

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY



KATTANKULATHUR, CHENNAI

4th Sem

DBIS

Online Gas Booking System

(DBMS MINI PROJECT)

Team Members

Adarsh Kumar Bohidar(RA2011042010083)

Aditya Rai(RA2011042010097)

Harsh Vardhan(RA2011042010082)

AIM:- To implement different SQL queries to create a database for an online gas booking system

Description

List of Entities and its attributes

1.Customer :- This entity includes attributes like name, customer_id , address and mobile no. Here the primary key will is customer_id. This entity is to show who will be receiving the gas or book the gas.

2.Agency :- This entity includes attributes like name, Agency_id. Its is the entity that receives the order and here the primary key is Agency_id.

3.Stock :- This entity includes attributes quantity, type, batch_no. and primary key here is batch_no.

4.LPG :- This entity includes attributes gid, weight, type and primary key is gid.

5.Bill :- This entity includes attributes Bill_id , Amount and customer_id where primary key is Bill_id and the foreign key is Customer_id

CREATION OF A NEW USER

```
SQL> SPOOL C:\Users\HP\OneDrive\Desktop\JOINS_VIEW.TXT;
SQL> create user C##Again identified by dbms123;

User created.

SQL> connect;
Enter user-name: system
Enter password:
Connected.
SQL> grant all privileges to C##Again;

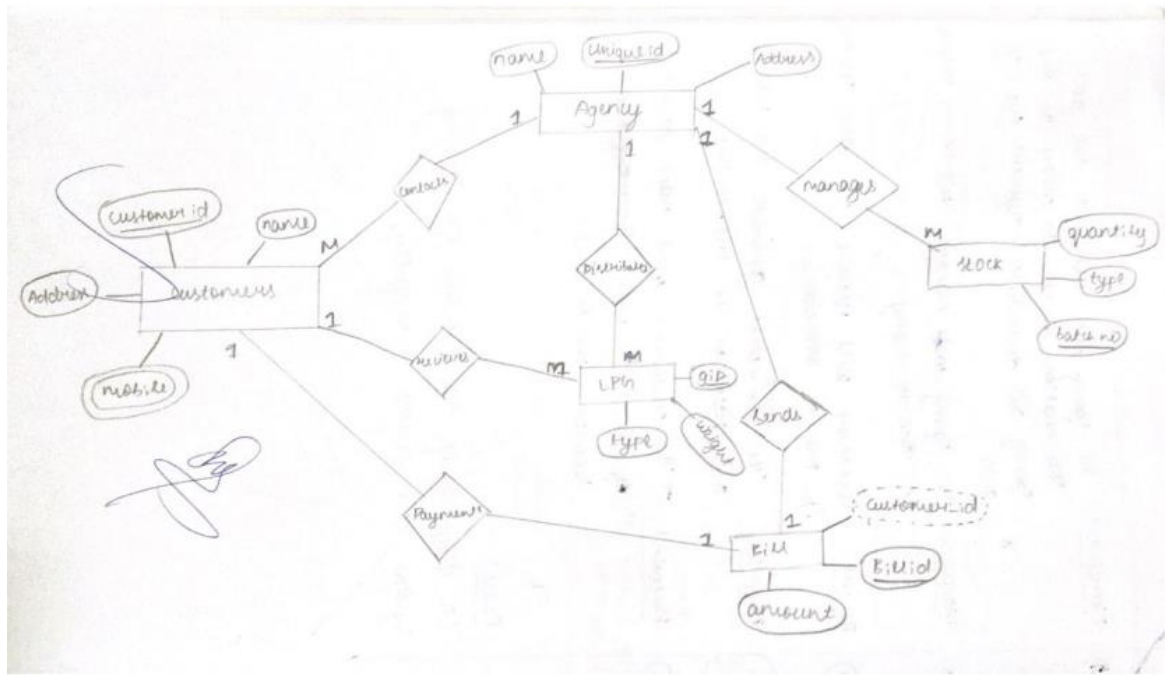
Grant succeeded.
```

CONNECTING TO NEW USER AND GIVING PRIVILEGES

```
SQL> connect;
Enter user-name: system
Enter password:
Connected.
SQL> grant all privileges to C##Again;

Grant succeeded.
```

ENTITY RELATION DIAGRAM FOR ONLINE GAS BOOKING SYSTEM:



