RELATIONAL DATABASE MANAGEMENT SYSTEM

(4330702)

MICRO-PROJECT

AIM: Railway Reservation System: -

SR.NO.	NAME	ENROLLMENT NO.
1	Dhami Parin Rajnibhai	226470307031
2	Diyora Khush Pareshbhai	226470307032
3	Gangani Het Arvindbhai	226470307039

INTRODUCTION:

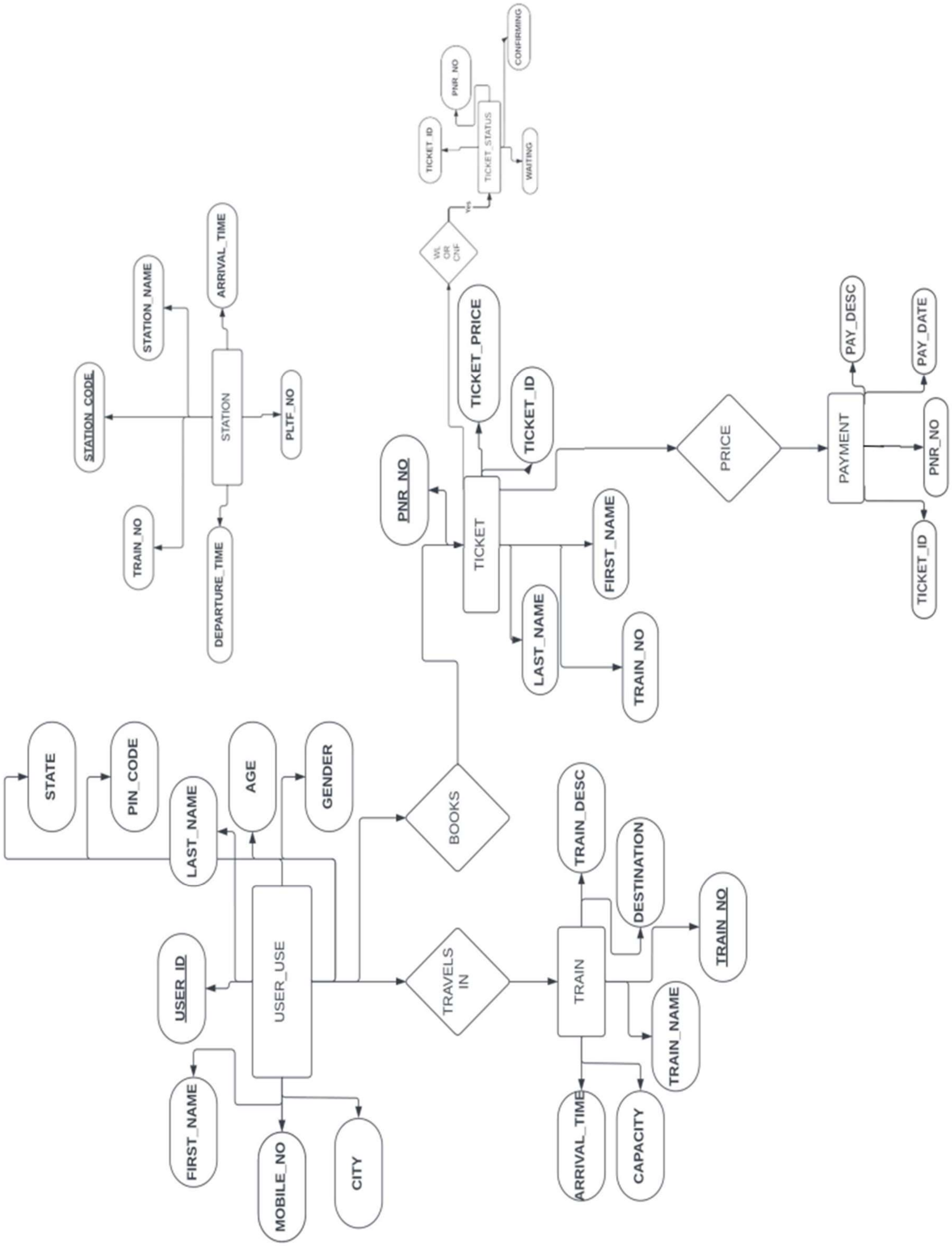
- This project includes the Railway reservation system

- There are 6 table which created in database such as, USER_USE, TRAIN, STATION, TICKET, TICKET_STATUS, PAYMENT

- Several Data are inserted in these tables and Procedural Language SQL is used to perform tasks such as update data on a database or retrieve data from database.

- ORDER-BY, SEQUENCE, VIEW, SUB-QUERY, GROUP-BY, HAVING CLAUSE, AGGREGATE FUNCTION, DML statements, etc. are also performed for manipulation of data and for proper execution of railway reservation system

ENTITY-RELATIONSHIP DIAGRAM

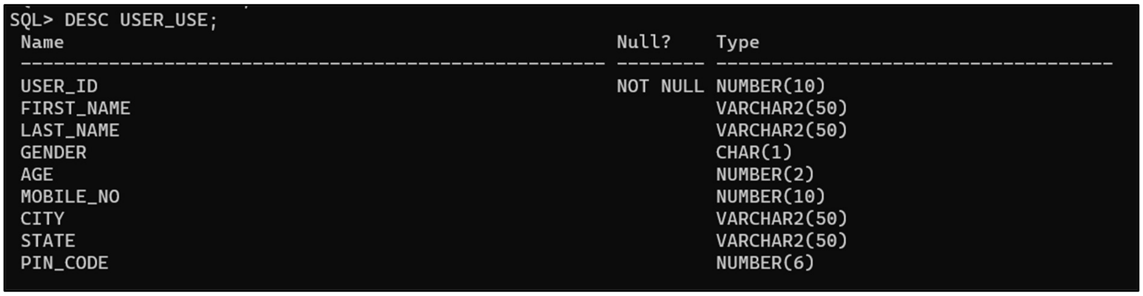


CREATE TABLE QUERY

TABLE 1: USER_USE

CREATE TABLE USER_USE(USER_ID NUMBER(10) PRIMARY KEY, FIRST_NAME VARCHAR2(50), LAST_NAME VARCHAR2(50), GENDER CHAR(1) DEFAULT 'M' CHECK(GENDER IN ('M','F')), AGE NUMBER(2), MOBILE_NO NUMBER(10), CITY VARCHAR2(50), STATE VARCHAR2(50), PIN_CODE NUMBER(6));

OUTPUT:

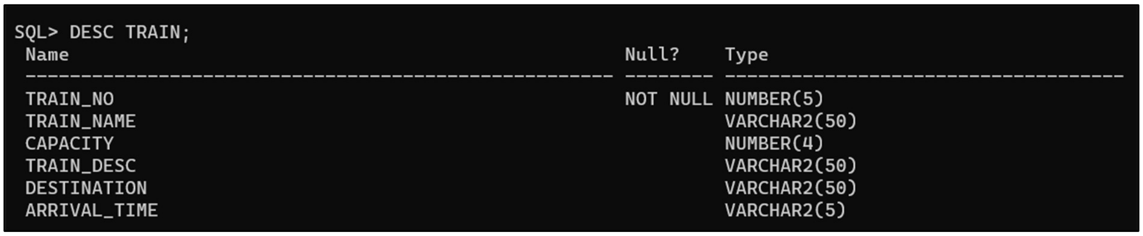


```
SQL> DESC USER_USE;
 Name                                                  Null?    Type
 ----------------------------------------------------- -------- ------------------------------------
 USER_ID                                               NOT NULL NUMBER(10)
 FIRST_NAME                                                     VARCHAR2(50)
 LAST_NAME                                                      VARCHAR2(50)
 GENDER                                                         CHAR(1)
 AGE                                                            NUMBER(2)
 MOBILE_NO                                                      NUMBER(10)
 CITY                                                           VARCHAR2(50)
 STATE                                                          VARCHAR2(50)
 PIN_CODE                                                       NUMBER(6)
```

TABLE 2: TRAIN

CREATE TABLE TRAIN(TRAIN_NO NUMBER(5) PRIMARY KEY, TRAIN_NAME VARCHAR2(50), CAPACITY NUMBER(4), TRAIN_DESC VARCHAR2(50), DESTINATION VARCHAR2(50), ARRIVAL_TIME VARCHAR2(5));

OUTPUT:

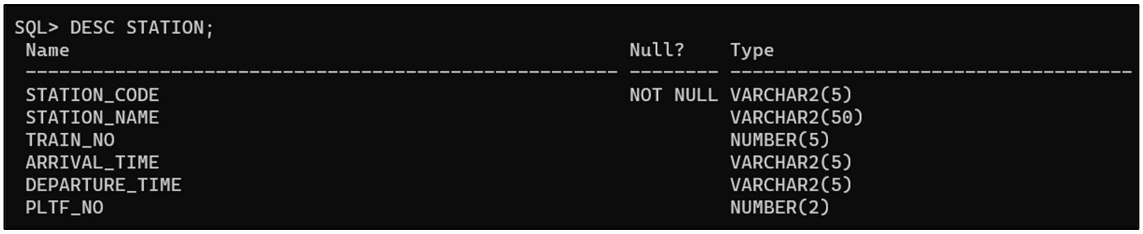


```
SQL> DESC TRAIN;
 Name                                                  Null?    Type
 ----------------------------------------------------- -------- ------------------------------------
 TRAIN_NO                                              NOT NULL NUMBER(5)
 TRAIN_NAME                                                     VARCHAR2(50)
 CAPACITY                                                       NUMBER(4)
 TRAIN_DESC                                                     VARCHAR2(50)
 DESTINATION                                                    VARCHAR2(50)
 ARRIVAL_TIME                                                   VARCHAR2(5)
```

TABLE 3: STATION

CREATE TABLE STATION (STATION_CODE VARCHAR2(5), STATION_NAME VARCHAR2(50), TRAIN_NO NUMBER(5) REFERENCES TRAIN(TRAIN_NO), ARRIVAL_TIME VARCHAR2(5), DEPARTURE_TIME VARCHAR2(5), PLTF_NO VARCHAR2(5));

OUTPUT:

