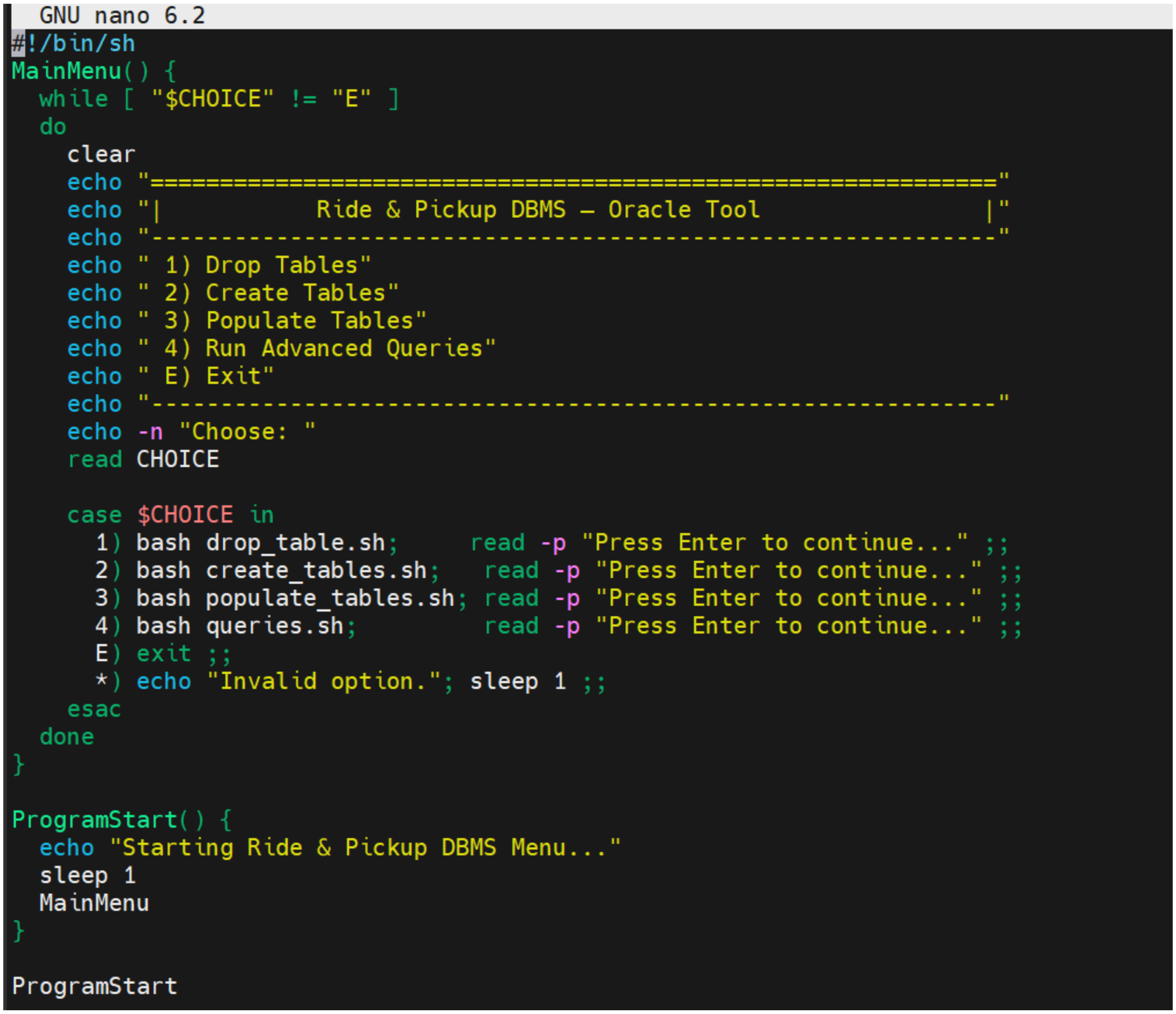
CPS 510 - Assignment 4

Shayaan Kirubakaran (Section 5)

Kirusanth Palakanthan (Section 10)

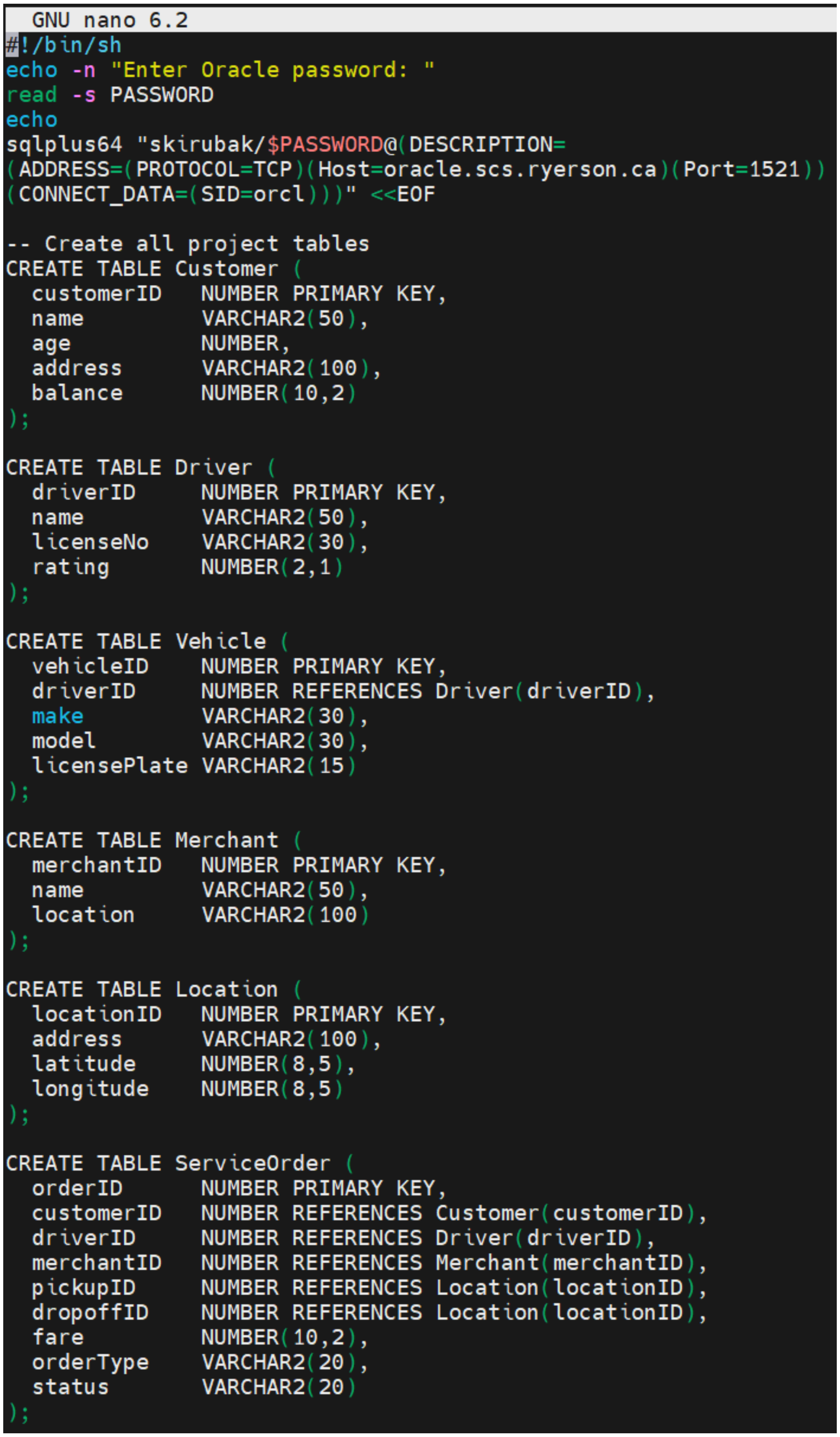
Ahmed Hasan (Section 5)

menu.sh

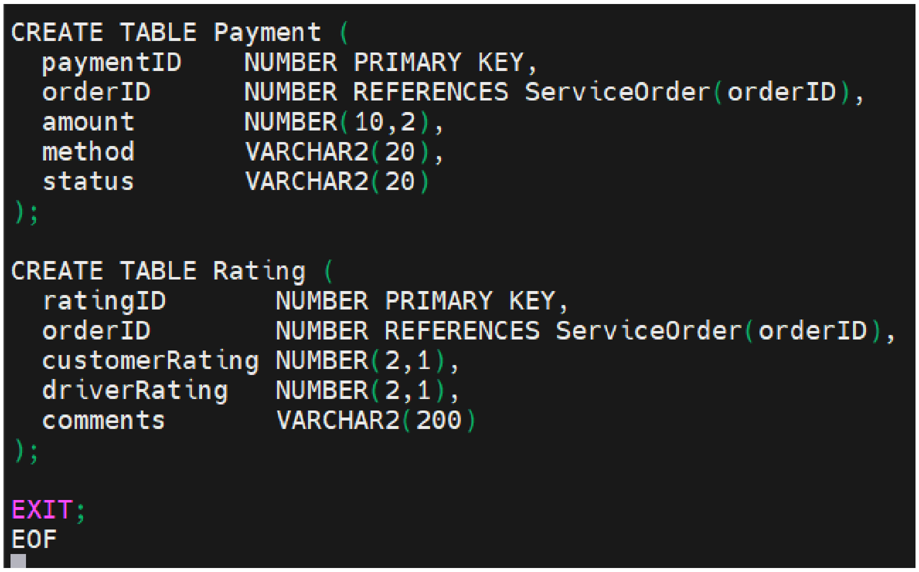


```
  GNU nano 6.2
#!/bin/sh
MainMenu() {
  while [ "$CHOICE" != "E" ]
  do
    clear
    echo "============================================================="
    echo "|           Ride & Pickup DBMS – Oracle Tool                |"
    echo "-------------------------------------------------------------"
    echo " 1) Drop Tables"
    echo " 2) Create Tables"
    echo " 3) Populate Tables"
    echo " 4) Run Advanced Queries"
    echo " E) Exit"
    echo "-------------------------------------------------------------"
