

مبادئ قواعد المعطيات

المحاضرة الثامنة

د. سمر الحلبي

قواعد المعطيات关系
لغة الاستفسارات البنوية SQL
(Structured Query Language)
الجزء الرابع

6

التعامل مع المعطيات

لغة التعامل مع المعطيات

- تستخدم لغة التعامل مع المعطيات من أجل :
 - ✓ إضافة أسطر إلى جدول
 - ✓ تعديل أسطر موجودة في جدول
 - ✓ حذف أسطر من جدول

إضافة أسطر إلى جدول

DEPARTMENTS				70 Public Relations	100	1700	New row
DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID				
10	Administration	200	1700				
20	Marketing	201	1800				
50	Shipping	124	1500				
60	IT	103	1400				
80	Sales	149	2500				
90	Executive	100	1700				
110	Accounting	205	1700				
190	Contracting		1700				

Insert a new row into the DEPARMENTS table.



DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
50	Shipping	124	1500
60	IT	103	1400
80	Sales	149	2500
90	Executive	100	1700
110	Accounting	205	1700
190	Contracting		1700
70	Public Relations	100	1700

تعليمية الإضافة INSERT

لإضافة سطر واحد إلى جدول نستخدم التعليمية **INSERT** والتي لها الشكل التالي:

```
Insert into <table_name> [(col1,col2,.....)]  
Values (val1, val2 ,...)
```

تعليمات الإضافة INSERT

ملاحظات هامة:

- إذا لم نحدد قائمة الأعمدة (val1,col1,col2,...) فإن قائمة المعطيات (...,...) يجب أن يحتوي على قيمة من أجل كل عمود في الجدول وبنفس ترتيب الأعمدة ضمن الجدول.
- يجب أن يكون نمط البيانات لكل قيمة يوافق(قابل للتحويل بشكل ضمني) نمط البيانات لكل عمود مقابل في قائمة الأعمدة.
- في حال حددنا قائمة الأعمدة فإن كل عمود غير موجود ضمن قائمة الأعمدة يجب أن يسمح إما بقيمة خالية null أو أن يكون لديه قيمة افتراضية وإلا ينتج لدينا خطأ .

تعليمية الإضافة INSERT

لإضافة سطر إلى جدول الأقسام تكون تعليمية الإضافة لها الشكل التالي:

```
INSERT INTO departments(department_id, department_name,  
                         manager_id, location_id)  
VALUES      (70, 'Public Relations', 100, 1700);  
1 row created.
```

تعليمية الإضافة INSERT

لإضافة سطر إلى جدول بقيم حقول معدومة فلدينا حلين:

- 1- لا يتم ذكر أسماء الأعمدة ذات القيم المعدومة وبالتالي لا يتم ذكر قيم لها .

مثال: إضافة سطر إلى جدول الأقسام دون ذكر حقل رقم المدير ورقم مكان

القسم وبالتالي يأخذان القيم Null

```
INSERT INTO departments (department_id,  
                        department_name) |||  
VALUES (30, 'Purchasing');  
1 row created.
```

تعليمية الإضافة INSERT

لإضافة سطر إلى جدول بقيم حقول معدومة فلدينا حلين:

- ٢- لا يتم ذكر أسماء الحقول ذات القيم المعدومة وبالتالي يتم ذكر قيمة Null لكل حقل.

مثال: إضافة سطر إلى جدول الأقسام دون ذكر أسماء الحقول وذكر قيمة Null للحقل المعدوم.

```
INSERT INTO departments
VALUES          (100, 'Finance', NULL, NULL);
1 row created.
```

تعديل أسطر موجودة في جدول

EMPLOYEES

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	HIRE_DATE	JOB_ID	SALARY	DEPARTMENT_ID	COMMIS
100	Steven	King	SKING	17-JUN-87	AD_PRES	24000	90	
101	Neena	Kochhar	NKOCHHAR	21-SEP-89	AD_VP	17000	90	
102	Lex	De Haan	LDEHAAN	13-JAN-93	AD_VP	17000	90	
103	Alexander	Hunold	AHUNOLD	03-JAN-90	IT_PROG	9000	60	
104	Bruce	Ernst	BERNST	21-MAY-91	IT_PROG	6000	60	
107	Diana	Lorentz	DLORENTZ	07-FEB-99	IT_PROG	4200	60	
124	Kevin	Mourgos	KMOURGOS	16-NOV-99	ST_MAN	5800	50	

Update rows in the EMPLOYEES table.



EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	HIRE_DATE	JOB_ID	SALARY	DEPARTMENT_ID	COMMIS
100	Steven	King	SKING	17-JUN-87	AD_PRES	24000	90	
101	Neena	Kochhar	NKOCHHAR	21-SEP-89	AD_VP	17000	90	
102	Lex	De Haan	LDEHAAN	13-JAN-93	AD_VP	17000	90	
103	Alexander	Hunold	AHUNOLD	03-JAN-90	IT_PROG	9000	30	
104	Bruce	Ernst	BERNST	21-MAY-91	IT_PROG	6000	30	
107	Diana	Lorentz	DLORENTZ	07-FEB-99	IT_PROG	4200	30	
124	Kevin	Mourgos	KMOURGOS	16-NOV-99	ST_MAN	5800	50	

قواعد تعليمية التعديل update

لتعديل أسطر موجودة في جدول نستخدم الـ **update**
 والتي لها الشكل التالي:

```
UPDATE      table
SET         column = value [, column = value, ...]
[WHERE      condition];
```

- يتم تعديل كل الأسطر التي تحقق الشرط

تعديل أسطر من جدول

- يجري تعديل أسطر محددة إذا استخدمنا عبارة `:where`

```
UPDATE employees
SET department_id = 70
WHERE employee_id = 113;
1 row updated.
```

- يجري تعديل كل أسطر الجدول إذا لم نستخدم `:where`

```
UPDATE copy_emp
SET department_id = 110;
22 rows updated.
```

حذف أسطر من جدول

DEPARTMENTS

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
70	Public Relations	100	1700
30	Purchasing		
50	Shipping	124	1500
60	IT	103	1400
100	Finance		
80	Sales	149	2500

Delete a row from the DEPARTMENTS table.

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
70	Public Relations	100	1700
30	Purchasing		
50	Shipping	124	1500
60	IT	103	1400
80	Sales	149	2500

قواعد تعليمية الحذف Delete

لحذف أسطر موجودة في جدول نستخدم التعليمية Delete والتي لها الشكل التالي:

```
DELETE [FROM]    table
[WHERE          condition] ;
```

- يتم حذف كل الأسطر التي تحقق الشرط

حذف أسطر من جدول

- يجري حذف أسطر محددة إذا استخدمنا عبارة `:where`

```
DELETE FROM departments
WHERE department_name = 'Finance';
1 row deleted.
```

- يجري حذف كل أسطر الجدول إذا لم نستخدم `:where`

```
DELETE FROM copy_emp;
22 rows deleted.
```

تعديل الأسطر : المحافظة على شروط التكامل

```
UPDATE employees  
SET department_id = 55  
WHERE department_id = 110;
```

```
UPDATE employees  
*  
ERROR at line 1:  
ORA-02291: integrity constraint (HR.EMP_DEPT_FK)  
violated - parent key not found
```

Department number 55 does not exist

حذف الأسطر : المحافظة على شروط التكامل

```
DELETE FROM departments  
WHERE department_id = 60;
```

```
DELETE FROM departments *  
ERROR at line 1:  
ORA-02292: integrity constraint (HR.EMP_DEPT_FK)  
violated - child record found
```

You cannot delete a row
that contains a primary key
that is used as a foreign key
in another table.

7

إنشاء الجداول

تعليمية إنشاء جدول

- لإنشاء جدول يجب تحديد اسم الجدول واسم كل عمود فيه مع نمط معطياته .
- الشكل العام لتعليمية إنشاء جدول:

```
CREATE TABLE [schema.]table  
    (column datatype [DEFAULT expr] [, . . .]);
```

تعليمية إنشاء جدول

- لإنشاء جدول الأقسام ذو الحقول رقم القسم ، اسم القسم واسم المكان .
بالتالي شكل تعليمية إنشاء جدول الأقسام لها الشكل التالي:

```
CREATE TABLE dept  
  (deptno NUMBER(2) ,  
   dname  VARCHAR2(14) ,  
   loc    VARCHAR2(13)) ;
```

Table created.

أنماط المعطيات

١. الأعداد الصحيحة :

Name	Size	Domain
Int	4 B	- 2^{31} to $2^{31}-1$
bigint	8 B	- 2^{63} to $2^{63}-1$
smallint	2 B	- 2^{15} to $2^{15}-1$
tinyint	1 B	0 to 255

٢. الأعداد الحقيقة :

Name	Size	Domain
float	8 B	- 1.79E+308 to -2.23E-308
real	4 B	-3.40E + 38 to -1.18E - 38

أنماط المعطيات

٣. السلاسل الحرفية (Character Strings)

Name	Size	Domain	Description
char[(n)]	n B	1 to 8000 char	معطيات حرفية بطول ثابت
varchar[(n max)]		1 to 8000 char	معطيات حرفية بطول متغير الحجم الأعظمي للتخزين وهو $2^{31}-1$ Max
text			معطيات حرفية بطول متغير

٤. السلاسل الحرفية من النوع Unicode (Unicode Character Strings)

Name	Size	Domain	Description
nchar[(n)]	2n B	1 to 4000 char	معطيات حرفية بطول ثابت ولكنها unicode
nvarchar[(n)]		1 to 4000 char	معطيات حرفية بطول متغير و unicode
ntext			معطيات حرفية بطول متغير و unicode

أنماط المعطيات

٥. الوقت والتاريخ : (Date and Time)

Name	Size	Domain	Description
time	5B	hh:mm:ss[.nnnnnnnn] 00:00:00:0000000 to 23:59:59:9999999	القيمة الافتراضية 00:00:00
date	3B	yyyy-mm-dd 0001-01-01 to 9999-12-31	القيمة الافتراضية 1900-01-01
smalldatetime		yyyy-mm-dd hh:mm:ss	

تعديل جدول

تتيح لك SQL عملية تعديل الجدول بعد إنشائه في الحالات التالية :

- ١ إضافة حقل جديد .
- ٢ تعديل مواصفات حقل .
- ٣ حذف المفتاح الأساسي من الجدول .

