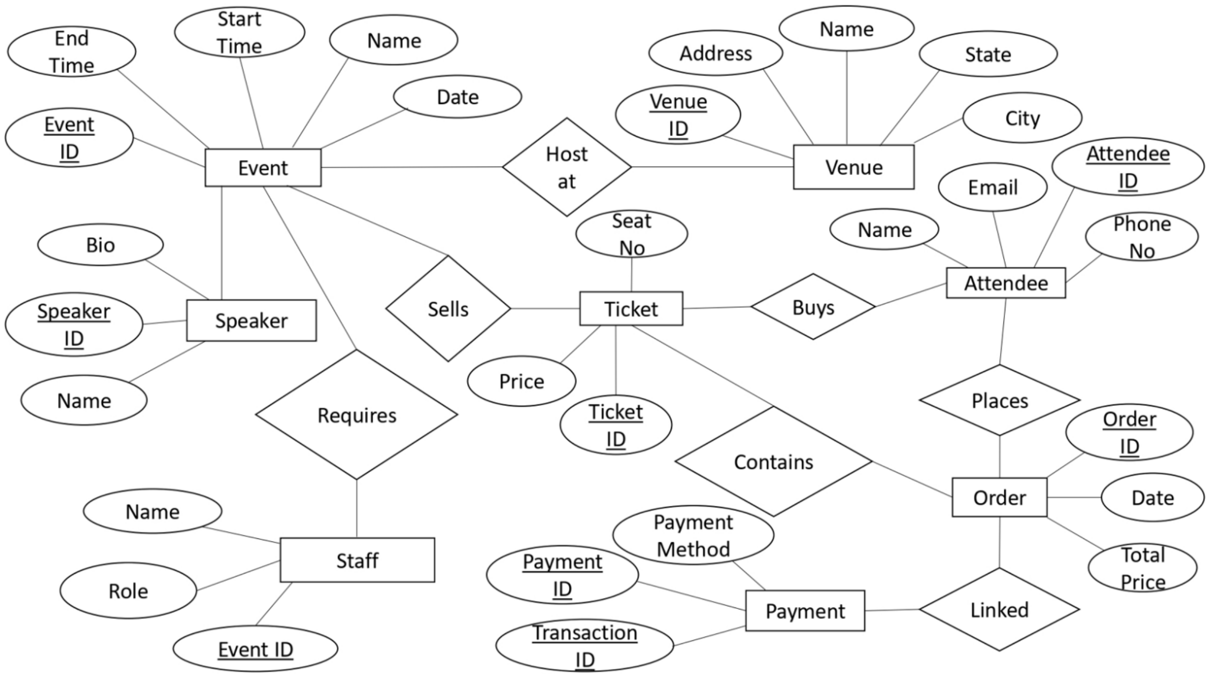
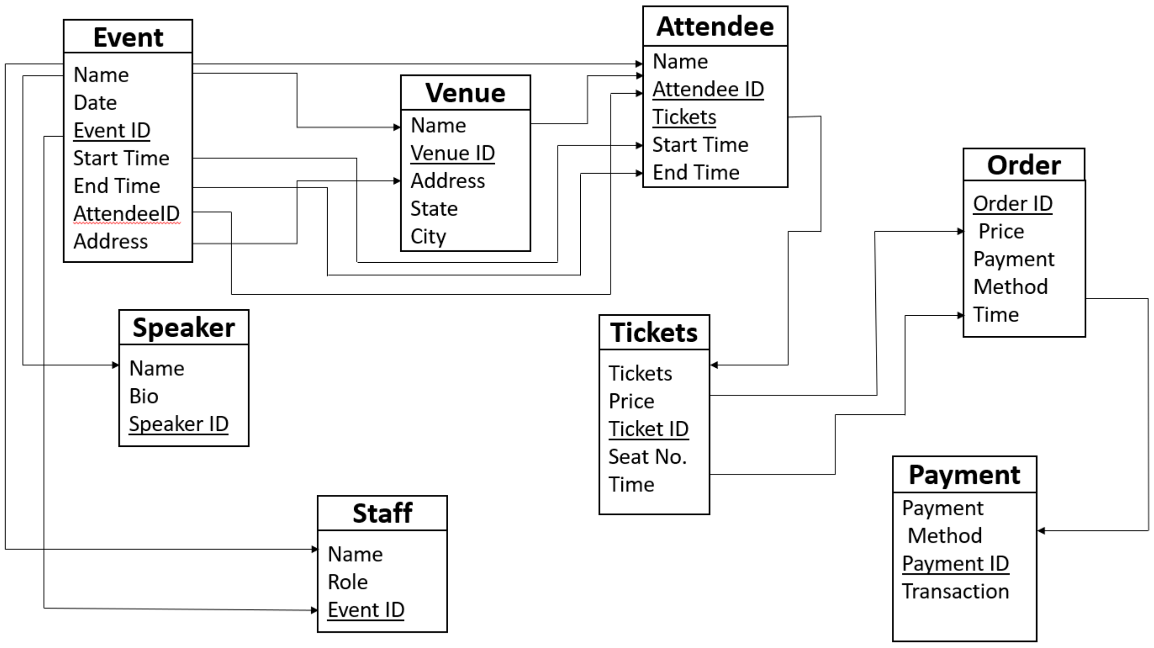
1.Draw an ER diagram and EER diagram and convert it into relational database and draw schema diagram.



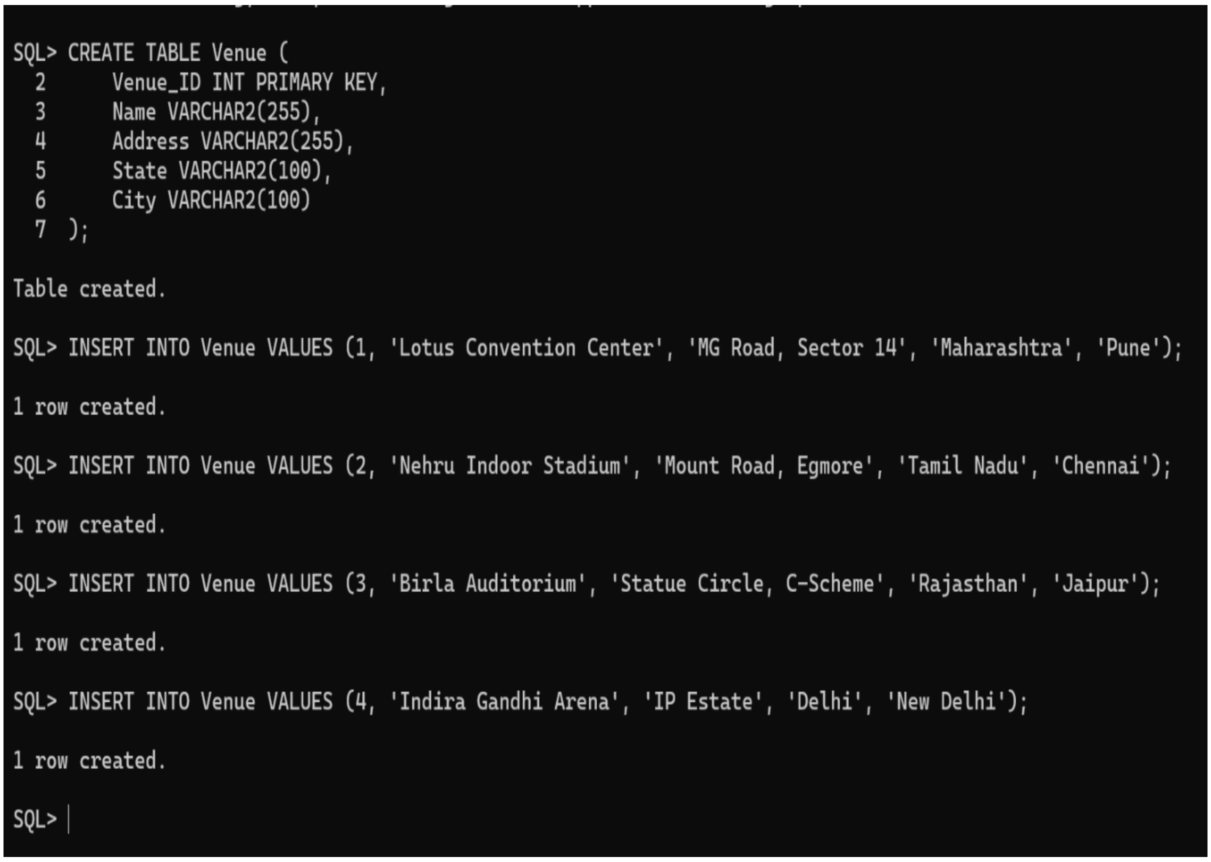
Entity-Relationship Model



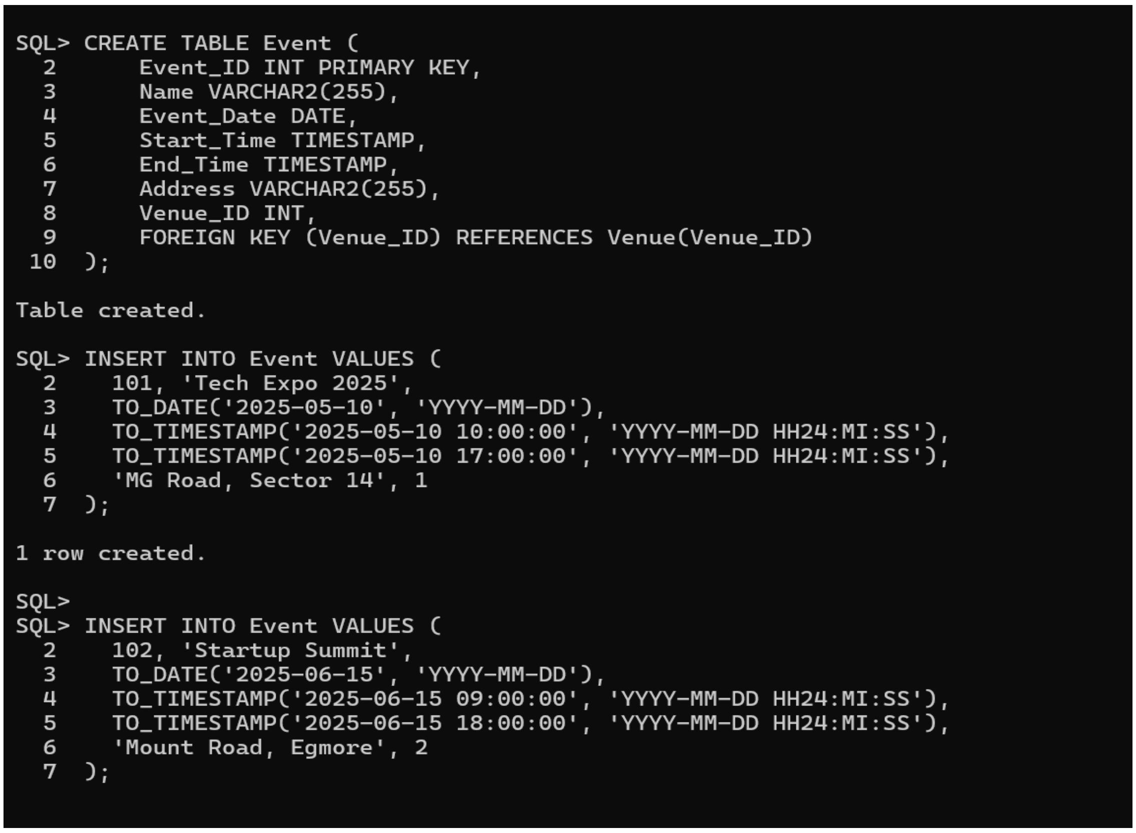
Schema Diagram

2. Write and execute basic SQL query- create, alter, insert, update and delete. (instructor should frame appropriate problem definition).

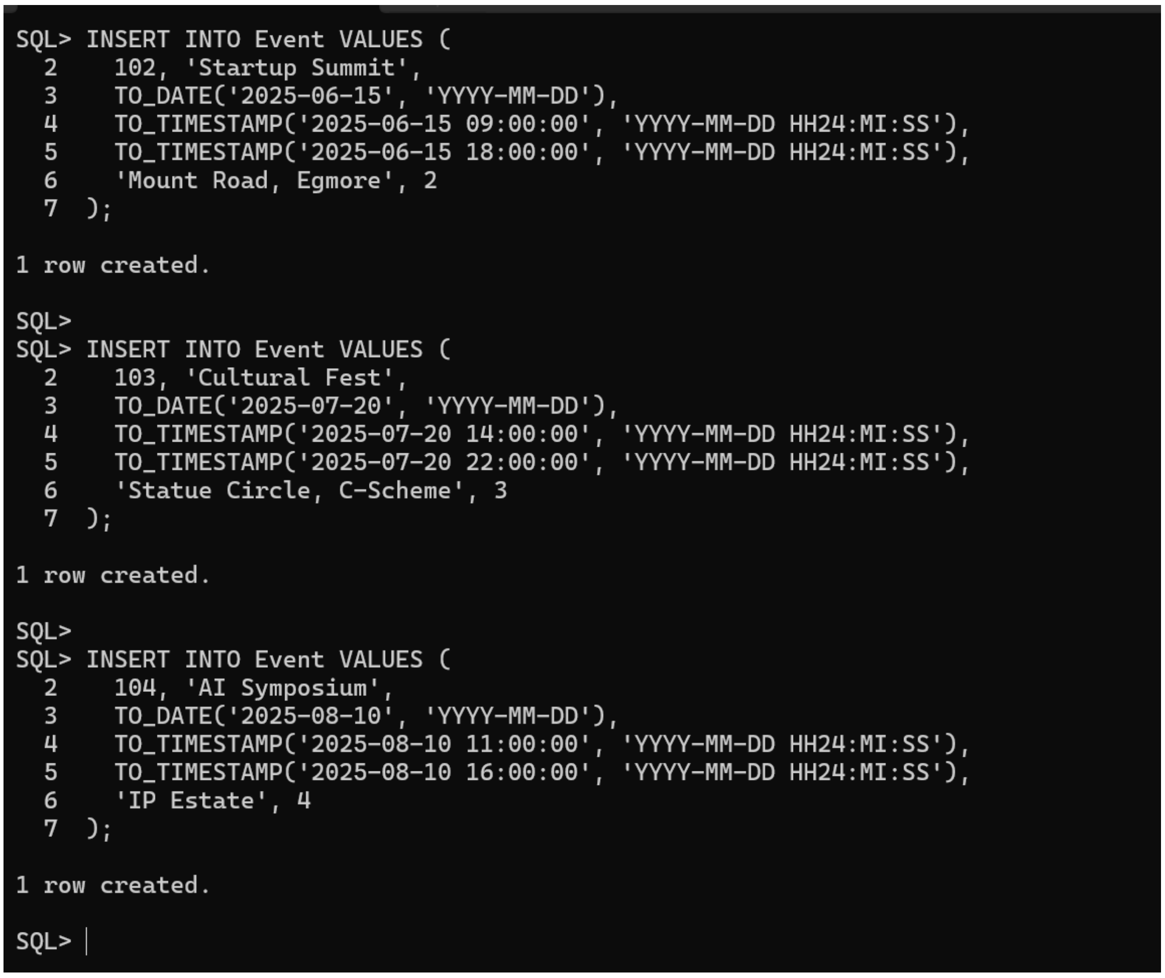
INSERT INTO statements add dummy data.



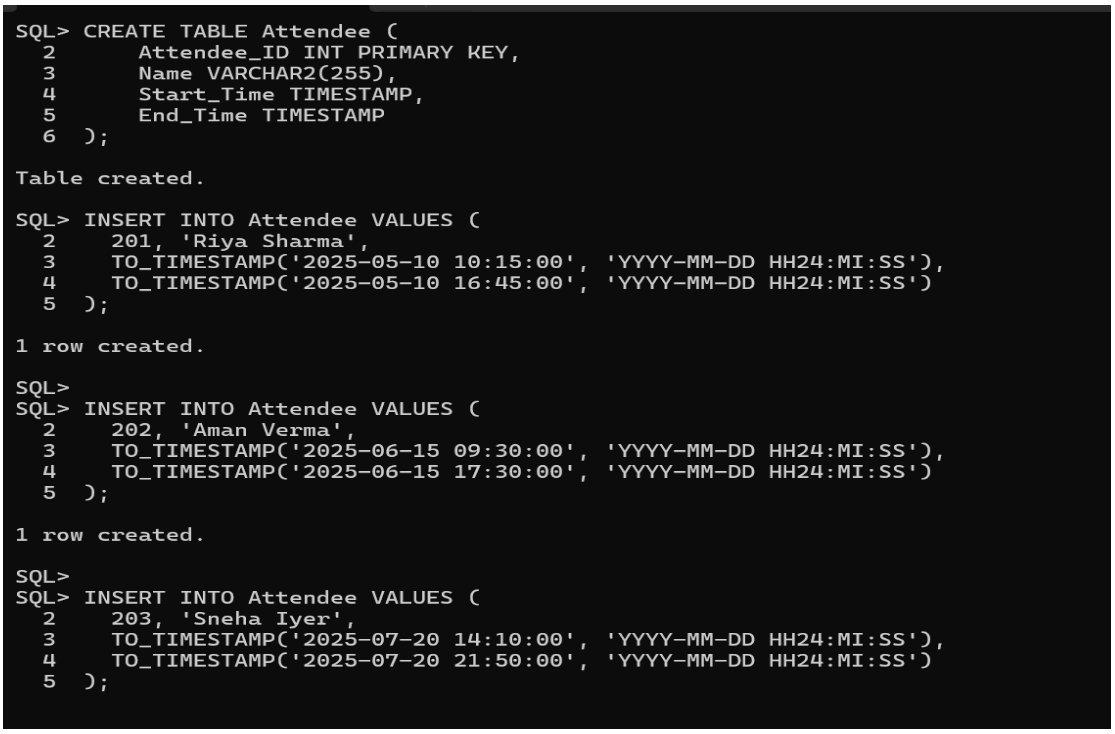
```
SQL> CREATE TABLE Venue (
  2      Venue_ID INT PRIMARY KEY,
  3      Name VARCHAR2(255),
  4      Address VARCHAR2(255),
  5      State VARCHAR2(100),
  6      City VARCHAR2(100)
  7  );

Table created.

SQL> INSERT INTO Venue VALUES (1, 'Lotus Convention Center', 'MG Road, Sector 14', 'Maharashtra', 'Pune');

1 row created.

SQL> INSERT INTO Venue VALUES (2, 'Nehru Indoor Stadium', 'Mount Road, Egmore', 'Tamil Nadu', 'Chennai');

1 row created.

SQL> INSERT INTO Venue VALUES (3, 'Birla Auditorium', 'Statue Circle, C-Scheme', 'Rajasthan', 'Jaipur');

1 row created.

SQL> INSERT INTO Venue VALUES (4, 'Indira Gandhi Arena', 'IP Estate', 'Delhi', 'New Delhi');

1 row created.

SQL>
```



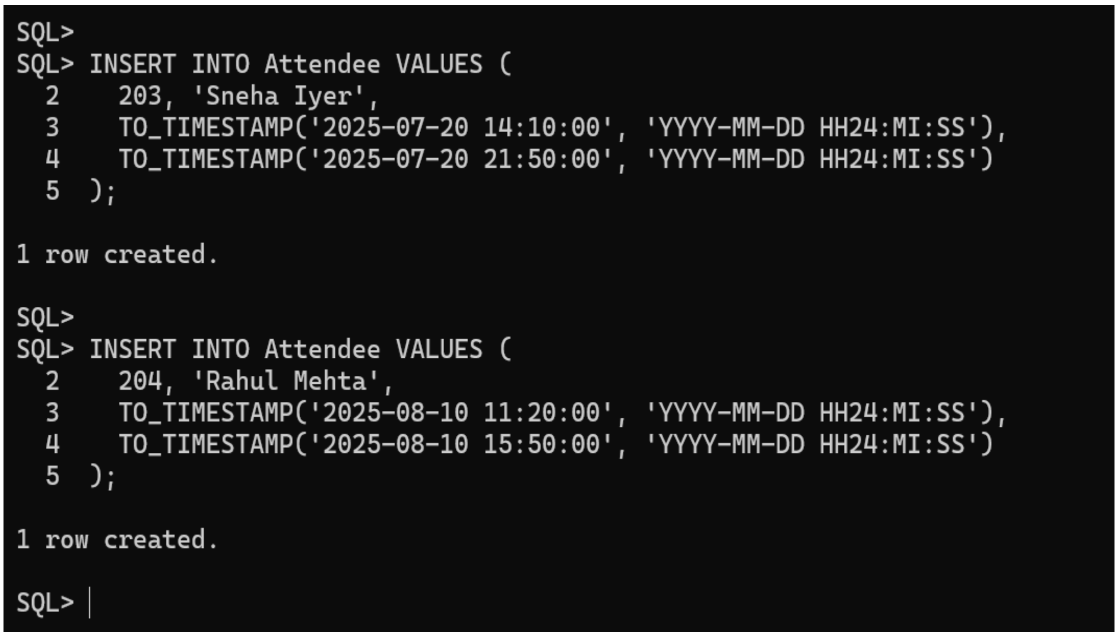
```
SQL> CREATE TABLE Event (
         Event_ID INT PRIMARY KEY,
         Name VARCHAR2(255),
         Event_Date DATE,
         Start_Time TIMESTAMP,
         End_Time TIMESTAMP,
         Address VARCHAR2(255),
         Venue_ID INT,
         FOREIGN KEY (Venue_ID) REFERENCES Venue(Venue_ID)
     );

Table created.

SQL> INSERT INTO Event VALUES (
       101, 'Tech Expo 2025',
       TO_DATE('2025-05-10', 'YYYY-MM-DD'),
       TO_TIMESTAMP('2025-05-10 10:00:00', 'YYYY-MM-DD HH24:MI:SS'),
       TO_TIMESTAMP('2025-05-10 17:00:00', 'YYYY-MM-DD HH24:MI:SS'),
       'MG Road, Sector 14', 1
     );

1 row created.

SQL>
SQL> INSERT INTO Event VALUES (
       102, 'Startup Summit',
       TO_DATE('2025-06-15', 'YYYY-MM-DD'),
       TO_TIMESTAMP('2025-06-15 09:00:00', 'YYYY-MM-DD HH24:MI:SS'),
       TO_TIMESTAMP('2025-06-15 18:00:00', 'YYYY-MM-DD HH24:MI:SS'),
       'Mount Road, Egmore', 2
     );
```



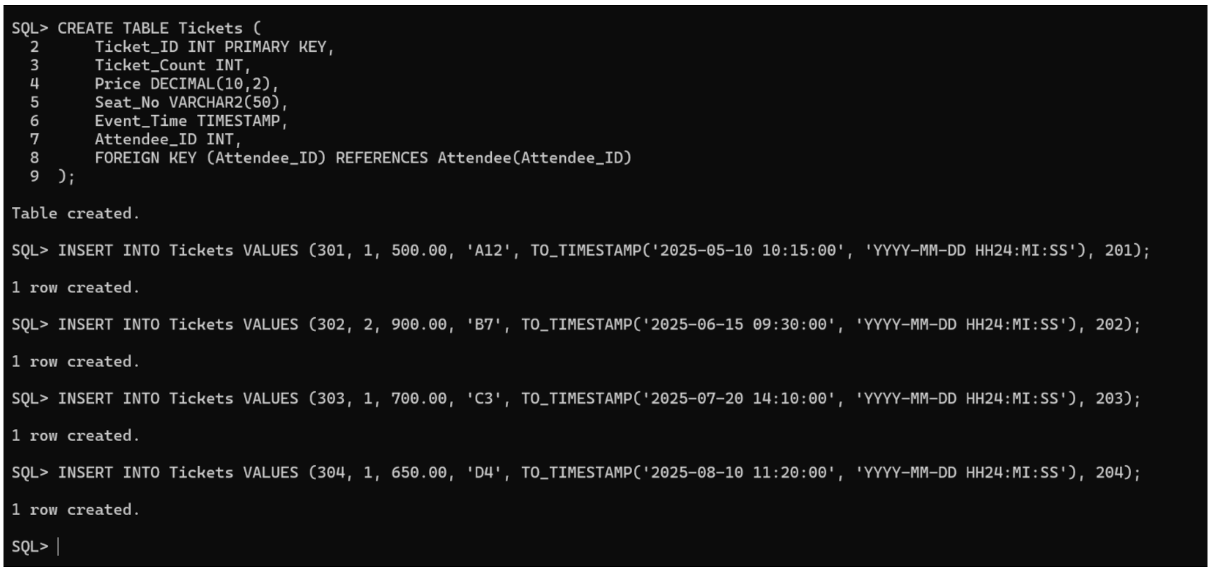
```
SQL> INSERT INTO Event VALUES (
       102, 'Startup Summit',
       TO_DATE('2025-06-15', 'YYYY-MM-DD'),
       TO_TIMESTAMP('2025-06-15 09:00:00', 'YYYY-MM-DD HH24:MI:SS'),
       TO_TIMESTAMP('2025-06-15 18:00:00', 'YYYY-MM-DD HH24:MI:SS'),
       'Mount Road, Egmore', 2
     );

1 row created.

SQL>
SQL> INSERT INTO Event VALUES (
       103, 'Cultural Fest',
       TO_DATE('2025-07-20', 'YYYY-MM-DD'),
       TO_TIMESTAMP('2025-07-20 14:00:00', 'YYYY-MM-DD HH24:MI:SS'),
       TO_TIMESTAMP('2025-07-20 22:00:00', 'YYYY-MM-DD HH24:MI:SS'),
       'Statue Circle, C-Scheme', 3
     );

1 row created.

SQL>
SQL> INSERT INTO Event VALUES (
       104, 'AI Symposium',
       TO_DATE('2025-08-10', 'YYYY-MM-DD'),
       TO_TIMESTAMP('2025-08-10 11:00:00', 'YYYY-MM-DD HH24:MI:SS'),
       TO_TIMESTAMP('2025-08-10 16:00:00', 'YYYY-MM-DD HH24:MI:SS'),
       'IP Estate', 4
     );

1 row created.

SQL>
```



