Modul Sistem Basis data Praktikum

Trigger

MODUL TRIGGER

- Apa sih yang dimaksud dengan Trigger?

  Trigger merupakan suatu block PL/SQL yang akan tersimpan didalam database. Trigger akan teraktivasi secara otomatis ketika terjadi suatu kejadian/event (INSERT, DELETE, UPDATE) terhadap tabel, view dan database. Biasanya Trigger digunakan untuk mengisi ataupun mengubah nilai kolom dala suatu tabel sehingga validasi nilai dari tabel tesebut akan terjaga.

- Jenis - Jenis Trigger.

  Trigger dapat dibagi menjadi 2 jenis, yaitu :

  1. Application Trigger

     Trigger yang akan aktif ketika terjadi suatu kejadian pada sebuah aplikasi tertentu.

  2. Database Trigger

     Trigger yang akan aktif ketika terjadi suatu operasi DML (INSERT, UPDATE, DELETE) pada tabel.

- Manfaat Trigger

  Dengan menggunakan Tigger integritas data serta konsistesi data dapat terjaga dan juga dapat mencegah terjadinya transaksi yang tidak valid. Selain itu standarisasi (penyeragaman) terhadap proses juga dapat tercapai karena Trigger dibuat satu kali dan tersimpan dalam database, sehingga setiap operasi yang dilakukan oleh siapapun terhadap sistem tersebut akan menggunakan Trigger yang sama.

- Batasan Trigger

  Trigger memiliki batasan - batasan tertentu seperti :

  1. Trigger tidak dapat memproses perintah commit dan rollback, selain itu juga tidak dapat memanggil prosedur, fungsi ataupun package yang menggunakan perintah tersebut.

2. Trigger tidak dapat digunakan pada kolom dalam suatu tabel yang terdapat constraint didalamnya.

- Struktur Dasar Trigger

Untuk membuat Trigger pada Oracle telah disediakan statement CREATE TRIGGER yang selanjutnya dapat digunakan untuk kejadian tertentu. Secara umum terdapat 2 waktu agar sebuah Trigger dapat teraktivasi, yaitu :

1. BEFORE

Waktu dimana sebuah Trigger akan aktif sebelum terjadinya suatu proses DML pada suatu tabel. Dapat dibagi menjadi :

a. BEFORE INSERT :Trigger akan aktif sebelum kejadian insert pada suatu tabel.

b. BEFORE UPDATE :Trigger akan aktif sebelum kejadian update pada suatu tabel,

c. BEFORE DELETE :Trigger akan aktif sebelum kejadian delete pada suatu tabel.

2. AFTER

Waktu dimana sebuah Trigger akan aktif sesudah proses DML pada suatu tabel dilakukan. Dapat dibagi menjadi :

d. AFTER INSERT : Trigger akan aktif setelah kejadian insert pada suatu tabel.

e. AFTER UPDATE : Trigger akan aktif setelah kejadian update pada suatu tabel,

f. AFTER DELETE : Trigger akan aktif setelah kejadian delete pada suatu tabel.

- Struktur Penulisan Trigger

SINTAKS

```
CREATE OR REPLACE TRIGGER <trigger_name>
    <trigger_timing> [BEFORE|AFTER]
    <trigger_event> [INSERT|UPDATE|DELETE|...] ON <table_name>
    <trigger_type> [FOR EACH ROW]
    <trigger_restriction> (optional)
DECLARE
    <variable_declaration>
BEGIN
    <trigger_body>
END;
/
```

Penjelasan :

Bagian Trigger	Keterangan	Nilai
Trigger Timing	Menunjukkan kapan Trigger akan dieksekusi terhadap suatu event/kejadian.	BEFORE, AFTER
Trigger Event	Jenis manipulasi data pada tabel/ view yang menyebabkan Trigger terpacu.	INSERT, UPDATE, DELETE, CREATE, ALTER, DROP
Trigger Restriction (optional)	Batasan yang mengizinkan pengeksekusianTrigger.	WHEN (Boolean value)
Trigger Type	Berapa kali body Trigger akan dieksekusi.Jika Trigger adalah ROW-level, maka akandinyalakan sekali untuk setiap baris yangdipengaruhioleh statementTrigger.Sedangkanapabila Trigger ialah STATEMENT-level, maka akan	ROW, STATEMENT

	dinyalakan sekali sebelum atau sesudah statement. Trigger ROW-level diidentifikasi oleh klausa FOR EACH ROW dalam pendefinisian Trigger.	
Trigger Body	Berisi algoritma yang akan dilakukan	Blok PL/SQL