CREATING NECESARRY TABLES:

CUSTOMERS

```
create table customers(  
    Customer_id int NOT NULL,  
    Name varchar(200),  
    Address varchar(200),  
    Mobile int NOT NULL,  
    PRIMARY KEY(Customer_id));
```

AGENCY

```
create table agency(  
    Agency_id int NOT NULL,  
    Name varchar(200),  
    PRIMARY KEY(Agency_id));
```

STOCK

```
create table stock(  
    Batch_no int,  
    type varchar(10),  
    quantity int,  
    PRIMARY KEY(Batch_no));
```

BILL

```
create table Bill(  
    Bill_id int,  
    Amount int,  
    Customer_id int,  
    PRIMARY KEY(Bill_id),  
    FOREIGN KEY(Customer_id) REFERENCES customers(Customer_id));
```

LPG

```
create table LPG(  
    Gid int,  
    weight varchar(5),  
    type varchar(5));
```

INSERTING DATA INTO THE CREATED TABLES:

1.INSERTING INTO CUSTOMERS TABLE

```
SQL> insert into customers values(&customer_id,&name,&address,&mobile);
Enter value for customer_id: 10387
Enter value for name: Anderson Silva
Enter value for address: KTR, Chennai
Enter value for mobile: 9874561235
old 1: insert into customers values(&customer_id,&name,&address,&mobile)
new 1: insert into customers values(10387,'Anderson Silva','KTR, Chennai',9874561235)

1 row created.

SQL> /
Enter value for customer_id: 10795
Enter value for name: Gelson Martins
Enter value for address: MCO, Chennai
Enter value for mobile: 6897451354
old 1: insert into customers values(&customer_id,&name,&address,&mobile)
new 1: insert into customers values(10795,'Gelson Martins','MCO, Chennai',6897451354)

1 row created.

SQL> /
Enter value for customer_id: 10732
Enter value for name: Inaki Williams
Enter value for address: Sevilla, Bangaluru
Enter value for mobile: 9684521654
old 1: insert into customers values(&customer_id,&name,&address,&mobile)
new 1: insert into customers values(10732,'Inaki Williams','Sevilla, Bangaluru',9684521654)

1 row created.

SQL> /
Enter value for customer_id: 10894
Enter value for name: Paulo Dybala
Enter value for address: Turin, Coimbatore
Enter value for mobile: 9988651423
old 1: insert into customers values(&customer_id,&name,&address,&mobile)
new 1: insert into customers values(10894,'Paulo Dybala','Turin, Coimbatore',9988651423)

1 row created.

SQL> /
Enter value for customer_id: 10338
Enter value for name: Gareth Bale
Enter value for address: Wales, Chennai
Enter value for mobile: 9874561325
old 1: insert into customers values(&customer_id,&name,&address,&mobile)
new 1: insert into customers values(10338,'Gareth Bale','Wales, Chennai',9874561325)

1 row created.

SQL> select * from customers;
```

```
SQL> select * from customers;
```

CUSTOMER_ID	NAME	ADDRESS	MOBILE
10387	Anderson Silva	KTR, Chennai	9874561235
10795	Gelson Martins	MCO, Chennai	6897451354
10732	Inaki Williams	Sevilla, Bangaluru	9684521654
10894	Paulo Dybala	Turin, Coimbatore	9988651423
10338	Gareth Bale	Wales, Chennai	9874561325

```
SQL> commit;
```

2.INSERTING INTO BILL TABLE

```
SQL> insert into bill values(&bill_id,&amount,&customer_id);
Enter value for bill_id: 2890
Enter value for amount: 3300
Enter value for customer_id: 10387
old 1: insert into bill values(&bill_id,&amount,&customer_id)
new 1: insert into bill values(2890,3300,10387)

1 row created.

SQL> /
Enter value for bill_id: 2892
Enter value for amount: 3100
Enter value for customer_id: 10795
old 1: insert into bill values(&bill_id,&amount,&customer_id)
new 1: insert into bill values(2892,3100,10795)

1 row created.

SQL> /
Enter value for bill_id: 2894
Enter value for amount: 2800
Enter value for customer_id: 10732
old 1: insert into bill values(&bill_id,&amount,&customer_id)
new 1: insert into bill values(2894,2800,10732)

1 row created.

SQL> /
Enter value for bill_id: 2896
Enter value for amount: 3700
Enter value for customer_id: 10894
old 1: insert into bill values(&bill_id,&amount,&customer_id)
new 1: insert into bill values(2896,3700,10894)

1 row created.

SQL> /
Enter value for bill_id: 2900
Enter value for amount: 4500
Enter value for customer_id: 10338
old 1: insert into bill values(&bill_id,&amount,&customer_id)
new 1: insert into bill values(2900,4500,10338)

1 row created.

SQL> select * from bill;

  BILL_ID    AMOUNT  CUSTOMER_ID
-----
    2890      3300      10387
    2892      3100      10795
    2894      2800      10732
    2896      3700      10894
    2900      4500      10338

SQL> commit;
```

