BIOMETRIC BASED AUTOMATED METRO RAIL SYSTEM

ROLL NO: 1602-18-737-097
NAME: B. SAI MANISHA

ABSTRACT:

The project is an application software developed for monitoring the biometric system which mainly focuses on basic operations like scanning the fingerprint, updating information, and generating online transactions. It is mainly developed for the sake of passengers who need to wait in the long queues for the ticket near the counter. This project helps to solve the problem by providing biometric system. The biometric scanner identifies the passengers who have subscribed for a metro card and directly deducts the corresponding amount from their card which makes the transactions easier and safer. This article aims to provide a structured approach to minimize the time duration of waiting in queues and also makes the system more secure.

REQUIREMENT ANALYSIS:

LIST OF TABLES

- Passengers
- Biometric_Scanners
- Scans
- Metro_Card
- Reserves
- Generates
- Transaction

LIST OF ATTRIBUTES WITH THEIR DOMAIN TYPES

Passengers:

Passenger id: p_id Number(20),

Mail id: mail_id varchar2(20),

Passenger name: name varchar2(20),

Passenger phone number: contact_no number(20)

Scans:

Time when scanned: when

Subscription: Subscription

Biometric_Scanners:

Scanner id: Scanner_id number(20),

Scanner name: name varchar2(20),

Cost: cost number(20),

Accuracy: accuracy number(20)

Reserves:

Subscription Status: Subscription_status varchar2(20)

Generates:

Destination location: destination_location varchar2(20),

Start location: start_location varchar2(20)

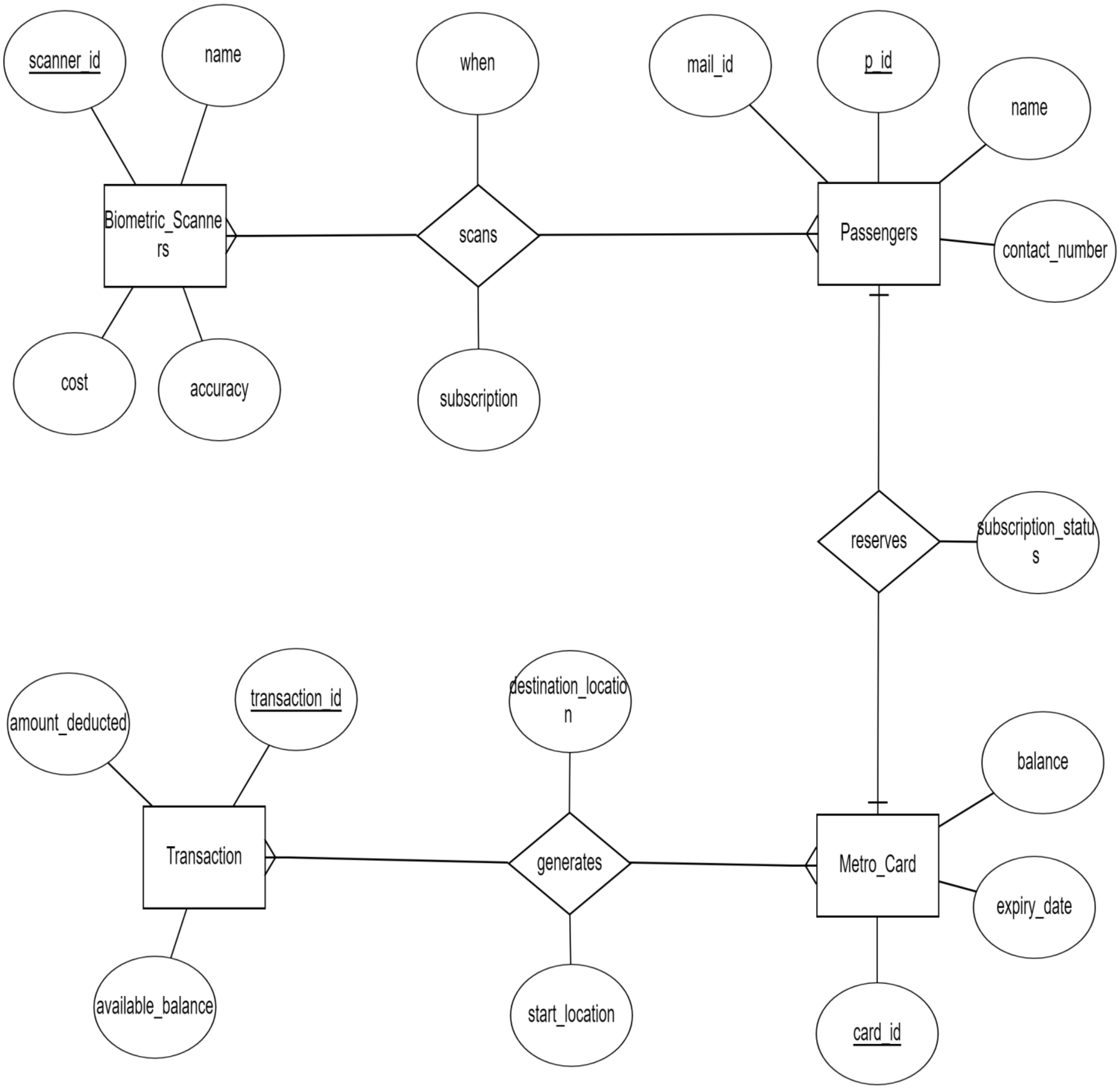
Transaction:

Transaction id: Transaction_id number(20),

Amount deducted: amount_deducted number(20),

Available balance: available_deducted number(20)

ENTITY RELATIONSHIP DIAGRAM:



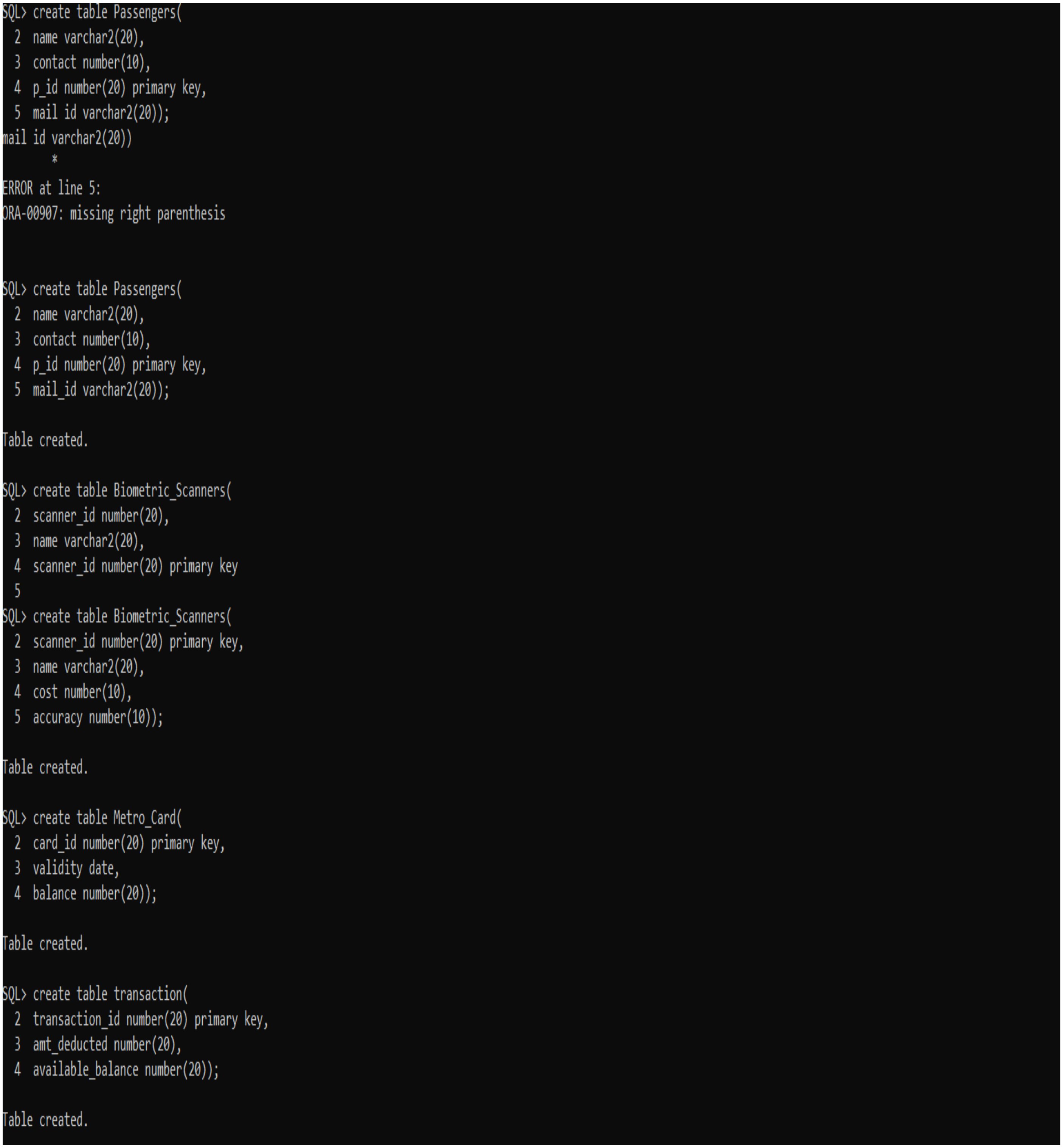
Mapping cardinalities and participation constraints:

A biometric scanner can scan any number of passengers and any number of passengers can also scan biometric scanner and hence many to many mapping cardinality is established between passengers and biometric scanners.

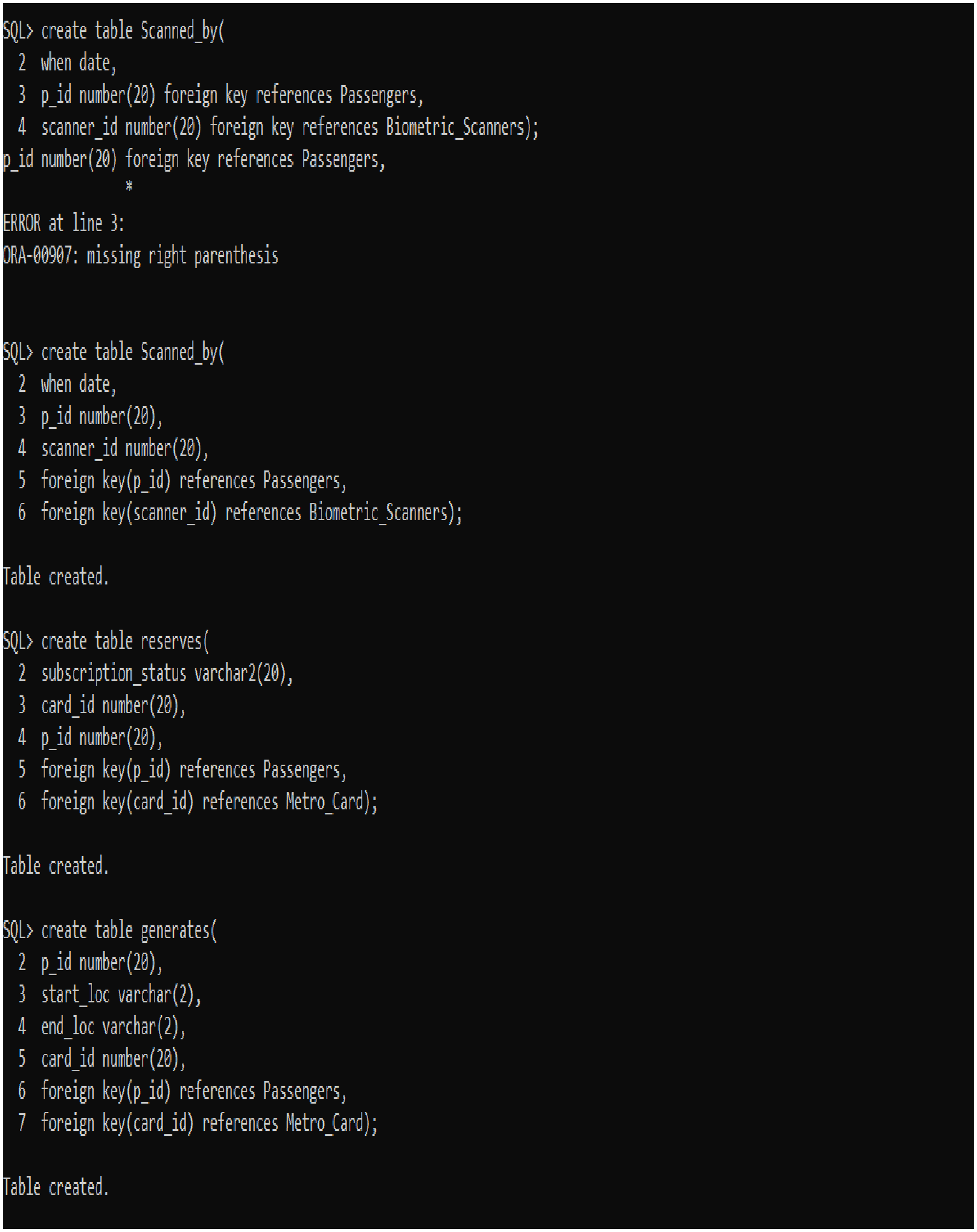
A passenger can reserve only one metro card and vice versa and hence one to one mapping cardinality is established between passengers and metro card.