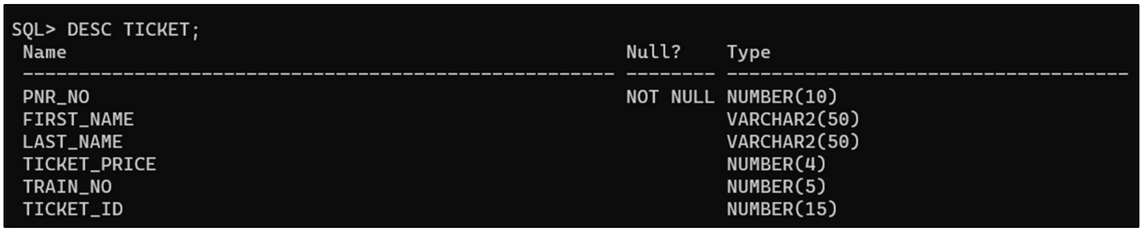


```
SQL> DESC STATION;
 Name                                                  Null?    Type
 ----------------------------------------------------- -------- ------------------------------------
 STATION_CODE                                          NOT NULL VARCHAR2(5)
 STATION_NAME                                                   VARCHAR2(50)
 TRAIN_NO                                                       NUMBER(5)
 ARRIVAL_TIME                                                   VARCHAR2(5)
 DEPARTURE_TIME                                                 VARCHAR2(5)
 PLTF_NO                                                        NUMBER(2)
```

TABLE 4: TICKET

CREATE TABLE TICKET (PNR_NO NUMBER(10) PRIMARY KEY, FIRST_NAME VARCHAR2(50), LAST_NAME VARCHAR2(50), TICKET_PRICE NUMBER(4), TRAIN_NO NUMBER(5) REFERENCES TRAIN(TRAIN_NO), TICKET_ID NUMBER(15));

OUTPUT:

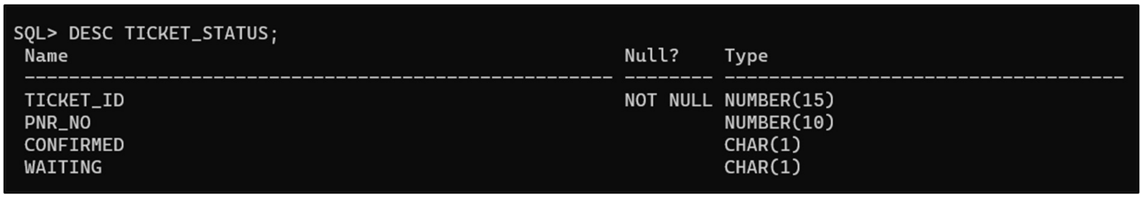


```
SQL> DESC TICKET;
 Name                                                  Null?    Type
 ----------------------------------------------------- -------- ------------------------------------
 PNR_NO                                                NOT NULL NUMBER(10)
 FIRST_NAME                                                     VARCHAR2(50)
 LAST_NAME                                                      VARCHAR2(50)
 TICKET_PRICE                                                   NUMBER(4)
 TRAIN_NO                                                       NUMBER(5)
 TICKET_ID                                                      NUMBER(15)
```

TABLE 5: TICKET_STATUS

CREATE TABLE TICKET_STATUS (TICKET_ID NUMBER(15) PRIMARY KEY, PNR_NO NUMBER(10) REFERENCES TICKET(PNR_NO), CONFIRMED CHAR(1) DEFAULT 'Y' CHECK(CONFIRMED IN ('Y','N')), WAITING CHAR(1) DEFAULT 'Y' CHECK(WAITING IN ('Y','N')));

OUTPUT:

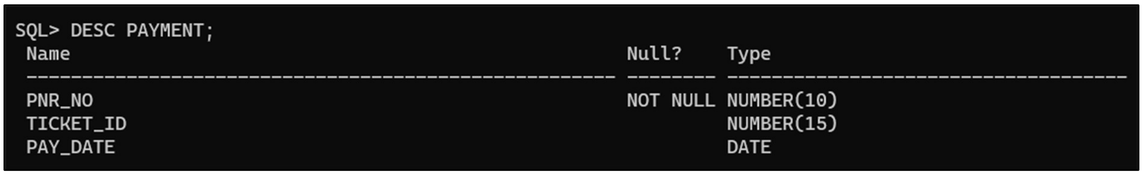


```
SQL> DESC TICKET_STATUS;
 Name                                                  Null?    Type
 ----------------------------------------------------- -------- ------------------------------------
 TICKET_ID                                             NOT NULL NUMBER(15)
 PNR_NO                                                         NUMBER(10)
 CONFIRMED                                                      CHAR(1)
 WAITING                                                        CHAR(1)
```

TABLE 6: PAYMENT

CREATE TABLE PAYMENT (PNR_NO NUMBER(10) PRIMARY KEY, TICKET_ID NUMBER(15) REFERENCES TICKET_STATUS(TICKET_ID), PAY_DATE DATE);

OUTPUT:



```
SQL> DESC PAYMENT;
 Name                                                  Null?    Type
 ----------------------------------------------------- -------- ------------------------------------
 PNR_NO                                                NOT NULL NUMBER(10)
 TICKET_ID                                                      NUMBER(15)
 PAY_DATE                                                       DATE
```

INSERT INTO QUERY

INSERT INTO TABLE 1: USER_USE

INSERT INTO USER_USE (USER_ID, FIRST_NAME, LAST_NAME, GENDER, AGE, MOBILE_NO, CITY, STATE, PIN_CODE) VALUES(1000000001, 'Parin', 'Dhami', 'M', 17, '7600498107', 'Ahmedabad', 'Gujarat', '380001');

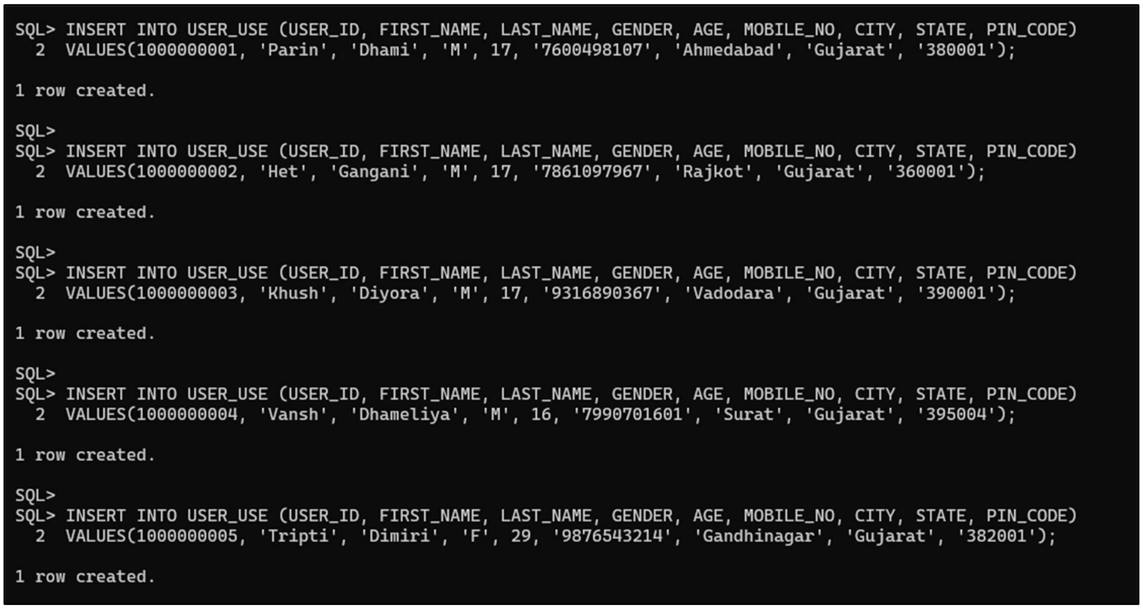
INSERT INTO USER_USE (USER_ID, FIRST_NAME, LAST_NAME, GENDER, AGE, MOBILE_NO, CITY, STATE, PIN_CODE) VALUES(1000000002, 'Het', 'Gangani', 'M', 17, '7861097967', 'Rajkot', 'Gujarat', '360001');

INSERT INTO USER_USE (USER_ID, FIRST_NAME, LAST_NAME, GENDER, AGE, MOBILE_NO, CITY, STATE, PIN_CODE) VALUES(1000000003, 'Khush', 'Diyora', 'M', 17, '9316890367', 'Vadodara', 'Gujarat', '390001');

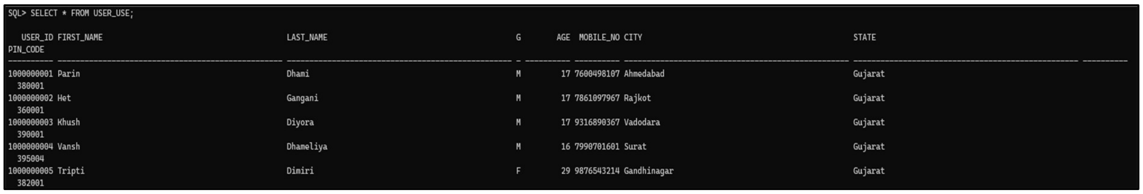
INSERT INTO USER_USE (USER_ID, FIRST_NAME, LAST_NAME, GENDER, AGE, MOBILE_NO, CITY, STATE, PIN_CODE) VALUES(1000000004, 'Vansh', 'Dhameliya', 'M', 16, '7990701601', 'Surat', 'Gujarat', '395004');

INSERT INTO USER_USE (USER_ID, FIRST_NAME, LAST_NAME, GENDER, AGE, MOBILE_NO, CITY, STATE, PIN_CODE) VALUES(1000000005, 'Tripti', 'Dimiri', 'F', 29, '9876543214', 'Gandhinagar', 'Gujarat', '382001');