    echo -n "Choose: "
    read CHOICE

    case $CHOICE in
      1) bash drop_table.sh;     read -p "Press Enter to continue..." ;;
      2) bash create_tables.sh;   read -p "Press Enter to continue..." ;;
      3) bash populate_tables.sh; read -p "Press Enter to continue..." ;;
      4) bash queries.sh;         read -p "Press Enter to continue..." ;;
      E) exit ;;
      *) echo "Invalid option."; sleep 1 ;;
    esac
  done
}

ProgramStart() {
  echo "Starting Ride & Pickup DBMS Menu..."
  sleep 1
  MainMenu
}

ProgramStart
```

create_tables.sh

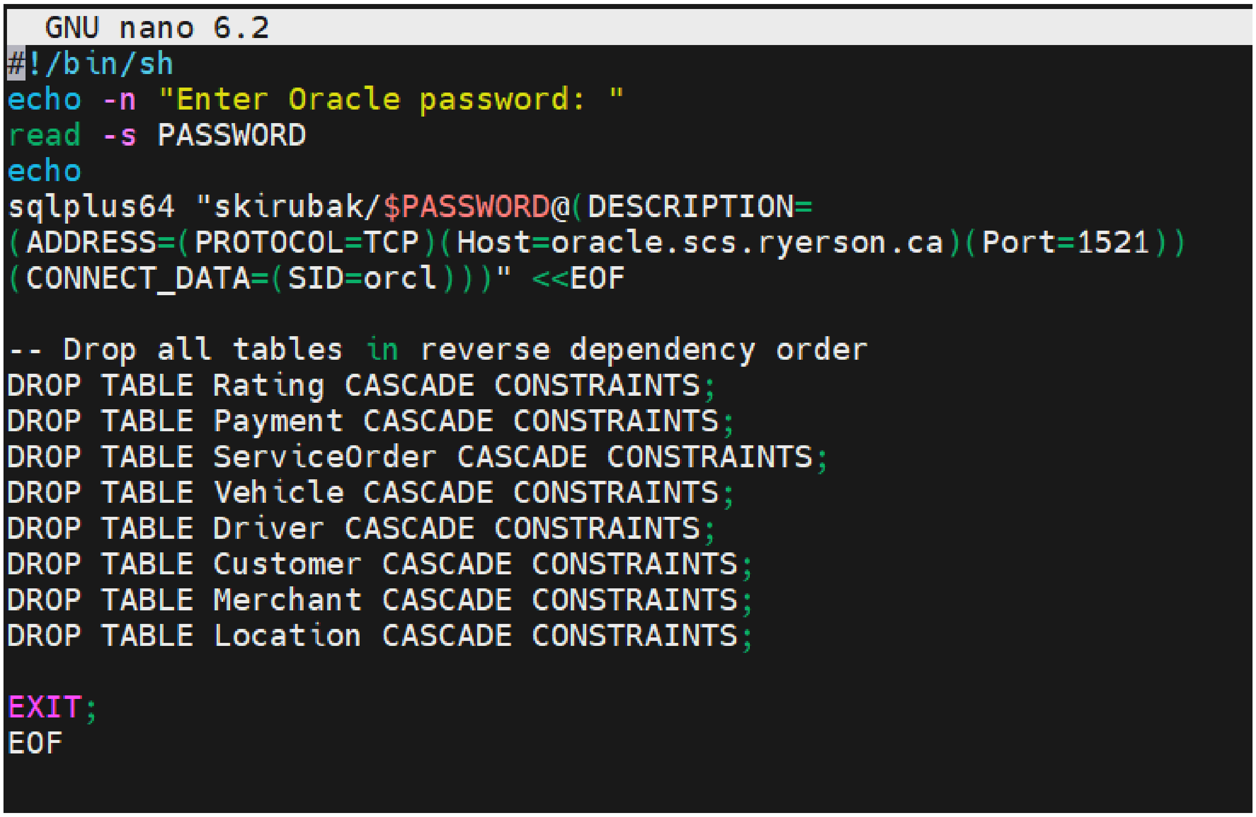


```
#!/bin/sh
echo -n "Enter Oracle password: "
read -s PASSWORD
echo
sqlplus64 "skirubak/$PASSWORD@(DESCRIPTION=