Any number of transactions can be made from metro card and a transaction can be generated from many number of metro cards and hence it is a many to many mapping cardinality.

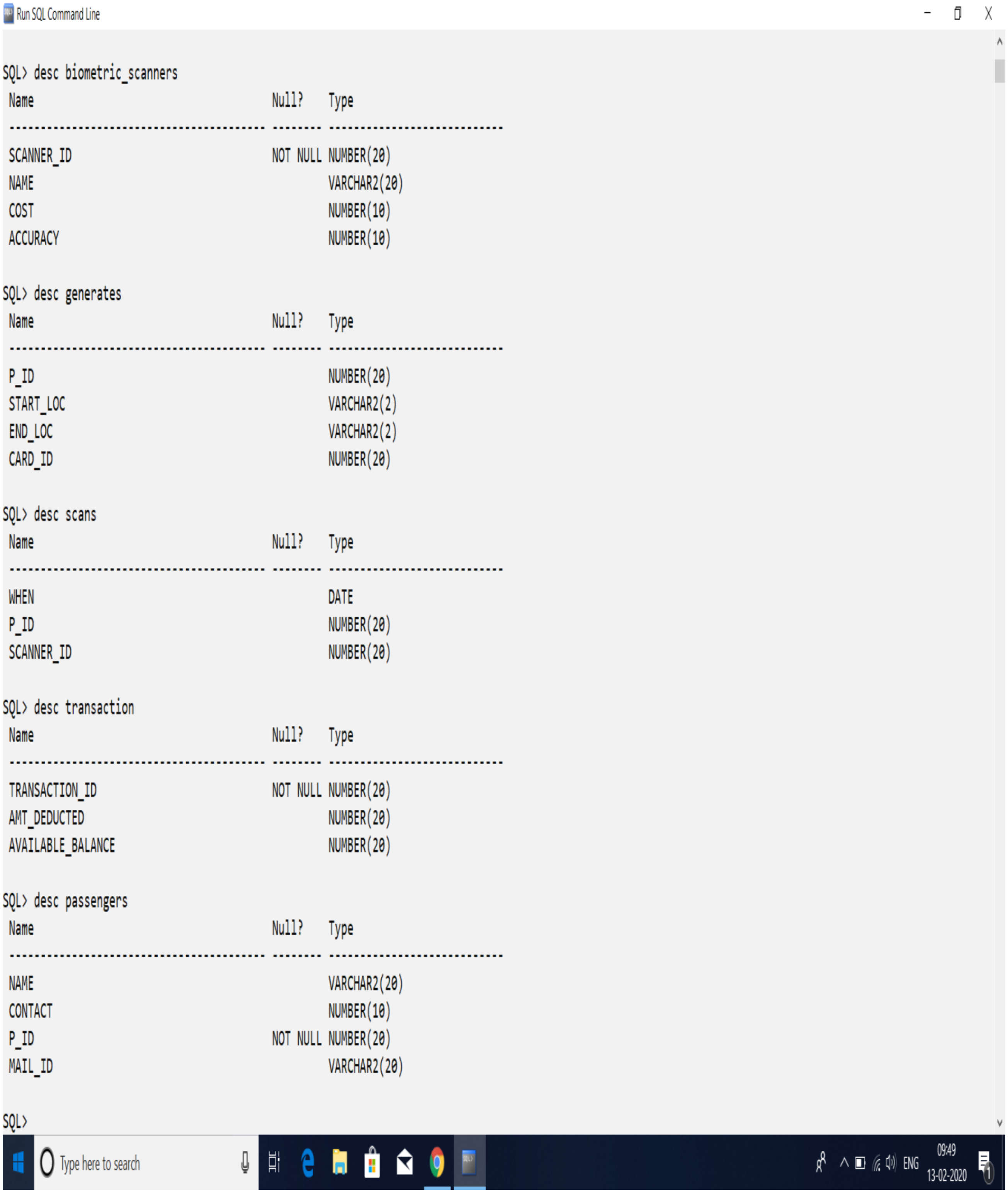
DDL COMMANDS



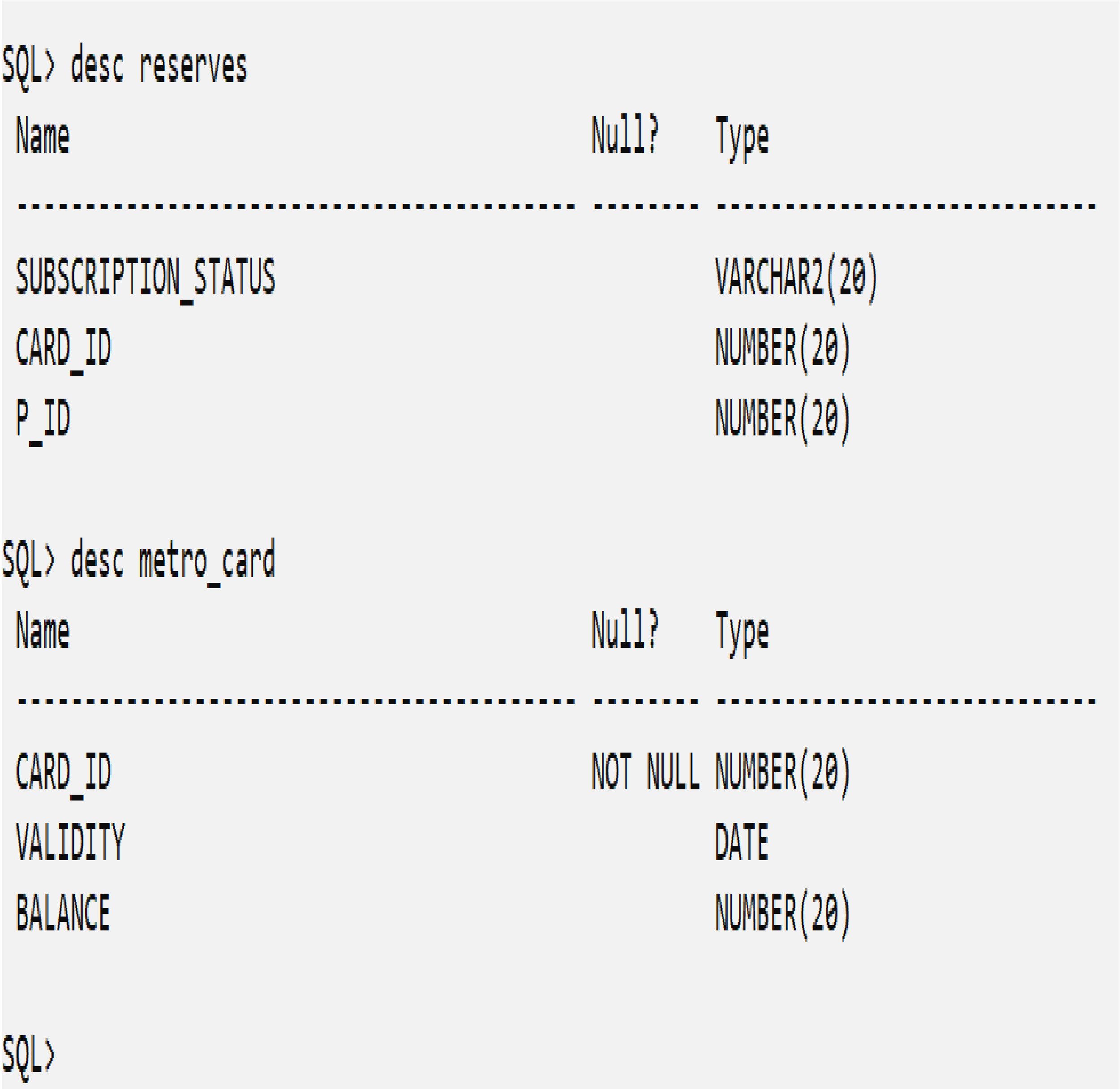
```
SQL> create table Passengers(
  2  name varchar2(20),
  3  contact number(10),
  4  p_id number(20) primary key,
  5  mail id varchar2(20));
mail id varchar2(20))
       *
ERROR at line 5:
ORA-00907: missing right parenthesis


SQL> create table Passengers(
  2  name varchar2(20),
  3  contact number(10),
  4  p_id number(20) primary key,
  5  mail_id varchar2(20));

Table created.

SQL> create table Biometric_Scanners(
  2  scanner_id number(20),
  3  name varchar2(20),
  4  scanner_id number(20) primary key
  5
SQL> create table Biometric_Scanners(
  2  scanner_id number(20) primary key,
  3  name varchar2(20),
  4  cost number(10),
  5  accuracy number(10));

Table created.

SQL> create table Metro_Card(
  2  card_id number(20) primary key,
  3  validity date,
  4  balance number(20));

Table created.

SQL> create table transaction(
  2  transaction_id number(20) primary key,
  3  amt_deducted number(20),
  4  available_balance number(20));

Table created.
```



```
SQL> create table Scanned_by(
  2  when date,
  3  p_id number(20) foreign key references Passengers,
  4  scanner_id number(20) foreign key references Biometric_Scanners);
p_id number(20) foreign key references Passengers,
                *
ERROR at line 3:
ORA-00907: missing right parenthesis


SQL> create table Scanned_by(
  2  when date,
  3  p_id number(20),
  4  scanner_id number(20),
  5  foreign key(p_id) references Passengers,
  6  foreign key(scanner_id) references Biometric_Scanners);

Table created.

SQL> create table reserves(
  2  subscription_status varchar2(20),
  3  card_id number(20),
  4  p_id number(20),
