

القيود

Constraints

نستخدم القيود على الجداول لتنظيم العمليات المختلفة التي تتم على الجداول من حذف وتعديل وإضافة .
ومن أنواع هذه القيود :

- ١- **NOT NULL** : وهذا القيد يمنع أن نترك عمود خالي بلا بيانات ويطبق على مستوى العمود فقط
- ٢- **UNIQUE** : يمنع تكرار البيانات داخل العمود ويطبق على مستوى العمود أو الجدول .
- ٣- **PRIMARY KEY** : ويستخدم هذا القيد لعمل مفتاح أساسي داخل الجدول ، أي لا يتكرر فيه القيم ، ولا نترك فيه قيمة فارغة أي يضم خصائص القيد السابقين ويطبق على مستوى العمود أو الجدول .
- ٤- **FOREIGN KEY** : وهو يستخدم لعمل ربط بين جدولين ويطبق على مستوى العمود أو الجدول.
- ٥- **CHECK** : يستخدم هذا القيم اختبار قيمة عمود حيث لا يقبل هذا العمود إلا شرط معين ويطبق على مستوى العمود أو الجدول .

وهناك طريقتين نستخدمهما في إنشاء القيود :

- ١- عند إنشاء الجدول . (على مستوى العمود - على مستوى الجدول) .
- ٢- بعد إنشاء الجدول .

أولا : Primary key :

وهو المفتاح الأساسي داخل الجدول ويتم إنشاء هذا القيد على مستوى العمود أو على مستوى الجدول ، ويتميز هذا القيد بأنه يضم خاصيتين كما ذكرنا :
الأولى : عدم السماح بترك قيمة خالية في العمود .
الثانية : عدم السماح بتكرار القيم داخل العمود .

وستعرض لطرق إنشاء القيود من خلال الأمثلة الآتية :

أولا : إنشاء القيد عند إنشاء الجدول :

- ١- تطبيقها على مستوى الأعمدة <<<<

مثال :

إنشاء جدول خاص بالمستخدمين وإنشاء القيود على مستوى الحقل .

Create table users

```
(
    User_no    number          primary key ,
    User_name  varchar2(20)    not null ,
    Password   varchar2(10)
);
```

قمنا في المثال السابق بإنشاء جدول للمستخدمين يتكون من رقم المستخدم واسمه ورقمه السري ، وقمنا بإنشاء قيدين على عمود رقم المستخدم وعلى العمود الآخر اسم المستخدم ، الأول مفتاح أساسي والثاني نشترط دخول اسم المستخدم وبالنسبة للقيد Not Null فيطبق فقط على مستوى العمود ولا يمكن تطبيقه على مستوى الجدول .

- ٢- تطبيقها على مستوى الجدول <<<<

سنقوم بعملية إنشاء نفس الجدول في المثال السابق ولكن القيود ستكون على مستوى الجدول .

Create table users

```
(
    User_no    number ,
    User_name  varchar2(20)    Not Null ,
    Password   varchar2(10)
    Constraint user_user_no_pk primary key (user_no )
);
```

في المثال السابق قمنا بعمل قيد Primary Key على مستوى الجدول وليس العمود ولقد قمنا بتسمية القيد ب user_user_no_pk ، وبالنسبة لقيد Not Null فكما ذكرنا لا يمكن عمل قيد له على مستوى الجدول فهو على مستوى العمود فقط كما كان في المثال السابق معنا .

إذا ما الفرق بين إنشاء القيد على مستوى العمود ومستوى الجدول ؟

أنه يمكننا حذف القيد على مستوى الجدول لأننا قمنا بعمل اسم له فيسهل علينا حذفه على عكس لو كان القيد على مستوى العمود .

ثانيا : Foreign key :

ويتم أيضا إنشائه على مستوى العمود أو على مستوى الجدول ، ومهمته هو الربط بين جدولين ويتم الربط على مفتاح أساسي في الجدول الآخر كما كان موجود في جدول Dept وجدل Emp كما بالشكل التالي :

EMP table				DEPT table		
EMPNO	ENAME	JOB	DEPTNO	DEPTNO	DNAME	LOC
7839	KING	PRESIDENT	10	10	ACCOUNT	NEW YORK
7698	BLAKE	MANAGER	30	20	RESEARCH	DALLAS
7782	CLARK	MANAGER	10	30	SALES	CHICAGO
7566	JONES	MANAGER	20	40	OPRATIONS	BOSTON

↑ Foreign Key ↑ Primary Key

مثال :

قم بإنشاء جدول تأجير السيارات به عمود أجنبي ويكون القيد على مستوى العمود .

Create table Rent_car

```
(
    Car_no          number    REFERNCES CAR(CAR_NO) ,
    RENTER_NO       NUMBER    REFERNCES RENTER(RENTER_NO)
);
```

في المثال السابق قمنا بإنشاء جدول خاص بإيجار السيارات وقمنا بعمل قيود على مستوى الأعمدة بالمفتاح الأجنبي ، فكل مفتاح أجنبي قام بربط العمود بجدول الآخر فنجد أن :
الحقل car_no : مربوط على حقل اسمه car_no في جدول السيارات car .
والحقل Renter_no مربوط على حقل اسمه Renter_no في جدول المستأجرين Renter .
يمكننا تنفيذ المثال السابق ولكن على مستوى الجدول كالتالي :