3.INSERTING INTO STOCK TABLE :

```
SQL> insert into stock values(&batch_no,&'type',&quantity);
Enter value for batch_no: 34
Enter value for type: S-CYL
Enter value for quantity: 234
old 1: insert into stock values(&batch_no,&'type',&quantity)
new 1: insert into stock values(34,'S-CYL',234)

1 row created.

SQL> /
Enter value for batch_no: 72
Enter value for type: L-CYL
Enter value for quantity: 187
old 1: insert into stock values(&batch_no,&'type',&quantity)
new 1: insert into stock values(72,'L-CYL',187)

1 row created.

SQL> /
Enter value for batch_no: 25
Enter value for type: PIPES
Enter value for quantity: 481
old 1: insert into stock values(&batch_no,&'type',&quantity)
new 1: insert into stock values(25,'PIPES',481)

1 row created.

SQL> /
Enter value for batch_no: 53
Enter value for type: Regulators
Enter value for quantity: 785
old 1: insert into stock values(&batch_no,&'type',&quantity)
new 1: insert into stock values(53,'Regulators',785)

1 row created.

SQL> /
Enter value for batch_no: 45
Enter value for type: Stoves
Enter value for quantity: 200
old 1: insert into stock values(&batch_no,&'type',&quantity)
new 1: insert into stock values(45,'Stoves',200)

1 row created.

SQL> select * from stock;

  BATCH_NO TYPE          QUANTITY
-----
         34 S-CYL             234
         72 L-CYL             187
         25 PIPES            481
         53 Regulators       785
         45 Stoves           200

SQL> commit;

Commit complete.
```

4.INSERTING INTO LPG TABLE:

```
SQL> insert into lpg values(&gid,&weight,&type');
Enter value for gid: 234
Enter value for weight: 15
Enter value for type: S-CYL
old 1: insert into lpg values(&gid,&weight,&type')
new 1: insert into lpg values(234,'15','S-CYL')
ERROR:
ORA-01756: quoted string not properly terminated

SQL> insert into lpg values(&gid,&weight,'&type');
Enter value for gid: 234
Enter value for weight: 15
Enter value for type: S-CYL
old 1: insert into lpg values(&gid,&weight,'&type')
new 1: insert into lpg values(234,'15','S-CYL')

1 row created.

SQL> /
Enter value for gid: 238
Enter value for weight: 15
Enter value for type: S-CYL
old 1: insert into lpg values(&gid,&weight,'&type')
new 1: insert into lpg values(238,'15','S-CYL')

1 row created.

SQL> /
Enter value for gid: 456
Enter value for weight: 20
Enter value for type: L-CYL
old 1: insert into lpg values(&gid,&weight,'&type')
new 1: insert into lpg values(456,'20','L-CYL')

1 row created.

SQL> /
Enter value for gid: 472
Enter value for weight: 20
Enter value for type: L-CYL
old 1: insert into lpg values(&gid,&weight,'&type')
new 1: insert into lpg values(472,'20','L-CYL')

1 row created.

SQL> /
Enter value for gid: 489
Enter value for weight: 20
Enter value for type: L-CYL
old 1: insert into lpg values(&gid,&weight,'&type')
new 1: insert into lpg values(489,'20','L-CYL')

1 row created.
```



```
SQL> SELECT * FROM LPG;

      GID WEIGH TYPE
-----
      234  15   S-CYL
      238  15   S-CYL
      456  20   L-CYL
      472  20   L-CYL
      489  20   L-CYL

SQL> COMMIT;

Commit complete.
```