جملة تعديل الحقول : ALTER Command

تستخدم هذه الجملة لإضافة حقول أو لتعديل حقول في جدول أو حذفها ويشمل هذا التعديل اسم الحقل ونوعه الحقل وحجم بيانته أو إلهاق عبارة فارغ أو غير فارغ إلى الحقل .

تعدیل جدول

إضافة حقل جديد : New Field

الشكل العام :

```
ALERT TABLE tablename  
ADD (field Type (size) [NOT NULL] | [NULL])
```

حيث أن

اسم الجدول المراد تعديله tablename

اسم الحقل field

نوع البيانات Type

مثال: إضافة حقل العمل إلى جدول القسم

```
SQL> ALTER TABLE dept30  
  2 ADD          (job VARCHAR2(9));  
Table altered.
```

تعديل جدول

تعديل مواصفات حقل : **Modify**

يمكن عن طريق جملة ALTER تعديل مواصفات حقل
الشكل العام :

```
ALTER TABLE tablename  
MODIFY Type (size) [NOT NULL] | [NULL] )
```

مثال: تعديل حجم نمط الحقل اسم الموظف

```
SQL> ALTER TABLE dept30  
      2 MODIFY ename VARCHAR2(15));  
Table altered.
```

تعديل جدول

حذف الحقل :**DROP**

يمكن حذف الحقل سواءً أكان به بيانات أم لم يكن، كذلك يجب أن يبقى حقل واحد في الجدول بعد تعديله ، وفي كل مرة نستخدم فيها ALTER يمكن حذف حقل واحد فقط ، والحقل المحذوف لا يمكن استرجاعه .

مثال: حذف حقل العمل من جدول الاقسام

```
SQL> ALTER TABLE      dept30
      2 DROP COLUMN    job ;
Table altered.
```

8

تعريف شروط التكامل

القيود

يتم عن طريق القيود فرض بعض القواعد على الحقول ، ويمكن أن ننشئ القيد أثناء إنشاء الجدول (ننشئ بعض القيود على مستوى الحقل و بعضها على مستوى الجدول)

أنواع القيود:

- NOT NULL (غير خالي)
- UNIQUE (متفرد(فريد) مع منع التكرار)
- PRIMARY KEY (حقل مفتاح اساسي)
- FOREIGN KEY (حقل مفتاح غريب)

القيود

الشكل العام لتعريف قيد :

```
column [CONSTRAINT constraint_name]  
constraint_type,
```

حيث أن :

اسم التقييد constraint_name

نوع التقييد constraint_type

القيود

1 - غير فارغ : **NOT NULL**

هذا الأمر لا يسمح بترك حقل فارغ.

مثال :

```
CREATE TABLE friend...
    phone  VARCHAR2(15) NOT NULL, ...
    last_name  VARCHAR2(25)
    CONSTRAINT friend_last_name_nn NOT NULL, ...
```

القيود

: **UNIQUE** ٢ - فريد :

هذا الأمر يجعل الحقل (العمود) فريداً فلا يسمح بتكرار القيم ضمن الحقل .

مثال :

```
... phone      VARCHAR2(10)
      CONSTRAINT s_emp_phone_uk UNIQUE, ...
```

القيود

٣ - مفتاح أساسى : Primary Key

لا يمكن لأى جدول أن يحتوى إلا على مفتاح أساسى واحد ، وإذا قمنا بتحويل حقل إلى مفتاح أساسى فهذا الحقل يعتبر حقل فريد ، ولكن لا يمكن أن يحتوى على قيم فارغة .

مثال :

```
... id NUMBER(7)
      CONSTRAINT s_emp_id_pk PRIMARY KEY, ...;
```

القيود

٤ - مفتاح خارجي : **Foreign Key**

انظر المثال التالي :

```
... dept_id NUMBER(7)
      CONSTRAINT s_emp_dept_id_fk
      FOREIGN KEY (department_id)
      REFERENCES s_dept(id)
      ON DELETE CASCADE;
```

: **Foreign Key - ١**

نكتب بجوارها اسم الحقل المراد أن يكون مفتاح خارجي في الجدول الابن

: **REFERENCES - ٢**

نكتب بجوارها الحقل الذي سيتم الربط به من الجدول (الأب) الآخر.

: **ON DELETE CASCADE - ٣**

إذا تم مسح أو حذف أحد الصفوف في الجدول الأب سيسمح الصف المقابل له في الجدول الابن (إذا تم حذف القسم في جدول معلومات الأقسام سيتم حذف جميع الموظفين التابعين لهذا القسم في جدول الموظفين).