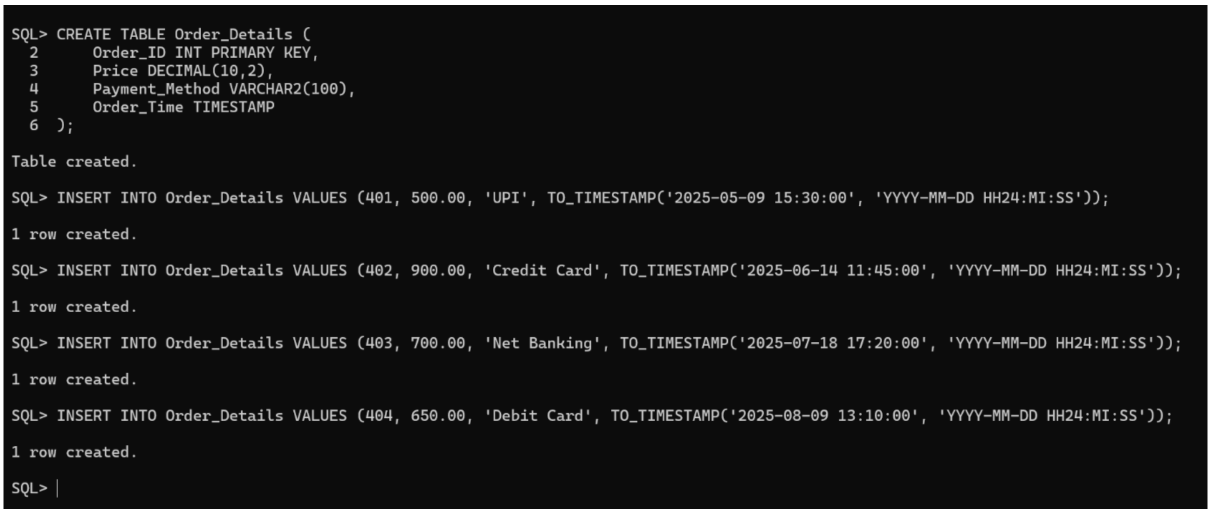
```
SQL> CREATE TABLE Attendee (
         Attendee_ID INT PRIMARY KEY,
         Name VARCHAR2(255),
         Start_Time TIMESTAMP,
         End_Time TIMESTAMP
     );

Table created.

SQL> INSERT INTO Attendee VALUES (
       201, 'Riya Sharma',
       TO_TIMESTAMP('2025-05-10 10:15:00', 'YYYY-MM-DD HH24:MI:SS'),
       TO_TIMESTAMP('2025-05-10 16:45:00', 'YYYY-MM-DD HH24:MI:SS')
     );

1 row created.

SQL>
SQL> INSERT INTO Attendee VALUES (
       202, 'Aman Verma',
       TO_TIMESTAMP('2025-06-15 09:30:00', 'YYYY-MM-DD HH24:MI:SS'),
       TO_TIMESTAMP('2025-06-15 17:30:00', 'YYYY-MM-DD HH24:MI:SS')
     );

1 row created.

SQL>
SQL> INSERT INTO Attendee VALUES (
       203, 'Sneha Iyer',
       TO_TIMESTAMP('2025-07-20 14:10:00', 'YYYY-MM-DD HH24:MI:SS'),
       TO_TIMESTAMP('2025-07-20 21:50:00', 'YYYY-MM-DD HH24:MI:SS')
     );
```



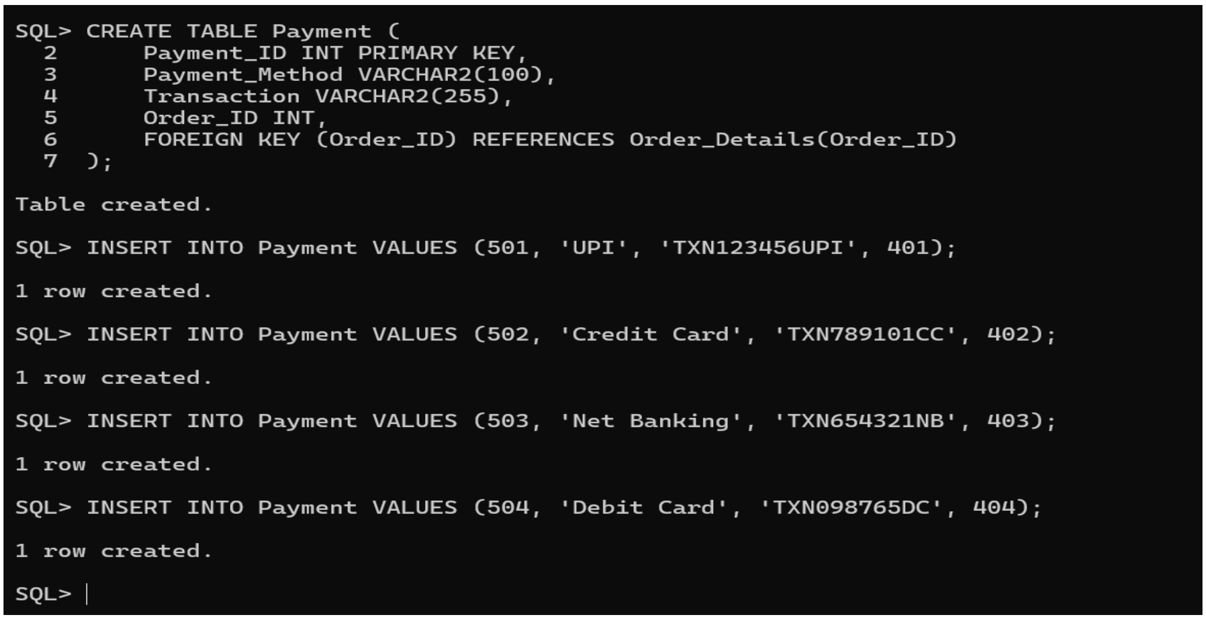
```
SQL>
SQL> INSERT INTO Attendee VALUES (
       203, 'Sneha Iyer',
       TO_TIMESTAMP('2025-07-20 14:10:00', 'YYYY-MM-DD HH24:MI:SS'),
       TO_TIMESTAMP('2025-07-20 21:50:00', 'YYYY-MM-DD HH24:MI:SS')
     );

1 row created.

SQL>
SQL> INSERT INTO Attendee VALUES (
       204, 'Rahul Mehta',
       TO_TIMESTAMP('2025-08-10 11:20:00', 'YYYY-MM-DD HH24:MI:SS'),
       TO_TIMESTAMP('2025-08-10 15:50:00', 'YYYY-MM-DD HH24:MI:SS')
     );

1 row created.

SQL>
```



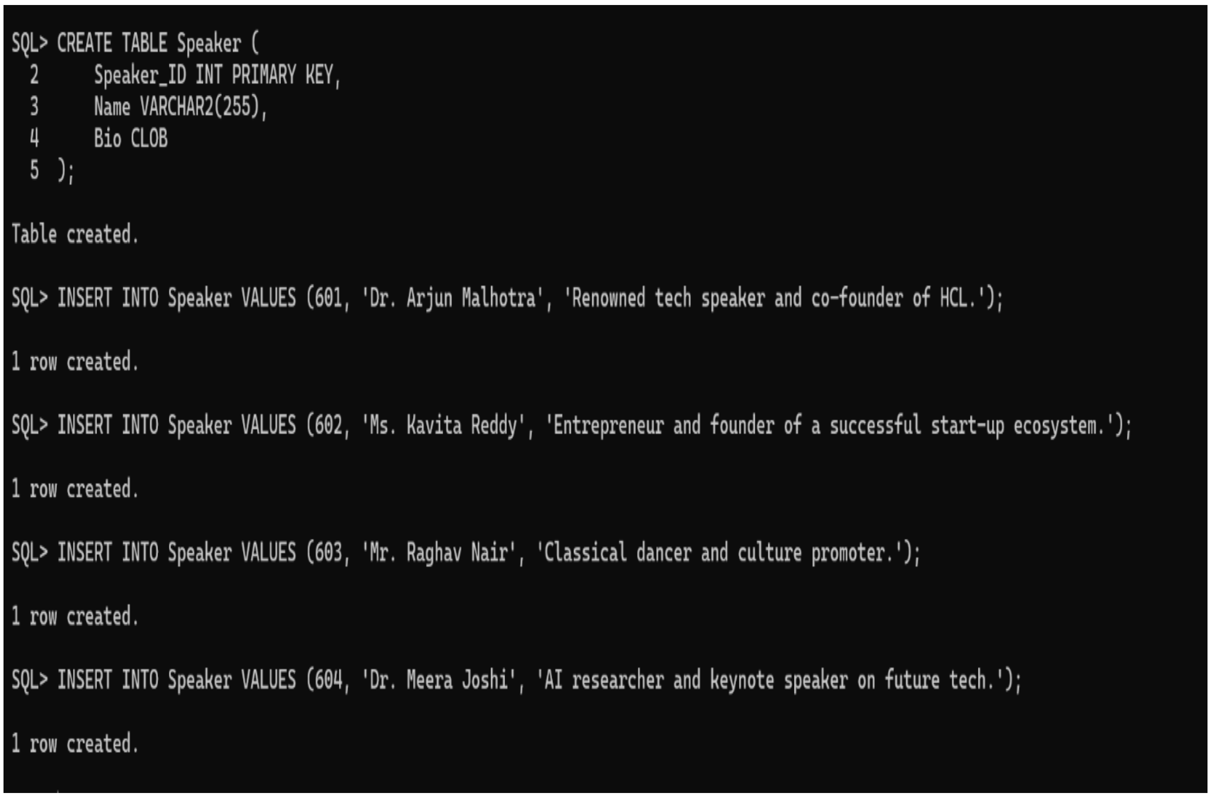
```
SQL> CREATE TABLE Tickets (
         Ticket_ID INT PRIMARY KEY,
         Ticket_Count INT,
         Price DECIMAL(10,2),
         Seat_No VARCHAR2(50),
         Event_Time TIMESTAMP,
         Attendee_ID INT,
         FOREIGN KEY (Attendee_ID) REFERENCES Attendee(Attendee_ID)
     );

Table created.

SQL> INSERT INTO Tickets VALUES (301, 1, 500.00, 'A12', TO_TIMESTAMP('2025-05-10 10:15:00', 'YYYY-MM-DD HH24:MI:SS'), 201);

1 row created.

SQL> INSERT INTO Tickets VALUES (302, 2, 900.00, 'B7', TO_TIMESTAMP('2025-06-15 09:30:00', 'YYYY-MM-DD HH24:MI:SS'), 202);

1 row created.

SQL> INSERT INTO Tickets VALUES (303, 1, 700.00, 'C3', TO_TIMESTAMP('2025-07-20 14:10:00', 'YYYY-MM-DD HH24:MI:SS'), 203);

1 row created.

SQL> INSERT INTO Tickets VALUES (304, 1, 650.00, 'D4', TO_TIMESTAMP('2025-08-10 11:20:00', 'YYYY-MM-DD HH24:MI:SS'), 204);

1 row created.

SQL> |
```



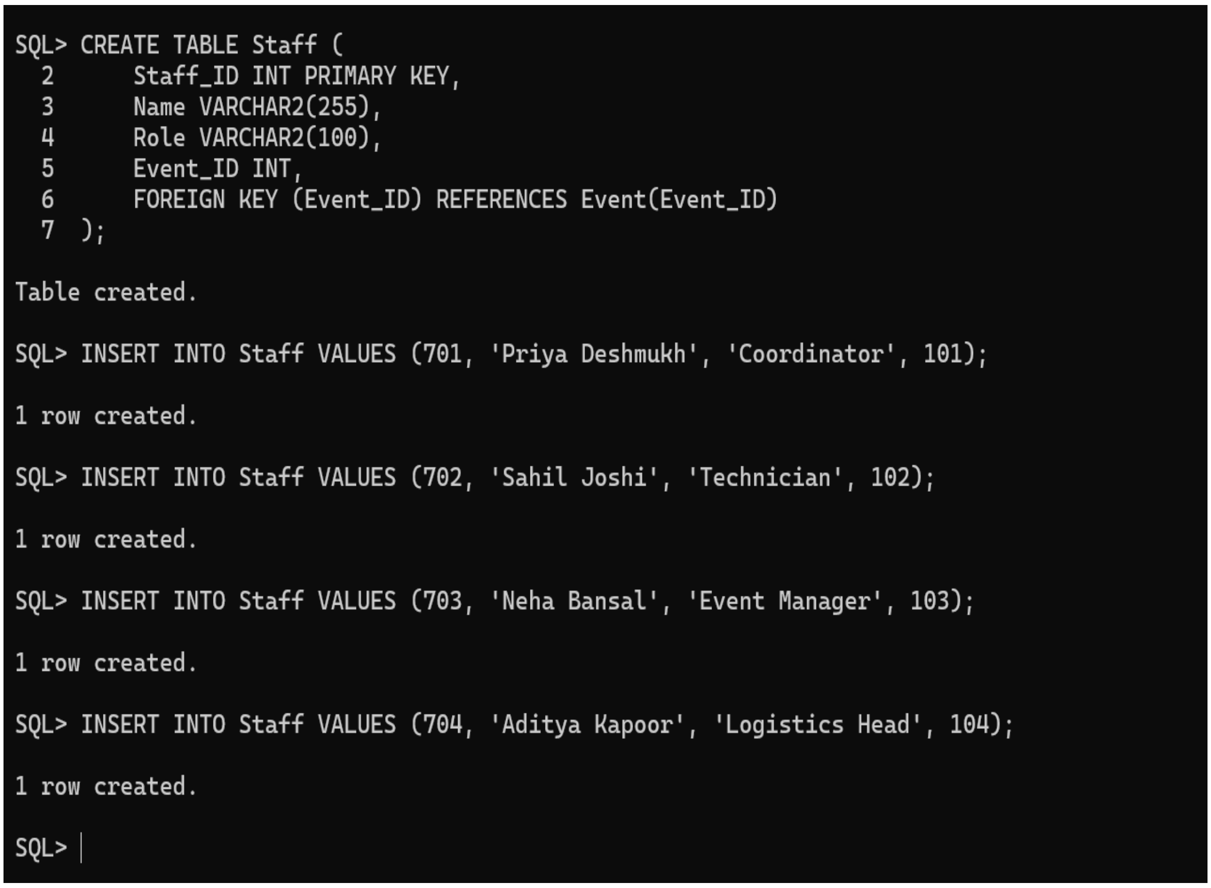
```
SQL> CREATE TABLE Order_Details (
         Order_ID INT PRIMARY KEY,
         Price DECIMAL(10,2),
         Payment_Method VARCHAR2(100),
         Order_Time TIMESTAMP
     );

Table created.

SQL> INSERT INTO Order_Details VALUES (401, 500.00, 'UPI', TO_TIMESTAMP('2025-05-09 15:30:00', 'YYYY-MM-DD HH24:MI:SS'));

1 row created.

SQL> INSERT INTO Order_Details VALUES (402, 900.00, 'Credit Card', TO_TIMESTAMP('2025-06-14 11:45:00', 'YYYY-MM-DD HH24:MI:SS'));

1 row created.

SQL> INSERT INTO Order_Details VALUES (403, 700.00, 'Net Banking', TO_TIMESTAMP('2025-07-18 17:20:00', 'YYYY-MM-DD HH24:MI:SS'));

1 row created.

SQL> INSERT INTO Order_Details VALUES (404, 650.00, 'Debit Card', TO_TIMESTAMP('2025-08-09 13:10:00', 'YYYY-MM-DD HH24:MI:SS'));

1 row created.

SQL> |
```



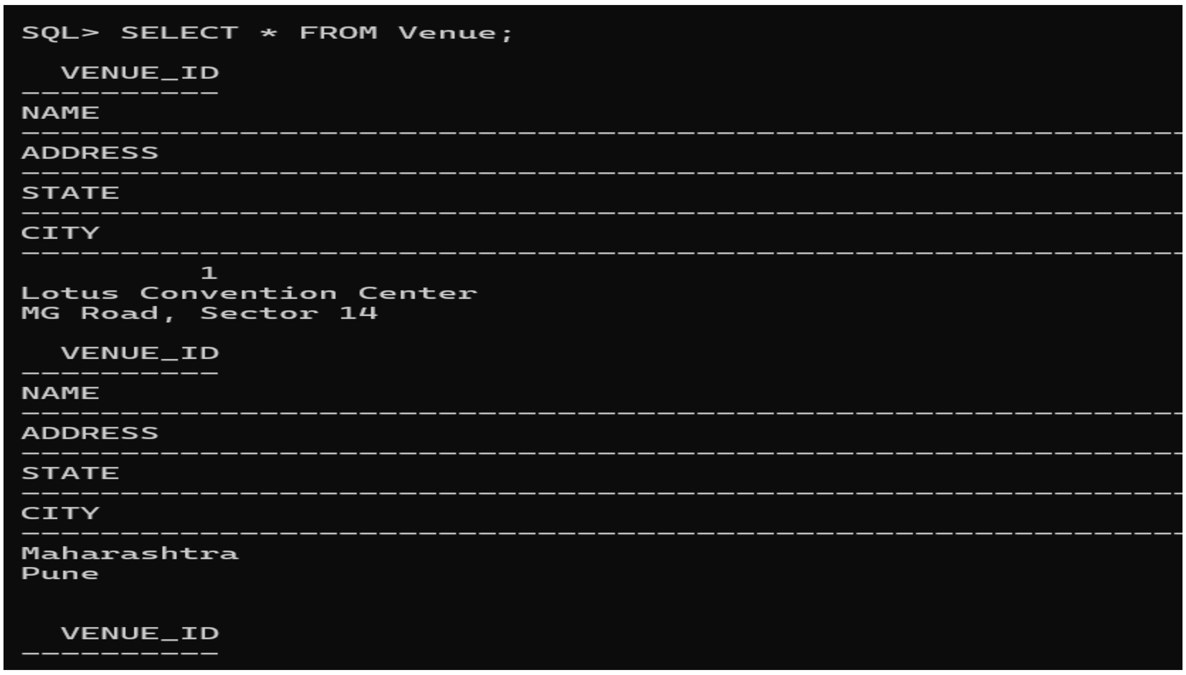
```
SQL> CREATE TABLE Payment (
         Payment_ID INT PRIMARY KEY,
         Payment_Method VARCHAR2(100),
         Transaction VARCHAR2(255),
         Order_ID INT,
         FOREIGN KEY (Order_ID) REFERENCES Order_Details(Order_ID)
     );

Table created.

SQL> INSERT INTO Payment VALUES (501, 'UPI', 'TXN123456UPI', 401);

1 row created.

SQL> INSERT INTO Payment VALUES (502, 'Credit Card', 'TXN789101CC', 402);

1 row created.

SQL> INSERT INTO Payment VALUES (503, 'Net Banking', 'TXN654321NB', 403);

1 row created.

SQL> INSERT INTO Payment VALUES (504, 'Debit Card', 'TXN098765DC', 404);

1 row created.

SQL> |
```



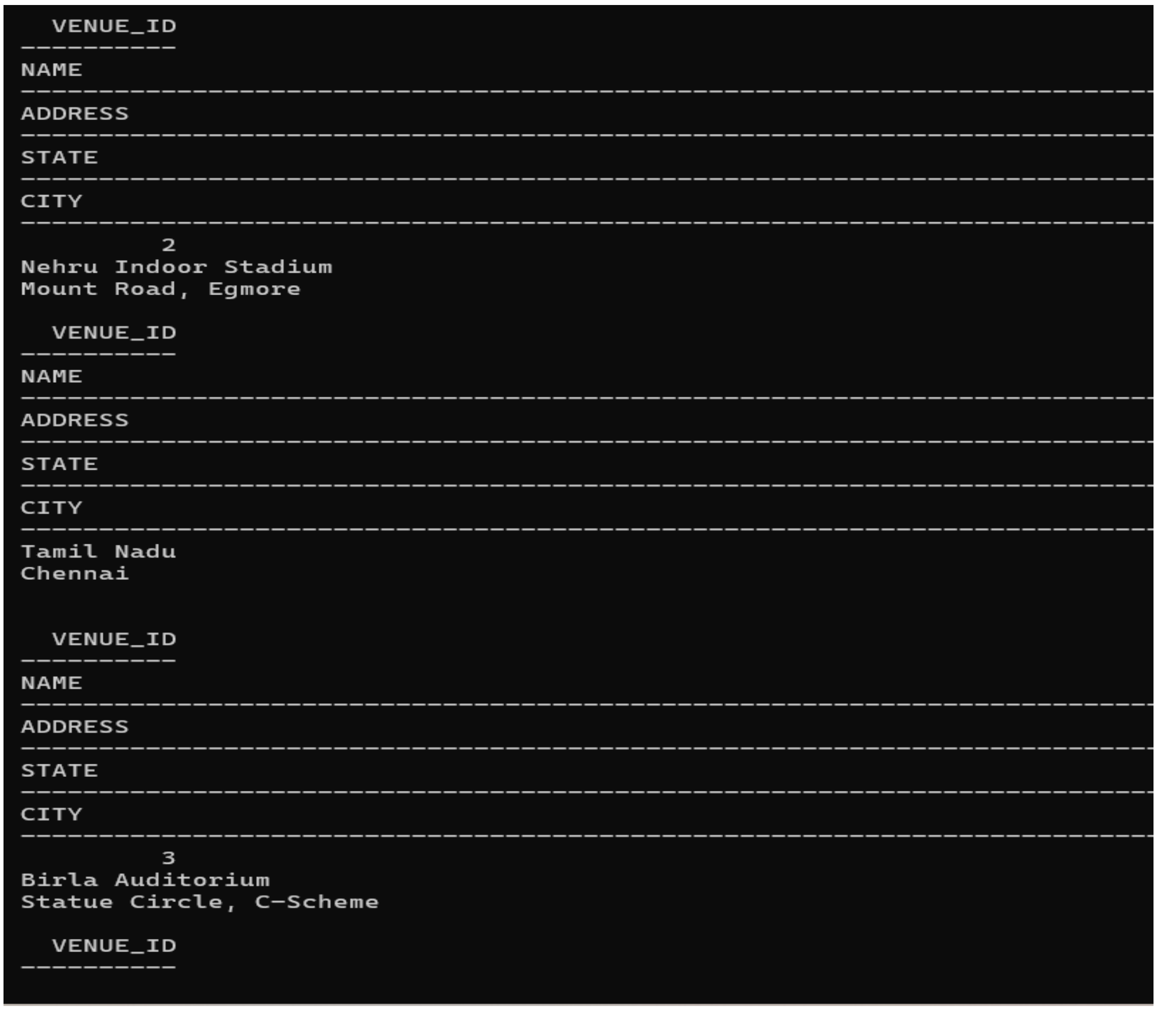
```
SQL> CREATE TABLE Speaker (
  2      Speaker_ID INT PRIMARY KEY,
  3      Name VARCHAR2(255),
  4      Bio CLOB
  5  );

Table created.

SQL> INSERT INTO Speaker VALUES (601, 'Dr. Arjun Malhotra', 'Renowned tech speaker and co-founder of HCL.');

1 row created.

SQL> INSERT INTO Speaker VALUES (602, 'Ms. Kavita Reddy', 'Entrepreneur and founder of a successful start-up ecosystem.');

1 row created.

SQL> INSERT INTO Speaker VALUES (603, 'Mr. Raghav Nair', 'Classical dancer and culture promoter.');

1 row created.

SQL> INSERT INTO Speaker VALUES (604, 'Dr. Meera Joshi', 'AI researcher and keynote speaker on future tech.');

1 row created.
```



