

مدخل للغة الإستفسارات SQL

مبادرة مهارات المستقبل

خالد البلوي



تعريفات عامة

المصطلح	الإختصار	المعنى
قواعد البيانات Database	DB	مجموعة من الجداول المترابطة
نظم إدارة قواعد البيانات Database Management	DBMS	أنظمة/برامج للتعامل مع قواعد البيانات
نظم قواعد البيانات Database System	DBS	هي DB + DBMS
مخطط العلاقات البينية Entity Relationship Diagram	ERD	مخطط لوصف علاقات الجداول في قاعدة البيانات
معرف النظام System Identifier	SID	أسم فريد لقواعد البيانات على النظام
الهيكلية (مخطط) Schema Diagram	-	عملياً هي حساب المستخدم على قواعد البيانات مع كل عناصرها من جداول ومناظير وكائنات برمجية

لغة الإستعلامات البيئوية SQL

- SQL هي إختصار ل Structured Query Language
- تُلفظ أيضاً SEQUEL كإختصار ل Structured English QUery Language
- تُعتبر مفتاح نجاح قواعد البيانات العلائقية.
- تتشابه مبادئها في أغلب بيئات قواعد البيانات على إختلافها.
- يمكن تضمينها في لغات عامة مثل C & Pascal, #C.

لغات قواعد البيانات

المصطلح	الإختصار	المعنى
Data Definition Language لغة تعريف البيانات	DDL	<ul style="list-style-type: none"> - تحتوي أوامر لبناء هيكلية ال Schema - مثل : Create, Drop, Alter, Rename - تنفذ مباشرة في قواعد البيانات
Data Manipulation Language لغة معالجة بالبيانات	DML	<ul style="list-style-type: none"> - تحتوي أوامر معالجة/تلاعب البيانات - مثل : Select, From, Where, Insert, Update, Delete - تحتاج إلى جُمْل التحكم بالتعاملات للتخزين بشكل دائم في قواعد البيانات
Data Control Language لغة التحكم بالبيانات	DCL	<ul style="list-style-type: none"> - تتعامل مع الصلاحيات - مثل : Grant, Revoke - تنفذ مباشرة في قواعد البيانات
Transaction Control Language لغة التحكم بالتعاملات	TCL	<ul style="list-style-type: none"> - تتعامل مع معاملات قواعد البيانات وتخزينها بشكل دائم - مثل : Commit, Rollback, Savepoint, Set Transaction

مخطط قواعد البيانات Schema

- يُخلط بين مفهوم المخطط Schema والحساب User. فالمخطط هو الحساب مع جميع عناصره التي يتم إنشاؤها من خلاله أو من خلال إرتباطها بعمليات حسابه.
- يمكن إنشاء المخطط من خلال شاشات Wizard أو بأوامر برمجية على مستوى قواعد البيانات (من مدير قواعد البيانات) بالجملة:

CREATE SCHEMA COMPANY AUTHORIZATION JSMITH

إنشاء الجداول Create Table

• يتطلب إنشاء الجدول على الأقل إمتلاك الحساب لصلاحيّة

TO GRANT CONNECT,RESOURCE,UNLIMITED TABLESPACE
;SCOTT IDENTIFIED BY TIGER

• جُملة الإنشاء تكون بحدها الأدنى:

CREATE TABLE MY_TABLE

(
MY_VAL DATATYPE
);

إنشاء الجداول Create Table

أنواع المتغيرات

نوع المتغير	نوع ال	ملاحظات
NUMERIC الرقمي	NUMBER, NUMBER(x), NUMBER(x,y)	يمكن تعريف المتغير كرقم صحيح بعدد خانات محدد أو كعدد كسري
STRING نصي	CHAR, CHAR(x), VARCHAR(x), VARCHAR2(x)	يمكن تعريف المتغير كحرف، أو مجموعة حروف بطول محدد، أو مجموعة حروف تتقيد بين حدود طول ما
BIT STRING النصوص الثنائية	BOOLEAN, BIT, BIT(x)	يمكن تعريف المتغير للقيم الثنائية التي تأخذ طابع (صواب/خطأ)
DATE-TIME التواريخ والوقت	DATE, TIME, DATETIME, TIMESTAMP	يمكن تحديد صيغ التواريخ أو الأوقات أو كليهما، أو حتى طابع الوقت والتاريخ بالنمط العالمي
LARGE OBJECT	LOB, BFILE	يمكن تحديدها للبيانات الضخمة كالوسائط والصور والملفات

إنشاء الجداول Create Table

قيود المتغيرات

- القيود تندرج بتصنيف لغة DDL ويتم تنفيذها دون الحاجة لل COMMIT
- تُكتب بعد تحديد نوع المتغير وقبل الفاصلة للمتغير التالي له.
- يمكن إستخدام أكثر من قيد على نفس المتغير، وتُكتب على التتابع دون الحاجة لوجود فواصل.