OUTPUT:



```
SQL> INSERT INTO USER_USE (USER_ID, FIRST_NAME, LAST_NAME, GENDER, AGE, MOBILE_NO, CITY, STATE, PIN_CODE)
  2  VALUES(1000000001, 'Parin', 'Dhami', 'M', 17, '7600498107', 'Ahmedabad', 'Gujarat', '380001');

1 row created.

SQL>
SQL> INSERT INTO USER_USE (USER_ID, FIRST_NAME, LAST_NAME, GENDER, AGE, MOBILE_NO, CITY, STATE, PIN_CODE)
  2  VALUES(1000000002, 'Het', 'Gangani', 'M', 17, '7861097967', 'Rajkot', 'Gujarat', '360001');

1 row created.

SQL>
SQL> INSERT INTO USER_USE (USER_ID, FIRST_NAME, LAST_NAME, GENDER, AGE, MOBILE_NO, CITY, STATE, PIN_CODE)
  2  VALUES(1000000003, 'Khush', 'Diyora', 'M', 17, '9316890367', 'Vadodara', 'Gujarat', '390001');

1 row created.

SQL>
SQL> INSERT INTO USER_USE (USER_ID, FIRST_NAME, LAST_NAME, GENDER, AGE, MOBILE_NO, CITY, STATE, PIN_CODE)
  2  VALUES(1000000004, 'Vansh', 'Dhameliya', 'M', 16, '7990701601', 'Surat', 'Gujarat', '395004');

1 row created.

SQL>
SQL> INSERT INTO USER_USE (USER_ID, FIRST_NAME, LAST_NAME, GENDER, AGE, MOBILE_NO, CITY, STATE, PIN_CODE)
  2  VALUES(1000000005, 'Tripti', 'Dimiri', 'F', 29, '9876543214', 'Gandhinagar', 'Gujarat', '382001');

1 row created.
```



```
SQL> SELECT * FROM USER_USE;

   USER_ID FIRST_NAME                                         LAST_NAME                                          G        AGE MOBILE_NO  CITY                                               STATE
PIN_CODE
---------- -------------------------------------------------- -------------------------------------------------- - ---------- ---------- -------------------------------------------------- -------------------------------------------------- ----------
1000000001 Parin                                              Dhami                                              M         17 7600498107 Ahmedabad                                          Gujarat
  380001
1000000002 Het                                                Gangani                                            M         17 7861097967 Rajkot                                             Gujarat
  360001
1000000003 Khush                                              Diyora                                             M         17 9316890367 Vadodara                                           Gujarat
  390001
1000000004 Vansh                                              Dhameliya                                          M         16 7990701601 Surat                                              Gujarat
  395004
1000000005 Tripti                                             Dimiri                                             F         29 9876543214 Gandhinagar                                        Gujarat
  382001
```

INSERT INTO TABLE 2: TRAIN

INSERT INTO TRAIN (TRAIN_NO, TRAIN_NAME, CAPACITY, TRAIN_DESC, DESTINATION, ARRIVAL_TIME)VALUES(20901, 'MMCT - GNC Vande Bharat Express', 1128, 'AC Chair Car', 'Gandhinagar Capital', '12:25');

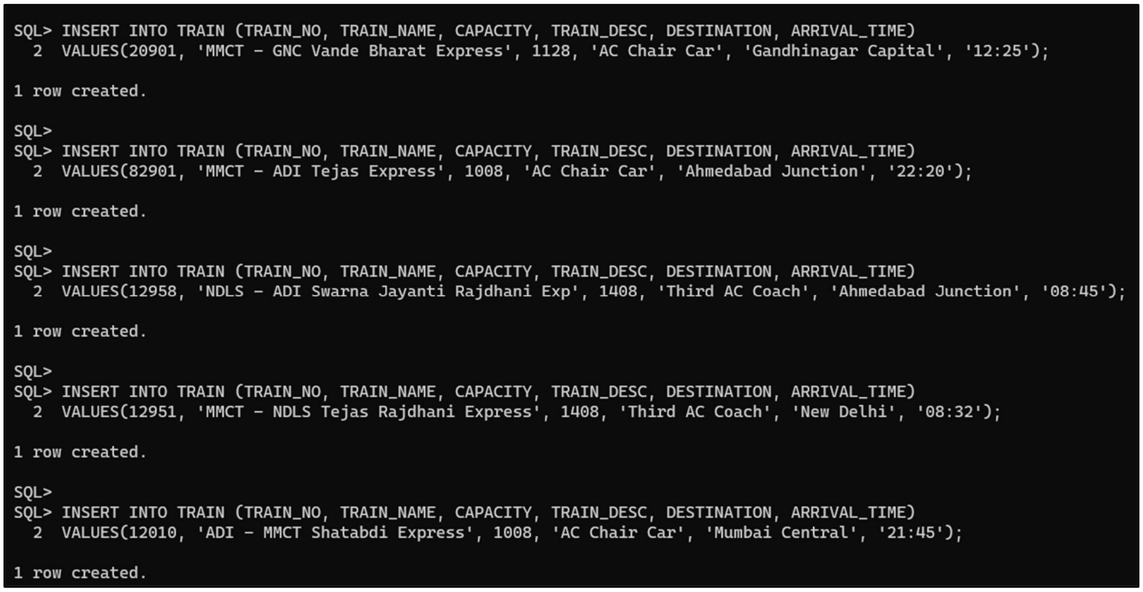
INSERT INTO TRAIN (TRAIN_NO, TRAIN_NAME, CAPACITY, TRAIN_DESC, DESTINATION, ARRIVAL_TIME)VALUES(82901, 'MMCT - ADI Tejas Express', 1008, 'AC Chair Car', 'Ahmedabad Junction', '22:20');

INSERT INTO TRAIN (TRAIN_NO, TRAIN_NAME, CAPACITY, TRAIN_DESC, DESTINATION, ARRIVAL_TIME)VALUES(12958, 'NDLS - ADI Swarna Jayanti Rajdhani Exp', 1408, 'Third AC Coach', 'Ahmedabad Junction', '08:45');

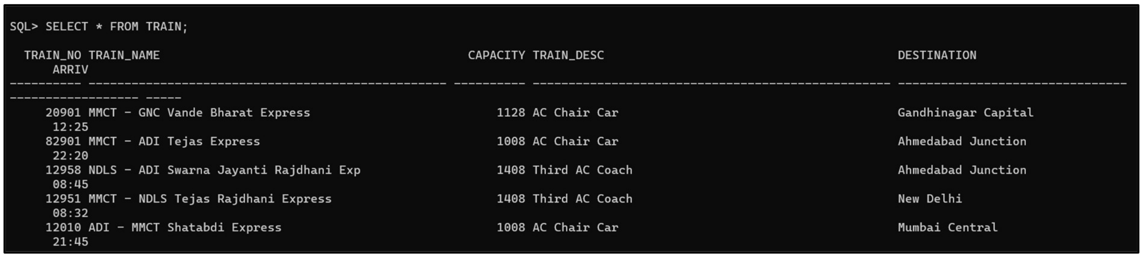
INSERT INTO TRAIN (TRAIN_NO, TRAIN_NAME, CAPACITY, TRAIN_DESC, DESTINATION, ARRIVAL_TIME)VALUES(12951, 'MMCT - NDLS Tejas Rajdhani Express', 1408, 'Third AC Coach', 'New Delhi', '08:32');

INSERT INTO TRAIN (TRAIN_NO, TRAIN_NAME, CAPACITY, TRAIN_DESC, DESTINATION, ARRIVAL_TIME)VALUES(12010, 'ADI - MMCT Shatabdi Express', 1008, 'AC Chair Car', 'Mumbai Central', '21:45');

OUTPUT:



```
SQL> INSERT INTO TRAIN (TRAIN_NO, TRAIN_NAME, CAPACITY, TRAIN_DESC, DESTINATION, ARRIVAL_TIME)
  2  VALUES(20901, 'MMCT - GNC Vande Bharat Express', 1128, 'AC Chair Car', 'Gandhinagar Capital', '12:25');

1 row created.

SQL>
SQL> INSERT INTO TRAIN (TRAIN_NO, TRAIN_NAME, CAPACITY, TRAIN_DESC, DESTINATION, ARRIVAL_TIME)
  2  VALUES(82901, 'MMCT - ADI Tejas Express', 1008, 'AC Chair Car', 'Ahmedabad Junction', '22:20');

1 row created.

SQL>
SQL> INSERT INTO TRAIN (TRAIN_NO, TRAIN_NAME, CAPACITY, TRAIN_DESC, DESTINATION, ARRIVAL_TIME)
  2  VALUES(12958, 'NDLS - ADI Swarna Jayanti Rajdhani Exp', 1408, 'Third AC Coach', 'Ahmedabad Junction', '08:45');

1 row created.

SQL>
SQL> INSERT INTO TRAIN (TRAIN_NO, TRAIN_NAME, CAPACITY, TRAIN_DESC, DESTINATION, ARRIVAL_TIME)
  2  VALUES(12951, 'MMCT - NDLS Tejas Rajdhani Express', 1408, 'Third AC Coach', 'New Delhi', '08:32');

1 row created.

SQL>
SQL> INSERT INTO TRAIN (TRAIN_NO, TRAIN_NAME, CAPACITY, TRAIN_DESC, DESTINATION, ARRIVAL_TIME)
  2  VALUES(12010, 'ADI - MMCT Shatabdi Express', 1008, 'AC Chair Car', 'Mumbai Central', '21:45');

1 row created.
```



```
SQL> SELECT * FROM TRAIN;

  TRAIN_NO TRAIN_NAME                                           CAPACITY TRAIN_DESC                                        DESTINATION
      ARRIV
---------- -------------------------------------------------- ---------- -------------------------------------------------- --------------------------------
------------------ -----
     20901 MMCT - GNC Vande Bharat Express                          1128 AC Chair Car                                      Gandhinagar Capital
      12:25
     82901 MMCT - ADI Tejas Express                                 1008 AC Chair Car                                      Ahmedabad Junction
      22:20
     12958 NDLS - ADI Swarna Jayanti Rajdhani Exp                   1408 Third AC Coach                                    Ahmedabad Junction
      08:45
     12951 MMCT - NDLS Tejas Rajdhani Express                       1408 Third AC Coach                                    New Delhi
      08:32
     12010 ADI - MMCT Shatabdi Express                              1008 AC Chair Car                                      Mumbai Central
      21:45
```

INSERT INTO TABLE 3: STATION

INSERT INTO STATION(STATION_CODE, STATION_NAME, TRAIN_NO, ARRIVAL_TIME, DEPARTURE_TIME, PLTF_NO)VALUES('MMCT', 'Mumbai Central', 20901, '06:10', '06:15', '1');

