

Alter table table\_name

Add constraint constraint\_name type\_of\_constraint (column)

5. Drop constraint : لازالة constraint من column كتالى

Alter table table\_name

Drop constraint constrain\_name

## Insert into

- تستخدم لاضافة قيم الى columns في الجدول

Insert into table\_name(columns)

## Values (the\_values)

- يمكن اضافة قيم بدون كتابة الcolumns ولاكن يجب كتابة قيم الاعمدة بترتيب وفي حالة اننا نريد اجتياز قيمة نضعها null

## Update

- نستخدم لتغيير محتوى column فى جدول
- يوجد فرق بينها وبين الalter حيث انها تقوم بتعديل فى محتوى الجدول بينما الalter تقوم بتعديل فى هيكل الجدول
- تكتب كالتالى :

Update table\_name

Set column\_name=new\_value

Where condition

- عند كتابة الكود بدون وضع where سيتم تطبيق التغيير على جميع قيم الcolumn

## Delete from

- وهو امر يستخدم لازالة محتوى الجدول
- يختلف عن drop حيث انه يقوم بحذف محتوى الصفحة ولاكن drop تقوم بحذف هيكل الصفحة ويكتب كالتالى:

Delete from table\_name

- يمكن اضافة where مع شرط لتحديد المحتوى الذى سيتم حذفه

## Select

- نستخدم لاسترجاع البيانات وتكتب كالتالى :

Select columns

From table\_name

- يمكن اضافة where لتحديد شرط العرض
- يمكن استخدام and و or لوضعها بين اكثر من شرط حيث ان and ستقوم بعرض النتائج التى تتحقق فيها الشرطان فقط بينما or ستعرض النتائج المحققة لشرط الاول او لشرط الثانى او للثانين معا
- يمكن استخدام الoperator in بدل استخدام or فى حالة ان الشرط لنفس الcolumns مثال :

Where state='ny' or state='tc' or state='ax'

Where state in ('ny','tc','ax')

- لعرض النتائج باستخدام select ولاكن بدون تكرار فى القيم نستخدم كلمة distinct بعد كلمة select
- يمكن استخدام like مع where فى select للبحث باستخدام معلومة غير كاملة بحيث :
- 1. % تستخدم لاستبدال بلاشئ او بحرف واحد او اكثر كالتالى :

Where name like 'z%'

- 2. \_ تستخدم لاستبدال بحرف واحد
- يمكن وضع not like لظهار النتائج التى لن تظر عند استخدام like
- يمكن اضافة order by فى الselect لترتيب النتائج وفقا لcolumn معين كالتالى

Order by column\_name

- Asc هو نوع الترتيب التصاعدى ويكتب بعد اسم العمود
- Desc هو الترتيب التنازلى
- عند عدم كتابة نوع الترتيب يكون تصاعدى بشكل تلقائى
- يمكن استخدام اكثر من عمود فى الترتيب حيث ترتب النتائج اولا حسب العمود الاول ثم حسب العمود الثانى

- نكتب order by كآخر جزء في ال select

## Join

- تستخدم عند الاستعلام عن بيانات موجودة في أكثر من جدول
- يقوم عمل ال join على اضافة صف من جدول الى صف اخر تحت وجود شرط ويوجد انواع عديدة من ال join
- يمكن تسمية الجداول باستخدام as اثناء تنفيذ ال select
- يمكن كتابة join كالتالي :

Select first\_name , last\_name , email ,order\_id,order\_status,order\_date

From orders as o , staffs as s

Where o.staff\_id = s.staff\_id

- first\_name , last\_name , email هي اعمدة من الجدول orders و order\_id,order\_status,order\_date هي اعمدة من الجدول staffs
- شكل اخر من كتابة ال join

Select first\_name , last\_name , email ,order\_id,order\_status,order\_date

From orders as o join staffs as s

on o.staff\_id = s.staff\_id

- في حالة ان العمود الذي يربط بين الجدولين بنفس الاسم في الجدولين مثل المثال السابق يمكننا استخدام natural join حيث انها تكتب مكان ال join ولاكنها لا تحتاج الى الشرط
- natural join لاتدعم Sql server
- Outer join وهي نوع من join تقوم بعرض النتائج المترتبة على الشرط الموجود بعد on وتعرض ايضا النتائج الغير مترتبة على النتائج من جدول واحد فقط
- يلزم وضع left او right قبل ال outer join لتحديد اي من الجدولين سيعرض حتى ولم يتحقق الشرط مثال :

Select first\_name , last\_name , email ,order\_id,order\_status,order\_date

From orders as o right outer join staffs as s

on o.staff\_id = s.staff\_id

هنا سيقوم بعرض نتائج ال join التي تحقق الشرط الموجود بعد on وسيقوم بعرض النتائج الموجود فقط في الجدول الموجود على يمين ال join

- Full outer join هو join يقوم بعرض النتائج التي تحقق الشرط والنتائج التي لا تحقق الشرط في الجدول الايمن ثم التي في الجدول الايسر
- هذه الامثلة على جدولين ولاكن يمكن تطبيقها على اكثر من جدول
- توجد بعض ال functions في sql التي تقوم ببعض العمليات على ال columns كالتالي :
- 1. Avg تقوم بعمل متوسط للقيم الموجودة في عمود معين
- 2. Count تقوم بحساب العناصر الموجودة في الجدول
- 3. Max تأتي باكبر قيمة في العمود
- 4. Min تأتي باقل قيمة في العمود
- 5. Sum تعوم بعمل summation للقيم الموجودة في الجدول
- يمكن كتابتهم في select ويتم كتابتهم كالتالي :

Select avg (column\_name) from table\_name

- Group by تقوم بتقسيم المعلومات المعروضة في select الى مجموعات منفصلة حسب عمود معين
- يمكن تنفيذ order by و group by في join