- Pengaturan Trigger
  - Mengaktifkan Trigger

    ALTER TRIGGER <nama_trigger> ENABLE
  - Menonaktifkan Trigger

    ALTER TRIGGER <nama_trigger> DISABLE
  - Mengaktifkan atau menonaktifkan semua Trigger pada suatui tabel

    ALTER TABLE <nama_tabel> {ENABLE/DISABLE} ALL TRIGGERS
  - Menghapus Trigger

    DROP TRIGGER <nama_triger>

- Contoh Penggunaan Trigger

  Membuat simulasi peminjaman buku diperpustakaan. Sehingga Trigger yang ada didalam sistem ini berguna untuk mengurangi stok buku yang disaat buku tersebut dipinjam. Langkah pembuatannya:

1. Buatlah Tabel

   Buku

```
SQL> desc buku;
 Name                                      Null?    Type
 ----------------------------------------- -------- ------------
 ID_BUKU                                   NOT NULL VARCHAR2(7)
 JUDUL_BUKU                                NOT NULL VARCHAR2(25)
 PENGARANG                                 NOT NULL VARCHAR2(15)
 STOK                                      NOT NULL NUMBER(12)
```

Anggota Perpustakaan

```
SQL> desc anggota_perpus;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------
 ID_ANGGOTA                                NOT NULL NUMBER(10)
 NAMA                                      NOT NULL VARCHAR2(50)
 JK                                        NOT NULL VARCHAR2(2)
 NO_TELEPON                                NOT NULL VARCHAR2(15)

SQL> create sequence seq_id_anggota increment by 1;
```

Peminjaman

```
SQL> desc peminjaman;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------
 ID_PEMINJAMAN                             NOT NULL NUMBER(10)
 ID_ANGGOTA                                NOT NULL NUMBER(10)
 ID_BUKU                                   NOT NULL VARCHAR2(7)
 JUMLAH_BUKU                               NOT NULL NUMBER(10)
 TANGGAL_PEMINJAMAN                        NOT NULL DATE

SQL> create sequence seq_id_peminjaman increment by 1;
```

2. Masukan Data pada Tabel

BUKU

```
SQL> select * from buku;

ID_BUKU JUDUL_BUKU                    PENGARANG            STOK
------- ----------------------------- ---------------- ----------
B-01    Basis Data                    Fathansyah               20
B-02    Algoritma dan Pemograman      Rinaldi Munir            30
B-03    Sistem Basis Data             Fathansyah               10
```

Anggota

```
SQL> select*from anggota_perpus;

ID_ANGGOTA NAMA                                               JK NO_TELEPON
---------- -------------------------------------------------- -- ---------------
         1 Irma Ayu Aryani                                    P  085312341234
         2 Yusuf Fakhri                                       L  085312345678
         3 Wiwi Juwita                                        P  085312349876
```

3. Buatlah Triger peminjaman

```
--perintah untuk membuat atau mengganti trigger dengan nama trigger pinjaman
create or replace trigger pinjaman
--event/kejadian dimana trigger akan aktif setelah ada peminjaman yang terjadi
after
    --jenis perintah yang akan dilakukan pada tabel peminjaman
    insert or delete or update on peminjaman
for each row
begin
    --ketika terjadi peminjaman
    if inserting then
        --maka stok buku yang ada pada tabel buku akan berkurang sebanyak
        jumlah_buku pada tabel peminjaman, dengan syarat id_buku yang ada pada
        tabel peminjaman sama dengan tabel buku
        update buku set stok = stok - :new.jumlah_buku where id_buku = :new.
            id_buku;
    --ketika data pada tabel peminjaman dihapus, asusmsi buku telah dikembalikan
    elsif deleting then
        --maka stok buku yang ada pada tabel buku akan berkurang sebanyak
        jumlah_buku pada tabel peminjaman, dengan syarat id_buku yang ada pada
        tabel peminjaman sama dengan tabel buku
        update buku set stok = stok + :old.jumlah_buku where id_buku = :old.
            id_buku;
    --ketika data pada tabel peminjaman di update
    elsif updating then
        --maka pertama jumlah stok buku yang ada pada tabel buku dikembalikan
        ke nilai awal yang belum diganti kemudian dikurangi dengan jumlah buku
        yang baru yang terdapat di tabel peminjaman.
        update buku set stok = stok + :old.jumlah_buku where id_buku = :old.
            id_buku;
        update buku set stok = stok - :new.jumlah_buku where id_buku = :new.
            id_buku;
    end if;
end;
/
```