  5  foreign key(p_id) references Passengers,
  6  foreign key(card_id) references Metro_Card);

Table created.

SQL> create table generates(
  2  p_id number(20),
  3  start_loc varchar(2),
  4  end_loc varchar(2),
  5  card_id number(20),
  6  foreign key(p_id) references Passengers,
  7  foreign key(card_id) references Metro_Card);

Table created.
```

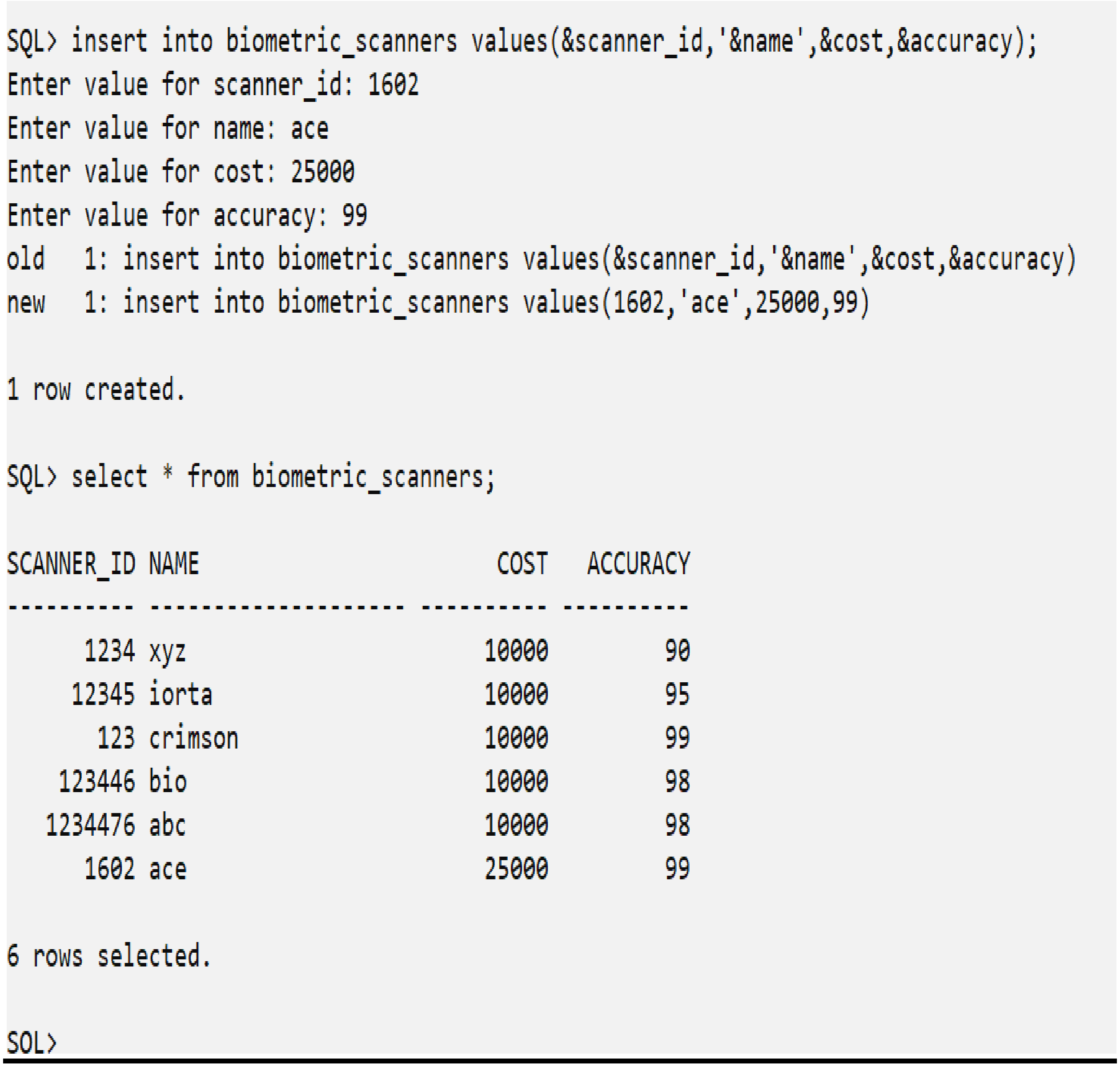


```
SQL> desc biometric_scanners
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 SCANNER_ID                                NOT NULL NUMBER(20)
 NAME                                               VARCHAR2(20)
 COST                                               NUMBER(10)
 ACCURACY                                           NUMBER(10)

SQL> desc generates
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 P_ID                                               NUMBER(20)
 START_LOC                                          VARCHAR2(2)
 END_LOC                                            VARCHAR2(2)
 CARD_ID                                            NUMBER(20)

SQL> desc scans
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 WHEN                                               DATE
 P_ID                                               NUMBER(20)
 SCANNER_ID                                         NUMBER(20)

SQL> desc transaction
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 TRANSACTION_ID                            NOT NULL NUMBER(20)
 AMT_DEDUCTED                                       NUMBER(20)
 AVAILABLE_BALANCE                                  NUMBER(20)

SQL> desc passengers
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 NAME                                               VARCHAR2(20)
 CONTACT                                            NUMBER(10)
 P_ID                                      NOT NULL NUMBER(20)
 MAIL_ID                                            VARCHAR2(20)

SQL>
```

