

## PERTEMUAN 11

### CONSTRAINT

#### Tujuan Pembelajaran :

- Memahami definisi Constraint
- Dapat Membuat Constraint
- Dapat Melakukan pemeliharaan Constraint

#### TEORI DAN PERCOBAAN

##### 11.1. Definisi Constraint

**Constraint** adalah batasan atau aturan yang ada pada table. Constraint mencegah penghapusan data dari suatu table yang mempunyai keterkaitan dengan table yang lain. Misal terdapat keterkaitan antara table department dengan table pegawai. Dimana pada table pegawai menyimpan informasi kolom nomer department yang juga terdapat pada table department. Jika baris dengan nomer department '10' akan dihapus dari table department, sedangkan terdapat data pegawai yang bekerja pada department tersebut, maka dengan adanya *constraint* antara dua table, penghapusan tersebut tidak bisa dilakukan.

Oracle menyediakan beberapa tipe constraint berikut :

- NOT NULL
- UNIQUE
- PRIMARY KEY
- FOREIGN KEY
- CHECK

##### 11.2. NOT NULL Constraint

Suatu kolom yang didefinisikan dengan constraint NOT NULL **tidak boleh** berisi nilai NULL.

Kolom yang berfungsi sebagai kunci primer (**primary key**) **otomatis tidak boleh NULL**.

Percobaan 1 : Berikut akan dicontohkan pembuatan table dengan constraint NOT NULL.

```
SQL> CREATE TABLE employee (  
2 empno NUMBER(4),  
3 ename VARCHAR2(10) NOT NULL,  
4 job VARCHAR2(9),  
5 deptno NUMBER(7,2) NOT NULL);
```

Table created.

### 11.3. UNIQUE Constraint

Constraint **UNIQUE** mendefinisikan suatu kolom menjadi bersifat **unik**.

Percobaan 2 : Berikut akan dicontohkan pembuatan table dengan constraint UNIQUE

```
SQL> CREATE TABLE department(  
2 deptno NUMBER(2),  
3 dname VARCHAR2(14),  
4 loc VARCHAR2(13),  
5 CONSTRAINT dept_name_uk UNIQUE(dname));
```

Table created.

### 11.4. PRIMARY KEY Constraint

Constraint **PRIMARY KEY** membentuk **key** yang **unik** untuk suatu table. Kolom yang didefinisikan sebagai PRIMARY KEY akan mengidentifikasi suatu baris data menjadi unik.

Percobaan 3 : Berikut akan dicontohkan pembuatan table dengan constraint PRIMARY KEY

```
SQL> CREATE TABLE department(  
2 deptno NUMBER(2),  
3 dname VARCHAR2(14),  
4 loc VARCHAR2(13),  
5 CONSTRAINT dept_dname_uk UNIQUE(dname),  
6 CONSTRAINT dept_deptno_pk PRIMARY KEY(deptno));
```

Table created.

### 11.5. FOREIGN KEY Constraint

FOREIGN KEY constraint didefinisikan pada suatu kolom yang ada pada suatu table, dimana kolom tersebut juga dimiliki oleh table yang lain sebagai suatu PRIMARY KEY.

Percobaan 4 : Berikut akan dicontohkan pembuatan table dengan constraint FOREIGN KEY

```
SQL> CREATE TABLE employee (  
2 empno NUMBER(4),  
3 ename VARCHAR2(10) NOT NULL,  
4 job VARCHAR2(9),  
5 deptno NUMBER(7,2) NOT NULL,  
6 CONSTRAINT emp_deptno_fk FOREIGN KEY(deptno)  
7 REFERENCES department(deptno));
```

Table created.

### 11.6. CHECK Constraint

Constraint CHECK digunakan untuk mendefinisikan suatu kondisi yang harus dipenuhi oleh tiap baris data dalam table.

Percobaan 5 : Constraint CHECK berikut ini akan ditambahkan pada table yang telah dibuat pada percobaan 4. Kondisi yang harus dipenuhi pada CHECK adalah bahwa nilai dari kolom deptno berada antara 10 dan 99.

```
SQL> ALTER TABLE employee  
2 ADD CONSTRAINT emp_deptno_ck  
3 CHECK (deptno BETWEEN 10 AND 99);
```

Table altered.

### 11.7. Menambahkan suatu Constraint

Untuk menambahkan suatu Constraint, digunakan perintah ALTER TABLE.

Cara penulisan :

```
ALTER TABLE table  
ADD [CONSTRAINT constraint] type (column);
```

Percobaan 6 : Akan ditambahkan kolom baru *mgr* *NUMBER(4)*, kemudian tambahkan constraint PRIMARY KEY pada kolom *empno*, dan terakhir tambahkan constraint FOREIGN KEY pada kolom *mgr*.

```
SQL> ALTER TABLE employee
  2  ADD (mgr NUMBER(4));

Table altered.

SQL> ALTER TABLE employee
  2  ADD CONSTRAINT empno_pk PRIMARY KEY(empno);

Table altered.

SQL> ALTER TABLE employee
  2  ADD CONSTRAINT emp_mgr_fk
  3      FOREIGN KEY(mgr) REFERENCES employee(empno);

Table altered.
```

### 11.8. Menghapus Constraint

Untuk menghapus suatu Constraint, juga digunakan perintah ALTER TABLE.