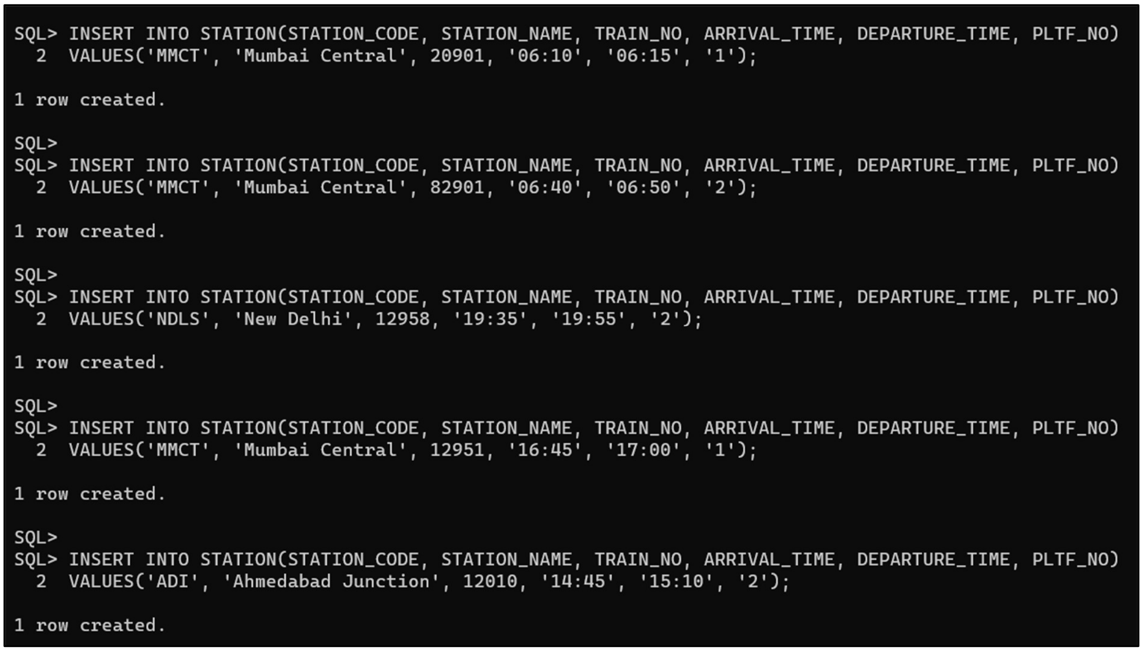
INSERT INTO STATION(STATION_CODE, STATION_NAME, TRAIN_NO, ARRIVAL_TIME, DEPARTURE_TIME, PLTF_NO)VALUES('MMCT', 'Mumbai Central', 82901, '06:40', '06:50', '2');

INSERT INTO STATION(STATION_CODE, STATION_NAME, TRAIN_NO, ARRIVAL_TIME, DEPARTURE_TIME, PLTF_NO)VALUES('NDLS', 'New Delhi', 12958, '19:35', '19:55', '2');

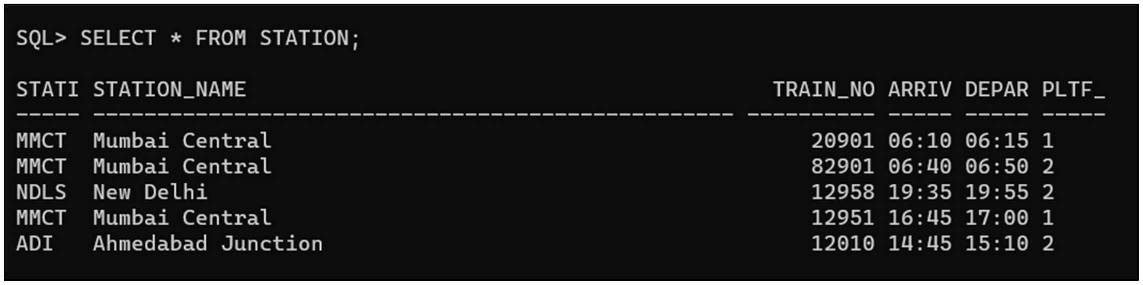
INSERT INTO STATION(STATION_CODE, STATION_NAME, TRAIN_NO, ARRIVAL_TIME, DEPARTURE_TIME, PLTF_NO)VALUES('MMCT', 'Mumbai Central', 12951, '16:45', '17:00', '1');

INSERT INTO STATION(STATION_CODE, STATION_NAME, TRAIN_NO, ARRIVAL_TIME, DEPARTURE_TIME, PLTF_NO)VALUES('ADI', 'Ahmedabad Junction', 12010, '14:45', '15:10', '2');

OUTPUT:



```
SQL> INSERT INTO STATION(STATION_CODE, STATION_NAME, TRAIN_NO, ARRIVAL_TIME, DEPARTURE_TIME, PLTF_NO)
  2  VALUES('MMCT', 'Mumbai Central', 20901, '06:10', '06:15', '1');

1 row created.

SQL>
SQL> INSERT INTO STATION(STATION_CODE, STATION_NAME, TRAIN_NO, ARRIVAL_TIME, DEPARTURE_TIME, PLTF_NO)
  2  VALUES('MMCT', 'Mumbai Central', 82901, '06:40', '06:50', '2');

1 row created.

SQL>
SQL> INSERT INTO STATION(STATION_CODE, STATION_NAME, TRAIN_NO, ARRIVAL_TIME, DEPARTURE_TIME, PLTF_NO)
  2  VALUES('NDLS', 'New Delhi', 12958, '19:35', '19:55', '2');

1 row created.

SQL>
SQL> INSERT INTO STATION(STATION_CODE, STATION_NAME, TRAIN_NO, ARRIVAL_TIME, DEPARTURE_TIME, PLTF_NO)
  2  VALUES('MMCT', 'Mumbai Central', 12951, '16:45', '17:00', '1');

1 row created.

SQL>
SQL> INSERT INTO STATION(STATION_CODE, STATION_NAME, TRAIN_NO, ARRIVAL_TIME, DEPARTURE_TIME, PLTF_NO)
  2  VALUES('ADI', 'Ahmedabad Junction', 12010, '14:45', '15:10', '2');

1 row created.
```



```
SQL> SELECT * FROM STATION;

STATI STATION_NAME                                         TRAIN_NO ARRIV DEPAR PLTF_
----- -------------------------------------------------- ---------- ----- ----- -----
MMCT  Mumbai Central                                          20901 06:10 06:15 1
MMCT  Mumbai Central                                          82901 06:40 06:50 2
NDLS  New Delhi                                               12958 19:35 19:55 2
MMCT  Mumbai Central                                          12951 16:45 17:00 1
ADI   Ahmedabad Junction                                      12010 14:45 15:10 2
```

INSERT INTO TABLE 4: TICKET

INSERT INTO TICKET (PNR_NO, FIRST_NAME, LAST_NAME, TICKET_PRICE, TRAIN_NO, TICKET_ID)VALUES('8836628295', 'Parin', 'Dhami', '1255', '20901', '200000932220827');

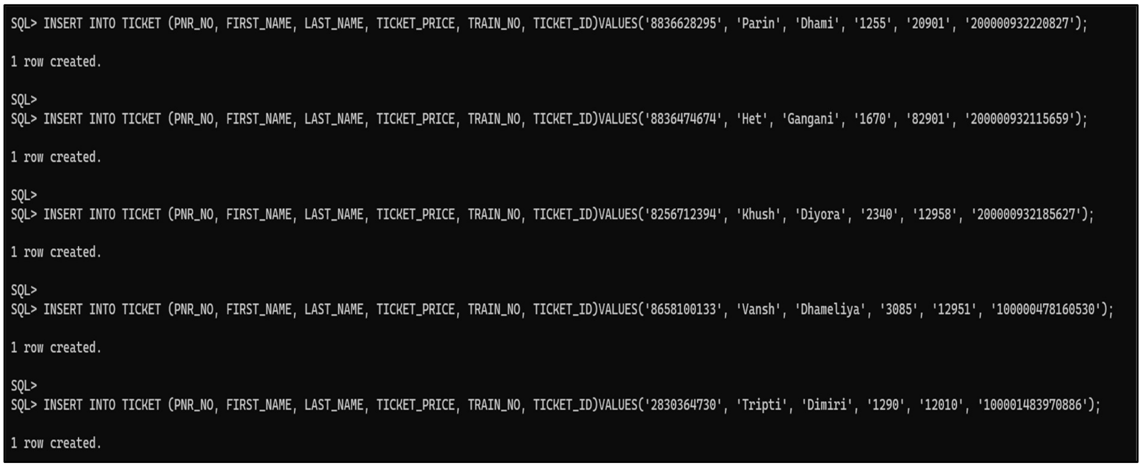
INSERT INTO TICKET (PNR_NO, FIRST_NAME, LAST_NAME, TICKET_PRICE, TRAIN_NO, TICKET_ID)VALUES('8836474674', 'Het', 'Gangani', '1670', '82901', '200000932115659');

INSERT INTO TICKET (PNR_NO, FIRST_NAME, LAST_NAME, TICKET_PRICE, TRAIN_NO, TICKET_ID)VALUES('8256712394', 'Khush', 'Diyora', '2340', '12958', '200000932185627');

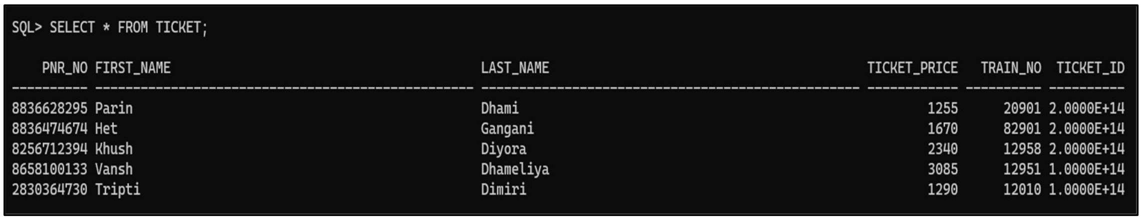
INSERT INTO TICKET (PNR_NO, FIRST_NAME, LAST_NAME, TICKET_PRICE, TRAIN_NO, TICKET_ID)VALUES('8658100133', 'Vansh', 'Dhameliya', '3085', '12951', '100000478160530');