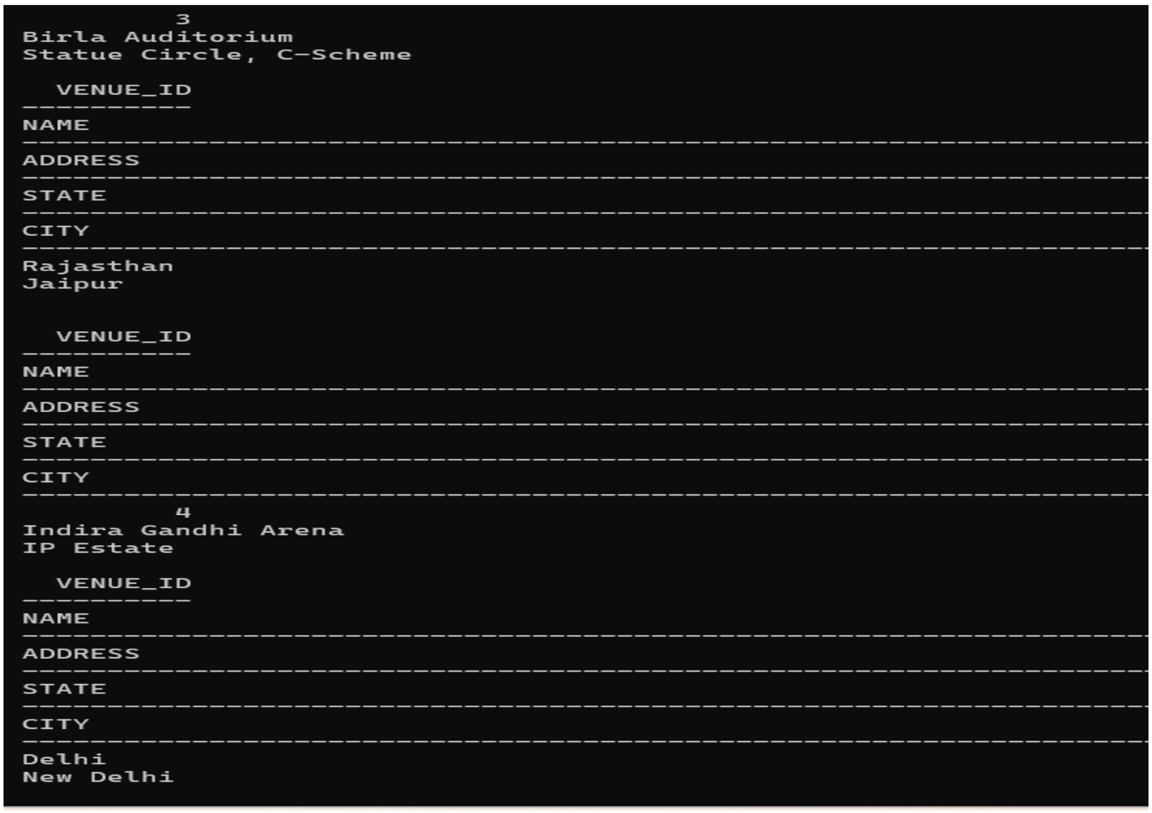
```
SQL> CREATE TABLE Staff (
  2      Staff_ID INT PRIMARY KEY,
  3      Name VARCHAR2(255),
  4      Role VARCHAR2(100),
  5      Event_ID INT,
  6      FOREIGN KEY (Event_ID) REFERENCES Event(Event_ID)
  7  );

Table created.

SQL> INSERT INTO Staff VALUES (701, 'Priya Deshmukh', 'Coordinator', 101);

1 row created.

SQL> INSERT INTO Staff VALUES (702, 'Sahil Joshi', 'Technician', 102);

1 row created.

SQL> INSERT INTO Staff VALUES (703, 'Neha Bansal', 'Event Manager', 103);

1 row created.

SQL> INSERT INTO Staff VALUES (704, 'Aditya Kapoor', 'Logistics Head', 104);

1 row created.

SQL>
```



```
SQL> SELECT * FROM Venue;

  VENUE_ID
----------
NAME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
STATE
--------------------------------------------------------------------------------
CITY
--------------------------------------------------------------------------------
         1
Lotus Convention Center
MG Road, Sector 14

  VENUE_ID
----------
NAME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
STATE
--------------------------------------------------------------------------------
CITY
--------------------------------------------------------------------------------
Maharashtra
Pune


  VENUE_ID
----------
```

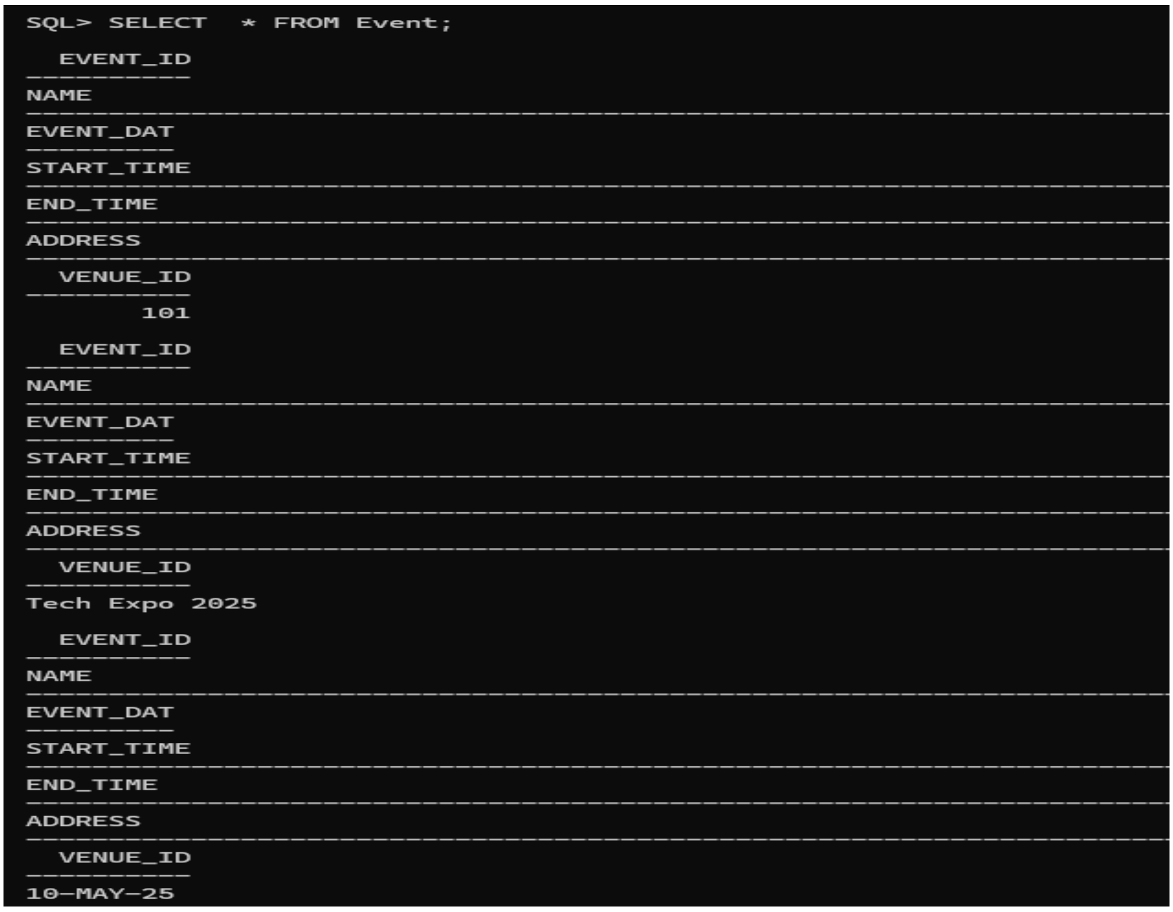


```
  VENUE_ID
----------
NAME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
STATE
--------------------------------------------------------------------------------
CITY
--------------------------------------------------------------------------------
         2
Nehru Indoor Stadium
Mount Road, Egmore

  VENUE_ID
----------
NAME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
STATE
--------------------------------------------------------------------------------
CITY
--------------------------------------------------------------------------------
Tamil Nadu
Chennai


  VENUE_ID
----------
NAME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
STATE
--------------------------------------------------------------------------------
CITY
--------------------------------------------------------------------------------
         3
Birla Auditorium
Statue Circle, C-Scheme

  VENUE_ID
----------
```

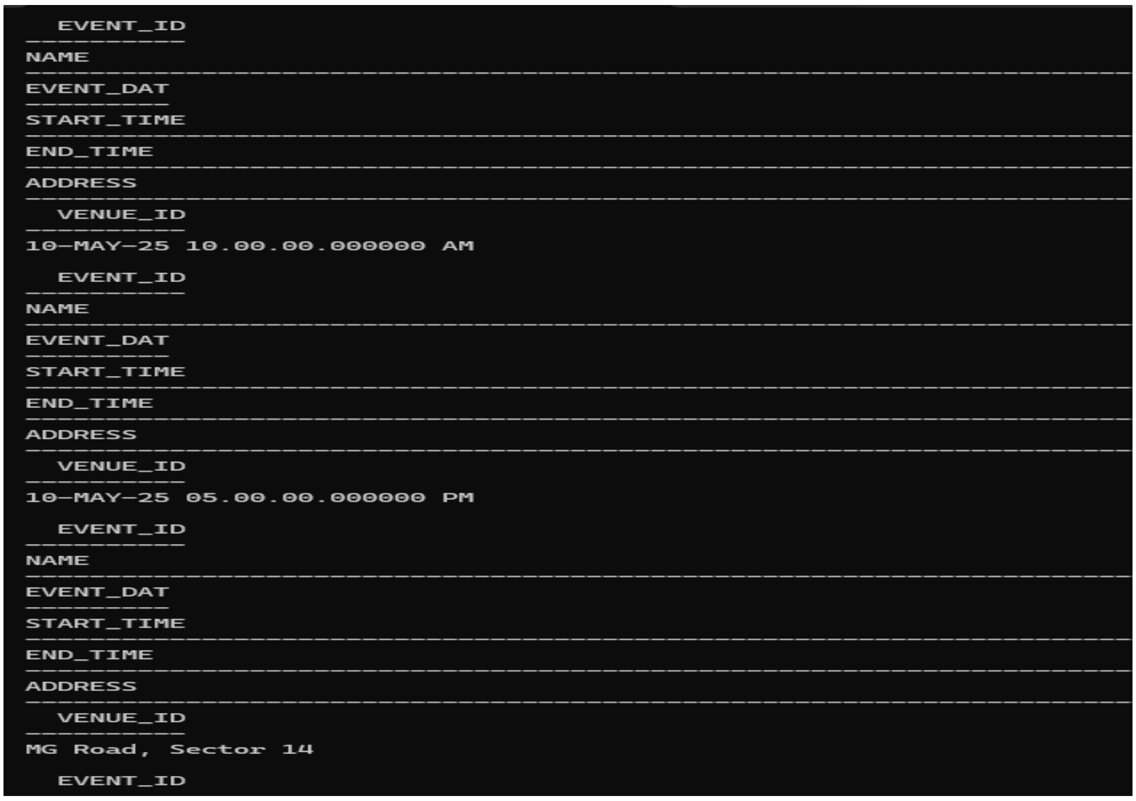


```
         3
Birla Auditorium
Statue Circle, C-Scheme

  VENUE_ID
----------
NAME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
STATE
--------------------------------------------------------------------------------
CITY
--------------------------------------------------------------------------------
Rajasthan
Jaipur


  VENUE_ID
----------
NAME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
STATE
--------------------------------------------------------------------------------
CITY
--------------------------------------------------------------------------------
         4
Indira Gandhi Arena
IP Estate

  VENUE_ID
----------
NAME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
STATE
--------------------------------------------------------------------------------
CITY
--------------------------------------------------------------------------------
Delhi
New Delhi
```

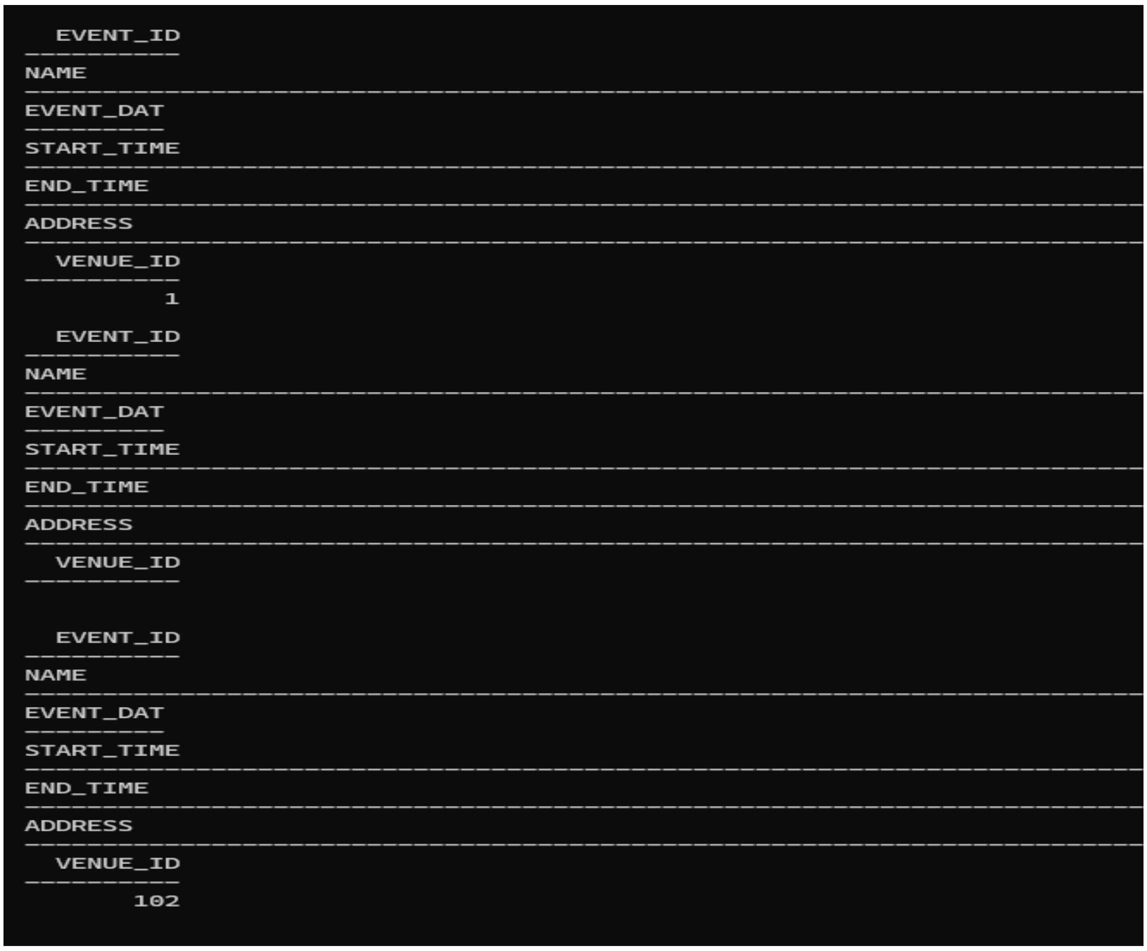
Table 1: Venue



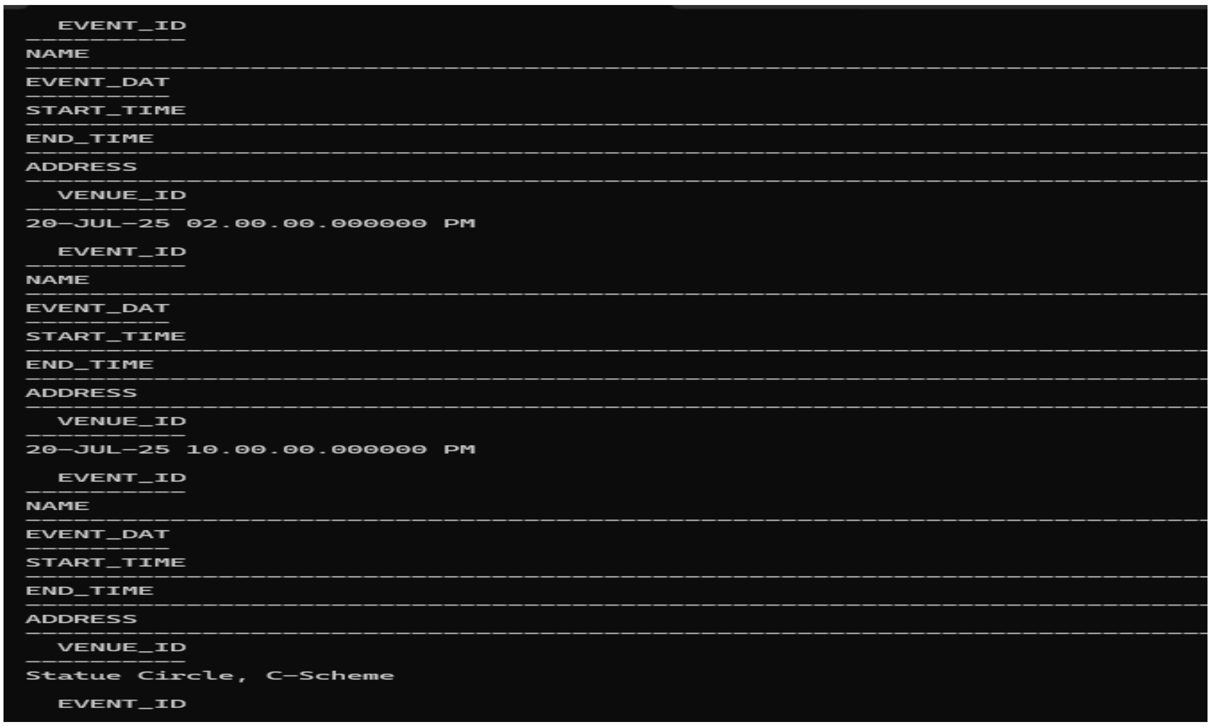
```
SQL> SELECT  * FROM Event;

  EVENT_ID
----------
NAME
--------------------------------------------------------------------------------
EVENT_DAT
---------
START_TIME
--------------------------------------------------------------------------------
END_TIME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
  VENUE_ID
----------
       101

  EVENT_ID
----------
NAME
--------------------------------------------------------------------------------
EVENT_DAT
---------
START_TIME
--------------------------------------------------------------------------------
END_TIME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
  VENUE_ID
----------
Tech Expo 2025

  EVENT_ID
----------
NAME
--------------------------------------------------------------------------------
EVENT_DAT
---------
START_TIME
--------------------------------------------------------------------------------
END_TIME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
  VENUE_ID
----------
10-MAY-25
```



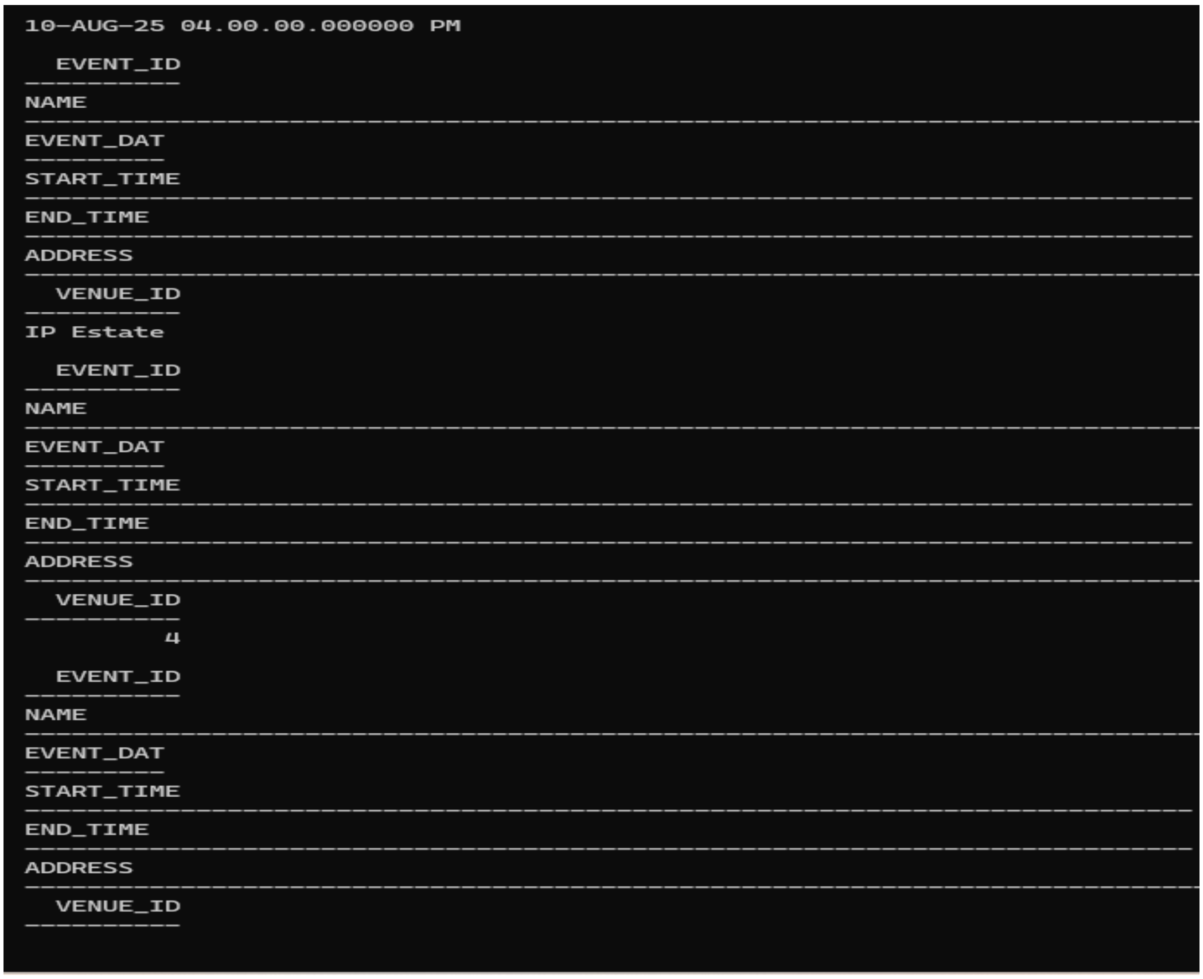
```
  EVENT_ID
----------
NAME
--------------------------------------------------------------------------------
EVENT_DAT
---------
START_TIME
--------------------------------------------------------------------------------
END_TIME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
  VENUE_ID
----------
10-MAY-25 10.00.00.000000 AM

  EVENT_ID
----------
NAME
--------------------------------------------------------------------------------
EVENT_DAT
---------
START_TIME
--------------------------------------------------------------------------------
END_TIME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
  VENUE_ID
----------
10-MAY-25 05.00.00.000000 PM

  EVENT_ID
----------
NAME
--------------------------------------------------------------------------------
EVENT_DAT
---------
START_TIME
--------------------------------------------------------------------------------
END_TIME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
  VENUE_ID
----------
MG Road, Sector 14

  EVENT_ID
```



```
  EVENT_ID
----------
NAME
--------------------------------------------------------------------------------
EVENT_DAT
---------
START_TIME
--------------------------------------------------------------------------------
END_TIME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
  VENUE_ID
----------
         1
  EVENT_ID
----------
NAME
--------------------------------------------------------------------------------
EVENT_DAT
---------
START_TIME
--------------------------------------------------------------------------------
END_TIME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
  VENUE_ID
----------

  EVENT_ID
----------
NAME
--------------------------------------------------------------------------------
EVENT_DAT
---------
START_TIME
--------------------------------------------------------------------------------
END_TIME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
  VENUE_ID
----------
       102
```