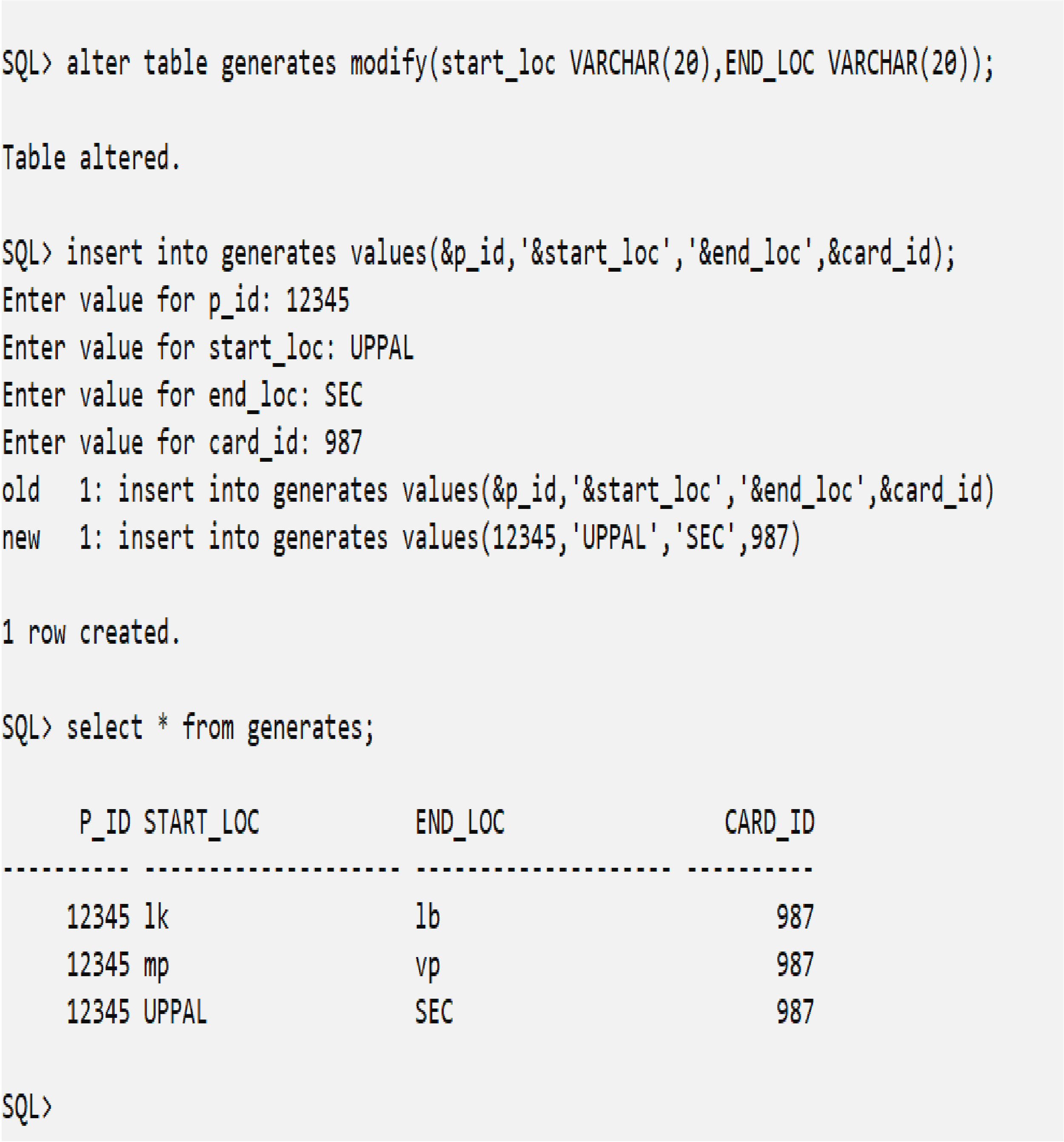


```
SQL> desc reserves
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 SUBSCRIPTION_STATUS                                VARCHAR2(20)
 CARD_ID                                            NUMBER(20)
 P_ID                                               NUMBER(20)

SQL> desc metro_card
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 CARD_ID                                   NOT NULL NUMBER(20)
 VALIDITY                                           DATE
 BALANCE                                            NUMBER(20)


SQL>
```