INSERT INTO TICKET (PNR_NO, FIRST_NAME, LAST_NAME, TICKET_PRICE, TRAIN_NO, TICKET_ID)VALUES('2830364730', 'Tripti', 'Dimiri', '1290', '12010', '100001483970886');

OUTPUT:



```
SQL> INSERT INTO TICKET (PNR_NO, FIRST_NAME, LAST_NAME, TICKET_PRICE, TRAIN_NO, TICKET_ID)VALUES('8836628295', 'Parin', 'Dhami', '1255', '20901', '200000932220827');

1 row created.

SQL>
SQL> INSERT INTO TICKET (PNR_NO, FIRST_NAME, LAST_NAME, TICKET_PRICE, TRAIN_NO, TICKET_ID)VALUES('8836474674', 'Het', 'Gangani', '1670', '82901', '200000932115659');

1 row created.

SQL>
SQL> INSERT INTO TICKET (PNR_NO, FIRST_NAME, LAST_NAME, TICKET_PRICE, TRAIN_NO, TICKET_ID)VALUES('8256712394', 'Khush', 'Diyora', '2340', '12958', '200000932185627');

1 row created.

SQL>
SQL> INSERT INTO TICKET (PNR_NO, FIRST_NAME, LAST_NAME, TICKET_PRICE, TRAIN_NO, TICKET_ID)VALUES('8658100133', 'Vansh', 'Dhameliya', '3085', '12951', '100000478160530');

1 row created.

SQL>
SQL> INSERT INTO TICKET (PNR_NO, FIRST_NAME, LAST_NAME, TICKET_PRICE, TRAIN_NO, TICKET_ID)VALUES('2830364730', 'Tripti', 'Dimiri', '1290', '12010', '100001483970886');

1 row created.
```



```
SQL> SELECT * FROM TICKET;

    PNR_NO FIRST_NAME                                         LAST_NAME                                          TICKET_PRICE   TRAIN_NO  TICKET_ID
---------- -------------------------------------------------- -------------------------------------------------- ------------ ---------- ----------
8836628295 Parin                                              Dhami                                                      1255      20901 2.0000E+14
8836474674 Het                                                Gangani                                                    1670      82901 2.0000E+14
8256712394 Khush                                              Diyora                                                     2340      12958 2.0000E+14
8658100133 Vansh                                              Dhameliya                                                  3085      12951 1.0000E+14
2830364730 Tripti                                             Dimiri                                                     1290      12010 1.0000E+14
```

INSERT INTO TABLE 5: TICKET_STATUS

INSERT INTO TICKET_STATUS (TICKET_ID, PNR_NO, CONFIRMED, WAITING)VALUES('200000932220827', '8836628295', 'Y', 'N');

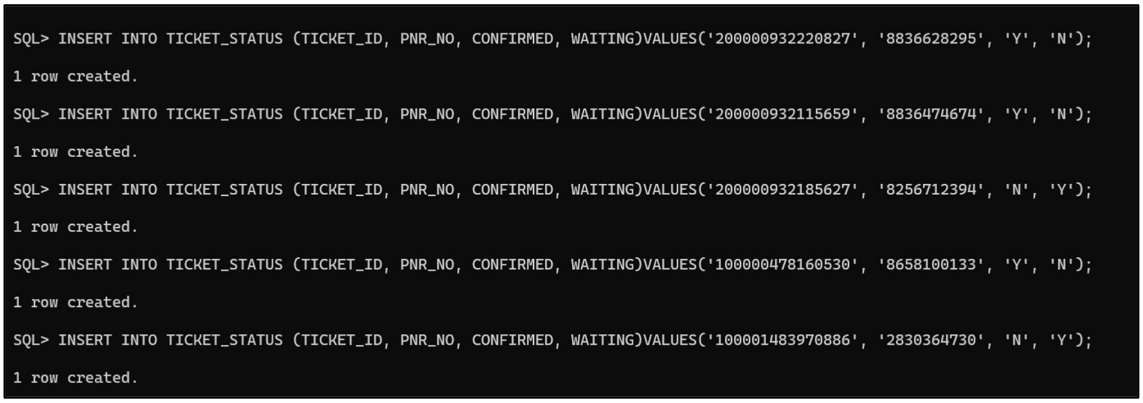
INSERT INTO TICKET_STATUS (TICKET_ID, PNR_NO, CONFIRMED, WAITING)VALUES('200000932115659', '8836474674', 'Y', 'N');

INSERT INTO TICKET_STATUS (TICKET_ID, PNR_NO, CONFIRMED, WAITING)VALUES('200000932185627', '8256712394', 'N', 'Y');

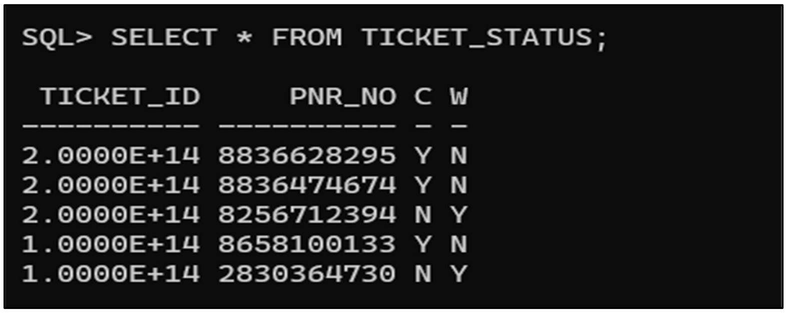
INSERT INTO TICKET_STATUS (TICKET_ID, PNR_NO, CONFIRMED, WAITING)VALUES('100000478160530', '8658100133', 'Y', 'N');

INSERT INTO TICKET_STATUS (TICKET_ID, PNR_NO, CONFIRMED, WAITING)VALUES('100001483970886', '2830364730', 'N', 'Y');

OUTPUT:



```
SQL> INSERT INTO TICKET_STATUS (TICKET_ID, PNR_NO, CONFIRMED, WAITING)VALUES('200000932220827', '8836628295', 'Y', 'N');

1 row created.

SQL> INSERT INTO TICKET_STATUS (TICKET_ID, PNR_NO, CONFIRMED, WAITING)VALUES('200000932115659', '8836474674', 'Y', 'N');

1 row created.

SQL> INSERT INTO TICKET_STATUS (TICKET_ID, PNR_NO, CONFIRMED, WAITING)VALUES('200000932185627', '8256712394', 'N', 'Y');

1 row created.

SQL> INSERT INTO TICKET_STATUS (TICKET_ID, PNR_NO, CONFIRMED, WAITING)VALUES('100000478160530', '8658100133', 'Y', 'N');

1 row created.

SQL> INSERT INTO TICKET_STATUS (TICKET_ID, PNR_NO, CONFIRMED, WAITING)VALUES('100001483970886', '2830364730', 'N', 'Y');

1 row created.
```



```
SQL> SELECT * FROM TICKET_STATUS;

 TICKET_ID     PNR_NO C W
---------- ---------- - -
2.0000E+14 8836628295 Y N
2.0000E+14 8836474674 Y N
2.0000E+14 8256712394 N Y
1.0000E+14 8658100133 Y N
1.0000E+14 2830364730 N Y
```

INSERT INTO TABLE 6: PAYMENT

INSERT INTO PAYMENT (PNR_NO, TICKET_ID, PAY_DATE)VALUES('8836628295','200000932220827', TO_DATE('20/10/2022', 'DD/MM/YYYY'));

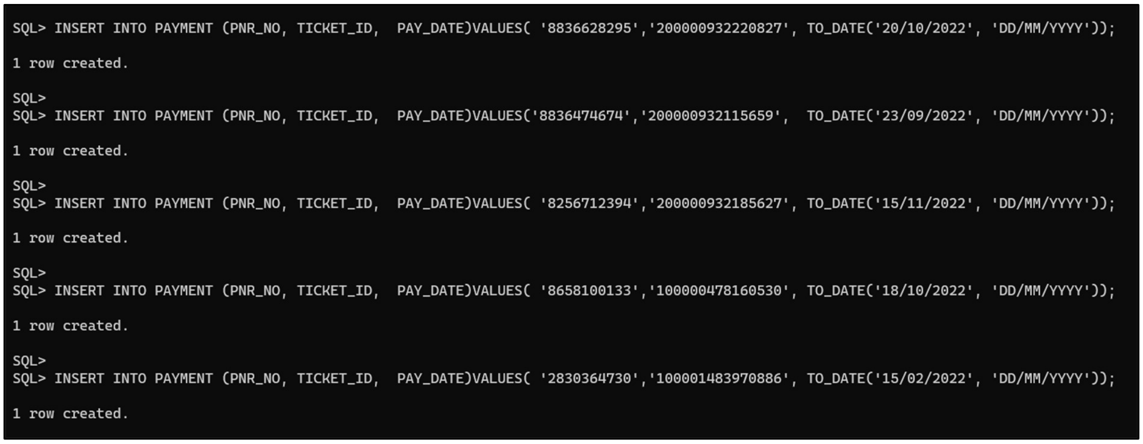
INSERT INTO PAYMENT (PNR_NO, TICKET_ID, PAY_DATE)VALUES('8836474674','200000932115659', TO_DATE('23/09/2022', 'DD/MM/YYYY'));

INSERT INTO PAYMENT (PNR_NO, TICKET_ID, PAY_DATE)VALUES('8256712394','200000932185627', TO_DATE('15/11/2022', 'DD/MM/YYYY'));

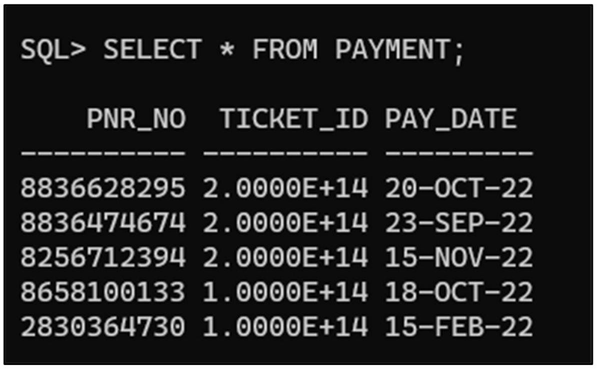
INSERT INTO PAYMENT (PNR_NO, TICKET_ID, PAY_DATE)VALUES('8658100133','100000478160530', TO_DATE('18/10/2022', 'DD/MM/YYYY'));