Create table Rent_car

```
(
    Car_no          number    ,
    RENTER_NO       NUMBER
Constraint renter_car_car_no_fk foreign key(car_no)
References car(car_no)
Constraint renter_car_renter_no_fk foreign key(renter_no)
References car(car_no)
);
```

ثالثا : Check :

يتم استخدام هذا القيد على مستوى العمود أو الجدول ، ونستخدمه عندما نريد تحديد مجال قيمة معينة للعمود في الجدول ، فلو أردنا أن نحدد القيمة المدخلة للعمود رقم المستأجر داخل المستأجرين بحيث تكون هذه القيمة محصورة بين (10,99) فإننا نستخدم القيد check في تطبيق هذا القيد .
مثال :

قم بإنشاء جدول الضمانات وقم بعمل قيد على حقل رقم الضمانة بحيث ينحصر الرقم ما بين (10,99)

Create table GUARANTEES

```
(
    Guaran_no          number    check (10,99),
    Guaran_comp_name    varchar2(30) ,
    Guaran_date         date    ,
    Guaran_record_no    number
);
```

في المثال السابق قمنا بعمل قيد check على مستوى عمود Guaran_no رقم الضمانة ويشترط الرقم أن يكون محصور بين الرقمين (10,99) من خلال القيد check .
يمكننا تطبيق المثال التالي على مستوى الجدول .

Create table GUARANTEES

```
(
    Guaran_no          number    ,
    Guaran_comp_name    varchar2(30) ,
    Guaran_date         date    ,
    Guaran_record_no    number
Constraint GUARANTEES_Guaran_no_ck check (Guaran_no between 10 and 90 )
```

رابعا : unique :

يستخدم هذا القيد في عدم السماح بتكرار القيم داخل العمود ويمكن أن يكون على مستوى العمود أو الجدول .
لو قمنا في المثال السابق بإضافة قيد unique الى عمود رقم السجل التجاري Guaran_record_no سيكون كالتالي :

Create table GUARANTEES

```
(
    Guaran_no          number    ,
    Guaran_comp_name    varchar2(30) ,
    Guaran_date         date    ,
    Guaran_record_no    number    unique
);
```

يمكننا تطبيق المثال السابق على مستوى الجدول كالتالي :

Create table GUARANTEES

```
(
    Guaran_no          number
    ,
    Guaran_comp_name   varchar2(30)
    ,
    Guaran_date        date
    ,
    Guaran_record_no   number
    Constraint GUARANTEES_Guaran_record_no_uk unique(Guaran_record_no)
);
```

ثانيا : إنشاء القيد بعد إنشاء الجدول :

يمكننا إضافة القيود بعد إنشاء الجداول باستخدام الأمر alter table بالصيغة التالية :

```
SQL > ALTER TABLE اسم الجدول
        ADD CONSTRAINT اسم القيد نوع القيد ;
```

أمثلة على إضافة القيود بعد إنشاء الجدول :

مثال ١ :

إضافة القيد primary key على العمود رقم الضمانة في جدول الضمانات .

Alter table GUARANTEES

Add constraint GUARANTEES_Guaran_no_pk primary key (Guaran_no) ;

مثال ٢ :

إضافة القيد foreign key على العمود رقم السيارة car_no في جدول الحجوزات RESERVATIONS

Alter table RESERVATIONS

Add constraint RESERVATIONS_car_no_fk foreign key (car_no)

References car(car_no) ;

- إزالة القيود من الجداول :

يمكننا إزالة القيد من الجدول أيضا عن طريق الأمر Alter Table كما بالشكل التالي :

```
SQL > ALTER TABLE emp
        DROP CONSTRAINT emp_mgr_fk ;
```

Table altered .

وكما ذكرنا سابقا لا يمكن إزالة القيود التي على مستوى العمود لكن يمكننا إزالة القيود التي أنشئناها على مستوى الجدول لان لها أسماء يمكننا استدعائها وعمل إزالة لها كما في المثال التالي :

Alter table RESERVATIONS

Drop constraint RESERVATIONS_car_no_fk ;

في المثال السابق قمنا بعمل حذف للقيد RESERVATIONS_car_no_fk من جدول الحجوزات .

ولكي نزيل أي علاقة كانت بين الجداول إذا كان هناك مفتاح مربوط على الآخر يمكننا استخدام أمر cascade على المفتاح الاساسي كما في المثال التالي :

Alter table renter

Drop primary key cascade ;

- يمكننا استعراض القيود التي أنشئناها على الجداول كالتالي :

```
SELECT constraint_name , constraint_type
FROM user_constraints
WHERE table_name = 'RENTER';
```

CONSTRAINT_NAME	CON
RENTED_NUM_PK	P
NATION_NUM_FK	R

قمنا في المثال السابق بعرض أسماء القيود وأنواعها من جدول المستأجرين RENTER والحروف التالية تدل على أنواع القيود المختلفة :

- الحرف P : يعني أن نوع القيد على هذا الحقل Primary Key .
- الحرف R : يعني أن نوع القيد على هذا الحقل Foreign key .
- الحرف C : يعني أن نوع القيد هو Check .
- الحرف U : يعني أن هذا القيد هو Unique .
- والقيد Not Null فيظهر كالقيد Check .

- ويمكننا أيضا استعراض القيود التي أنشئناها على مستوى الأعمدة كالتالي :

```
SELECT constraint_name column_name
FROM user_cons_column
WHERE table_name = 'USERS' ;
```

والله ولي التوفيق