استخدامه	القيد
يُجبر المستخدم على إدراج قيمة في الخانة.	NOT NULL
يُسند قيمة افتراضية في حال عدم إدراج قيمة من المستخدم	DEFAULT
يرفض أي قيمة في حال كانت موجودة سلفاً في نفس الجدول	UNIQUE
يقيد القيمة بمعادلة محددة أو بمجموعة قيم	CHECK
يجعل الحقل هو المفتاح الرئيسي للجدول	PRIMARY KEY

إنشاء الجداول Create Table

تعيين المفتاح الرئيسي PK

- المفتاح الرئيسي PRIMARY KEY يمكن تحديده بأحد الطرق:
 - بعد كتابة المتغير مباشرة.

```
<< MY_VAL CHAR(9) PRIMARY KEY
```

- بعد الإنتهاء من تعريف كافة المتغيرات وقبل إغلاق القوس للجدول كما يلي

```
<< PRIMARY KEY (MY_VAL)
```

- يمكن إسناد أسم للقيود (وهو الأفضل) مباشرة، ويمكن تحديد أكثر من حقل كذلك ليكون مفتاح رئيسي

```
<< CONSTRAINT CONST_NAME PRIMARY KEY (VAR01,VAR02)
```

- خارج جملة CREATE TABLE بأمر مستقل (وهو الأفضل)، كما يلي:

```
<< ALTER TABLE MY_TABLE ADD MY_VAL PRIMARY KEY
```

إنشاء الجداول Create Table

تعيين المفتاح الارتباطي FK

- يمكن تعريف المفتاح الارتباطي بأحد الطرق:
- كتابته بعد تعريف المتغيرات في الجدول وقبل إغلاق القوس.

>> **FOREIGN KEY**(VAL02) **REFERENCES** TABLE_Y(VAL01)

- ويمكن كتابته بتعيين أسم للقيد كذلك

>>**CONSTRAINT** FK_CONST_NAME **FOREIGN KEY**(VAL02) **REFERENCES** TABLE_Y(VAL01)

- كتابته خارج جملة بناء الجدول بشكل مستقل.

إنشاء الجداول Create Table

تعيين المفتاح الإرتباطي FK

- في تعريف المفتاح الإرتباطي FOREIGN KEY يمكننا أن نحدد ماذا يحصل لهذا الحقل لو تم تغيير قيمة الحقل الذي تم الإرتباط عليه...!!
- التغييرات التي تتم في الجدول الآخر يمكن أن تكون جملة تعديل UPDATE أو جملة حذف DELETE
- يتم التعامل مع التغييرات بأحد الخيارات (SET NULL, SET DEFAULT, CASCADE)
- يمكننا تحديد إحدى الحالتين أو كليهما.

**CONSTRAINT FK_CONST_NAME FOREIGN KEY (VAL02) REFERENCES TABLE_Y (VAL01) ON DELETE<<
;SET NULL ON UPDATE CASCADE**

إنشاء الجداول Create Table

إنشاء من جدول موجود سلفاً

• يمكن إنشاء جدول جديد بقيم موجودة في جدول/جداول أخرى بالأمر:

```
>>CREATE TABLE MY_TABLE AS  
(SELECT *  
FROM YOUR_TABLE  
WHERE VAR_A >= xxx);
```

• كما يمكن إنشاء جدول له نفس هيكلية جدول آخر، ويكون فارغاً دون قيم بالأمر:

```
>>CREATE TABLE MY_TABLE AS  
(SELECT *  
FROM YOUR_TABLE  
WHERE 1 = 2);
```

التعديل على أوامر DDL

- عند التعامل مع المخطط الجداول أو ما يتم كتابته بجمل DDL سيتم استخدام كلمة ALTER للتعديل. وكلمة DROP للحذف. و ADD للإضافة.
- مثلاً

```
ALTER TABLE MY_TABLE ADD VAR03 VARCHAR(12);
```

```
ALTER TABLE MY_TABLE DROP VAR02 CASCADE;
```

```
ALTER TABLE MY_TABLE ALTER VAR01 SET DEFAULT "NEW_VAL";
```

```
ALTER TABLE MY_TABLE DROP CONSTRAINT CONST_NAME  
CASCADE;
```

جُمْل ال DML

إدراج القيم INSERT

- إدراج القيم في الغالب- يتم من خلال الجداول بشكل مباشر.
- يمكن الإستغناء عن أسماء الحقول لو ضمنت أن القيم مكتوبة بنفس الترتيب للجدول، مثلاً جدول فيه حقلين (رقم الموظف/أسمه) يمكن كتابته بأحدى الطرق:

```
INSERT INTO MY_TABLE  
VALUES (xxx,'yyy');
```