Cara penulisan :

```
ALTER TABLE      table
DROP [CONSTRAINT constraint] type (column);
```

Percobaan 7 : Hapus constraint manager (percobaan 6) dari table employee

```
SQL> ALTER TABLE employee
  2  DROP CONSTRAINT emp_mgr_fk;

Table altered.
```

Untuk menghapus kolom yang merupakan PRIMARY KEY yang ada pada table parent yang berhubungan dengan suatu kolom pada table child -> kolom ini merupakan FOREIGN KEY dalam table parent, digunakan perintah CASCADE.

Percobaan 8 : Hapus PRIMARY KEY deptno yang ada pada table dept.

```
SQL> ALTER TABLE dept
  2  DROP PRIMARY KEY CASCADE;

Table altered.
```

### 11.9. Menonaktifkan (disabling) Constraint

Untuk menonaktifkan constraint digunakan klausa DISABLE.

Percobaan 9 : Nonaktifkan constraint *empno\_pk* yang telah dibuat pada percobaan 6

```
SQL> ALTER TABLE employee
  2  DISABLE CONSTRAINT empno_pk CASCADE;

Table altered.
```

### 11.10. Mengaktifkan (enabling) Constraint

Untuk mengaktifkan kembali constraint, digunakan klausa ENABLE

Percobaan 10 : Aktifkan kembali constraint *empno\_pk* yang telah dinonaktifkan pada percobaan 9

```
SQL> ALTER TABLE employee
  2  ENABLE CONSTRAINT empno_pk;

Table altered.
```

### 11.11. Menampilkan Constraint

Tabel USER\_CONSTRAINT menyimpan informasi tentang constraint.

Percobaan 11 : Tampilkan nama constraint, tipe constraint dan kondisi pencarian pada constraint.

```
SQL> COLUMN SEARCH_CONDITION FORMAT A20
SQL> SELECT constraint_name, constraint_type, search_condition
  2  FROM user_constraints
  3  where table_name='EMPLOYEE';
```

CONSTRAINT_NAME	C SEARCH_CONDITION
SYS_C0011512	C "ENAME" IS NOT NULL
SYS_C0011513	C "DEPTNO" IS NOT NULL
EMP_DEPTNO_FK	R
EMPNO_PK	P

Kolom tipe constraint (*constraint\_type*) dapat berisi : C untuk constraint CHECK, R untuk referential integrity atau FOREIGN KEY, P untuk PRIMARY KEY, U untuk constraint UNIQUE.

## LATIHAN SOAL

1. BUAT TABEL **DEPARTEMEN** DENGAN STRUKTUR SBB :

Nama kolom	Type	Keterangan
NODEPT	NUMBER(2)	Nomer departemen
NAMADEPT	VARCHAR2(14)	Nama departemen
LOKASI	VARCHAR2(13)	Lokasi departemen

Dengan constraint yang dibuat bersamaan dengan pembuatan tabel, sebagai berikut :

Nama constraint	Type Constraint
NODEPT_PK	PRIMARY KEY

2. BUAT TABEL **PEGAWAI** DENGAN STRUKTUR

Name	Type	Keterangan
NOPEG	NUMBER(4)	Nomer pegawai
NAMA	VARCHAR2(10)	Nama pegawai
GAJI	NUMBER(7,2)	Gaji pegawai
DEPTNO	NUMBER(2)	Nomer departemen tempat pegawai bekerja

Dengan constraint yang dibuat bersamaan dengan pembuatan tabel, sebagai berikut :

Nama constraint	Type constraint
Dibuat oleh sistem	PRIMARY KEY
GAJI_CK	CHECK (kondisi : GAJI > 0)

3. BUAT CONSTRAINT BARU PADA TABEL **DEPARTEMEN** :

Nama constraint	Type Constraint
NAMADEPT_UQ	UNIQUE

4. NONAKTIFKAN CONSTRAINT GAJI\_CK PADA TABEL **PEGAWAI** DENGAN PERINTAH DISABLE

5. TAMBAHKAN BARIS BARU KE DALAM TABEL **PEGAWAI** :

Nama kolom	Nilai
NOPEG	99
NAMA	Saya
GAJI	0
DEPTNO	10

Kalau baris baru tidak berhasil dibuat, apa yang sebelumnya harus ada pada tabel **DEPARTEMEN** ?

6. AKTIFKAN KEMBALI CONSTRAINT GAJI\_CK PADA TABEL **PEGAWAI** DENGAN PERINTAH ENABLE.

Mengapa perintah tersebut tidak berhasil dilakukan ?

Apa yang harus dihapus dari tabel **PEGAWAI**, sebelum constraint gaji\_ck bisa diaktifkan kembali ?