(ADDRESS=(PROTOCOL=TCP)(Host=oracle.scs.ryerson.ca)(Port=1521))
(CONNECT_DATA=(SID=orcl)))" <<EOF

-- Create all project tables
CREATE TABLE Customer (
  customerID   NUMBER PRIMARY KEY,
  name         VARCHAR2(50),
  age          NUMBER,
  address      VARCHAR2(100),
  balance      NUMBER(10,2)
);

CREATE TABLE Driver (
  driverID     NUMBER PRIMARY KEY,
  name         VARCHAR2(50),
  licenseNo    VARCHAR2(30),
  rating       NUMBER(2,1)
);

CREATE TABLE Vehicle (
  vehicleID    NUMBER PRIMARY KEY,
  driverID     NUMBER REFERENCES Driver(driverID),
  make         VARCHAR2(30),
  model        VARCHAR2(30),
  licensePlate VARCHAR2(15)
);

CREATE TABLE Merchant (
  merchantID   NUMBER PRIMARY KEY,
  name         VARCHAR2(50),
  location     VARCHAR2(100)
);

CREATE TABLE Location (
  locationID   NUMBER PRIMARY KEY,
  address      VARCHAR2(100),
  latitude     NUMBER(8,5),
  longitude    NUMBER(8,5)
);

CREATE TABLE ServiceOrder (
  orderID      NUMBER PRIMARY KEY,
  customerID   NUMBER REFERENCES Customer(customerID),
  driverID     NUMBER REFERENCES Driver(driverID),
  merchantID   NUMBER REFERENCES Merchant(merchantID),
  pickupID     NUMBER REFERENCES Location(locationID),
  dropoffID    NUMBER REFERENCES Location(locationID),
  fare         NUMBER(10,2),
  orderType    VARCHAR2(20),
  status       VARCHAR2(20)
);
```



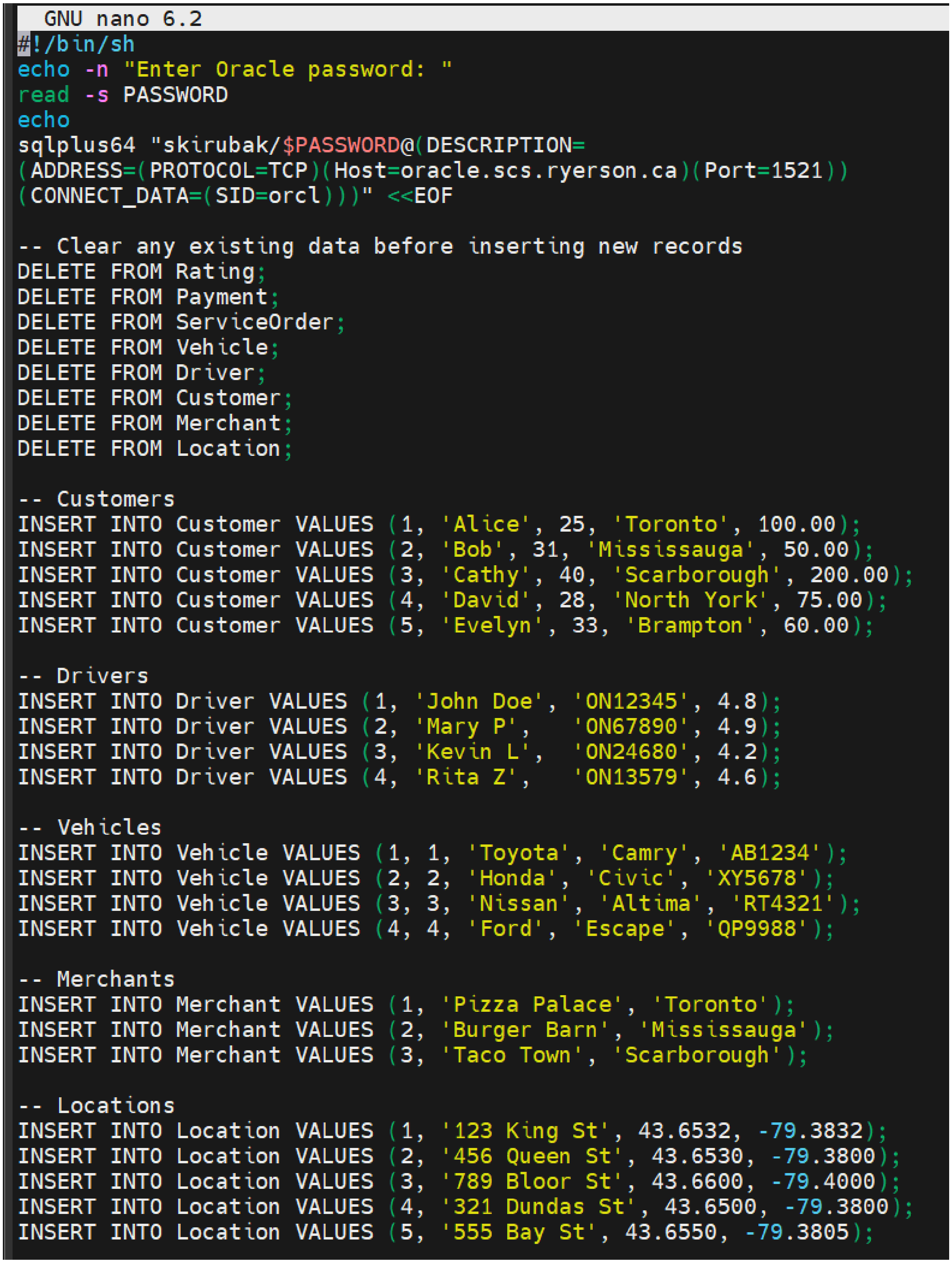
```
CREATE TABLE Payment (
  paymentID    NUMBER PRIMARY KEY,