5.INSRTING INTO AGENCY TABLE:

```

SQL> insert into agency values(&agency_id,&name,&address');
Enter value for agency_id: 305
Enter value for name: ADARSH AGENCY
Enter value for address: CHENNAI
old 1: insert into agency values(&agency_id,&name,&address')
new 1: insert into agency values(305,'ADARSH AGENCY','CHENNAI')
1 row created.

SQL> /
Enter value for agency_id: 320
Enter value for name: ADITYA AGENCY
Enter value for address: INDORE
old 1: insert into agency values(&agency_id,&name,&address')
new 1: insert into agency values(320,'ADITYA AGENCY','INDORE')
1 row created.

SQL> /
Enter value for agency_id: 335
Enter value for name: JOBBY AGENCY
Enter value for address: BENGALURU
old 1: insert into agency values(&agency_id,&name,&address')
new 1: insert into agency values(335,'JOBBY AGENCY','BENGALURU')
1 row created.

SQL> /
Enter value for agency_id: 350
Enter value for name: RAHUL AGENCY
Enter value for address: COIMBATORE
old 1: insert into agency values(&agency_id,&name,&address')
new 1: insert into agency values(350,'RAHUL AGENCY','COIMBATORE')
1 row created.

SQL> /
Enter value for agency_id: 365
Enter value for name: ANIKET AGENCY
Enter value for address: BILANGIR
old 1: insert into agency values(&agency_id,&name,&address')
new 1: insert into agency values(365,'ANIKET AGENCY','BILANGIR')
1 row created.

SQL> SELECT * FROM AGENCY;

AGENCY_ID NAME                ADDRESS
-----
305 ADARSH AGENCY            CHENNAI
320 ADITYA AGENCY            INDORE
335 JOBBY AGENCY             BENGALURU
350 RAHUL AGENCY             COIMBATORE
365 ANIKET AGENCY            BILANGIR

SQL> COMMIT;
Commit complete.

```

```

SQL> insert into agency values(345,'PED AGENCY','MORADABAD');
1 row created.

SQL> DELETE FROM AGENCY WHERE AGENCY_ID=345;
1 row deleted.

SQL> ALTER TABLE AGENCY
  2  ADD INFO VARCHAR(10);
Table altered.

SQL> ALTER TABLE AGENCY
  2  DROP COLUMN INFO;
ALTER TABLE AGENCY
*
ERROR at line 1:
ORA-12988: cannot drop column from table owned by SYS

SQL> COMMIT;
Commit complete.

SQL> SAVEPOINT S1;
Savepoint created.

```

TCL COMMANDS

```

SQL> COMMIT;
Commit complete.

SQL> SAVEPOINT S1;
Savepoint created.

```

BUILT-IN FUNCTIONS

```

SQL> SELECT SUM(AMOUNT) FROM BILL;

SUM(AMOUNT)
-----
      17400

SQL> SELECT AVG(WEIGHT) FROM LPG;

AVG(WEIGHT)
-----
         18

```

SET OPERATORS

```
SQL> SELECT NAME FROM CUSTOMERS
2 UNION
3 SELECT NAME FROM AGENCY;

NAME
-----
ADARSH AGENCY
ADITYA AGENCY
ANIKET AGENCY
Anderson Silva
Gareth Bale
Gelson Martins
Inaki Williams
JOBBY AGENCY
Paulo Dybala
RAHUL AGENCY

10 rows selected.

SQL> SELECT CUSTOMER_ID FROM CUSTOMERS
2 INTERSECT
3 SELECT CUSTOMER_ID FROM BILL;

CUSTOMER_ID
-----
10338
10387
10732
10795
10894

SQL> SELECT AMOUNT FROM BILL
2 UNION ALL;///
3

SQL> SELECT ADDRESS FROM CUSTOMERS
2 UNION ALL
3 SELECT ADDRESS FROM AGENCY;

ADDRESS
-----
KTR, Chennai
MCO, Chennai
Sevilla, Bangaluru
Turin, Coimbatore
Wales, Chennai
CHENNAI
INDORE
BENGALURU
COIMBATORE
BILANGIR

10 rows selected.
```

