

فى هذا الجزء من الدرس سوف نقوم بشرح Data Definition Language (DDL)

هو القسم المسئول عن : تعريف البيانات

- ١- **Create Database** : لإنشاء قاعدة بيانات جديد
- ٢- **Create Table** : لإنشاء جدول داخل قاعدة بيانات
- ٣- **ALTER TABLE** : للتعديل فى الجدول
- ٤- **DROP TABLE** : لحذف الجدول من قاعدة البيانات
- ٥- **CREATE INDEX** : لإنشاء مفتاح للبحث (الفهارس)
- ٦- **DROP INDEX** : لحذف مفتاح البحث

اولا انشاء قاعدة بيانات **Create Database** :

بناء الجملة كالتالى

```
CREATE DATABASE database_name
```

طبعا نحدد اسم قاعدة البيانات

قاعدة البيانات ديه هتكون فاضيه يعنى مفيش جواها اى
جداول .

لأنشاء جدول داخل قاعده البيانات **Create Table**:

يكون بناء الجمله كالتالى :

```
CREATE TABLE table_name  
(  
column_name1 data_type,  
column_name2 data_type,  
.....  
)
```

مثال على أنشاء جدول :

```
CREATE TABLE Person  
(  
LastName text(30),  
FirstName text(30),  
Address text(150),  
Age (Number)  
)
```

الان يمكننا انشاء جدوا وتحديد الحقول وانواع وحجم البيانات فيها .

الآن انشاء الفهارس CREATE INDEX :

الفهرس يصمم فى الجدول حتى يجعل عمليه الاستعلام أسرع كما يمكن انشاء اكثر من فهرس نفس الجدول المستخدم لا يرى هذه الفهارس انما هى لتسرع عمليه الاستعلام فقط .

هناك نوعين من الفهارس النوع الأول لا يمكن ان يتكرر به البيانات اما النوع\ع الثانى يمكن تكرار البيانات به .

البناء للفهرس من النوع الفريد (الذى لا يتكرر) Unique Index

```
CREATE UNIQUE INDEX index_name  
ON table_name (column_name)
```

البناء للفهرس من النوع العادى Simple Index

```
CREATE INDEX index_name  
ON table_name (column_name)
```

مثال :

```
CREATE INDEX PersonIndex  
ON Person (LastName)
```

لانشاء فهرس بترتيب عكسى :

```
CREATE INDEX PersonIndex  
ON Person (LastName DESC)
```

لانشاء فهرس فى حقلين فى نفس الجدول :

```
CREATE INDEX PersonIndex  
ON Person (LastName, FirstName)
```

الأمر Drop

Drop ترجمتها الحرفيه إلقاء ولكننا هنا نستخدمها كأمر حذف ولكن حذف ايه (قاعدة بيانات - فهرس - جدول)

اولا حذف قاعد بيانات : يكون بناء الجملة كالتالى

```
DROP DATABASE database_name
```

ثانيا حذف جدول من قاعده البيانات : يكون بناء الجملة كالتالى :

```
DROP TABLE table_name
```

ثالثا حذف فهرس : ويختلف البناء حسب نوع قاعده البيانات
مثال :

نوع قاعده البيانات : Syntax for Microsoft SQLJet (and Microsoft Access):

```
DROP INDEX index_name ON table_name
```

نوع قاعده البيانات : Syntax for MS SQL Server:

```
DROP INDEX table_name.index_name
```

نوع قاعده البيانات : Syntax for IBM DB2 and Oracle:

```
DROP INDEX index_name
```

نوع قاعده البيانات : Syntax for MySQL:

```
ALTER TABLE table_name DROP INDEX index_name
```

أخير امر حذف البيانات من داخل الجدول دون حذف الجدول :

```
TRUNCATE TABLE table_name
```

الأمر ALTER TABLE

يستخدم في التعديل على الجدول من إضافه وحذف أعمده (حقول) .