  orderID      NUMBER REFERENCES ServiceOrder(orderID),
  amount       NUMBER(10,2),
  method       VARCHAR2(20),
  status       VARCHAR2(20)
);

CREATE TABLE Rating (
  ratingID       NUMBER PRIMARY KEY,
  orderID        NUMBER REFERENCES ServiceOrder(orderID),
  customerRating NUMBER(2,1),
  driverRating   NUMBER(2,1),
  comments       VARCHAR2(200)
);

EXIT;
EOF
```

drop_table.sh

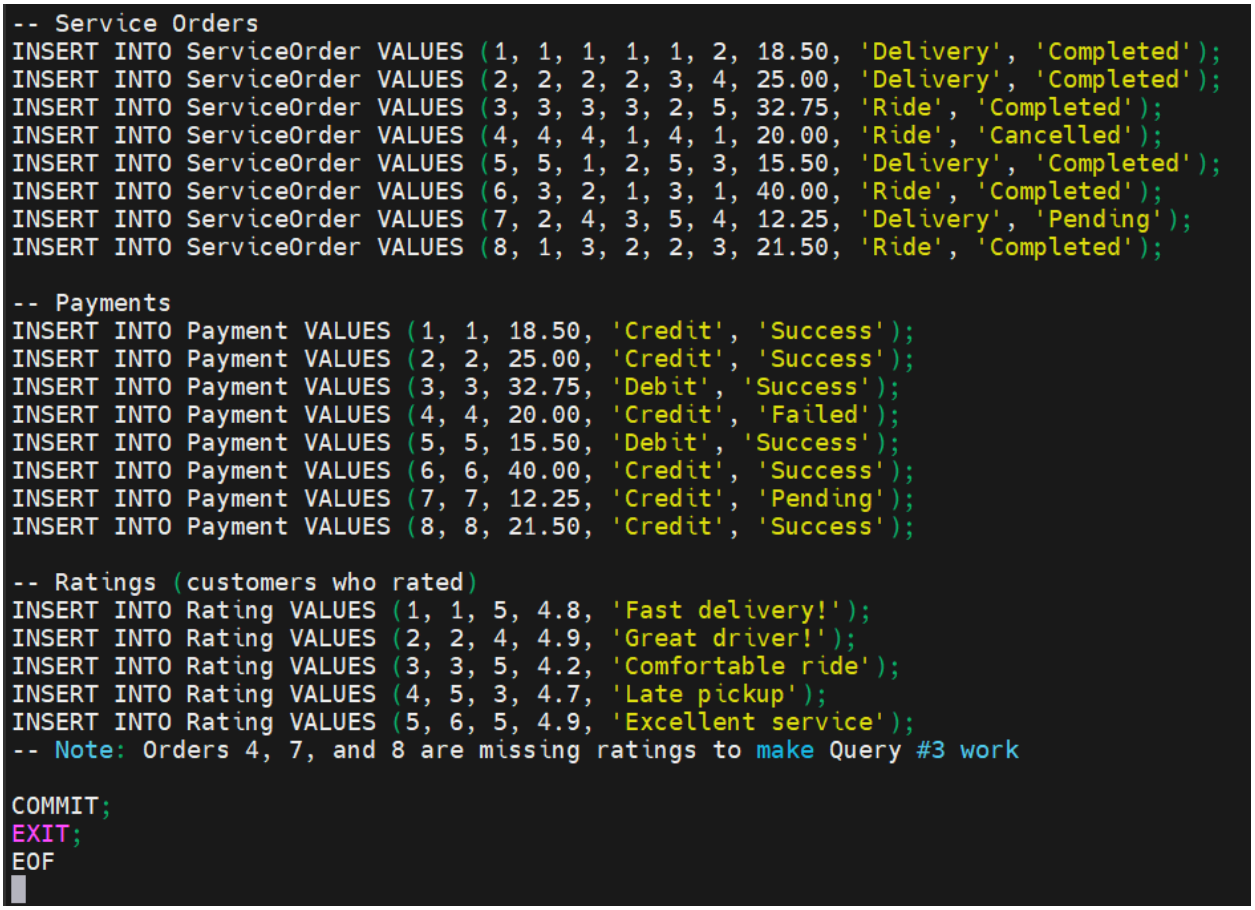


```
#!/bin/sh
echo -n "Enter Oracle password: "
read -s PASSWORD
echo
sqlplus64 "skirubak/$PASSWORD@(DESCRIPTION=
(ADDRESS=(PROTOCOL=TCP)(Host=oracle.scs.ryerson.ca)(Port=1521))
(CONNECT_DATA=(SID=orcl)))" <<EOF

-- Drop all tables in reverse dependency order
DROP TABLE Rating CASCADE CONSTRAINTS;
DROP TABLE Payment CASCADE CONSTRAINTS;