SUB QUERY

```
SQL> SELECT BATCH_NO, QUANTITY FROM STOCK WHERE QUANTITY = (SELECT MAX(QUANTITY) FROM STOCK);

  BATCH_NO  QUANTITY
-----
        53         785

SQL> SELECT AGENCY_ID,NAME FROM AGENCY WHERE AGENCY_ID IN(SELECT AGENCY_ID FROM AGENCY WHERE ADDRESS='BENGALURU');

 AGENCY_ID NAME
-----
        335 JOBBY AGENCY

SQL> SELECT AGENCY_ID,NAME FROM AGENCY WHERE AGENCY_ID=ANY(SELECT AGENCY_ID FROM AGENCY WHERE ADDRESS='CHENNAI');

 AGENCY_ID NAME
-----
        305 ADARSH AGENCY
```

CREATING DIFFERENT VIEWS WITH JOIN FOR THE DATABASE:

```
SQL> CREATE VIEW BILLSVIEW
  2 AS SELECT a.customer_id,name,bill_id,amount
  3 from customers a,bill b
  4 where a.customer_id=b.customer_id;
```

View created.

```
SQL> select * from BILLSVIEW;
```

CUSTOMER_ID	NAME	BILL_ID	AMOUNT
10387	Anderson Silva	2890	3300
10795	GELSON MARTINS	2892	3100
10732	INAKI WILLIAMS	2894	2800
10894	PAULO DYBALA	2896	3700
10338	GARETH BALE	2900	4500

```
SQL> select * from customers natural join bill;
```

CUSTOMER_ID	NAME	ADDRESS	MOBILE	BILL_ID
10387	Anderson Silva	KTR, CHENNAI	9874561235	2890
10795	GELSON MARTINS	MCO,CHENNAI	6897451354	2892
10732	INAKI WILLIAMS	SEVILLA, BENGALURU	9684521654	2894
10894	PAULO DYBALA	TURIN, COIMBATORE	9988651423	2896
10338	GARETH BALE	WALES, CHENNAI	987461325	2900

```
SQL> select * from agency natural join stock;
```

AGENCY_ID	NAME	ADDRESS	BATCH_NO	TYPE

QUANTITY				

305 234	ADARSH AGENCY	CHENNAI	34	S-CYL
305 187	ADARSH AGENCY	CHENNAI	72	L-CYL
305 481	ADARSH AGENCY	CHENNAI	25	PIPES
AGENCY_ID	NAME	ADDRESS	BATCH_NO	TYPE

QUANTITY				

305 785	ADARSH AGENCY	CHENNAI	53	REGULATORS
305 200	ADARSH AGENCY	CHENNAI	45	STOVES
320 234	ADITYA AGENCY	INDORE	34	S-CYL
AGENCY_ID	NAME	ADDRESS	BATCH_NO	TYPE

QUANTITY				

320 187	ADITYA AGENCY	INDORE	72	L-CYL
320 481	ADITYA AGENCY	INDORE	25	PIPES
320 785	ADITYA AGENCY	INDORE	53	REGULATORS
AGENCY_ID	NAME	ADDRESS	BATCH_NO	TYPE

QUANTITY				

320 200	ADITYA AGENCY	INDORE	45	STOVES
335 234	JOBBY AGENCY	BENGALURU	34	S-CYL
335 187	JOBBY AGENCY	BENGALURU	72	L-CYL

AGENCY_ID	NAME	ADDRESS	BATCH_NO	TYPE
335	JOBBY AGENCY	BENGALURU	25	PIPES
481				
335	JOBBY AGENCY	BENGALURU	53	REGULATORS
785				
335	JOBBY AGENCY	BENGALURU	45	STOVES
200				
AGENCY_ID	NAME	ADDRESS	BATCH_NO	TYPE
350	RAHUL AGENCY	COIMBATORE	34	S-CYL
234				
350	RAHUL AGENCY	COIMBATORE	72	L-CYL
187				
350	RAHUL AGENCY	COIMBATORE	25	PIPES
481				
AGENCY_ID	NAME	ADDRESS	BATCH_NO	TYPE
350	RAHUL AGENCY	COIMBATORE	53	REGULATORS
785				
350	RAHUL AGENCY	COIMBATORE	45	STOVES
200				
369	ANIKET AGENCY	BILANGIR	34	S-CYL
234				
AGENCY_ID	NAME	ADDRESS	BATCH_NO	TYPE
369	ANIKET AGENCY	BILANGIR	72	L-CYL
187				
369	ANIKET AGENCY	BILANGIR	25	PIPES
481				
369	ANIKET AGENCY	BILANGIR	53	REGULATORS
785				

AGENCY_ID	NAME	ADDRESS	BATCH_NO	TYPE
369	ANIKET AGENCY	BILANGIR	45	STOVES
200				

25 rows selected.