INSERT INTO PAYMENT (PNR_NO, TICKET_ID, PAY_DATE)VALUES('2830364730','100001483970886', TO_DATE('15/02/2022', 'DD/MM/YYYY'));

OUTPUT:



```
SQL> INSERT INTO PAYMENT (PNR_NO, TICKET_ID,  PAY_DATE)VALUES( '8836628295','200000932220827', TO_DATE('20/10/2022', 'DD/MM/YYYY'));

1 row created.

SQL>
SQL> INSERT INTO PAYMENT (PNR_NO, TICKET_ID,  PAY_DATE)VALUES('8836474674','200000932115659',  TO_DATE('23/09/2022', 'DD/MM/YYYY'));

1 row created.

SQL>
SQL> INSERT INTO PAYMENT (PNR_NO, TICKET_ID,  PAY_DATE)VALUES( '8256712394','200000932185627', TO_DATE('15/11/2022', 'DD/MM/YYYY'));

1 row created.

SQL>
SQL> INSERT INTO PAYMENT (PNR_NO, TICKET_ID,  PAY_DATE)VALUES( '8658100133','100000478160530', TO_DATE('18/10/2022', 'DD/MM/YYYY'));

1 row created.

SQL>
SQL> INSERT INTO PAYMENT (PNR_NO, TICKET_ID,  PAY_DATE)VALUES( '2830364730','100001483970886', TO_DATE('15/02/2022', 'DD/MM/YYYY'));

1 row created.
```



```
SQL> SELECT * FROM PAYMENT;

    PNR_NO  TICKET_ID PAY_DATE
---------- ---------- ---------
8836628295 2.0000E+14 20-OCT-22
8836474674 2.0000E+14 23-SEP-22
8256712394 2.0000E+14 15-NOV-22
8658100133 1.0000E+14 18-OCT-22
2830364730 1.0000E+14 15-FEB-22
```

ORDER-BY QUERY

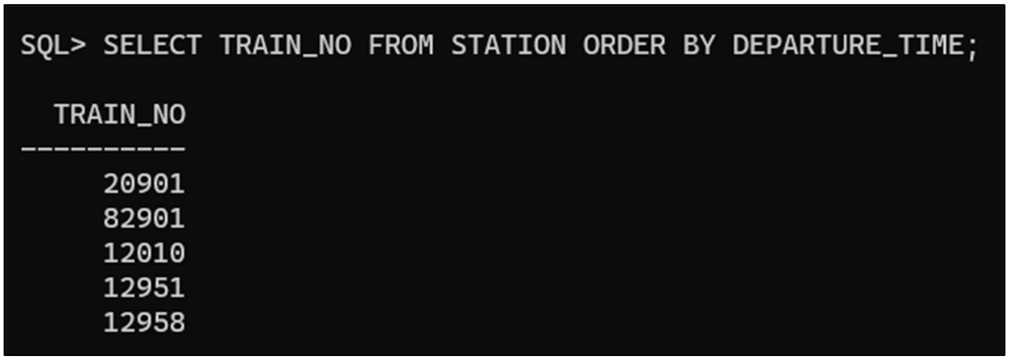
The ORDER BY keyword is used to sort the result-set in ascending or descending order.

The ORDER BY keyword sorts the records in ascending order by default. To sort the records in descending order, use the DESC keyword.

1. **Display TRAIN_NO according to their DEPARTURE_TIME.**

   SELECT TRAIN_NO FROM STATION ORDER BY DEPARTURE_TIME;

   OUTPUT:

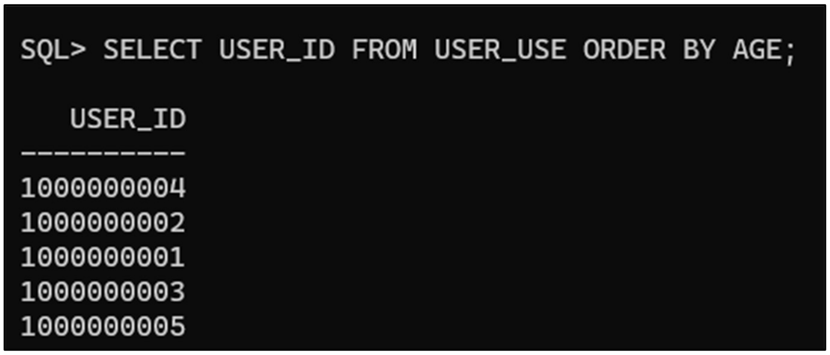


```
SQL> SELECT TRAIN_NO FROM STATION ORDER BY DEPARTURE_TIME;

  TRAIN_NO
----------
     20901
     82901
     12010
     12951
     12958
```

2. **Display USER_ID according to the User's Age.**

   SELECT USER_ID FROM USER_AGE ORDER BY AGE;

   OUTPUT:



```
SQL> SELECT USER_ID FROM USER_USE ORDER BY AGE;

   USER_ID
----------
1000000004
1000000002
1000000001
1000000003
1000000005
```

SEQUENCE QUERY

A sequence is a user defined schema bound object that generates a sequence of numeric values.

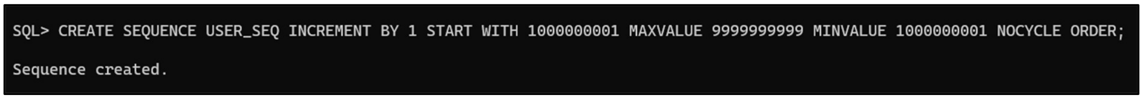
Sequences are frequently used in many databases because many applications require each row in a table to contain a unique value and sequences provides an easy way to generate them.

The sequence of numeric values is generated in an ascending or descending order at defined intervals and can be configured to restart when exceeds max_value.

Create a Sequence USER_SEQ for USER_ID in USER Table.

CREATE SEQUENCE USER_SEQ INCREMENT BY 1 START WITH 1000000001 MAXVALUE 9999999999 MINVALUE 1000000001 NOCYCLE ORDER;

OUTPUT:

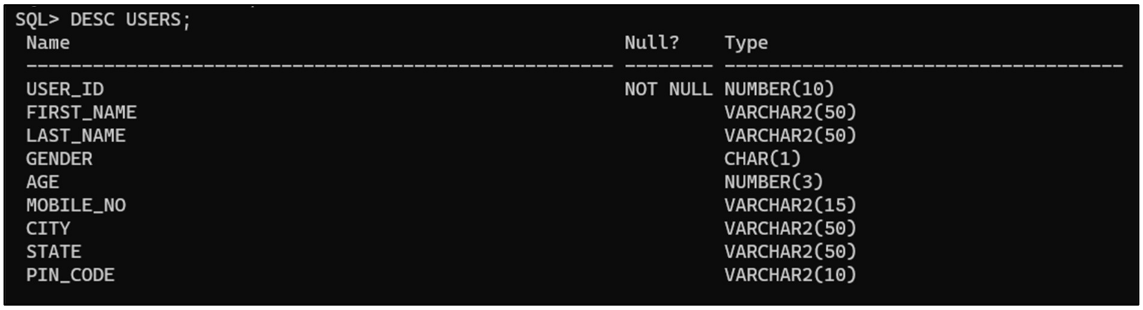


```
SQL> CREATE SEQUENCE USER_SEQ INCREMENT BY 1 START WITH 1000000001 MAXVALUE 9999999999 MINVALUE 1000000001 NOCYCLE ORDER;

Sequence created.
```

For view sequence create table users

CREATE TABLE USERS (USER_ID NUMBER(10) DEFAULT USER_SEQ.NEXTVAL PRIMARY KEY, FIRST_NAME VARCHAR2(50), LAST_NAME VARCHAR2(50), GENDER CHAR(1), AGE NUMBER(3), MOBILE_NO VARCHAR2(15), CITY VARCHAR2(50), STATE VARCHAR2(50), PIN_CODE VARCHAR2(10));

OUTPUT:



```
SQL> DESC USERS;
 Name                                                  Null?    Type
 ----------------------------------------------------- -------- ------------------------------------
 USER_ID                                               NOT NULL NUMBER(10)
 FIRST_NAME                                                     VARCHAR2(50)
 LAST_NAME                                                      VARCHAR2(50)
 GENDER                                                         CHAR(1)
 AGE                                                            NUMBER(3)
 MOBILE_NO                                                      VARCHAR2(15)
 CITY                                                           VARCHAR2(50)
 STATE                                                          VARCHAR2(50)
 PIN_CODE                                                       VARCHAR2(10)
```

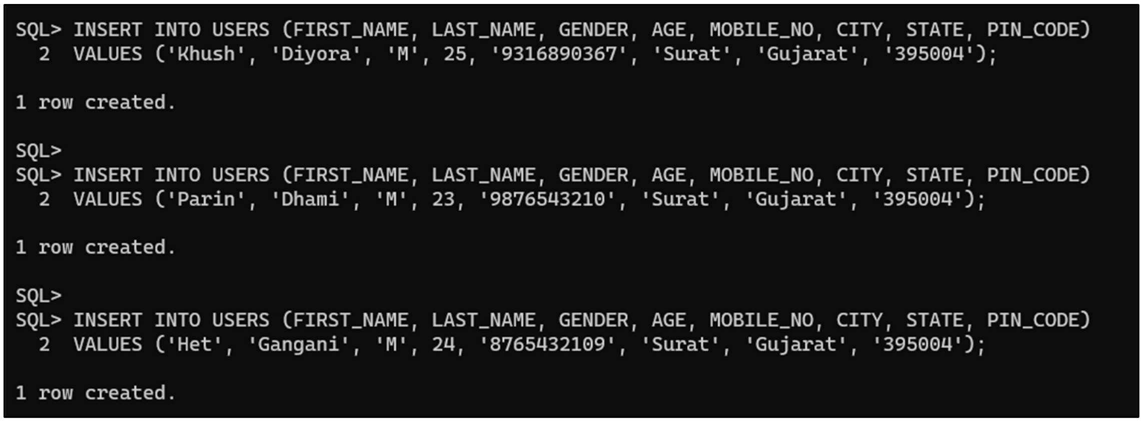
Insert into query

INSERT INTO USERS (FIRST_NAME, LAST_NAME, GENDER, AGE, MOBILE_NO, CITY, STATE, PIN_CODE)VALUES ('Khush', 'Diyora', 'M', 25, '9316890367', 'Surat', 'Gujarat', '395004');