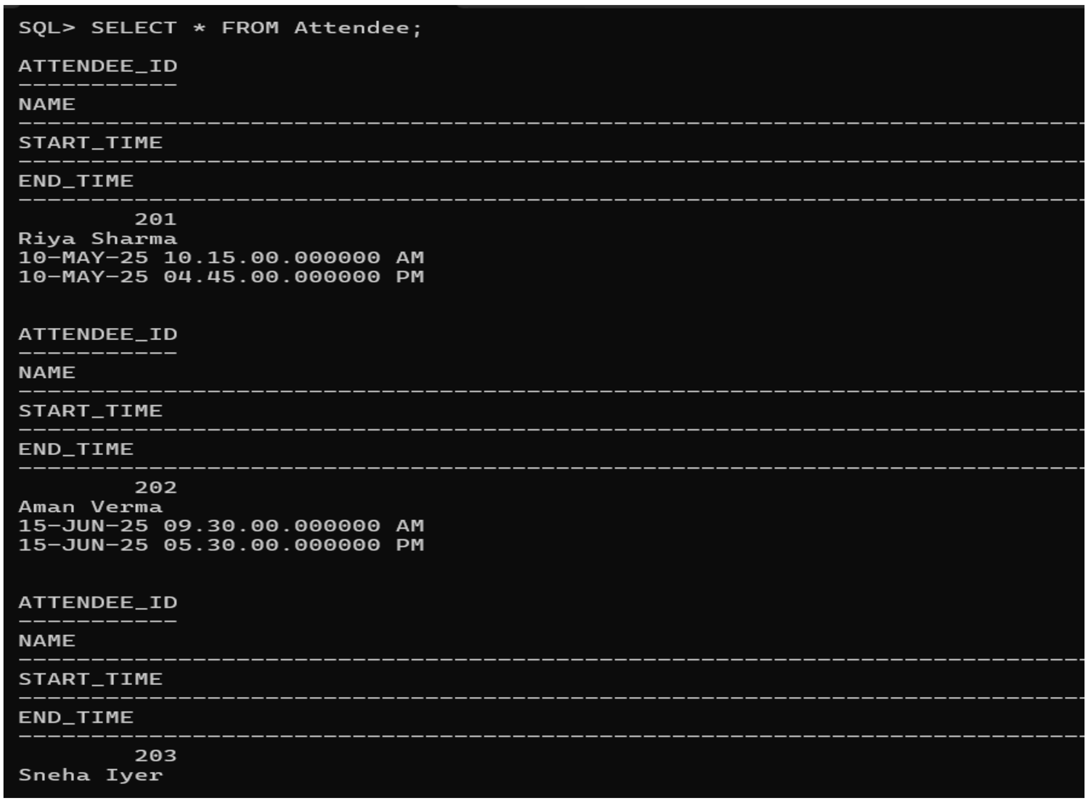


```
  EVENT_ID
----------
NAME
--------------------------------------------------------------------------------
EVENT_DAT
---------
START_TIME
--------------------------------------------------------------------------------
END_TIME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
  VENUE_ID
----------
20-JUL-25 02.00.00.000000 PM
  EVENT_ID
----------
NAME
--------------------------------------------------------------------------------
EVENT_DAT
---------
START_TIME
--------------------------------------------------------------------------------
END_TIME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
  VENUE_ID
----------
20-JUL-25 10.00.00.000000 PM
  EVENT_ID
----------
NAME
--------------------------------------------------------------------------------
EVENT_DAT
---------
START_TIME
--------------------------------------------------------------------------------
END_TIME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
  VENUE_ID
----------
Statue Circle, C-Scheme
  EVENT_ID
```

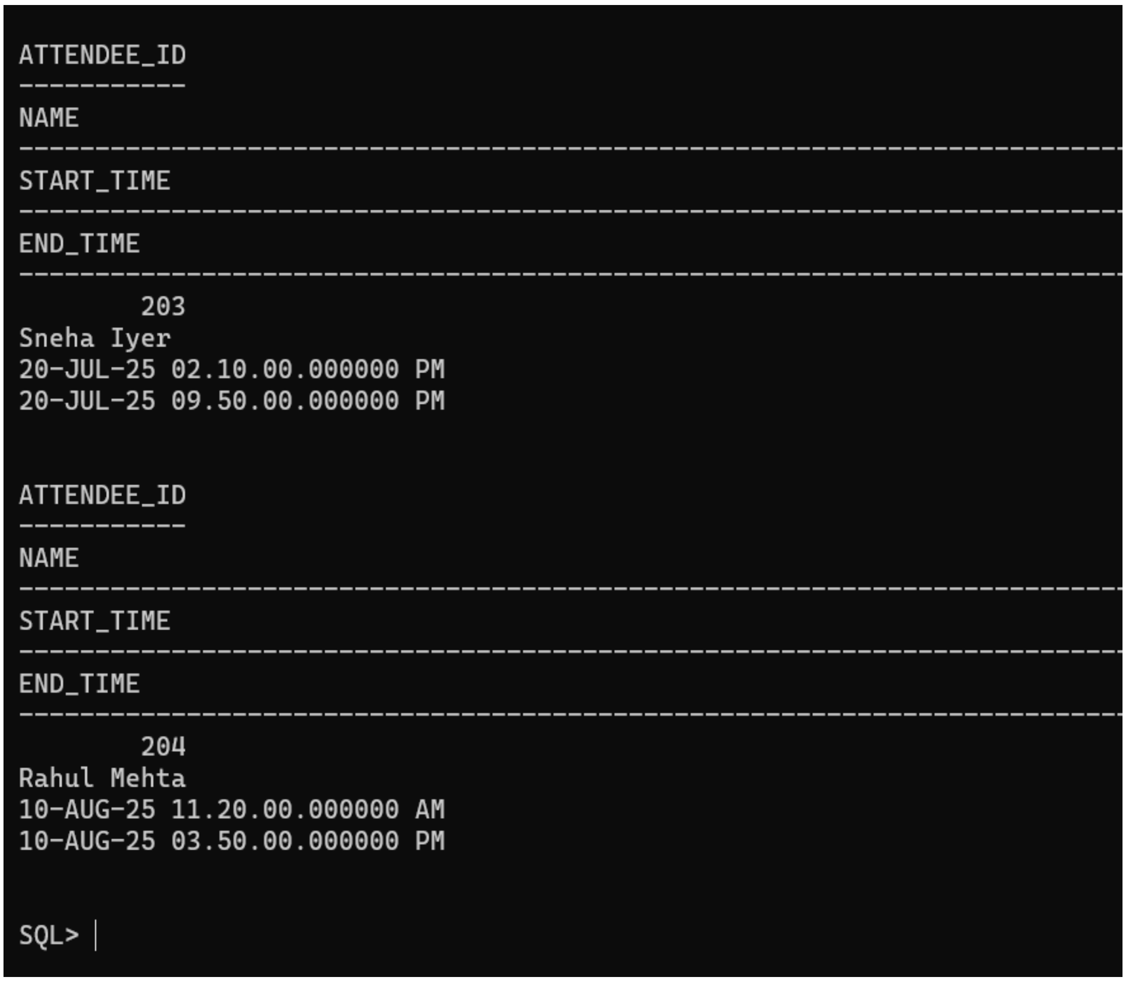


```
10-AUG-25 04.00.00.000000 PM

  EVENT_ID
----------
NAME
--------------------------------------------------------------------------------
EVENT_DAT
---------
START_TIME
--------------------------------------------------------------------------------
END_TIME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
  VENUE_ID
----------
IP Estate

  EVENT_ID
----------
NAME
--------------------------------------------------------------------------------
EVENT_DAT
---------
START_TIME
--------------------------------------------------------------------------------
END_TIME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
  VENUE_ID
----------
         4

  EVENT_ID
----------
NAME
--------------------------------------------------------------------------------
EVENT_DAT
---------
START_TIME
--------------------------------------------------------------------------------
END_TIME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
  VENUE_ID
----------
```

Table 2: Event

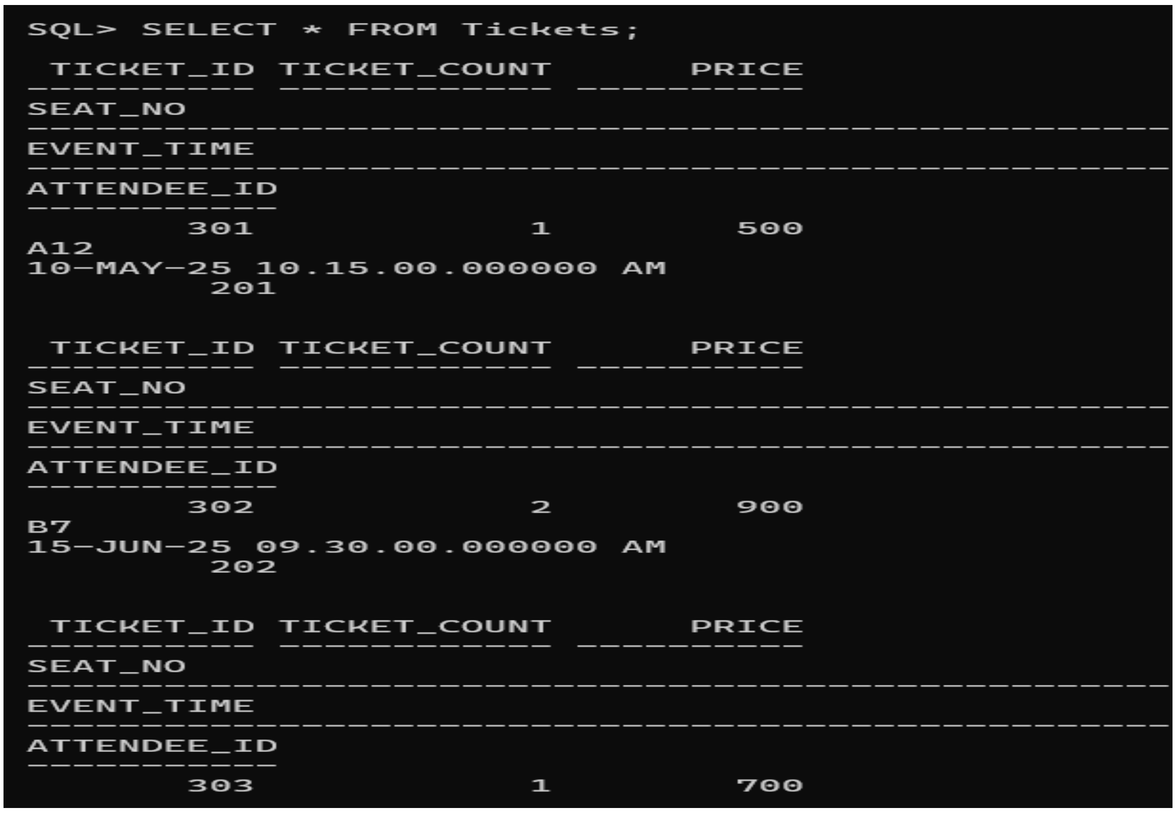


```
SQL> SELECT * FROM Attendee;

ATTENDEE_ID
-----------
NAME
--------------------------------------------------------------------------------
START_TIME
--------------------------------------------------------------------------------
END_TIME
--------------------------------------------------------------------------------
        201
Riya Sharma
10-MAY-25 10.15.00.000000 AM
10-MAY-25 04.45.00.000000 PM


ATTENDEE_ID
-----------
NAME
--------------------------------------------------------------------------------
START_TIME
--------------------------------------------------------------------------------
END_TIME
--------------------------------------------------------------------------------
        202
Aman Verma
15-JUN-25 09.30.00.000000 AM
15-JUN-25 05.30.00.000000 PM


ATTENDEE_ID
-----------
NAME
--------------------------------------------------------------------------------
START_TIME
--------------------------------------------------------------------------------
END_TIME
--------------------------------------------------------------------------------
        203
Sneha Iyer
```

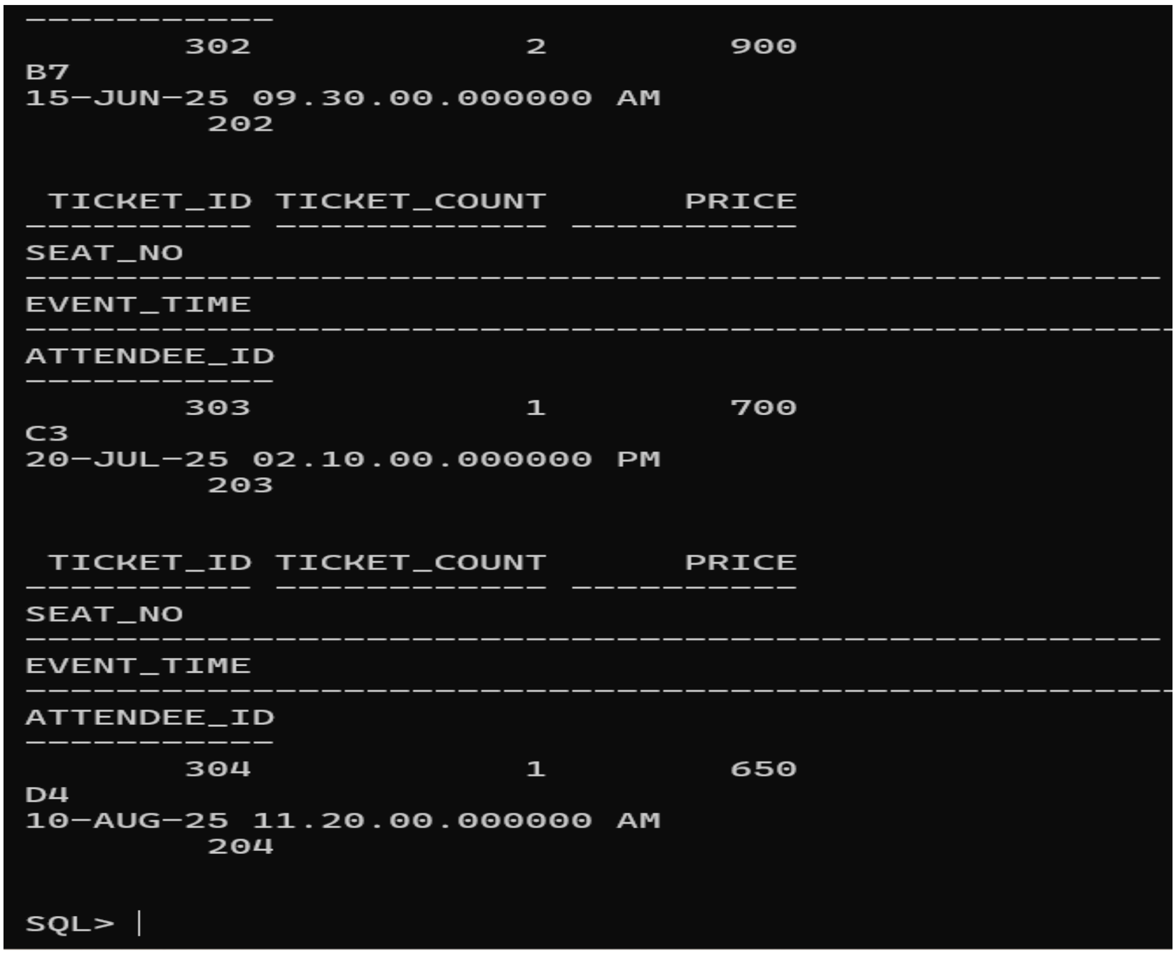


```
ATTENDEE_ID
-----------
NAME
--------------------------------------------------------------------------------
START_TIME
--------------------------------------------------------------------------------
END_TIME
--------------------------------------------------------------------------------
        203
Sneha Iyer
20-JUL-25 02.10.00.000000 PM
20-JUL-25 09.50.00.000000 PM


ATTENDEE_ID
-----------
NAME
--------------------------------------------------------------------------------
START_TIME
--------------------------------------------------------------------------------
END_TIME
--------------------------------------------------------------------------------
        204
Rahul Mehta
10-AUG-25 11.20.00.000000 AM
10-AUG-25 03.50.00.000000 PM


SQL>
```

Table 3: Attendee

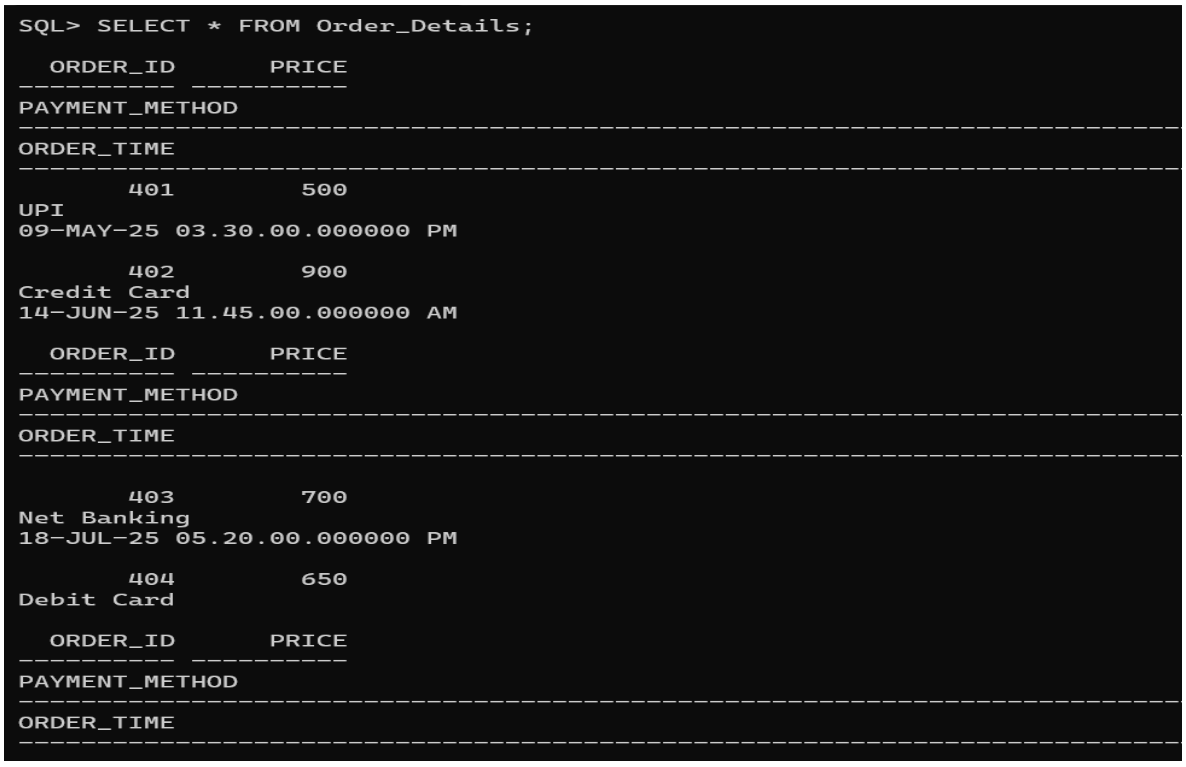


```
SQL> SELECT * FROM Tickets;

 TICKET_ID TICKET_COUNT      PRICE
---------- ------------ ----------
SEAT_NO
--------------------------------------------------
EVENT_TIME
--------------------------------------------------
ATTENDEE_ID
-----------
       301            1        500
A12
10-MAY-25 10.15.00.000000 AM
        201

 TICKET_ID TICKET_COUNT      PRICE
---------- ------------ ----------
SEAT_NO
--------------------------------------------------
EVENT_TIME
--------------------------------------------------
ATTENDEE_ID
-----------
       302            2        900
B7
15-JUN-25 09.30.00.000000 AM
        202

 TICKET_ID TICKET_COUNT      PRICE
---------- ------------ ----------
SEAT_NO
--------------------------------------------------
EVENT_TIME
--------------------------------------------------
ATTENDEE_ID
-----------
       303            1        700
```

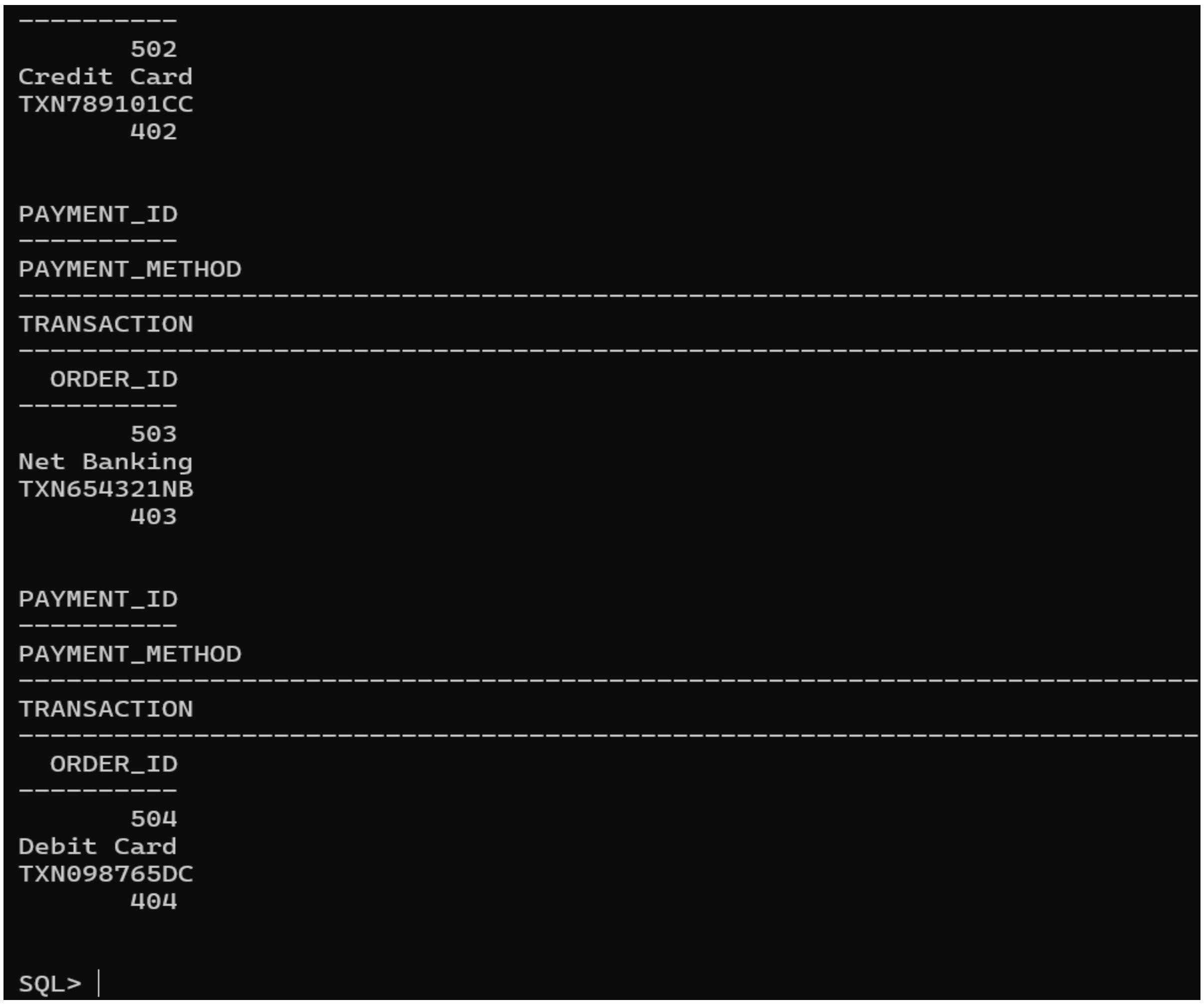


```
-----------
       302              2         900
B7
15-JUN-25 09.30.00.000000 AM
        202

 TICKET_ID TICKET_COUNT       PRICE
---------- ------------ ----------
SEAT_NO
--------------------------------------------------
EVENT_TIME
---------------------------------------------------
ATTENDEE_ID
-----------
       303              1         700
C3
20-JUL-25 02.10.00.000000 PM
        203

 TICKET_ID TICKET_COUNT       PRICE
---------- ------------ ----------
SEAT_NO
--------------------------------------------------
EVENT_TIME
---------------------------------------------------
ATTENDEE_ID
-----------
       304              1         650
D4
10-AUG-25 11.20.00.000000 AM
        204


SQL> |
```