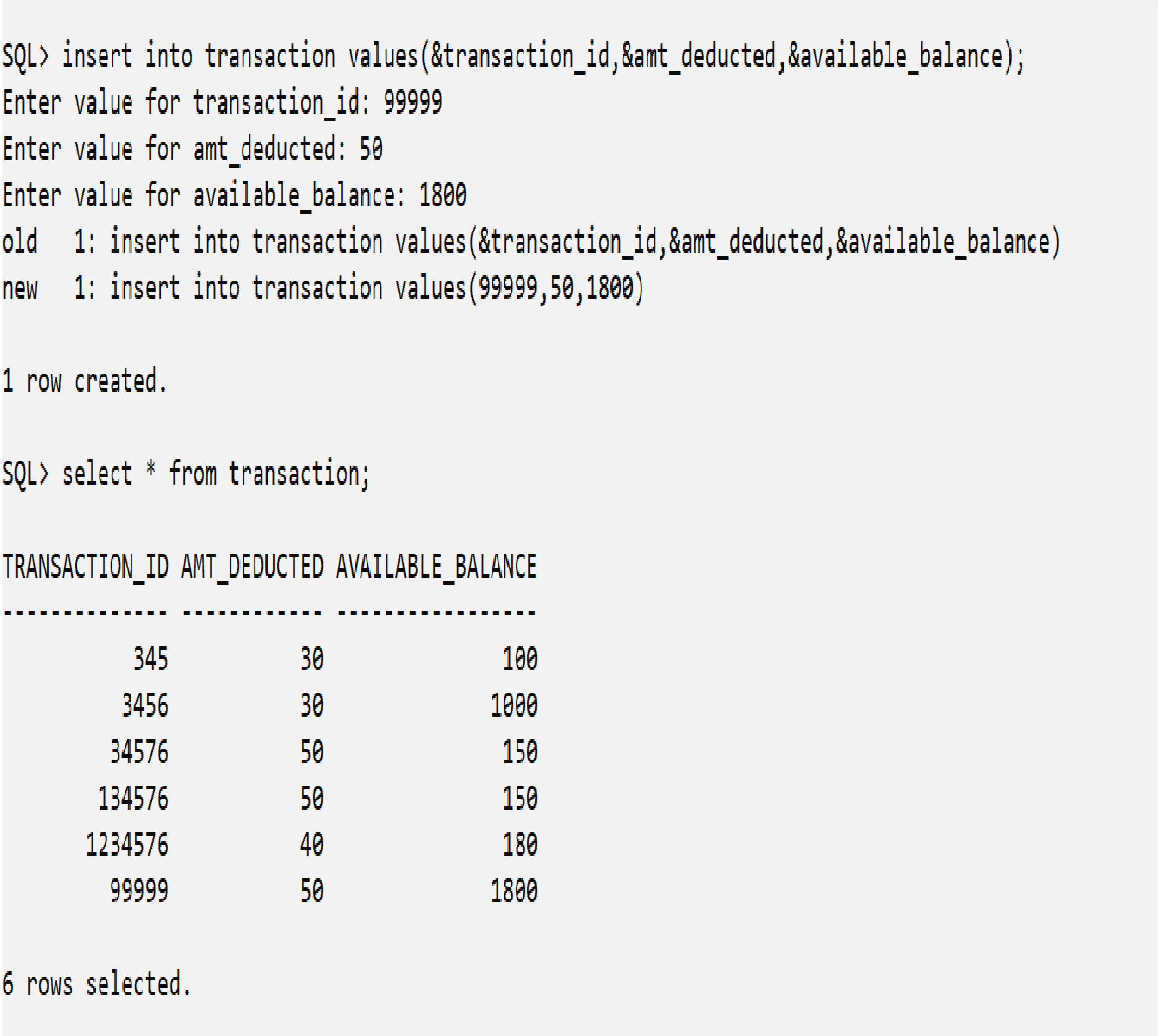
DML COMMANDS



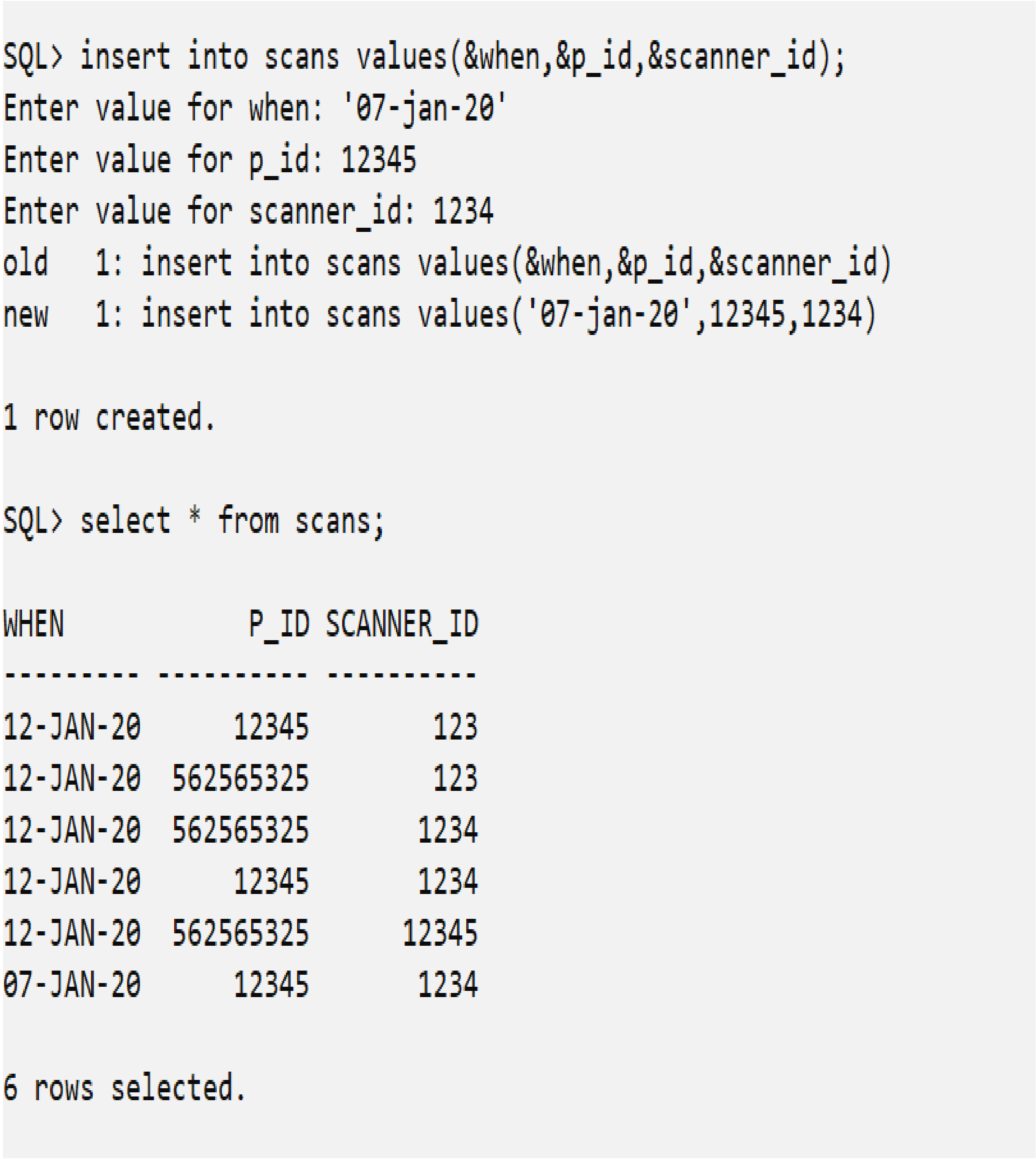
```
SQL> insert into biometric_scanners values(&scanner_id,'&name',&cost,&accuracy);
Enter value for scanner_id: 1602
Enter value for name: ace
Enter value for cost: 25000
Enter value for accuracy: 99
old   1: insert into biometric_scanners values(&scanner_id,'&name',&cost,&accuracy)
new   1: insert into biometric_scanners values(1602,'ace',25000,99)

1 row created.

SQL> select * from biometric_scanners;

SCANNER_ID NAME                       COST   ACCURACY
---------- -------------------- ---------- ----------
      1234 xyz                       10000         90
     12345 iorta                     10000         95
       123 crimson                   10000         99
    123446 bio                       10000         98
   1234476 abc                       10000         98
      1602 ace                       25000         99

6 rows selected.

SQL>
```