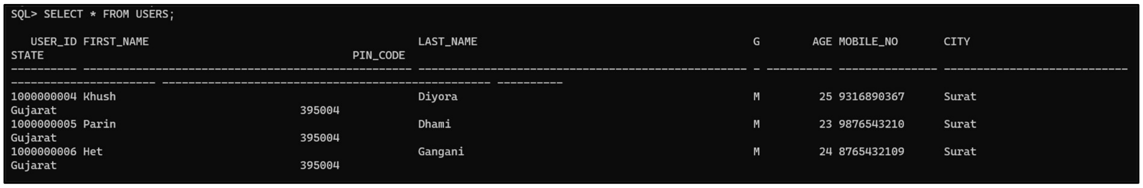
INSERT INTO USERS (FIRST_NAME, LAST_NAME, GENDER, AGE, MOBILE_NO, CITY, STATE, PIN_CODE)VALUES ('Parin', 'Dhami', 'M', 23, '9876543210', 'Surat', 'Gujarat', '395004');

INSERT INTO USERS (FIRST_NAME, LAST_NAME, GENDER, AGE, MOBILE_NO, CITY, STATE, PIN_CODE)VALUES ('Het', 'Gangani', 'M', 24, '8765432109', 'Surat', 'Gujarat', '395004');

OUTPUT:



```
SQL> INSERT INTO USERS (FIRST_NAME, LAST_NAME, GENDER, AGE, MOBILE_NO, CITY, STATE, PIN_CODE)
  2  VALUES ('Khush', 'Diyora', 'M', 25, '9316890367', 'Surat', 'Gujarat', '395004');

1 row created.

SQL>
SQL> INSERT INTO USERS (FIRST_NAME, LAST_NAME, GENDER, AGE, MOBILE_NO, CITY, STATE, PIN_CODE)
  2  VALUES ('Parin', 'Dhami', 'M', 23, '9876543210', 'Surat', 'Gujarat', '395004');

1 row created.

SQL>
SQL> INSERT INTO USERS (FIRST_NAME, LAST_NAME, GENDER, AGE, MOBILE_NO, CITY, STATE, PIN_CODE)
  2  VALUES ('Het', 'Gangani', 'M', 24, '8765432109', 'Surat', 'Gujarat', '395004');

1 row created.
```



```
SQL> SELECT * FROM USERS;

   USER_ID FIRST_NAME                                         LAST_NAME                                          G        AGE MOBILE_NO       CITY
STATE                                           PIN_CODE
---------- -------------------------------------------------- -------------------------------------------------- - ---------- --------------- ----------------------------
---------------------- -------------------------------------------------- ----------
1000000004 Khush                                              Diyora                                             M         25 9316890367      Surat
Gujarat                                     395004
1000000005 Parin                                              Dhami                                              M         23 9876543210      Surat
Gujarat                                     395004
1000000006 Het                                                Gangani                                            M         24 8765432109      Surat
Gujarat                                     395004
```

VIEW QUERY

In SQL, a view is a virtual table based on the result-set of an SQL statement.

A view contains rows and columns, just like a real table. The fields in a view are fields from one or more real tables in the database.

View helps in flexible enforcement of security and simplification of complex Query.

There are 2 types of View:

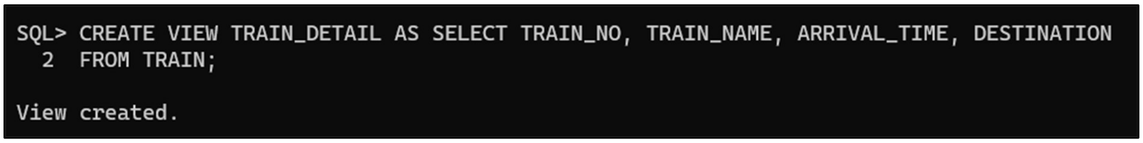
1.) Read-Only View

2.) Updateable View

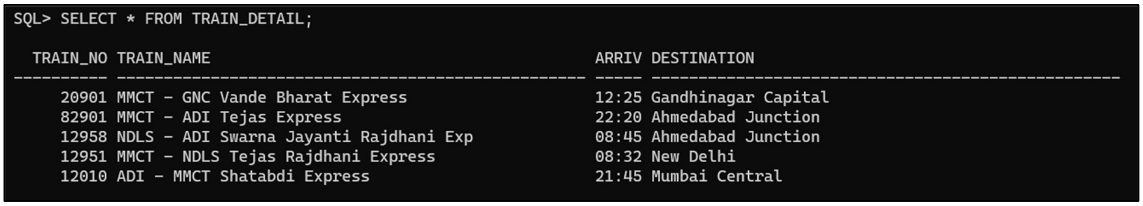
Create a view TRAIN_DETAIL on TRAIN_NO, TRAIN_NAME, ARRIVAL_TIME, DESTINATION of the TRAIN Table.

CREATE VIEW TRAIN_DETAIL AS SELECT TRAIN_NO, TRAIN_NAME, ARRIVAL_TIME, DESTINATION FROM TRAIN;

OUTPUT:



```
SQL> CREATE VIEW TRAIN_DETAIL AS SELECT TRAIN_NO, TRAIN_NAME, ARRIVAL_TIME, DESTINATION
  2  FROM TRAIN;

View created.
```



```
SQL> SELECT * FROM TRAIN_DETAIL;

  TRAIN_NO TRAIN_NAME                                         ARRIV DESTINATION
---------- -------------------------------------------------- ----- --------------------------------------------------
     20901 MMCT - GNC Vande Bharat Express                    12:25 Gandhinagar Capital
     82901 MMCT - ADI Tejas Express                           22:20 Ahmedabad Junction
     12958 NDLS - ADI Swarna Jayanti Rajdhani Exp             08:45 Ahmedabad Junction
     12951 MMCT - NDLS Tejas Rajdhani Express                 08:32 New Delhi
     12010 ADI - MMCT Shatabdi Express                        21:45 Mumbai Central
```

SUB-QUERY

Sub Query is a form of SQL Statement that appears inside another SQL Statement. It is also called as Nested Query.

Sub Query are used with WHERE and HAVING Clause.

The SQL statement in which Sub Query is included is called the main Query or the outer query.

Types of Sub Query:

1.) Single Row Sub Query

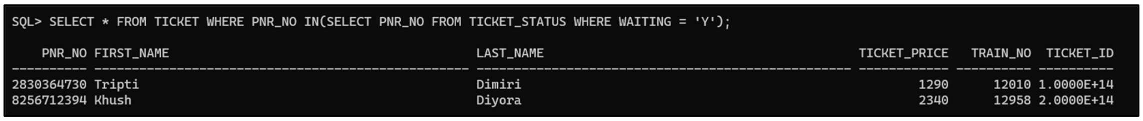
2.) Multiple Row Sub Query

3.) Correlated Sub Query

Give PNR_NO of Passenger whose ticket is in waiting list.

SELECT * FROM TICKET WHERE PNR_NO IN(SELECT PNR_NO FROM TICKET_STATUS WHERE WAITING = 'Y');

OUTPUT:



```
SQL> SELECT * FROM TICKET WHERE PNR_NO IN(SELECT PNR_NO FROM TICKET_STATUS WHERE WAITING = 'Y');

    PNR_NO FIRST_NAME                                         LAST_NAME                                          TICKET_PRICE   TRAIN_NO  TICKET_ID
---------- -------------------------------------------------- -------------------------------------------------- ------------ ---------- ----------
2830364730 Tripti                                             Dimiri                                                     1290      12010 1.0000E+14
8256712394 Khush                                              Diyora                                                     2340      12958 2.0000E+14
```

GROUP-BY QUERY

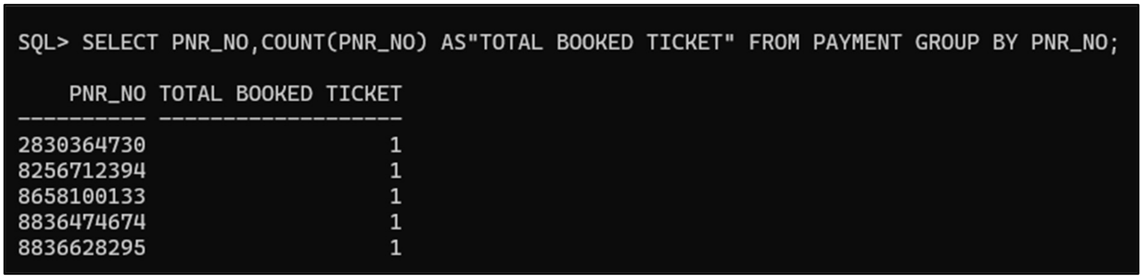
The GROUP BY statement groups rows that have the same values into summary rows, like "find the number of customers in each country".

The GROUP BY statement is often used with aggregate functions (COUNT(), MAX(), MIN(), SUM(), AVG()) to group the result-set by one or more columns.

Sort the tickets based on how many passengers have booked them overall.