Table 4: Tickets

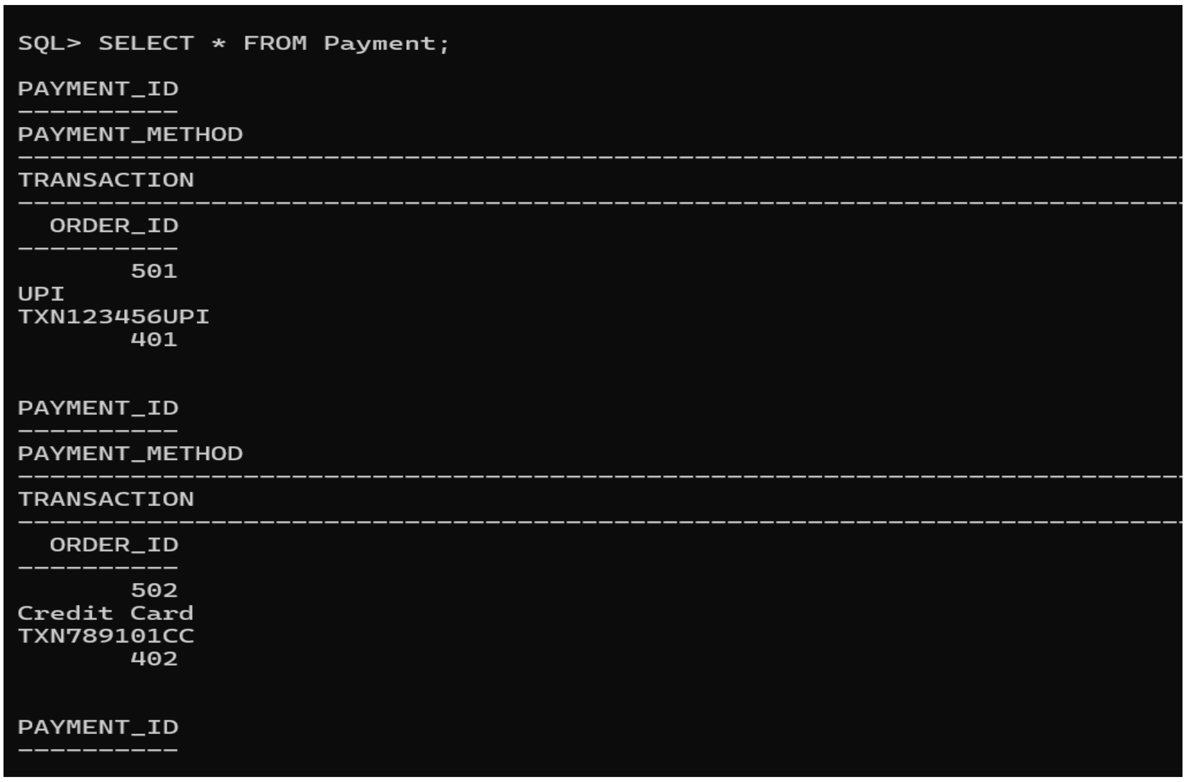


```
SQL> SELECT * FROM Order_Details;

  ORDER_ID      PRICE
---------- ----------
PAYMENT_METHOD
------------------------------------------------------------------------------
ORDER_TIME
------------------------------------------------------------------------------
       401        500
UPI
09-MAY-25 03.30.00.000000 PM

       402        900
Credit Card
14-JUN-25 11.45.00.000000 AM

  ORDER_ID      PRICE
---------- ----------
PAYMENT_METHOD
------------------------------------------------------------------------------
ORDER_TIME
------------------------------------------------------------------------------

       403        700
Net Banking
18-JUL-25 05.20.00.000000 PM

       404        650
Debit Card

  ORDER_ID      PRICE
---------- ----------
PAYMENT_METHOD
------------------------------------------------------------------------------
ORDER_TIME
------------------------------------------------------------------------------
```

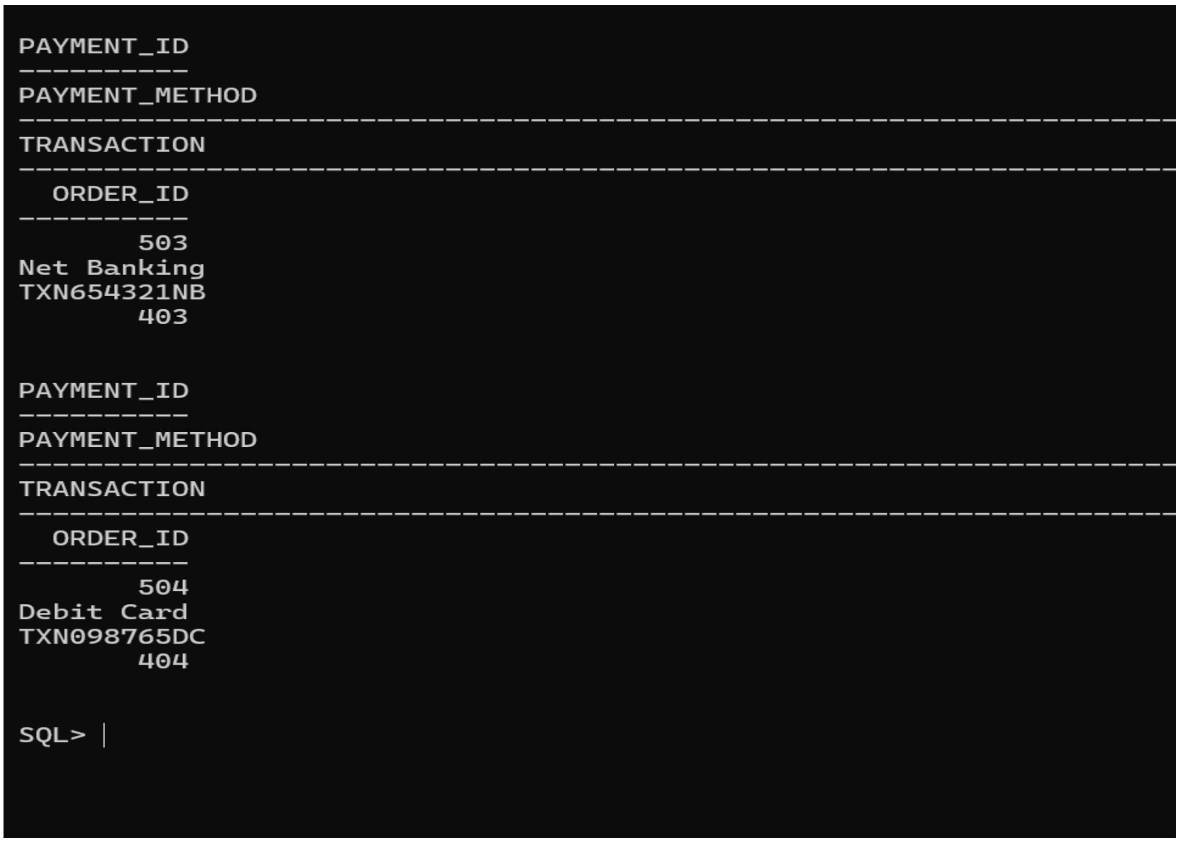


```
----------
       502
Credit Card
TXN789101CC
       402


PAYMENT_ID
----------
PAYMENT_METHOD
--------------------------------------------------------------------------
TRANSACTION
--------------------------------------------------------------------------
  ORDER_ID
----------
       503
Net Banking
TXN654321NB
       403


PAYMENT_ID
----------
PAYMENT_METHOD
--------------------------------------------------------------------------
TRANSACTION
--------------------------------------------------------------------------
  ORDER_ID
----------
       504
Debit Card
TXN098765DC
       404


SQL>
```

Table 5: Order_Details

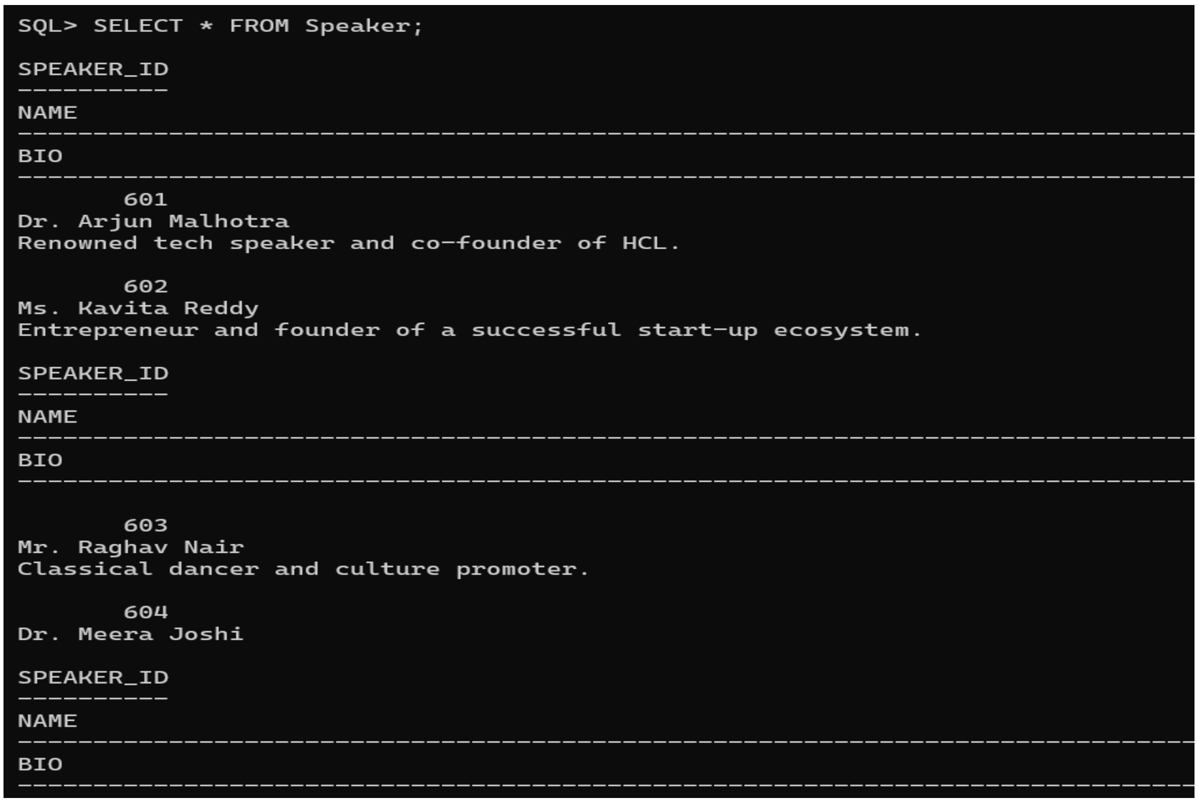


```
SQL> SELECT * FROM Payment;

PAYMENT_ID
----------
PAYMENT_METHOD
--------------------------------------------------------------------------
TRANSACTION
--------------------------------------------------------------------------
  ORDER_ID
----------
       501
UPI
TXN123456UPI
       401


PAYMENT_ID
----------
PAYMENT_METHOD
--------------------------------------------------------------------------
TRANSACTION
--------------------------------------------------------------------------
  ORDER_ID
----------
       502
Credit Card
TXN789101CC
       402


PAYMENT_ID
----------
```



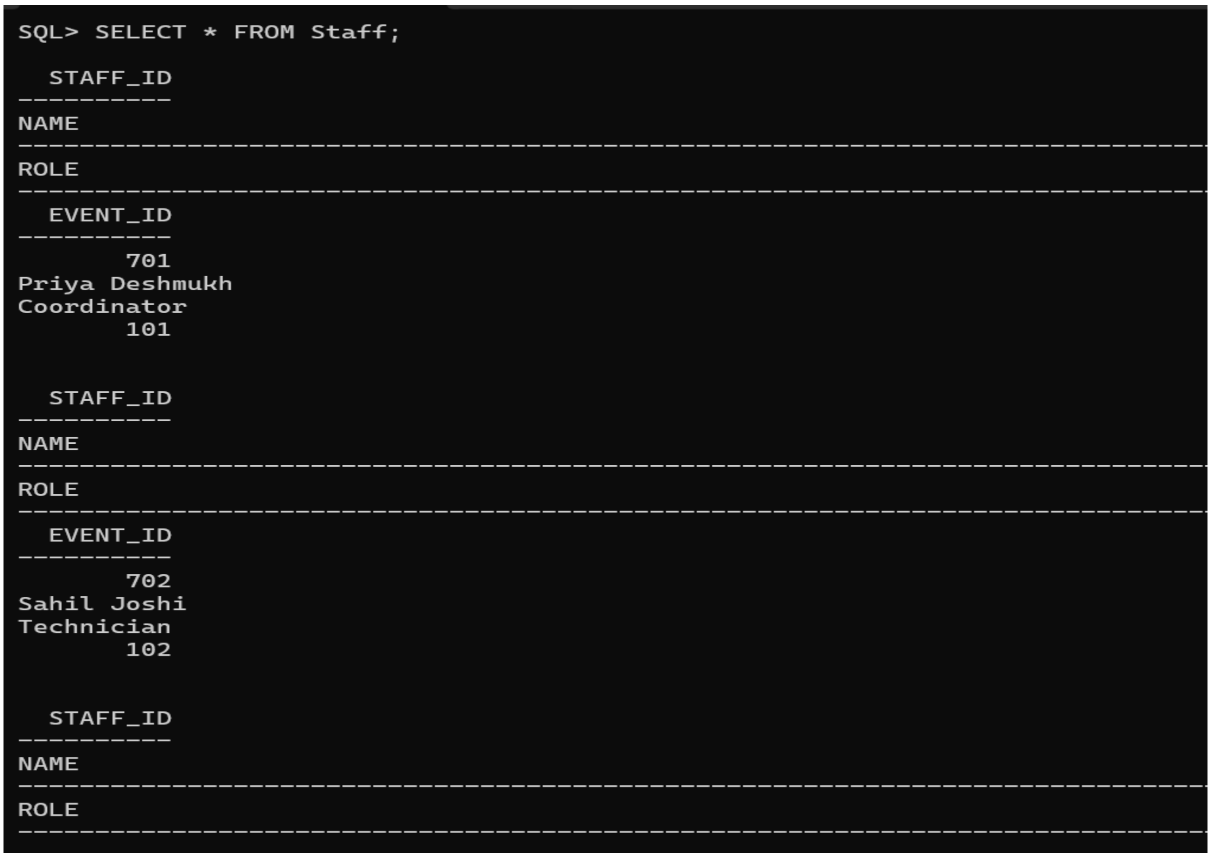
```
PAYMENT_ID
----------
PAYMENT_METHOD
--------------------------------------------------------------------------------
TRANSACTION
--------------------------------------------------------------------------------
  ORDER_ID
----------
       503
Net Banking
TXN654321NB
       403


PAYMENT_ID
----------
PAYMENT_METHOD
--------------------------------------------------------------------------------
TRANSACTION
--------------------------------------------------------------------------------
  ORDER_ID
----------
       504
Debit Card
TXN098765DC
       404


SQL> |
```

Table 6: Payment

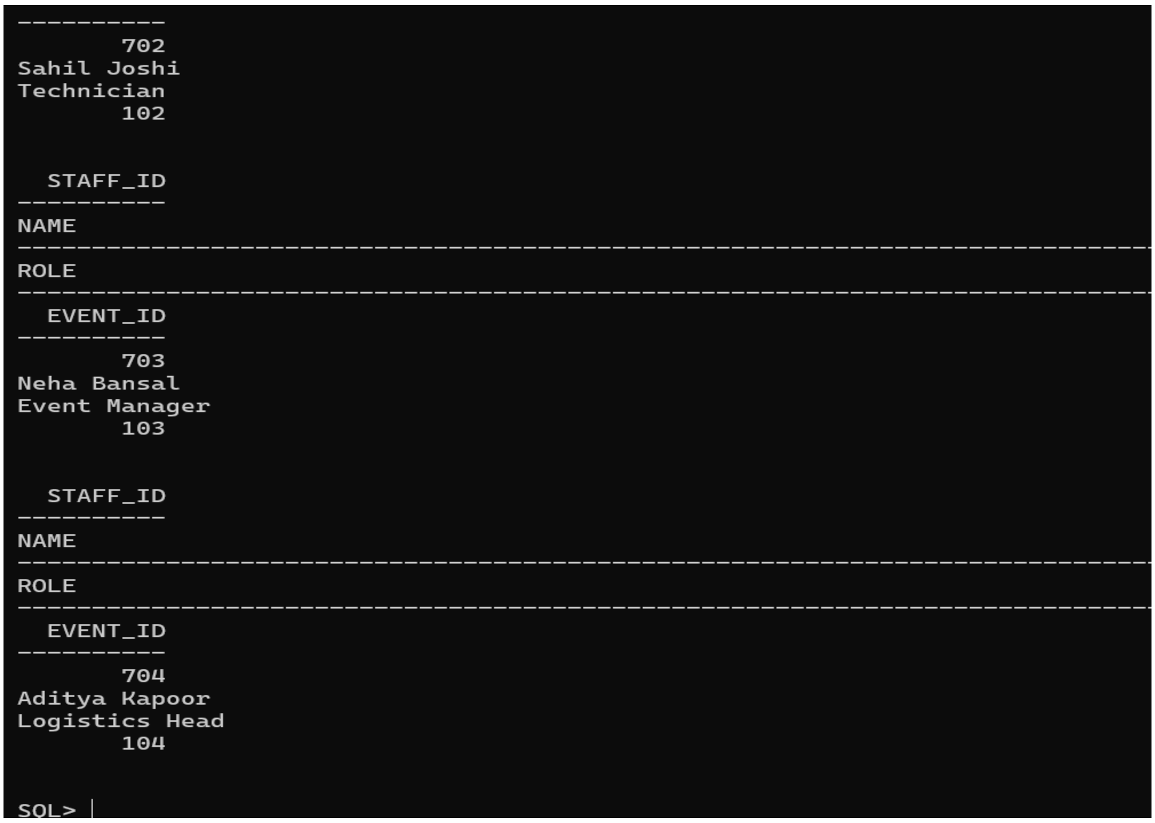


```
SQL> SELECT * FROM Speaker;

SPEAKER_ID
----------
NAME
--------------------------------------------------------------------------------
BIO
--------------------------------------------------------------------------------
       601
Dr. Arjun Malhotra
Renowned tech speaker and co-founder of HCL.

       602
Ms. Kavita Reddy
Entrepreneur and founder of a successful start-up ecosystem.

SPEAKER_ID
----------
NAME
--------------------------------------------------------------------------------
BIO
--------------------------------------------------------------------------------

       603
Mr. Raghav Nair
Classical dancer and culture promoter.

       604
Dr. Meera Joshi

SPEAKER_ID
----------
NAME
--------------------------------------------------------------------------------
BIO
--------------------------------------------------------------------------------
```

Table 7: Speaker

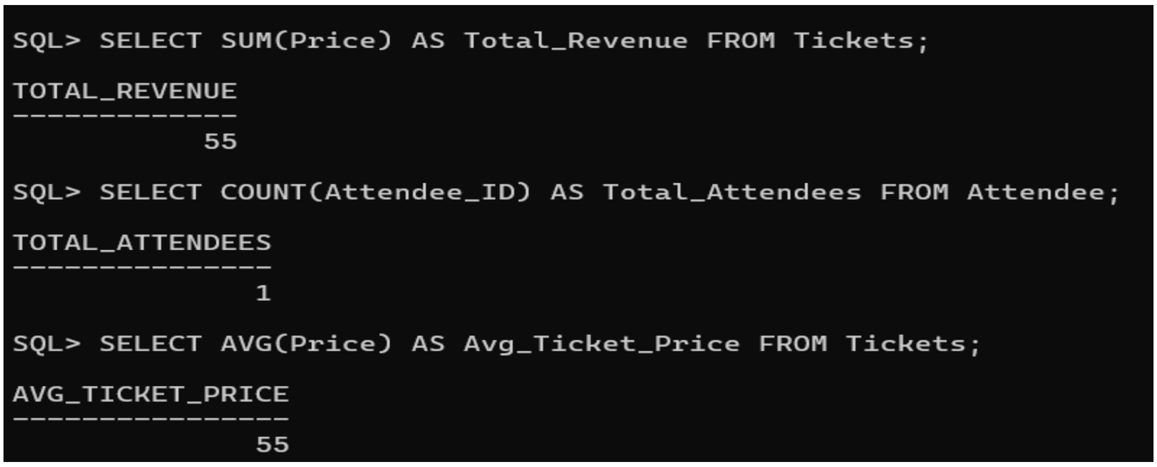


```
SQL> SELECT * FROM Staff;

  STAFF_ID
----------
NAME
-----------------------------------------------------------------------------
ROLE
-----------------------------------------------------------------------------
  EVENT_ID
----------
       701
Priya Deshmukh
Coordinator
       101


  STAFF_ID
----------
NAME
-----------------------------------------------------------------------------
ROLE
-----------------------------------------------------------------------------
  EVENT_ID
----------
       702
Sahil Joshi
Technician
       102


  STAFF_ID
----------
NAME
-----------------------------------------------------------------------------
ROLE
-----------------------------------------------------------------------------
```

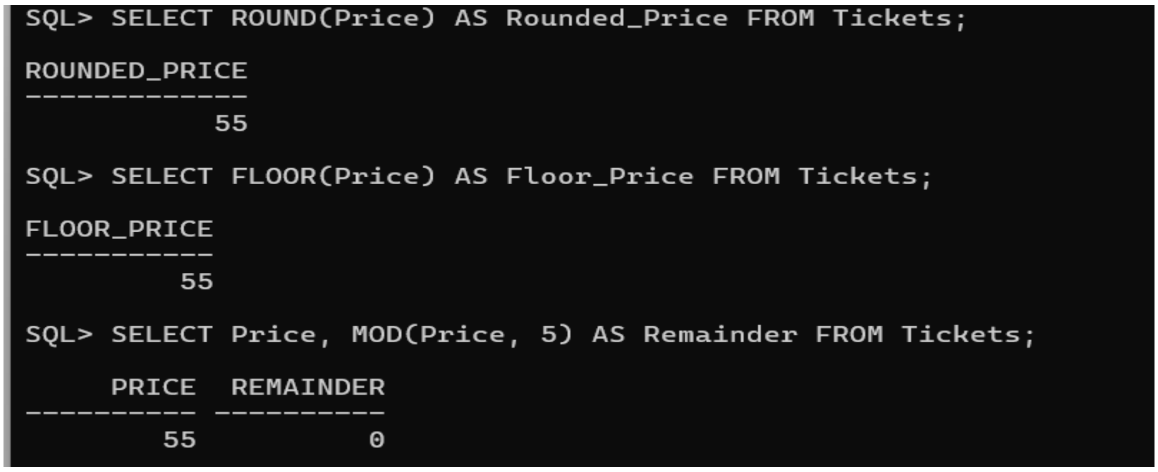


```
----------
       702
Sahil Joshi
Technician
       102


  STAFF_ID
----------
NAME
--------------------------------------------------------------------------
ROLE
--------------------------------------------------------------------------
  EVENT_ID
----------
       703
Neha Bansal
Event Manager
       103


  STAFF_ID
----------
NAME
--------------------------------------------------------------------------
ROLE
--------------------------------------------------------------------------
  EVENT_ID
----------
       704
Aditya Kapoor
Logistics Head
       104


SQL> 
```

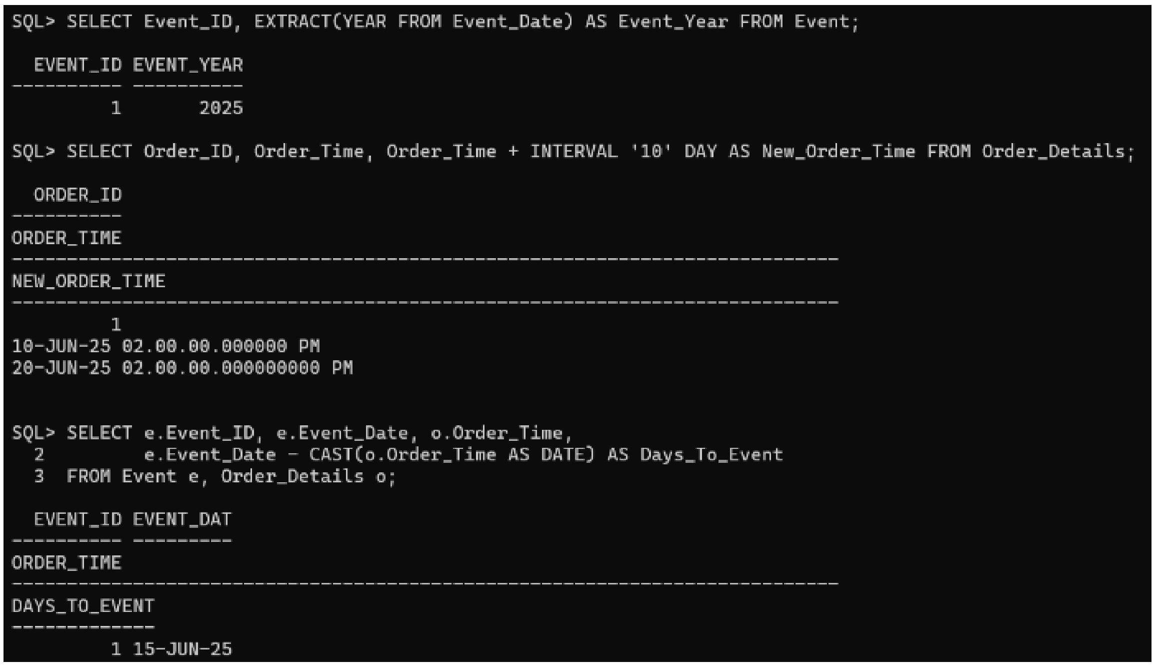
3.Write and execute SQL functions- aggregate, numeric, date, string, and conversion.