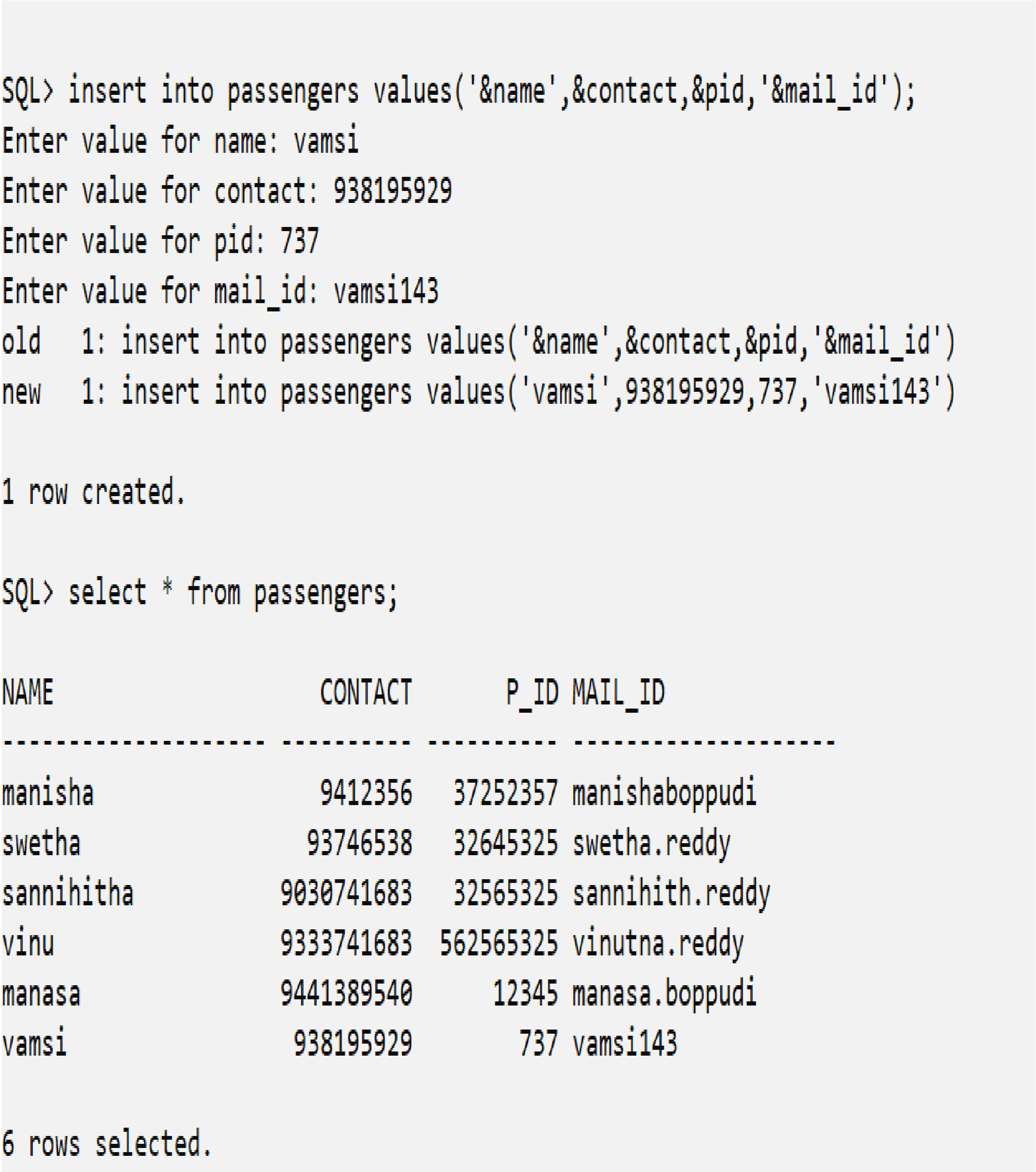
```
SQL> alter table generates modify(start_loc VARCHAR(20),END_LOC VARCHAR(20));

Table altered.

SQL> insert into generates values(&p_id,'&start_loc','&end_loc',&card_id);
Enter value for p_id: 12345
Enter value for start_loc: UPPAL
Enter value for end_loc: SEC
Enter value for card_id: 987
old   1: insert into generates values(&p_id,'&start_loc','&end_loc',&card_id)
new   1: insert into generates values(12345,'UPPAL','SEC',987)

1 row created.

SQL> select * from generates;

      P_ID START_LOC            END_LOC                 CARD_ID
---------- -------------------- -------------------- ----------
     12345 lk                   lb                          987
     12345 mp                   vp                          987
     12345 UPPAL                SEC                         987

SQL>
```