4. Lakukan Perintah Insert, Update dan Delete pada Tabel Peminjaman

INSERT

```
/*insert*/
insert into peminjaman values(seq_id_peminjaman.nextval, 1, 'B-03', 5, sysdate);
insert into peminjaman values(seq_id_peminjaman.nextval, 2, 'B-02', 10, sysdate);
```

Hasil

```
SQL> select * from peminjaman;

ID_PEMINJAMAN ID_ANGGOTA ID_BUKU JUMLAH_BUKU TANGGAL_P
------------- ---------- ------- ---------- ---------
            1          1 B-03             5 01-NOV-15
            2          2 B-02            10 01-NOV-15

SQL> select * from buku;

ID_BUKU JUDUL_BUKU                    PENGARANG            STOK
------- ----------------------------- -------------------- ----------
B-01    Basis Data                    Fathansyah                   20
B-02    Algoritma dan Pemograman      Rinaldi Munir                20
B-03    Sistem Basis Data             Fathansyah                    5
```

UPDATE

```
/*update*/
update peminjaman set jumlah_buku = 7 where id_anggota = 1;
```

Hasil

```
SQL> select * from peminjaman;

ID_PEMINJAMAN ID_ANGGOTA ID_BUKU JUMLAH_BUKU TANGGAL_P
------------- ---------- -------- ----------- ---------
            1          1 B-03              7 01-NOV-15
            2          2 B-02             10 01-NOV-15

SQL> select * from buku;

ID_BUKU JUDUL_BUKU                    PENGARANG            STOK
------- ----------------------------- ------------------ -----
B-01    Basis Data                    Fathansyah            20
B-02    Algoritma dan Pemograman      Rinaldi Munir         20
B-03    Sistem Basis Data             Fathansyah             3
```

DELETE

```
/*delete*/
delete from peminjaman where id_anggota = 2;
```

Hasil

```
SQL> select * from peminjaman;

ID_PEMINJAMAN ID_ANGGOTA ID_BUKU JUMLAH_BUKU TANGGAL_P
------------- ---------- -------- ----------- ---------
            1          1 B-03              7 01-NOV-15

SQL> select * from buku;

ID_BUKU JUDUL_BUKU                    PENGARANG            STOK
------- ----------------------------- ------------------ -----
B-01    Basis Data                    Fathansyah            20
B-02    Algoritma dan Pemograman      Rinaldi Munir         30
B-03    Sistem Basis Data             Fathansyah             3
```

Karena record yang dihapus pada table peminjaman ialah dengan id_anggota nomer 2 yang sebelumnnya meminjam buku dengan id_buku = B-02 sebanyak 10 setelah dihapus jumlah stok pada table buku akan kembali lagi kr kondidsi awal

Latihan !

Buatlah triger untuk mengenghitung jumlah pendaftar setiap lomba yang ada pada kegiatan dinamik 12

Tabel lomba

```
SQL> select * from buku;

ID_BUKU JUDUL_BUKU                 PENGARANG            STOK
------- -------------------------- ---------------- ----------
B-01    Basis Data                 Fathansyah           20
B-02    Algoritma dan Pemograman   Rinaldi Munir        30
B-03    Sistem Basis Data          Fathansyah           10
```

Tabel pendaftaran

```
SQL> desc pendaftaran;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------

 ID_PENDAFTARAN                            NOT NULL NUMBER(10)
 NAMA                                      NOT NULL VARCHAR2(50)
 ID_LOMBA                                  NOT NULL VARCHAR2(7)
 JUMLAH                                    NOT NULL NUMBER(2)
 TANGGAL                                   NOT NULL DATE
```

Buatlah Trigger untuk menambah jumlah pendaftar bahkan jumlah penfadtar pada tabel sesuai dengan lomba yang diikutinya dan jumlah peserta yang di inputkan

```
ID_PENDAFTARAN NAMA                                               ID_LOMB JUMLAH    TANGGAL
-------------- -------------------------------------------------- ------- ---------- ----------

            24 deni                                               DSTAR   2          14-10-2016
```

```
SQL> select*from lomba;

ID_LOMB NAMA_LOMBA                 JUMLAH_PENDAFTAR
------- -------------------------- ----------------
DSTAR   Dinamik Star                              2
OTIK    Olompiade TIK                             0
LCW     Lomba Cipta Web                           0
DRDH    Donor Darah                               0
```

SELESAI !!!

SEMANGAT

REFERENSI

MODUL PRAKTIKUM TRIGGER TAHUN 2013