PL/SQL

TRIGGER:-

```
SQL> set serveroutput on;
SQL> create or replace trigger new_bill
  2 before
  3 insert or update
  4 on bill
  5 for each row
  6 when(NEW.bill_id>0)
  7 begin
  8 :NEW.amount:= :NEW.amount-:NEW.amount*0.3;
  9 END;
10 /

Trigger created.
```

OUTPUT:-

```
SQL> insert into customers values(10882,'Fabien Centonze','Nice,France',8561237854);

1 row created.

SQL> insert into bill values (2902,4000,10882);

1 row created.

SQL> select * from bill;

  BILL_ID    AMOUNT  CUSTOMER_ID
-----
    2890       3300       10387
    2892       3100       10795
    2894       2800       10732
    2896       3700       10894
    2900       4500       10338
    2902       2800       10882

6 rows selected.
```

PROCEDURE WITH EXCEPTION

HANDLING:-

```
SQL> create or replace procedure increase_quantity(batch_id number) IS
  2  current_quantity number;
  3  empty_stock exception;
  4  begin
  5  select quantity into current_quantity from stock
  6  where batch_no=batch_id;
  7  IF current_quantity=0 THEN
  8  Raise empty_stock;
  9  ELSE
 10  update stock
 11  set quantity=quantity*2
 12  where batch_no=batch_id;
 13  END IF;
 14  EXCEPTION
 15  WHEN NO_DATA_FOUND THEN
 16  dbms_output.put_line('No such batch number');
 17  WHEN empty_stock THEN
 18  dbms_output.put_line('Stock is empty');
 19  END increase_quantity;
 20  /
```

Procedure created.

OUTPUT:-

```
SQL> declare
  2  x number;
  3  begin
  4  x:=45;
  5  increase_quantity(x);
  6  end;
  7  /
```

PL/SQL procedure successfully completed.

```
SQL> select * from stock;
```

BATCH_NO	TYPE	QUANTITY
34	S-CYL	234
72	L-CYL	187
25	PIPES	481
53	REGULATORS	785
45	STOVES	400

FUNCTION:-

```
SQL> create or replace function total_quantity
 2  return number
 3  IS
 4  total_stock number(8) :=0;
 5  begin
 6  select sum(quantity) into total_stock
 7  from stock;
 8  return total_stock;
 9  end;
10  /
```

Function created.

OUTPUT:-

```
SQL> declare
 2  x number(8);
 3  begin
 4  x:=total_quantity();
 5  dbms_output.put_line('Total quantity in stock is : '||x);
 6  end;
 7  /
```

Total quantity in stock is : 2087

PL/SQL procedure successfully completed.

CURSOR:-

i) EXPLICIT CURSOR

```
SQL> declare
  2  cid customers.customer_id%type;
  3  cname customers.name%type;
  4  caddr customers.address%type;
  5  cursor c_customers is
  6  select customer_id,name,address from customers;
  7  begin
  8  open c_customers;
  9  LOOP
 10  fetch c_customers into cid,cname,caddr;
 11  exit when c_customers%notfound;
 12  dbms_output.put_line(cid || ' ' || cname || ' ' || caddr);
 13  end loop;
 14  close c_customers;
 15  end;
 16  /
10387 Anderson Silva KTR, CHENNAI
10795 GELSON MARTINS MCO,CHENNAI
10732 INAKI WILLIAMS SEVILLA, BENGALURU
10894 PAULO DYBALA TURIN, COIMBATORE
10338 GARETH BALE WALES, CHENNAI
10882 Fabien Centonze Nice,France

PL/SQL procedure successfully completed.
```

ii) IMPLICIT CURSOR

```
PL/SQL procedure successfully completed.

SQL> declare
  2  total number(8);
  3  begin
  4  update stock
  5  set quantity=quantity+50;
  6  if sql%notfound then
  7  dbms_output.put_line('No quantity updation');
  8  elsif sql%found then
  9  total:= sql%rowcount;
 10  dbms_output.put_line(total || ' quantity type updated');
 11  end if;
 12  end;
 13  /
5 quantity type updated

PL/SQL procedure successfully completed.
```