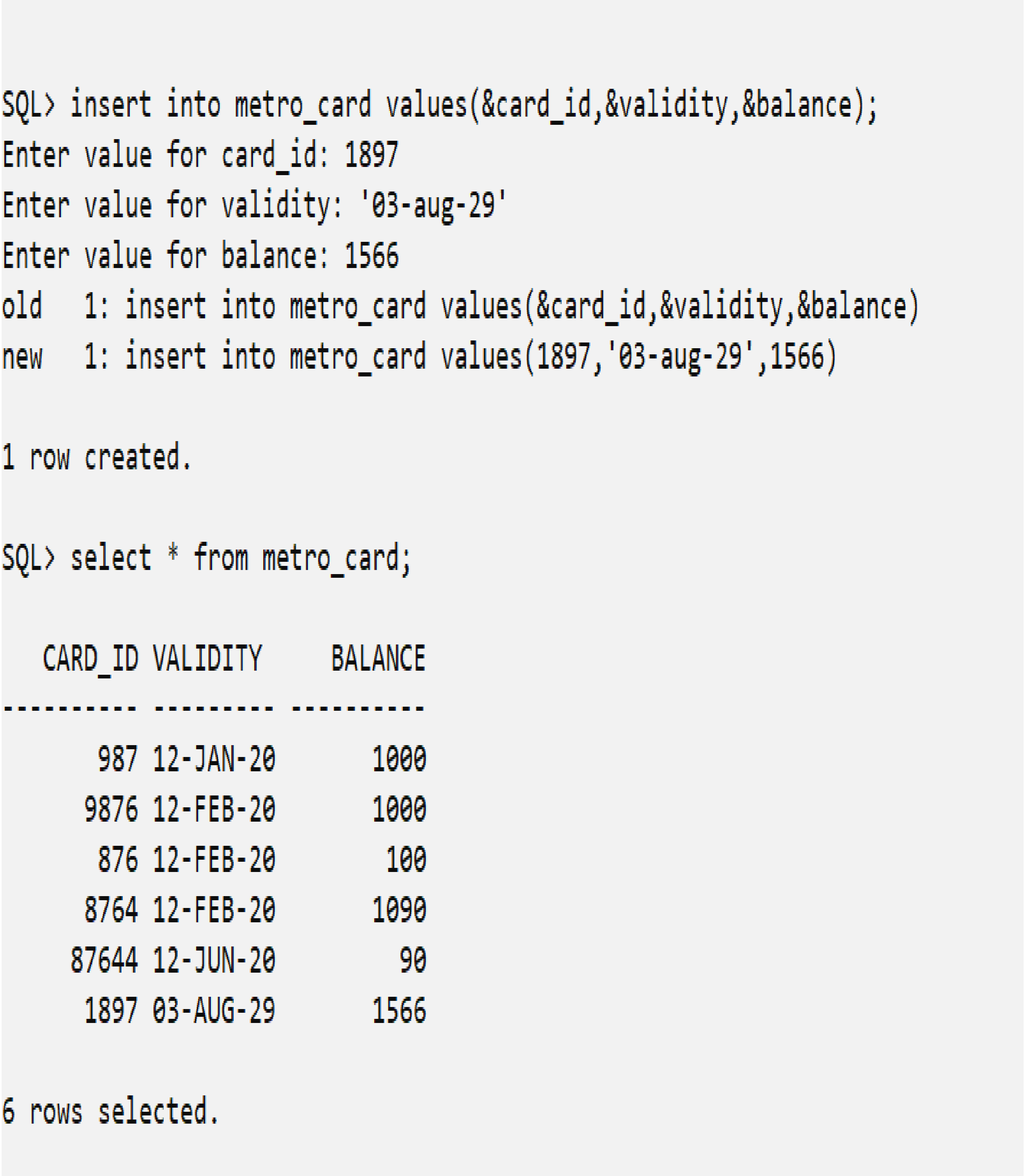
```
SQL> insert into transaction values(&transaction_id,&amt_deducted,&available_balance);
Enter value for transaction_id: 99999
Enter value for amt_deducted: 50
Enter value for available_balance: 1800
old   1: insert into transaction values(&transaction_id,&amt_deducted,&available_balance)
new   1: insert into transaction values(99999,50,1800)

1 row created.

SQL> select * from transaction;

TRANSACTION_ID AMT_DEDUCTED AVAILABLE_BALANCE
-------------- ------------ -----------------
           345           30               100
          3456           30              1000
         34576           50               150
        134576           50               150
       1234576           40               180
         99999           50              1800

6 rows selected.
```



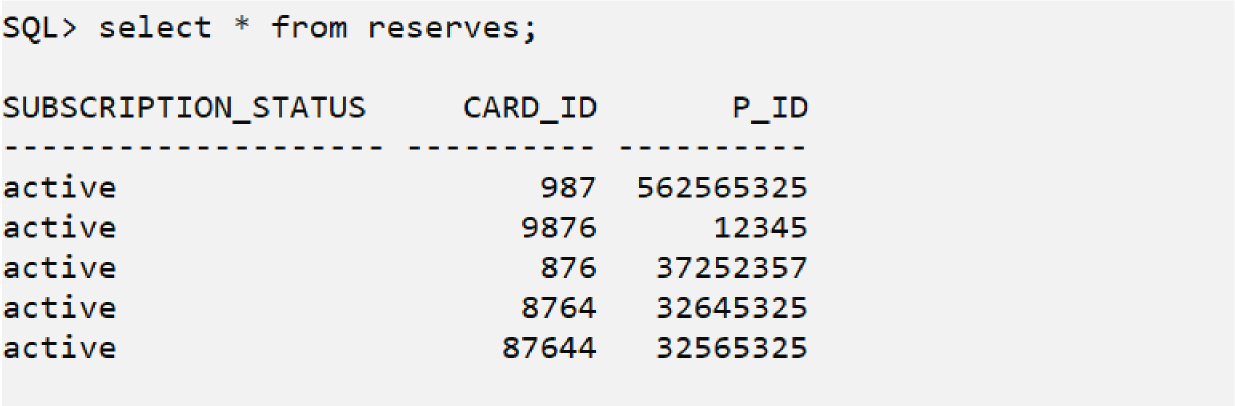
```
SQL> insert into scans values(&when,&p_id,&scanner_id);
Enter value for when: '07-jan-20'
Enter value for p_id: 12345
Enter value for scanner_id: 1234
old   1: insert into scans values(&when,&p_id,&scanner_id)
new   1: insert into scans values('07-jan-20',12345,1234)

1 row created.

SQL> select * from scans;

WHEN            P_ID SCANNER_ID
--------- ---------- ----------
12-JAN-20      12345        123
12-JAN-20  562565325        123
12-JAN-20  562565325       1234
12-JAN-20      12345       1234
12-JAN-20  562565325      12345
07-JAN-20      12345       1234

6 rows selected.
```



```
SQL> insert into passengers values('&name',&contact,&pid,'&mail_id');
Enter value for name: vamsi
Enter value for contact: 938195929
Enter value for pid: 737
Enter value for mail_id: vamsi143
old   1: insert into passengers values('&name',&contact,&pid,'&mail_id')
new   1: insert into passengers values('vamsi',938195929,737,'vamsi143')

1 row created.

SQL> select * from passengers;

NAME                    CONTACT       P_ID MAIL_ID
-------------------- ---------- ---------- --------------------
manisha                 9412356   37252357 manishaboppudi
swetha                 93746538   32645325 swetha.reddy
sannihitha           9030741683   32565325 sannihith.reddy
vinu                 9333741683  562565325 vinutna.reddy
manasa               9441389540      12345 manasa.boppudi
vamsi                 938195929        737 vamsi143

6 rows selected.
```



```
SQL> insert into metro_card values(&card_id,&validity,&balance);
Enter value for card_id: 1897
Enter value for validity: '03-aug-29'
Enter value for balance: 1566
old   1: insert into metro_card values(&card_id,&validity,&balance)
new   1: insert into metro_card values(1897,'03-aug-29',1566)

1 row created.

SQL> select * from metro_card;

   CARD_ID VALIDITY     BALANCE
---------- --------- ----------
       987 12-JAN-20       1000
      9876 12-FEB-20       1000
       876 12-FEB-20        100
      8764 12-FEB-20       1090
     87644 12-JUN-20         90
      1897 03-AUG-29       1566

6 rows selected.
```



```
SQL> select * from reserves;

SUBSCRIPTION_STATUS     CARD_ID       P_ID
-------------------- ---------- ----------
active                      987  562565325
active                     9876      12345
active                      876   37252357
active                     8764   32645325
active                    87644   32565325
```