لإضافه حقل : يكون بناء الجملة كالتالى

```
ALTER TABLE table_name  
ADD column_name datatype
```

لحذف حقل : يكون بناء الجملة كالتالى :

```
ALTER TABLE table_name  
DROP COLUMN column_name
```

مثال : على الجدول التالى

Person:

LastName	FirstName	Address
Pettersen	Kari	Storgt 20

لإضافه حقل جديد :

To add a column named "City" in the "Person" table:

```
ALTER TABLE Person ADD City varchar(30)
```

النتيجه Result:

LastName	FirstName	Address	City
Pettersen	Kari	Storgt 20	

لحذف حقل من الجدول :

To drop the "Address" column in the "Person" table:

```
ALTER TABLE Person DROP COLUMN Address
```

النتيجه Result:

LastName	FirstName	City
Pettersen	Kari	

جمله SELECT INTO

تستخدم عادة في حفظ نسخة احتياطية من الجدول
بناء الجمل كالتالي :

```
SELECT column_name(s) INTO newtable [IN externaldatabase]  
FROM source
```

لعمل نسخة إحتياطية من الجدول في نفس القاعده :

```
SELECT * INTO Persons_backup  
FROM Persons
```

لنسخ الجدول لقاعده بيانات أخرى : نستخدم **IN**

```
SELECT Persons.* INTO Persons IN 'Backup.mdb'  
FROM Persons
```

يمكن أيضا نسخ حقول محدده فقط :

```
SELECT LastName,FirstName INTO Persons_backup  
FROM Persons
```

نفس المثال السابق ولكن مع إدخال شرط : **Where**

```
SELECT LastName,Firstname INTO Persons_backup  
FROM Persons  
WHERE City='Sandnes'
```

إذا اردنا نسخ الجدول لكن في وجود علاق مع جدول آخر :

```
SELECT Employees.Name,Orders.Product  
INTO Empl_Ord_backup  
FROM Employees  
INNER JOIN Orders  
ON Employees.Employee_ID=Orders.Employee_ID
```

جمله CREATE VIEW

تستخدم لإنشاء جملة تعرض بيانات تحت شرط معين
يمكن استخدام أي من جمل SQL بها مثل عمل علاقه
أو إضافه داله من الدوال .

يكون بنائها العام كالتالى :

```
CREATE VIEW view_name AS  
SELECT column_name(s)  
FROM table_name  
WHERE condition
```

ملاحظات هامه : 💡

- ١- قاعده البيانات لا تتأثر بما يتم تنفيذه من شروط او دوال
- ٢- قاعده البيانات لا تقوم بحفظ الناتج فى جدول وانما يقوم
محرك قاعده البيانات بتنفيذ الجملة كل مره تقوم
بتشغيله فيها

بعض الأمثله المبنيه على قاعده البيانات Northwind :

مثال ١ :

```
CREATE VIEW [Current Product List] AS  
SELECT ProductID,ProductName  
FROM Products  
WHERE Discontinued=No
```

كيف يتم تشغيله : عن طريق الجملة الآتيه

```
SELECT * FROM [Current Product List]
```

مثال ٢ :

```
CREATE VIEW [Products Above Average Price] AS  
SELECT ProductName,UnitPrice  
FROM Products  
WHERE UnitPrice>(SELECT AVG(UnitPrice) FROM Products)
```

يتم عرضه بالجملة:

```
SELECT * FROM [Products Above Average Price]
```

مثال ٣ :

```
CREATE VIEW [Category Sales For 1997] AS  
SELECT DISTINCT CategoryName,Sum(ProductSales) AS CategorySales  
FROM [Product Sales for 1997]  
GROUP BY CategoryName
```

يتم عرضه بالجملة :

```
SELECT * FROM [Category Sales For 1997]
```

ويمكن استخدام الشرط اثناء عرض البيانات :

```
SELECT * FROM [Category Sales For 1997]  
WHERE CategoryName='Beverages'
```

تم بحمد الله الانتهاء من الدروس
وندعوا الله ان يوفك الجميع لما فيه الخير
اسئلكم الدعاء بظهر الغائب
تحياتى للجميع أحمد حامد

GENIUS-IT@HOTMAIL.COM

الفريق العربى للبرمجه
WWW.ARABTEAM2000.COM