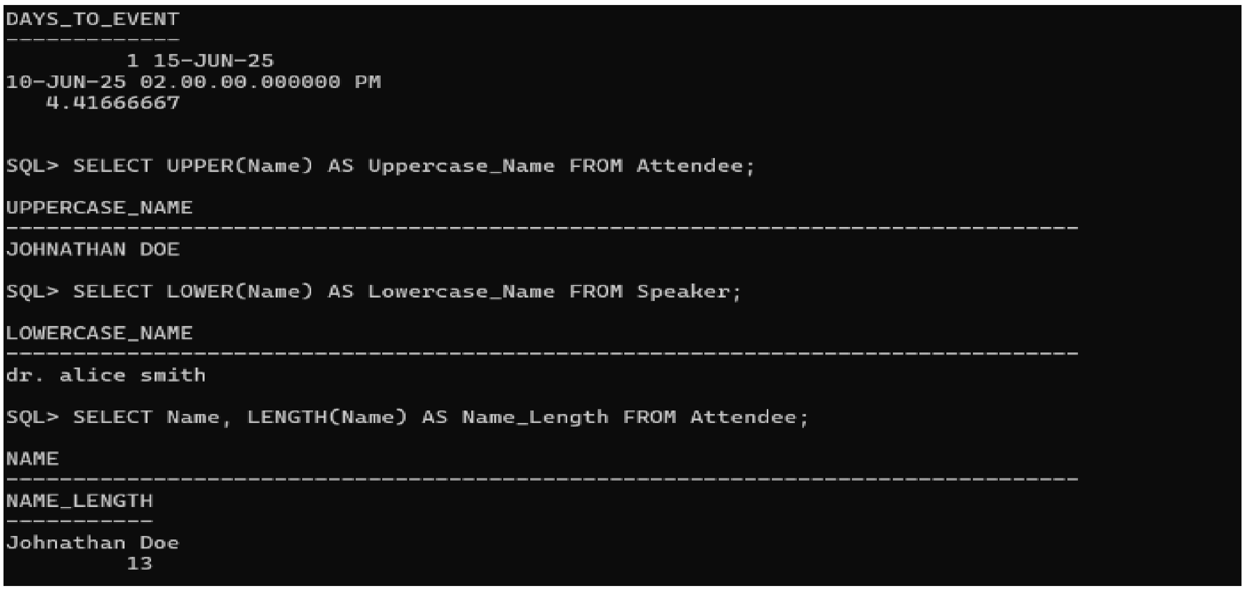
```
SQL> SELECT SUM(Price) AS Total_Revenue FROM Tickets;

TOTAL_REVENUE
-------------
           55

SQL> SELECT COUNT(Attendee_ID) AS Total_Attendees FROM Attendee;

TOTAL_ATTENDEES
---------------
              1

SQL> SELECT AVG(Price) AS Avg_Ticket_Price FROM Tickets;

AVG_TICKET_PRICE
----------------
              55
```

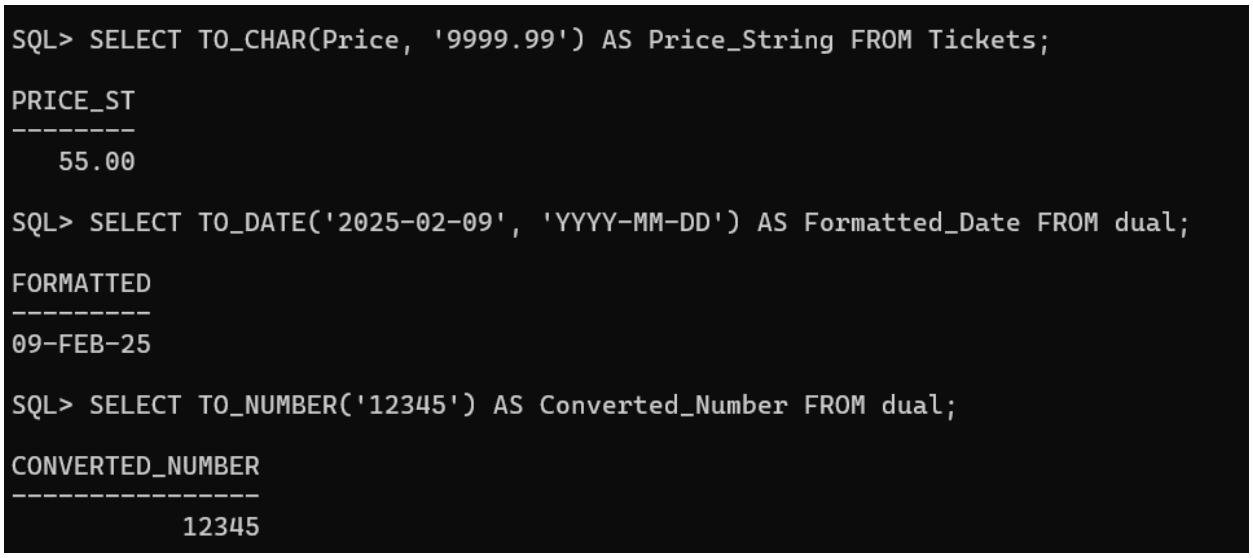


```
SQL> SELECT ROUND(Price) AS Rounded_Price FROM Tickets;

ROUNDED_PRICE
-------------
           55

SQL> SELECT FLOOR(Price) AS Floor_Price FROM Tickets;

FLOOR_PRICE
-----------
         55

SQL> SELECT Price, MOD(Price, 5) AS Remainder FROM Tickets;

     PRICE  REMAINDER
---------- ----------
        55          0
```

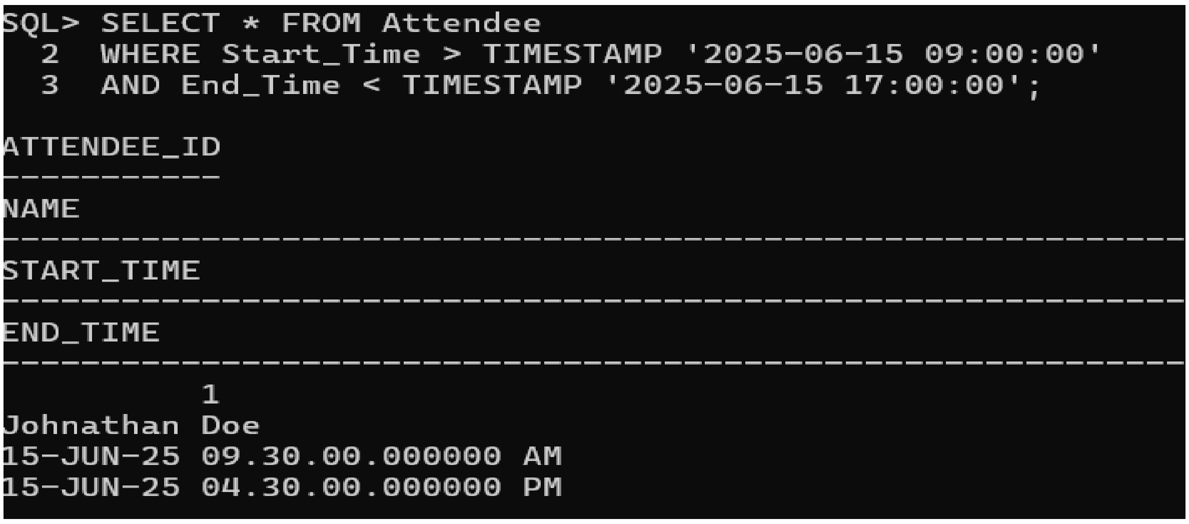


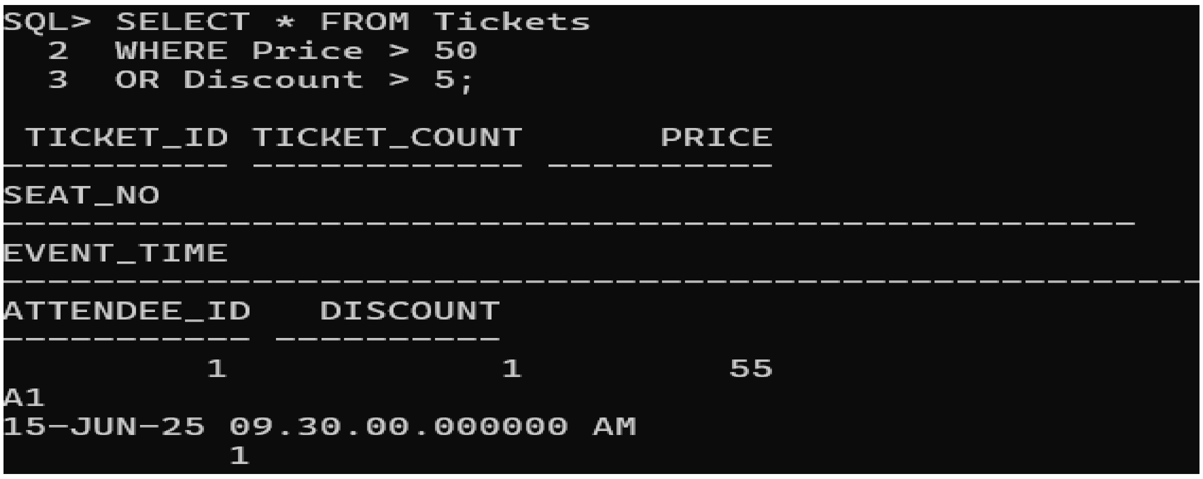
```
SQL> SELECT Event_ID, EXTRACT(YEAR FROM Event_Date) AS Event_Year FROM Event;

  EVENT_ID EVENT_YEAR
---------- ----------
         1       2025

SQL> SELECT Order_ID, Order_Time, Order_Time + INTERVAL '10' DAY AS New_Order_Time FROM Order_Details;

  ORDER_ID
----------
ORDER_TIME
---------------------------------------------------------------------------
NEW_ORDER_TIME
---------------------------------------------------------------------------
         1
10-JUN-25 02.00.00.000000 PM
20-JUN-25 02.00.00.000000000 PM


SQL> SELECT e.Event_ID, e.Event_Date, o.Order_Time,
  2         e.Event_Date - CAST(o.Order_Time AS DATE) AS Days_To_Event
  3  FROM Event e, Order_Details o;

  EVENT_ID EVENT_DAT
---------- ---------
ORDER_TIME
---------------------------------------------------------------------------
DAYS_TO_EVENT
-------------
         1 15-JUN-25
```


```
DAYS_TO_EVENT
-------------
         1 15-JUN-25
10-JUN-25 02.00.00.000000 PM
   4.41666667


SQL> SELECT UPPER(Name) AS Uppercase_Name FROM Attendee;

UPPERCASE_NAME
--------------------------------------------------------------------------------
JOHNATHAN DOE

SQL> SELECT LOWER(Name) AS Lowercase_Name FROM Speaker;

LOWERCASE_NAME
--------------------------------------------------------------------------------
dr. alice smith

SQL> SELECT Name, LENGTH(Name) AS Name_Length FROM Attendee;

NAME
--------------------------------------------------------------------------------
NAME_LENGTH
-----------
Johnathan Doe
         13
```

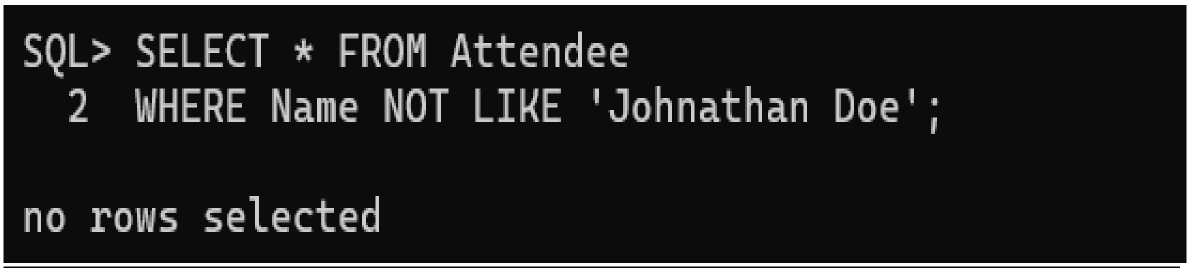

```
SQL> SELECT TO_CHAR(Price, '9999.99') AS Price_String FROM Tickets;

PRICE_ST
--------
   55.00

SQL> SELECT TO_DATE('2025-02-09', 'YYYY-MM-DD') AS Formatted_Date FROM dual;

FORMATTED
---------
09-FEB-25

SQL> SELECT TO_NUMBER('12345') AS Converted_Number FROM dual;

CONVERTED_NUMBER
----------------
           12345
```

4. Write and execute SQL queries- Operators (and, or, not, like, between, in)

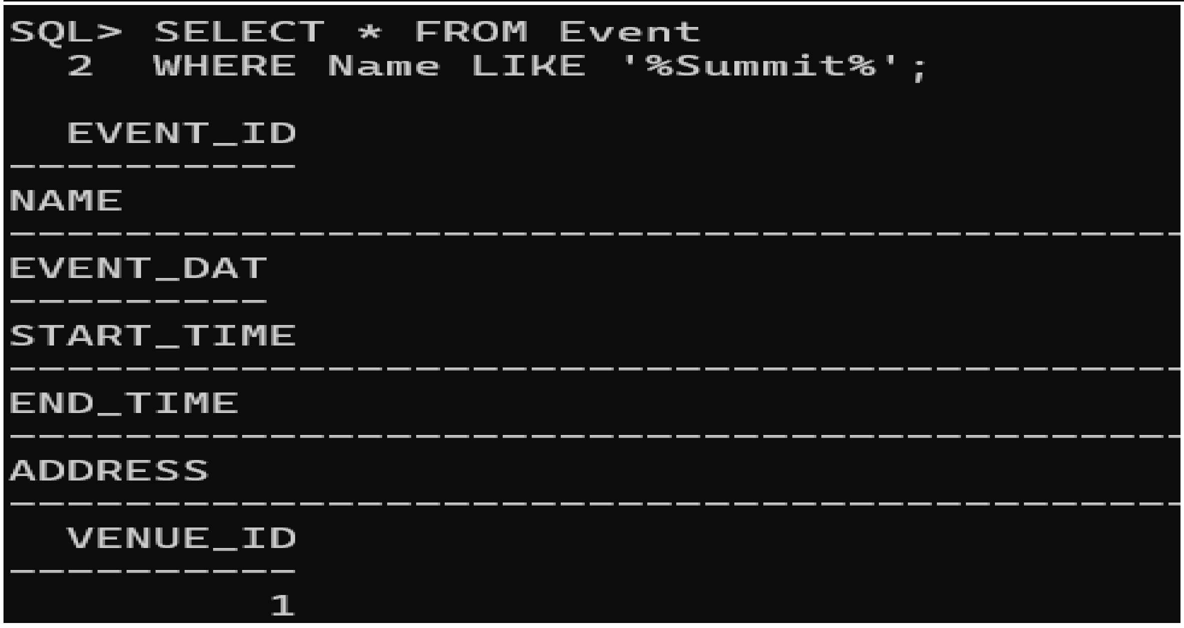

```
SQL> SELECT * FROM Attendee
  2  WHERE Start_Time > TIMESTAMP '2025-06-15 09:00:00'
  3  AND End_Time < TIMESTAMP '2025-06-15 17:00:00';

ATTENDEE_ID
-----------
NAME
-----------------------------------------------------------
START_TIME
-----------------------------------------------------------
END_TIME
-----------------------------------------------------------
          1
Johnathan Doe
15-JUN-25 09.30.00.000000 AM
15-JUN-25 04.30.00.000000 PM
```



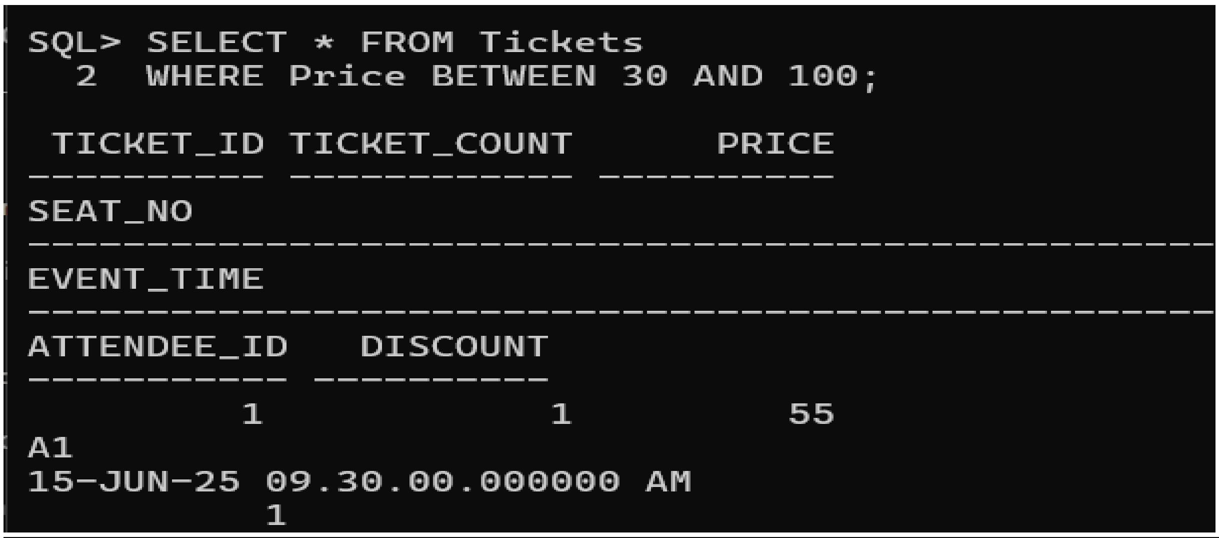
```
SQL> SELECT * FROM Tickets
  2  WHERE Price > 50
  3  OR Discount > 5;

 TICKET_ID TICKET_COUNT      PRICE
---------- ------------ ----------
SEAT_NO
--------------------------------------------------
EVENT_TIME
---------------------------------------------------------------------------
ATTENDEE_ID   DISCOUNT
----------- ----------
         1            1         55
A1
15-JUN-25 09.30.00.000000 AM
          1
```



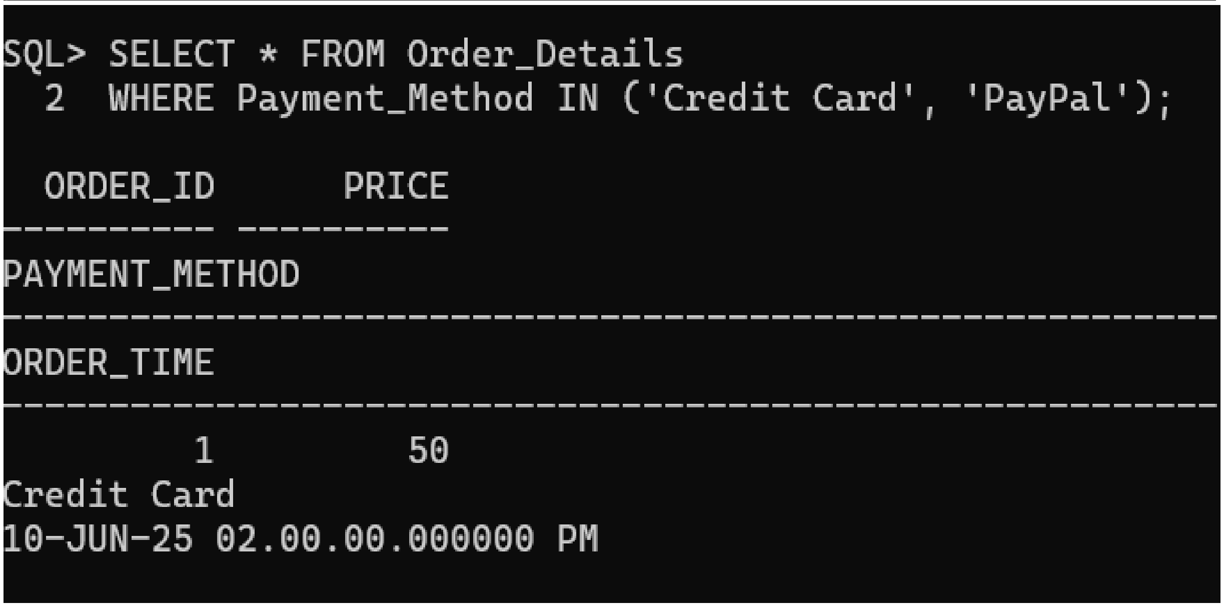
```
SQL> SELECT * FROM Attendee
  2  WHERE Name NOT LIKE 'Johnathan Doe';

no rows selected
```



```
SQL> SELECT * FROM Event
  2  WHERE Name LIKE '%Summit%';

  EVENT_ID
----------
NAME
--------------------------------------------------------------------------------
EVENT_DAT
---------
START_TIME
--------------------------------------------------------------------------------
END_TIME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
  VENUE_ID
----------
         1
```

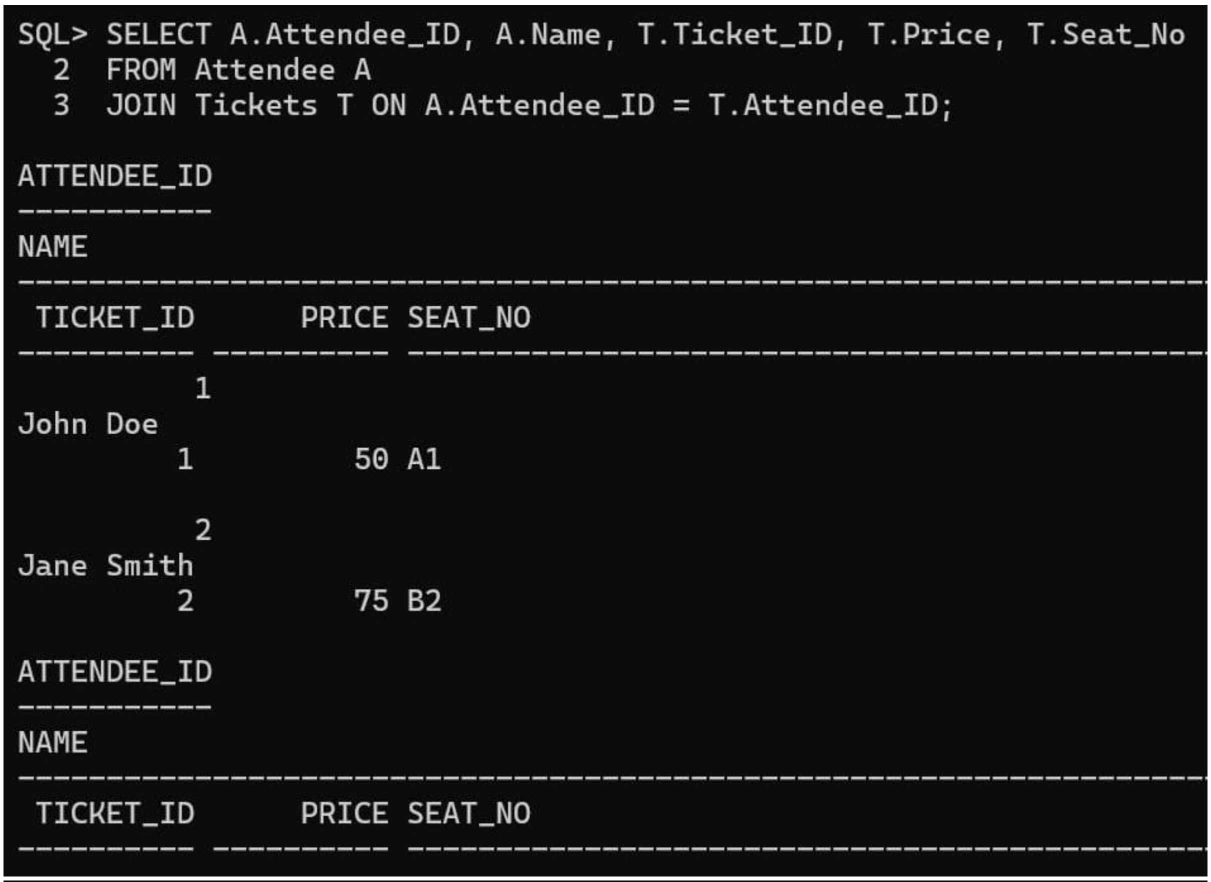


```
SQL> SELECT * FROM Tickets
  2  WHERE Price BETWEEN 30 AND 100;

 TICKET_ID TICKET_COUNT      PRICE
---------- ------------ ----------
SEAT_NO
--------------------------------------------------
EVENT_TIME
--------------------------------------------------
ATTENDEE_ID   DISCOUNT
----------- ----------
         1            1         55
A1
15-JUN-25 09.30.00.000000 AM
          1
```