SELECT PNR_NO,COUNT(PNR_NO) AS"TOTAL BOOKED TICKET" FROM PAYMENT GROUP BY PNR_NO;

OUTPUT:

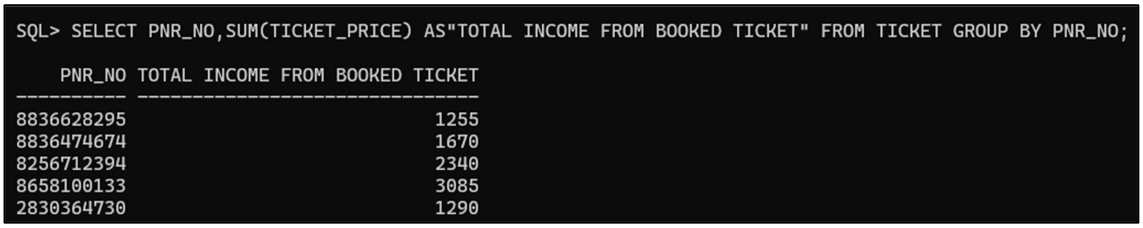


```
SQL> SELECT PNR_NO,COUNT(PNR_NO) AS"TOTAL BOOKED TICKET" FROM PAYMENT GROUP BY PNR_NO;

    PNR_NO TOTAL BOOKED TICKET
---------- -------------------
2830364730                   1
8256712394                   1
8658100133                   1
8836474674                   1
8836628295                   1
```

Total income from the booked ticket group by PNR_NO.

SELECT PNR_NO,SUM(TICKET_PRICE) AS"TOTAL INCOME FROM BOOKED TICKET" FROM TICKET GROUP BY PNR_NO;

OUTPUT:



```
SQL> SELECT PNR_NO,SUM(TICKET_PRICE) AS"TOTAL INCOME FROM BOOKED TICKET" FROM TICKET GROUP BY PNR_NO;

    PNR_NO TOTAL INCOME FROM BOOKED TICKET
---------- -------------------------------
8836628295                            1255
8836474674                            1670
8256712394                            2340
8658100133                            3085
2830364730                            1290
```

HAVING-CLAUSE QUERY

The HAVING clause was added to SQL because the WHERE keyword cannot be used with aggregate functions.

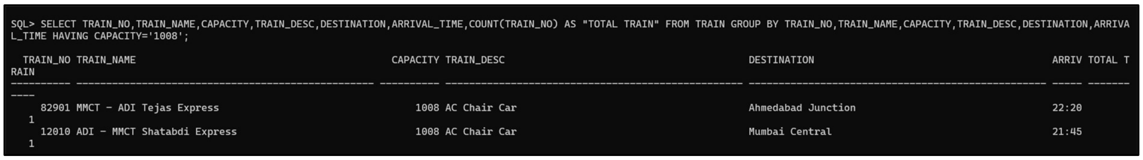
SYNTAX:

SELECT column_name(s) FROM table_name WHERE condition GROUP BY column_name(s) HAVING condition ORDER BY column_name(s);

Retrieve the train details whose capacity having ‘1008’

SELECT TRAIN_NO,TRAIN_NAME,CAPACITY,TRAIN_DESC,DESTINATION,ARRIVAL_TIME,COUNT(TRAIN_NO) AS "TOTAL TRAIN" FROM GROUP BY TRAIN_NO,TRAIN_NAME,CAPACITY,TRAIN_DESC,DESTINATION,ARRIVAL_TIME HAVING CAPACITY='1008';

OUTPUT:

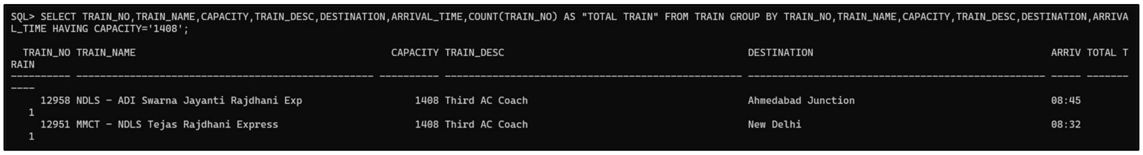


```
SQL> SELECT TRAIN_NO,TRAIN_NAME,CAPACITY,TRAIN_DESC,DESTINATION,ARRIVAL_TIME,COUNT(TRAIN_NO) AS "TOTAL TRAIN" FROM TRAIN GROUP BY TRAIN_NO,TRAIN_NAME,CAPACITY,TRAIN_DESC,DESTINATION,ARRIVA
L_TIME HAVING CAPACITY='1008';

  TRAIN_NO TRAIN_NAME                                           CAPACITY TRAIN_DESC                                         DESTINATION                                        ARRIV TOTAL T
RAIN
---------- -------------------------------------------------- ---------- -------------------------------------------------- -------------------------------------------------- ----- -------
----
     82901 MMCT - ADI Tejas Express                                 1008 AC Chair Car                                       Ahmedabad Junction                                 22:20
   1
     12010 ADI - MMCT Shatabdi Express                              1008 AC Chair Car                                       Mumbai Central                                     21:45
   1
```

Retrieve the train details whose capacity having ‘1408’

SELECT TRAIN_NO,TRAIN_NAME,CAPACITY,TRAIN_DESC,DESTINATION,ARRIVAL_TIME,COUNT(TRAIN_NO) AS "TOTAL TRAIN" FROM TRAIN GROUP BY TRAIN_NO,TRAIN_NAME,CAPACITY,TRAIN_DESC,DESTINATION,ARRIVAL_TIME HAVING CAPACITY='1408';

OUTPUT:



```
SQL> SELECT TRAIN_NO,TRAIN_NAME,CAPACITY,TRAIN_DESC,DESTINATION,ARRIVAL_TIME,COUNT(TRAIN_NO) AS "TOTAL TRAIN" FROM TRAIN GROUP BY TRAIN_NO,TRAIN_NAME,CAPACITY,TRAIN_DESC,DESTINATION,ARRIVA
L_TIME HAVING CAPACITY='1408';

  TRAIN_NO TRAIN_NAME                                           CAPACITY TRAIN_DESC                                         DESTINATION                                        ARRIV TOTAL T
RAIN
---------- -------------------------------------------------- ---------- -------------------------------------------------- -------------------------------------------------- ----- -------
----
     12958 NDLS - ADI Swarna Jayanti Rajdhani Exp                   1408 Third AC Coach                                     Ahmedabad Junction                                 08:45
   1
     12951 MMCT - NDLS Tejas Rajdhani Express                       1408 Third AC Coach                                     New Delhi                                          08:32
   1
```

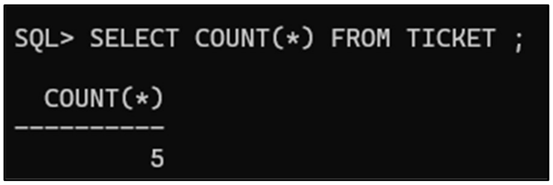
AGGREGATE FUNCTION

There are five types of Aggregate functions are listed below:

COUNT(*)

SELECT COUNT(*) FROM TICKET ;

OUTPUT:

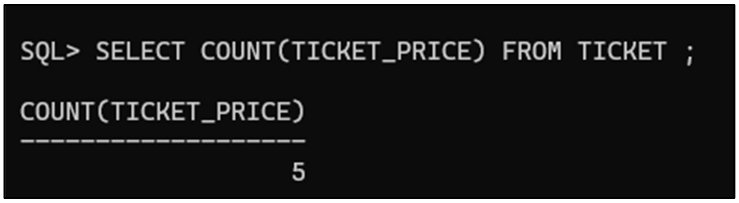


```
SQL> SELECT COUNT(*) FROM TICKET ;

  COUNT(*)
----------
         5
```

COUNT(COLUMNNAME)

SELECT COUNT(TICKET_PRICE) FROM TICKET ;

OUTPUT:

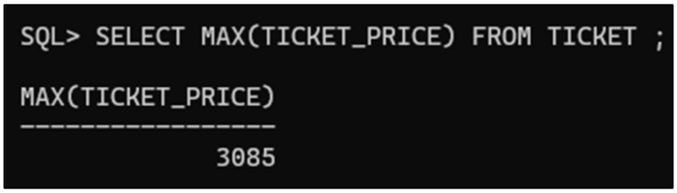


```
SQL> SELECT COUNT(TICKET_PRICE) FROM TICKET ;

COUNT(TICKET_PRICE)
-------------------
                  5
```

MAX()

SELECT MAX(TICKET_PRICE) FROM TICKET ;

OUTPUT:

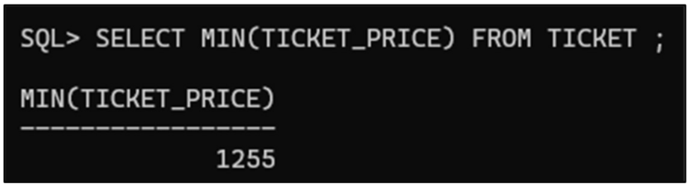


```
SQL> SELECT MAX(TICKET_PRICE) FROM TICKET ;

MAX(TICKET_PRICE)
-----------------
             3085
```

MIN()

SELECT MIN(TICKET_PRICE) FROM TICKET ;

OUTPUT:

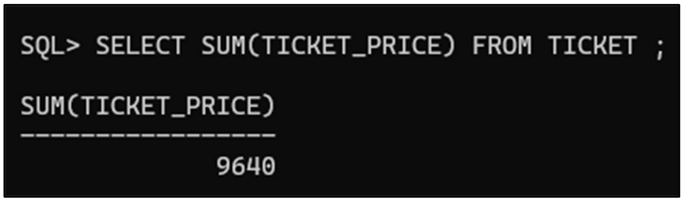


```
SQL> SELECT MIN(TICKET_PRICE) FROM TICKET ;

MIN(TICKET_PRICE)
-----------------
             1255
```

SUM()

SELECT SUM(TICKET_PRICE) FROM TICKET ;

OUTPUT:

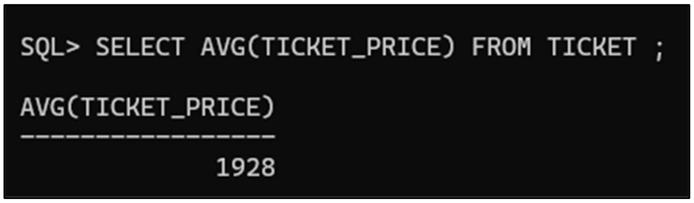


```
SQL> SELECT SUM(TICKET_PRICE) FROM TICKET ;

SUM(TICKET_PRICE)
-----------------
             9640
```

AVG()

SELECT AVG(TICKET_PRICE) FROM TICKET ;

OUTPUT:



```
SQL> SELECT AVG(TICKET_PRICE) FROM TICKET ;

AVG(TICKET_PRICE)
-----------------
             1928
```