DROP TABLE ServiceOrder CASCADE CONSTRAINTS;
DROP TABLE Vehicle CASCADE CONSTRAINTS;
DROP TABLE Driver CASCADE CONSTRAINTS;
DROP TABLE Customer CASCADE CONSTRAINTS;
DROP TABLE Merchant CASCADE CONSTRAINTS;
DROP TABLE Location CASCADE CONSTRAINTS;

EXIT;
EOF
```

populate_tables.sh

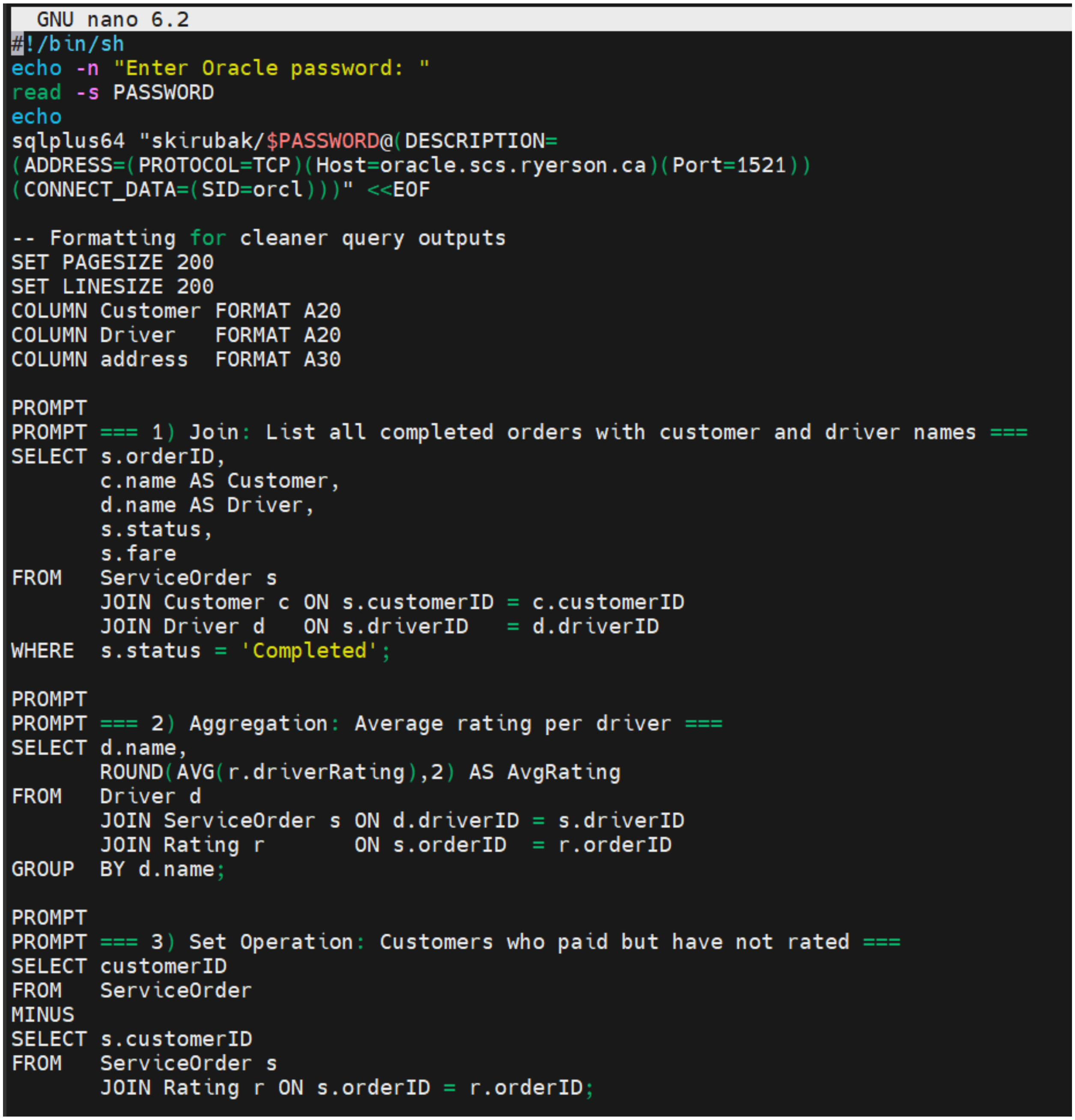


```
  GNU nano 6.2
#!/bin/sh
echo -n "Enter Oracle password: "
read -s PASSWORD
echo
sqlplus64 "skirubak/$PASSWORD@(DESCRIPTION=
(ADDRESS=(PROTOCOL=TCP)(Host=oracle.scs.ryerson.ca)(Port=1521))
(CONNECT_DATA=(SID=orcl)))" <<EOF

-- Clear any existing data before inserting new records
DELETE FROM Rating;
DELETE FROM Payment;
DELETE FROM ServiceOrder;
DELETE FROM Vehicle;
DELETE FROM Driver;