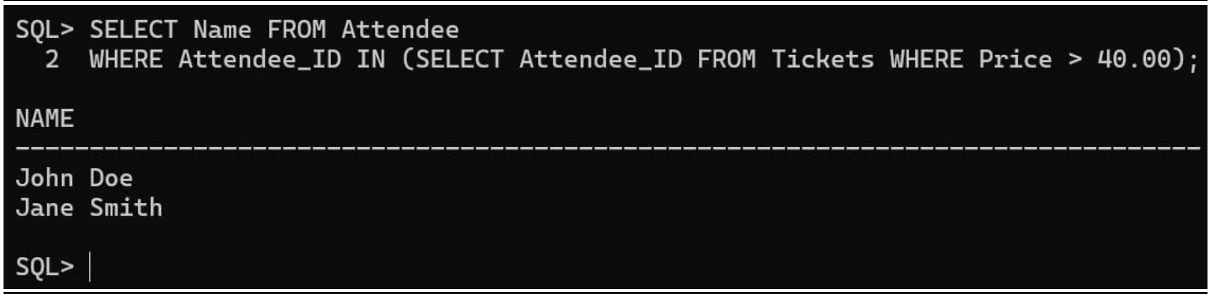


```
SQL> SELECT * FROM Order_Details
  2  WHERE Payment_Method IN ('Credit Card', 'PayPal');

  ORDER_ID      PRICE
---------- ----------
PAYMENT_METHOD
--------------------------------------------------
ORDER_TIME
--------------------------------------------------
         1         50
Credit Card
10-JUN-25 02.00.00.000000 PM
```

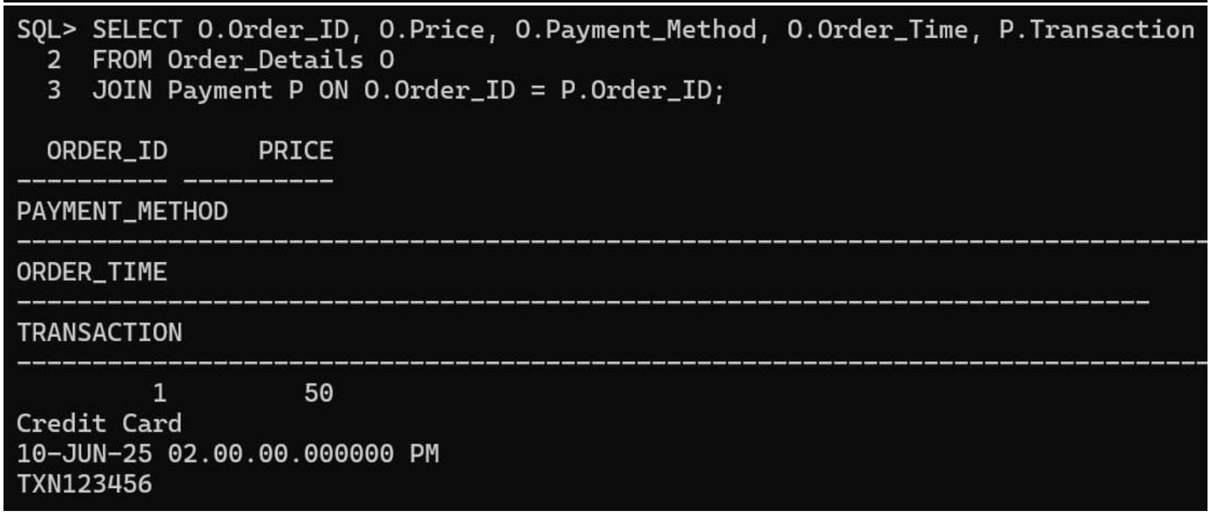
5.Write and execute SQL queries- subqueries, joins.



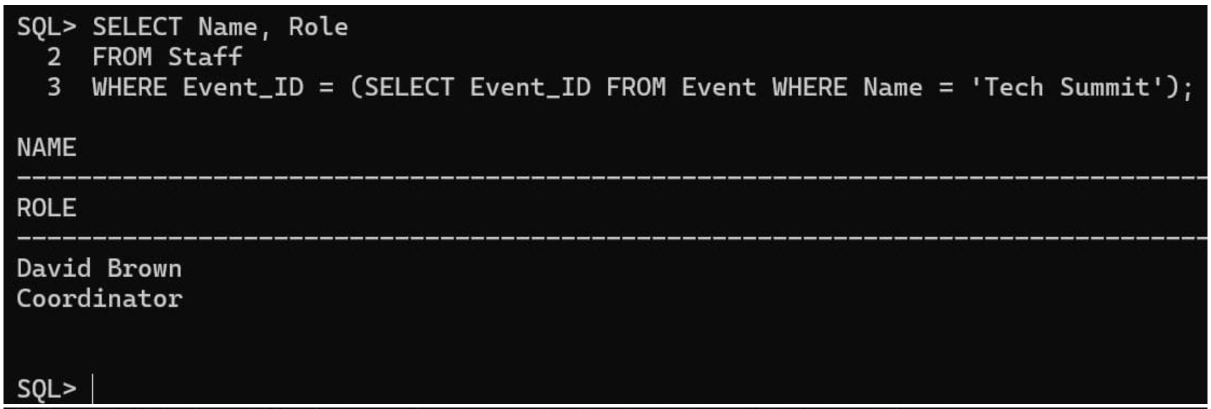
```
SQL> SELECT A.Attendee_ID, A.Name, T.Ticket_ID, T.Price, T.Seat_No
     FROM Attendee A
     JOIN Tickets T ON A.Attendee_ID = T.Attendee_ID;

ATTENDEE_ID
-----------
NAME
--------------------------------------------------------------------------------
 TICKET_ID      PRICE SEAT_NO
---------- ---------- --------------------------------------------------------
          1
John Doe
         1         50 A1

          2
Jane Smith
         2         75 B2

ATTENDEE_ID
-----------
NAME
--------------------------------------------------------------------------------
 TICKET_ID      PRICE SEAT_NO
---------- ---------- --------------------------------------------------------
```



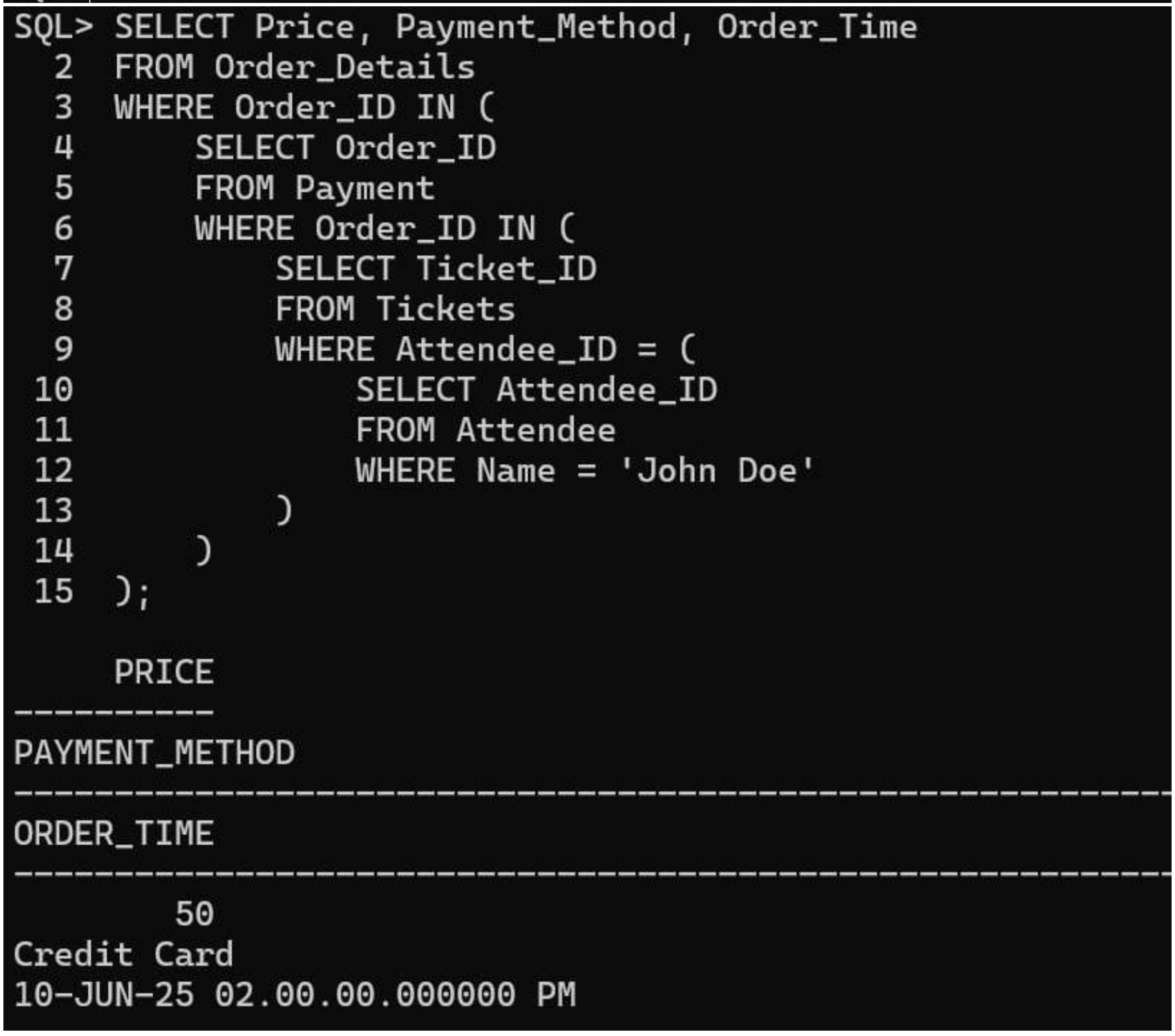
```
SQL> SELECT Name FROM Attendee
     WHERE Attendee_ID IN (SELECT Attendee_ID FROM Tickets WHERE Price > 40.00);

NAME
--------------------------------------------------------------------------------
John Doe
Jane Smith

SQL>
```



```
SQL> SELECT O.Order_ID, O.Price, O.Payment_Method, O.Order_Time, P.Transaction
     FROM Order_Details O
     JOIN Payment P ON O.Order_ID = P.Order_ID;

  ORDER_ID      PRICE
---------- ----------
PAYMENT_METHOD
--------------------------------------------------------------------------------
ORDER_TIME
---------------------------------------------------------------------------
TRANSACTION
--------------------------------------------------------------------------------
         1         50
Credit Card
10-JUN-25 02.00.00.000000 PM
TXN123456
```

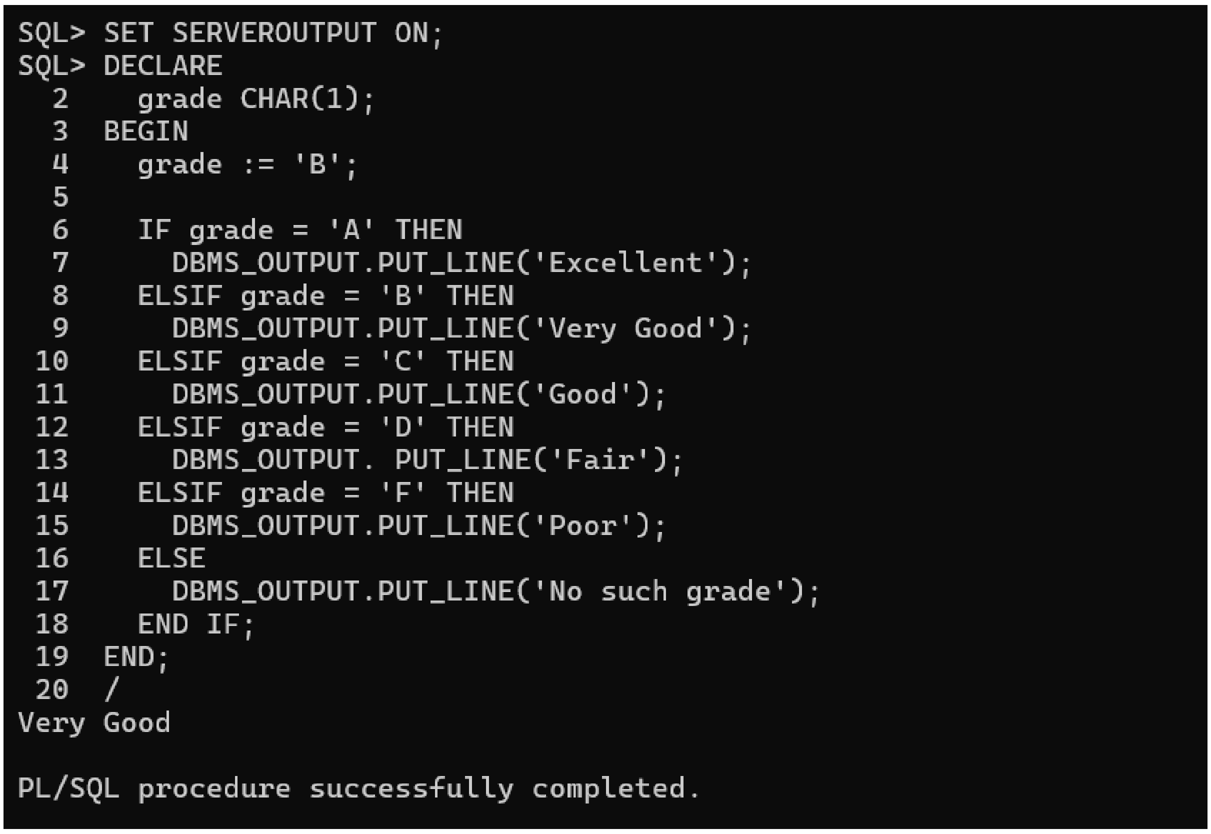


```
SQL> SELECT Name, Role
     FROM Staff
     WHERE Event_ID = (SELECT Event_ID FROM Event WHERE Name = 'Tech Summit');

NAME
--------------------------------------------------------------------------------
ROLE
--------------------------------------------------------------------------------
David Brown
Coordinator


SQL>
```



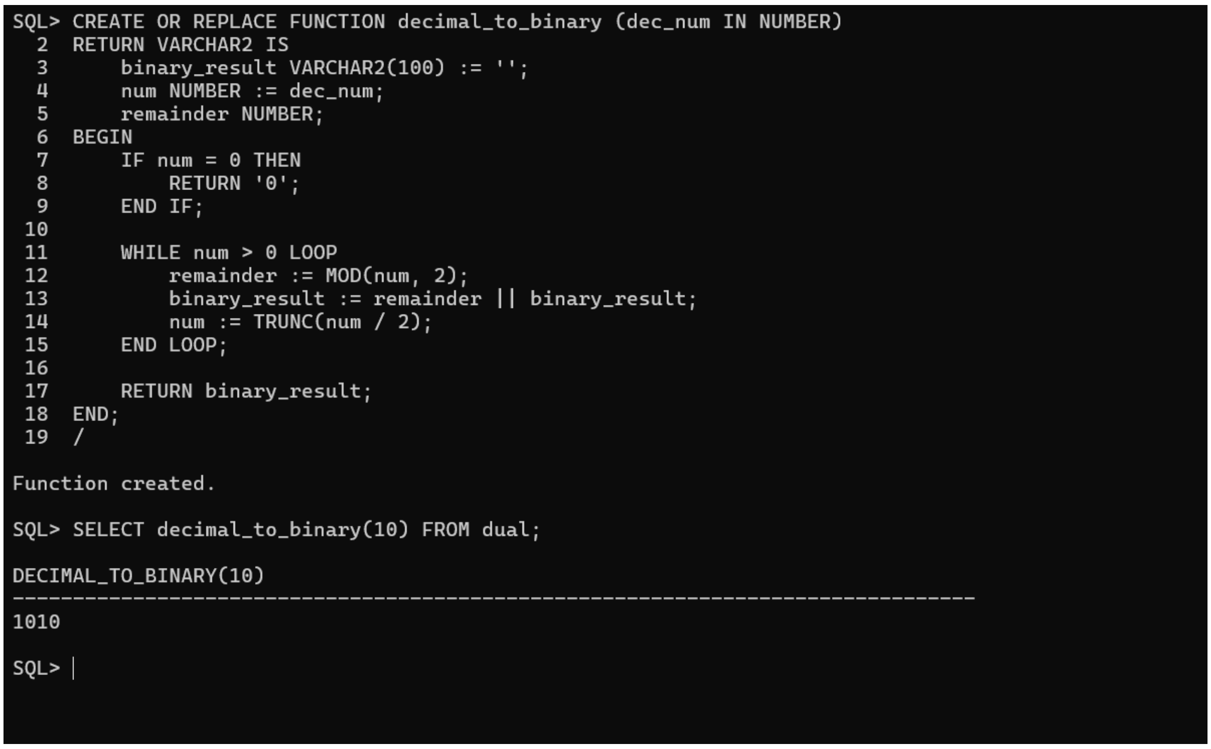
```
SQL> SELECT Price, Payment_Method, Order_Time
     FROM Order_Details
     WHERE Order_ID IN (
         SELECT Order_ID
         FROM Payment
         WHERE Order_ID IN (
             SELECT Ticket_ID
             FROM Tickets
             WHERE Attendee_ID = (
                 SELECT Attendee_ID
                 FROM Attendee
                 WHERE Name = 'John Doe'
             )
         )
     );

     PRICE
----------
PAYMENT_METHOD
----------------------------------------------------------
ORDER_TIME
----------------------------------------------------------
        50
Credit Card
10-JUN-25 02.00.00.000000 PM
```

6. Write and execute basic Pl/SQL programs - simple program, condition statements and loops.



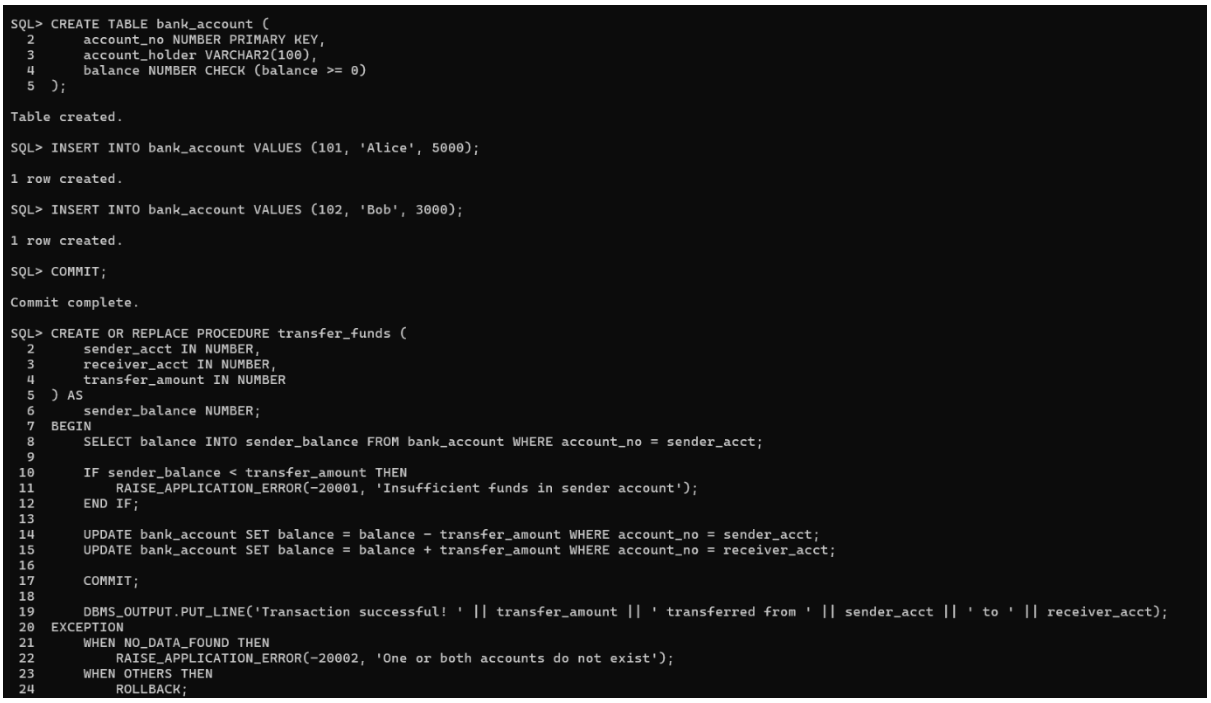
```
SQL> SET SERVEROUTPUT ON;
SQL> DECLARE
    grade CHAR(1);
  BEGIN
    grade := 'B';

    IF grade = 'A' THEN
      DBMS_OUTPUT.PUT_LINE('Excellent');
    ELSIF grade = 'B' THEN
      DBMS_OUTPUT.PUT_LINE('Very Good');
    ELSIF grade = 'C' THEN
      DBMS_OUTPUT.PUT_LINE('Good');
    ELSIF grade = 'D' THEN
      DBMS_OUTPUT. PUT_LINE('Fair');
    ELSIF grade = 'F' THEN
      DBMS_OUTPUT.PUT_LINE('Poor');
    ELSE
      DBMS_OUTPUT.PUT_LINE('No such grade');
    END IF;
  END;
  /
Very Good

PL/SQL procedure successfully completed.
```

7.Write and execute Pl/SQL function to print /return binary equivalent of decimal number.

