CAB BOOKING

TRIGGERS

APRIL 16, 2019

DBMS PL/SQL

Driver Triggers:

1. 'Driver' removal will require removing his record from 'driver_cab' table first.

```
SQL> create or replace trigger rem_driver
   before delete on driver
   for each row
   begin
   delete from driver_cab where driver_cab.driver_id=:old.driver_id;
   end;
   /
```

2. Getting driver id while registering the driver.

```
SQL> create or replace trigger bef_reg_driver
before insert on driver
for each row
begin
select drivid.NEXTVAL into :new.driver_id
from dual;
end;
/
```

Cab_ride Triggers:

3. Generating payment id using payid sequence.

```
SQL> create or replace trigger pay_tri
   before insert on cab_ride
   for each row
```

```
begin
select payid.NEXTVAL into :new.payment_id from dual;
end;
/
```

4. Inserting into payment_type table after booking of cab.

```
SQL> create or replace trigger post_cab
after insert on cab_ride
for each row
begin
insert into payment_type values(:new.payment_id,'&type_name');
end;
/
```

5. for Deleting a record from cab_ride table , cab_ride_history should be updated and data should be removed from payment_type table.

```
SQL> create or replace trigger rem_cab_ride
before delete on cab_ride
for each row
declare
cbid cab_ride.cab_id%type;
cid cab_ride.cust_id%type;
s cab_ride.ride_start_time%type;
e cab_ride.ride_end_time%type;
j cab_ride.jfrom%type;
t cab_ride.jto%type;
c cab_ride.cancelled%type;
pid cab_ride.payment_id%type;
charges cab_ride.charges%type;
```

```
begin
cbid := :old.cab_id;
cid := :old.cust_id;
s := :old.ride_start_time;
e := :old.ride_end_time;
j := :old.jfrom;
t := :old.jto;
c := :old.cancelled;
pid := :old.payment_id;
charges := :old.charges;
reg_cab_ride_history(cbid,cid,s,e,j,t,c,pid,charges);
delete from payment_type where payment_id = pid;
end;
/
```

Customer Triggers:

6. Generating customer id using cusid sequence.

```
SQL> create or replace trigger cus_tri
  before insert on customer
  for each row
  begin
  select cusid.NEXTVAL into :new.cust_id
  from dual;
  end;
  /
```

7. Customer removal will require removing his/her record from 'cab_ride' table first .

```
SQL> create or replace trigger rem_customer
   before delete on customer
   for each row
   begin
   delete from cab_ride where cab_ride.cust_id = :old.cust_id;
   end;
   /
```

Cab Triggers:

8. Getting cab id while registering the cab.

```
SQL> create or replace trigger bef_reg_cab
before insert on cab
for each row
begin
select cabid.NEXTVAL into :new.cab_id
from dual;
end;
/
```

9. Removing cab from DB will need to first delete its record from Driver_cab Table.

```
SQL> create or replace trigger rem_cab
   before delete on cab
   for each row
   begin
   delete from driver_cab where driver_cab.cab_id=:old.cab_id;
   end;  /
```

Owner Triggers:

10. Getting owner id while registering the owner

```
SQL> create or replace trigger bef_reg_owner before insert on owner
for each row
begin
select ownid.NEXTVAL into :new.owner_id
from dual;
end;    /
```

11. Removing an owner from DB will need to first delete its record from cab Table

```
SQL> create or replace trigger rem_owner
before delete on owner
for each row
begin
delete from cab where cab.owner_id=:old.owner_id;
end;
/
```

Car_model Triggers:

12. Removing car model from DB will need to first delete its record from cab Table

```
SQL> create or replace trigger rem_car_model
   before delete on car_model
   for each row
   begin
   delete from cab where cab.model_name=:old.model_name;
   end;    /
```