- يجب أن تنتهي الجملة بفاصلة منقوطة.
- **COMMIT;** لتخزين القيمة بشكل دائم في الذاكرة يجب أن تتم عملية
- إذا استخدمت أسماء الحقول يمكن تغيير ترتيب الإدراج

```
INSERT INTO MY_TABLE(VAR02,VAR01)  
VALUES ('yyy',xxx);
```

جُمْل ال DML

إدراج القيم INSERT

- يمكن إدراج مجموعة قيم من خلال جداول موجودة سلفاً في النظام.
- مثلاً

```
INSERT INTO MY_TABLE(VAR01,VAR04,VAR02)
SELECT TBL_X.VAR_A , TBL_Y.VAR_B, COUNT(*)
FROM TBL_X, TBL_Y
WHERE .....
GROUP BY TBL_X.VAR_A, TBL_Y.VAR_B;
```

جُمْل ال DML

حذف القيم DELETE

- بمثل باقي جُمْل DML لا يتم التخزين بشكل دائم مالم يتم عمل COMMIT بعدها.
- صيغة الجملة التقليدية:

```
>> DELETE FROM MY_TABLE  
WHERE .....
```

• لحذف كل القيم في الجدول << DELETE FROM MY_TABLE

- يمكن استخدام جُمْل استعلام فرعية معها

```
>> DELETE FROM MY_TABLE  
WHERE VAR_A IN (SELECT VAR_Z  
FROM TBL_Y);
```


جُمْل ال DML

تعديل القيم UPDATE

- بمثل باقي جُمْل DML لا يتم التخزين بشكل دائم مالم يتم عمل COMMIT بعدها.
- صيغة الجملة التقليدية:

```
>> UPDATE MY_TABLE  
SET VAR_A = xxx  
WHERE .....
```

```
>> UPDATE MY_TABLE  
SET VAR_A = xxx;
```

```
>> UPDATE MY_TABLE  
SET VAR_A = xxx  
WHERE VAR_B NOT IN (SELECT VAR_Z  
FROM TBL_Y);
```

- لتعديل كل الحقول في الجدول بنفس القيمة

- يمكن استخدام جُمْل استعمال فرعية معها

جُمْل ال DML الاستعلام SELECT

• الصياغة العامة هي:

```
>> SELECT *  
      FROM MY_TABLE  
      WHERE MY_CONDITIONS;
```

```
>> SELECT  
      FROM  
      WHERE
```

ماذا؟؟ ماذا سترى من نتائج (حقول)

من أين؟ من أين مصدر تلك البيانات (جداول أو مناظير أو جُمْل فرعية)

ماهو؟ ماهو شرط تلك القيم التي ستظهر (قيم مقارنة أو جُمْل فرعية)

جُمْل ال DML

الاستعلام SELECT

• إختيار كافة القيم يكون بإستخدام الرمز *

• لتحديد القيم بشكل خاص يتم كتابة أسماء الحقول بعد ال SELECT

```
SELECT    VAR02, VAR07, ....  
FROM    TABLE_A  
WHERE    .....
```

• لإستبعاد التكرار في النتائج نستخدم DISTINCT

```
SELECT    DISTINCT VAR02, VAR07, ....  
FROM    TABLE_A  
WHERE    .....
```

جُمْل ال DML

الاستعلام SELECT

- يمكن دمج نتائج جُمْلتي استعلام بالأمر UNION مع ملاحظة أن الدمج يتطلب بالضرورة تشابه حقول جُمْلتي الإستعلام:

```
>> (SELECT  VAR_X_INT, VAR_Y_CHAR  
FROM      TABLE_A  
WHERE     .....)  
UNION  
(SELECT  VAR_X_INT, VAR_Y_CHAR  
FROM      TABLE_A  
WHERE     .....);
```

جُمْل ال DML

الاستعلام SELECT

- يمكن دمج نتائج جُمْلتي استعلام مع استثناء التكرار بالنتائج بالأمر MINUS مع ملاحظة أن الدمج يتطلب بالضرورة تشابه قول جُمْلتي الإستعلام:

```
>> (SELECT  VAR_X_INT, VAR_Y_CHAR  
FROM      TABLE_A  
WHERE     .....)  
MINUS  
(SELECT  VAR_X_INT, VAR_Y_CHAR  
FROM      TABLE_A  
WHERE     .....);
```

شكراً



وزارة الاتصالات
وتقنية المعلومات
MINISTRY OF COMMUNICATIONS
AND INFORMATION TECHNOLOGY