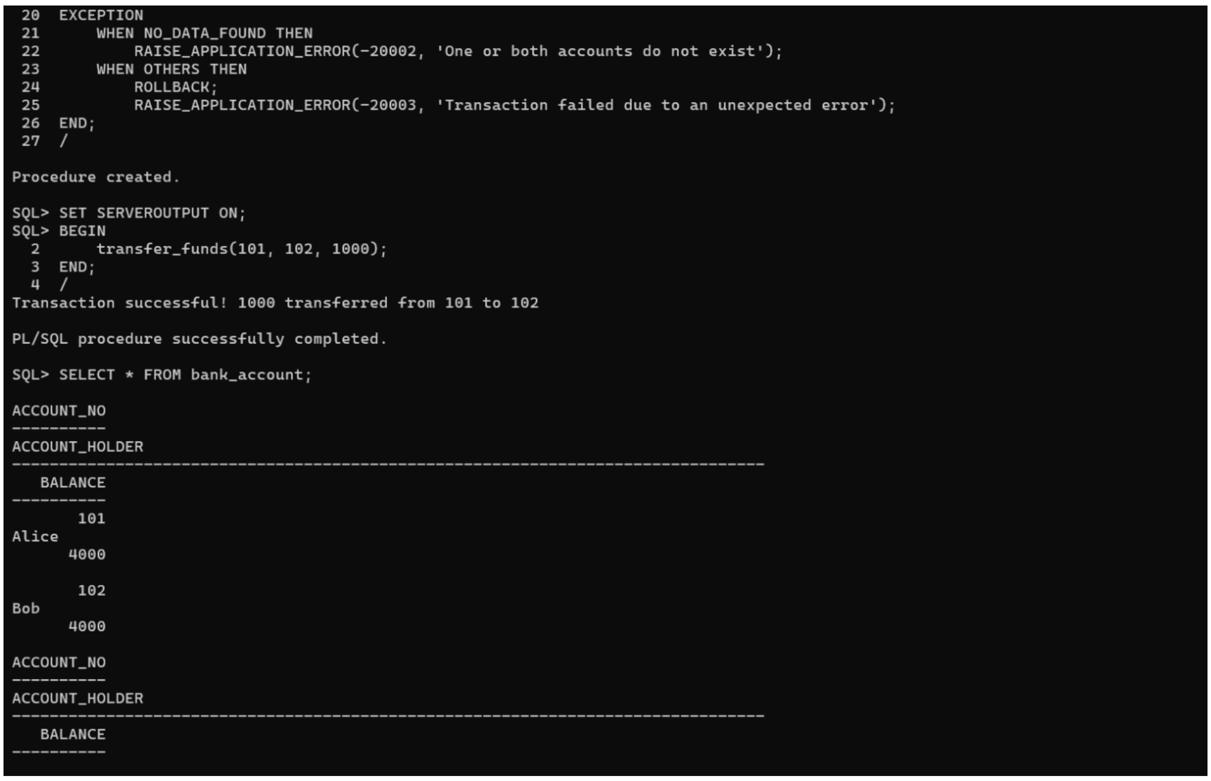


```
SQL> CREATE OR REPLACE FUNCTION decimal_to_binary (dec_num IN NUMBER)
  RETURN VARCHAR2 IS
      binary_result VARCHAR2(100) := '';
      num NUMBER := dec_num;
      remainder NUMBER;
  BEGIN
      IF num = 0 THEN
          RETURN '0';
      END IF;

      WHILE num > 0 LOOP
          remainder := MOD(num, 2);
          binary_result := remainder || binary_result;
          num := TRUNC(num / 2);
      END LOOP;

      RETURN binary_result;
  END;
  /

Function created.

SQL> SELECT decimal_to_binary(10) FROM dual;

DECIMAL_TO_BINARY(10)
--------------------------------------------------------------------------------
1010

SQL> |
```

8. Write and execute PL/SQL procedure to transfer fund from one account to another.

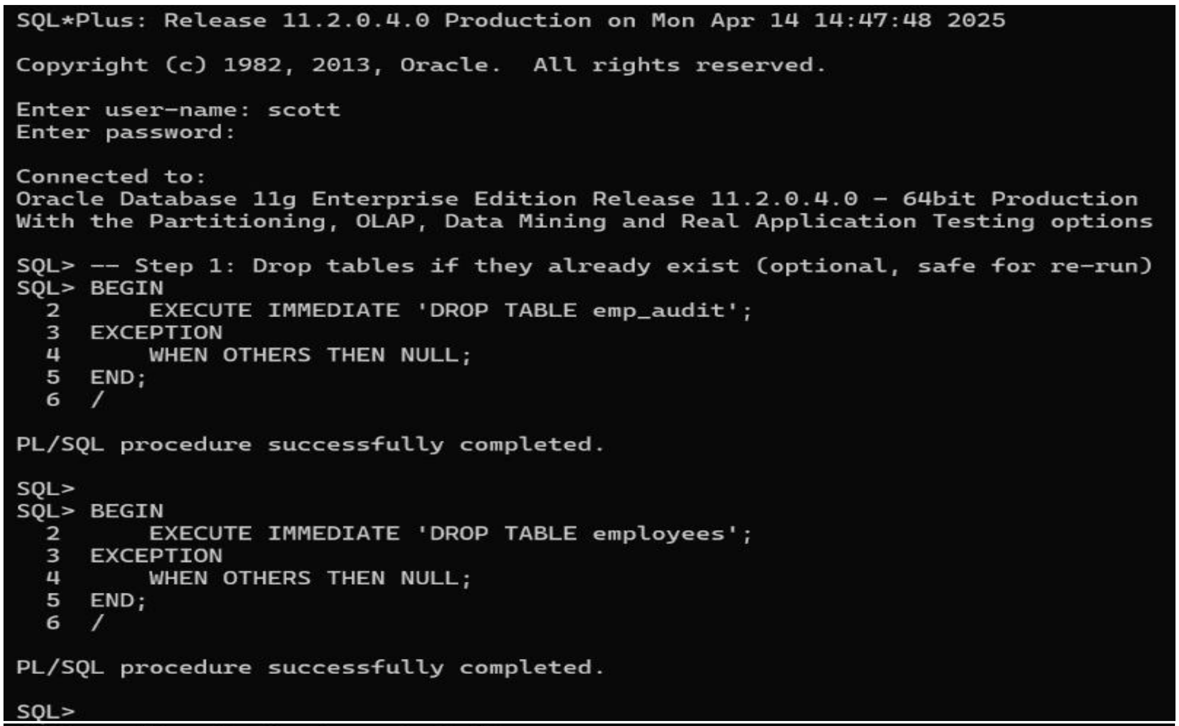


```
SQL> CREATE TABLE bank_account (
         account_no NUMBER PRIMARY KEY,
         account_holder VARCHAR2(100),
         balance NUMBER CHECK (balance >= 0)
     );

Table created.

SQL> INSERT INTO bank_account VALUES (101, 'Alice', 5000);

1 row created.

SQL> INSERT INTO bank_account VALUES (102, 'Bob', 3000);

1 row created.

SQL> COMMIT;

Commit complete.

SQL> CREATE OR REPLACE PROCEDURE transfer_funds (
         sender_acct IN NUMBER,
         receiver_acct IN NUMBER,
         transfer_amount IN NUMBER
     ) AS
         sender_balance NUMBER;
     BEGIN
         SELECT balance INTO sender_balance FROM bank_account WHERE account_no = sender_acct;

         IF sender_balance < transfer_amount THEN
             RAISE_APPLICATION_ERROR(-20001, 'Insufficient funds in sender account');
         END IF;

         UPDATE bank_account SET balance = balance - transfer_amount WHERE account_no = sender_acct;
         UPDATE bank_account SET balance = balance + transfer_amount WHERE account_no = receiver_acct;

         COMMIT;

         DBMS_OUTPUT.PUT_LINE('Transaction successful! ' || transfer_amount || ' transferred from ' || sender_acct || ' to ' || receiver_acct);
     EXCEPTION
         WHEN NO_DATA_FOUND THEN
             RAISE_APPLICATION_ERROR(-20002, 'One or both accounts do not exist');
         WHEN OTHERS THEN
             ROLLBACK;
```

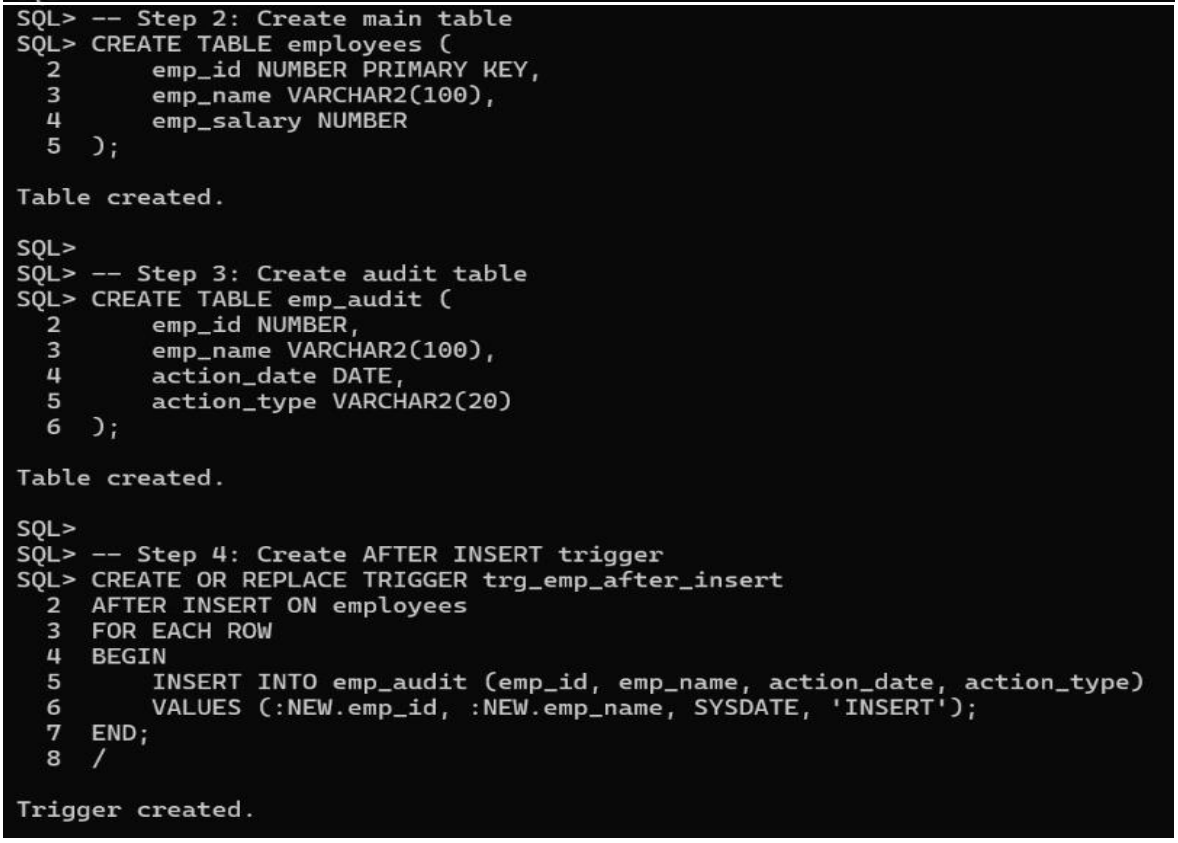


```
     EXCEPTION
         WHEN NO_DATA_FOUND THEN
             RAISE_APPLICATION_ERROR(-20002, 'One or both accounts do not exist');
         WHEN OTHERS THEN
             ROLLBACK;
             RAISE_APPLICATION_ERROR(-20003, 'Transaction failed due to an unexpected error');
     END;
     /

Procedure created.

SQL> SET SERVEROUTPUT ON;
SQL> BEGIN
         transfer_funds(101, 102, 1000);
     END;
     /
Transaction successful! 1000 transferred from 101 to 102

PL/SQL procedure successfully completed.

SQL> SELECT * FROM bank_account;

ACCOUNT_NO
----------
ACCOUNT_HOLDER
--------------------------------------------------------------------------------
   BALANCE
----------
       101
Alice
      4000

       102
Bob
      4000

ACCOUNT_NO
----------
ACCOUNT_HOLDER
--------------------------------------------------------------------------------
   BALANCE
----------
```

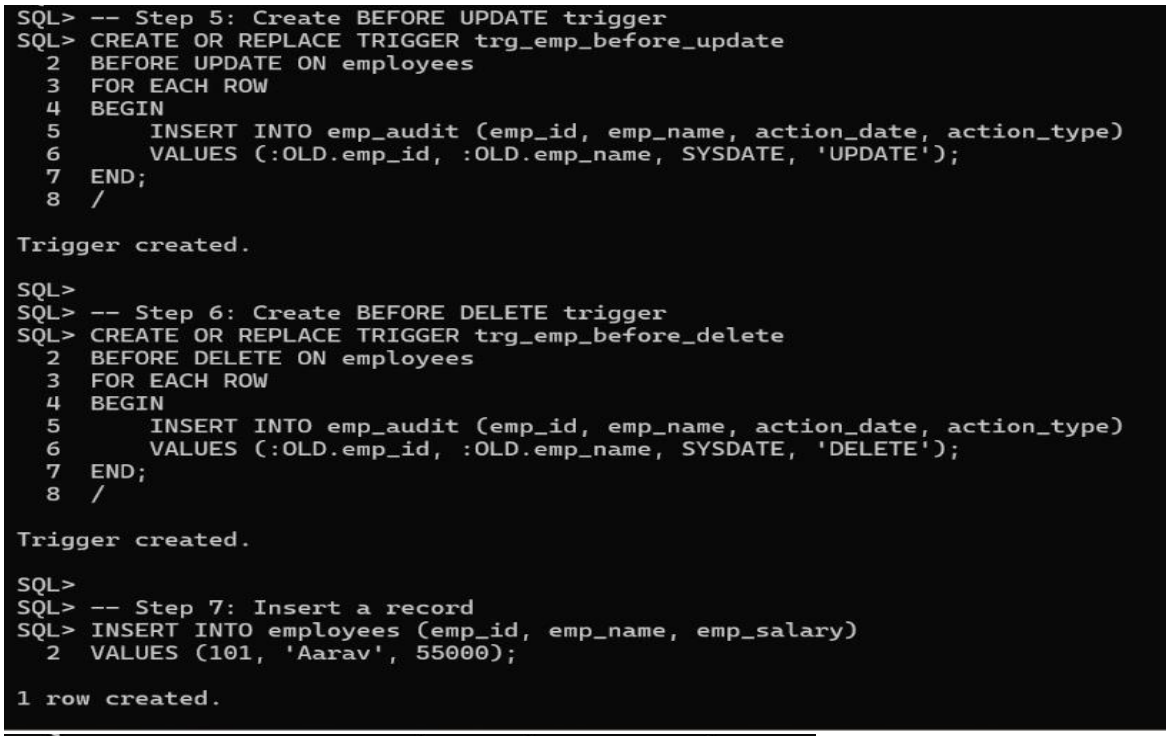
9.Write and execute triggers using PL/SQL.



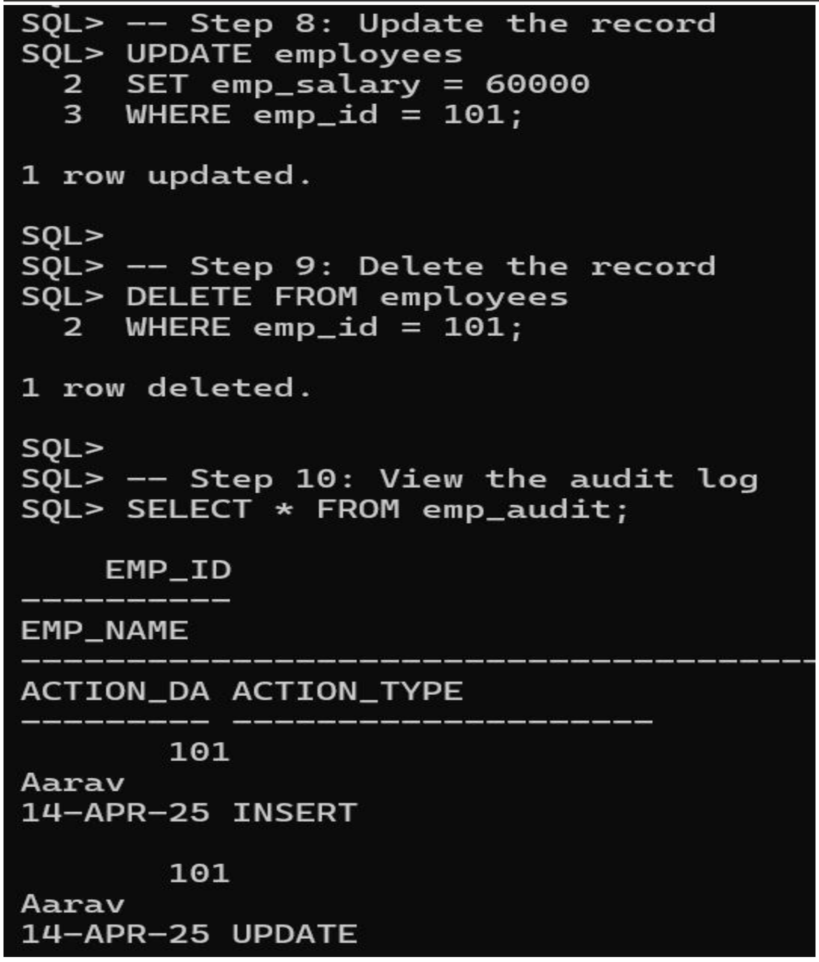
```
SQL*Plus: Release 11.2.0.4.0 Production on Mon Apr 14 14:47:48 2025

Copyright (c) 1982, 2013, Oracle.  All rights reserved.

Enter user-name: scott
Enter password:

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.4.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL> -- Step 1: Drop tables if they already exist (optional, safe for re-run)
SQL> BEGIN
  2      EXECUTE IMMEDIATE 'DROP TABLE emp_audit';
  3  EXCEPTION
  4      WHEN OTHERS THEN NULL;
  5  END;
  6  /

PL/SQL procedure successfully completed.

SQL>
SQL> BEGIN
  2      EXECUTE IMMEDIATE 'DROP TABLE employees';
  3  EXCEPTION
  4      WHEN OTHERS THEN NULL;
  5  END;
  6  /

PL/SQL procedure successfully completed.

SQL>
```



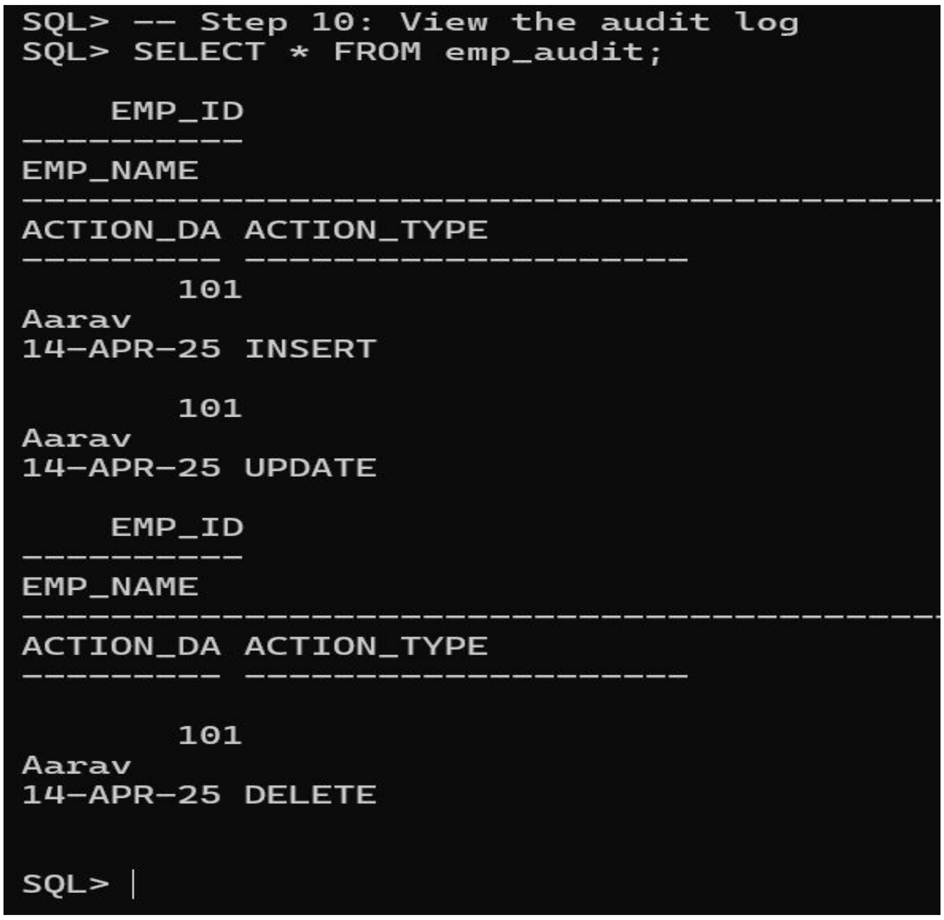
```
SQL> -- Step 2: Create main table
SQL> CREATE TABLE employees (
  2      emp_id NUMBER PRIMARY KEY,
  3      emp_name VARCHAR2(100),
  4      emp_salary NUMBER
  5  );

Table created.

SQL>
SQL> -- Step 3: Create audit table
SQL> CREATE TABLE emp_audit (
  2      emp_id NUMBER,
  3      emp_name VARCHAR2(100),
  4      action_date DATE,
  5      action_type VARCHAR2(20)
  6  );

Table created.

SQL>
SQL> -- Step 4: Create AFTER INSERT trigger
SQL> CREATE OR REPLACE TRIGGER trg_emp_after_insert
  2  AFTER INSERT ON employees
  3  FOR EACH ROW
  4  BEGIN
  5      INSERT INTO emp_audit (emp_id, emp_name, action_date, action_type)
  6      VALUES (:NEW.emp_id, :NEW.emp_name, SYSDATE, 'INSERT');
  7  END;
  8  /

Trigger created.
```



```
SQL> -- Step 5: Create BEFORE UPDATE trigger
SQL> CREATE OR REPLACE TRIGGER trg_emp_before_update
     BEFORE UPDATE ON employees
     FOR EACH ROW
     BEGIN
         INSERT INTO emp_audit (emp_id, emp_name, action_date, action_type)
         VALUES (:OLD.emp_id, :OLD.emp_name, SYSDATE, 'UPDATE');
     END;
     /

Trigger created.

SQL>
SQL> -- Step 6: Create BEFORE DELETE trigger
SQL> CREATE OR REPLACE TRIGGER trg_emp_before_delete
     BEFORE DELETE ON employees
     FOR EACH ROW
     BEGIN
         INSERT INTO emp_audit (emp_id, emp_name, action_date, action_type)
         VALUES (:OLD.emp_id, :OLD.emp_name, SYSDATE, 'DELETE');
     END;
     /

Trigger created.

SQL>
SQL> -- Step 7: Insert a record
SQL> INSERT INTO employees (emp_id, emp_name, emp_salary)
     VALUES (101, 'Aarav', 55000);

1 row created.
```

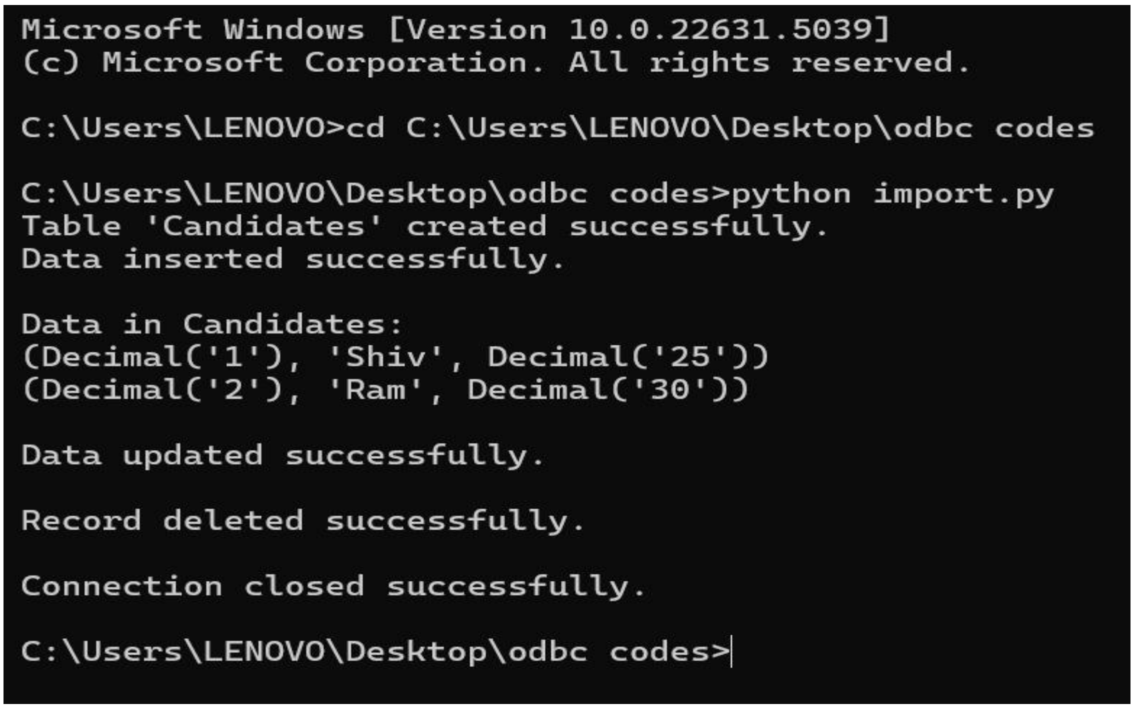


```
SQL> -- Step 8: Update the record
SQL> UPDATE employees
     SET emp_salary = 60000
     WHERE emp_id = 101;

1 row updated.

SQL>
SQL> -- Step 9: Delete the record
SQL> DELETE FROM employees
     WHERE emp_id = 101;

1 row deleted.

SQL>
SQL> -- Step 10: View the audit log
SQL> SELECT * FROM emp_audit;

    EMP_ID
----------
EMP_NAME
--------------------------------------
ACTION_DA ACTION_TYPE
--------- --------------------
       101
Aarav
14-APR-25 INSERT

       101
Aarav
14-APR-25 UPDATE
```



```
SQL> -- Step 10: View the audit log
SQL> SELECT * FROM emp_audit;

    EMP_ID
----------
EMP_NAME
------------------------------------------
ACTION_DA ACTION_TYPE
--------- --------------------
       101
Aarav
14-APR-25 INSERT

       101
Aarav
14-APR-25 UPDATE

    EMP_ID
----------
EMP_NAME
------------------------------------------
ACTION_DA ACTION_TYPE
--------- --------------------

       101
Aarav
14-APR-25 DELETE


SQL> |
```

10. Create and perform database operations using ODBC.



```
Microsoft Windows [Version 10.0.22631.5039]
(c) Microsoft Corporation. All rights reserved.

C:\Users\LENOVO>cd C:\Users\LENOVO\Desktop\odbc codes

C:\Users\LENOVO\Desktop\odbc codes>python import.py
Table 'Candidates' created successfully.
Data inserted successfully.

Data in Candidates:
(Decimal('1'), 'Shiv', Decimal('25'))
(Decimal('2'), 'Ram', Decimal('30'))

Data updated successfully.

Record deleted successfully.

Connection closed successfully.

C:\Users\LENOVO\Desktop\odbc codes>
```