DELETE FROM Customer;
DELETE FROM Merchant;
DELETE FROM Location;

-- Customers
INSERT INTO Customer VALUES (1, 'Alice', 25, 'Toronto', 100.00);
INSERT INTO Customer VALUES (2, 'Bob', 31, 'Mississauga', 50.00);
INSERT INTO Customer VALUES (3, 'Cathy', 40, 'Scarborough', 200.00);
INSERT INTO Customer VALUES (4, 'David', 28, 'North York', 75.00);
INSERT INTO Customer VALUES (5, 'Evelyn', 33, 'Brampton', 60.00);

-- Drivers
INSERT INTO Driver VALUES (1, 'John Doe', 'ON12345', 4.8);
INSERT INTO Driver VALUES (2, 'Mary P',   'ON67890', 4.9);
INSERT INTO Driver VALUES (3, 'Kevin L', 'ON24680', 4.2);
INSERT INTO Driver VALUES (4, 'Rita Z',   'ON13579', 4.6);

-- Vehicles
INSERT INTO Vehicle VALUES (1, 1, 'Toyota', 'Camry', 'AB1234');
INSERT INTO Vehicle VALUES (2, 2, 'Honda', 'Civic', 'XY5678');
INSERT INTO Vehicle VALUES (3, 3, 'Nissan', 'Altima', 'RT4321');
INSERT INTO Vehicle VALUES (4, 4, 'Ford', 'Escape', 'QP9988');

-- Merchants
INSERT INTO Merchant VALUES (1, 'Pizza Palace', 'Toronto');
INSERT INTO Merchant VALUES (2, 'Burger Barn', 'Mississauga');
INSERT INTO Merchant VALUES (3, 'Taco Town', 'Scarborough');

-- Locations
INSERT INTO Location VALUES (1, '123 King St', 43.6532, -79.3832);
INSERT INTO Location VALUES (2, '456 Queen St', 43.6530, -79.3800);
INSERT INTO Location VALUES (3, '789 Bloor St', 43.6600, -79.4000);
INSERT INTO Location VALUES (4, '321 Dundas St', 43.6500, -79.3800);
INSERT INTO Location VALUES (5, '555 Bay St', 43.6550, -79.3805);
```

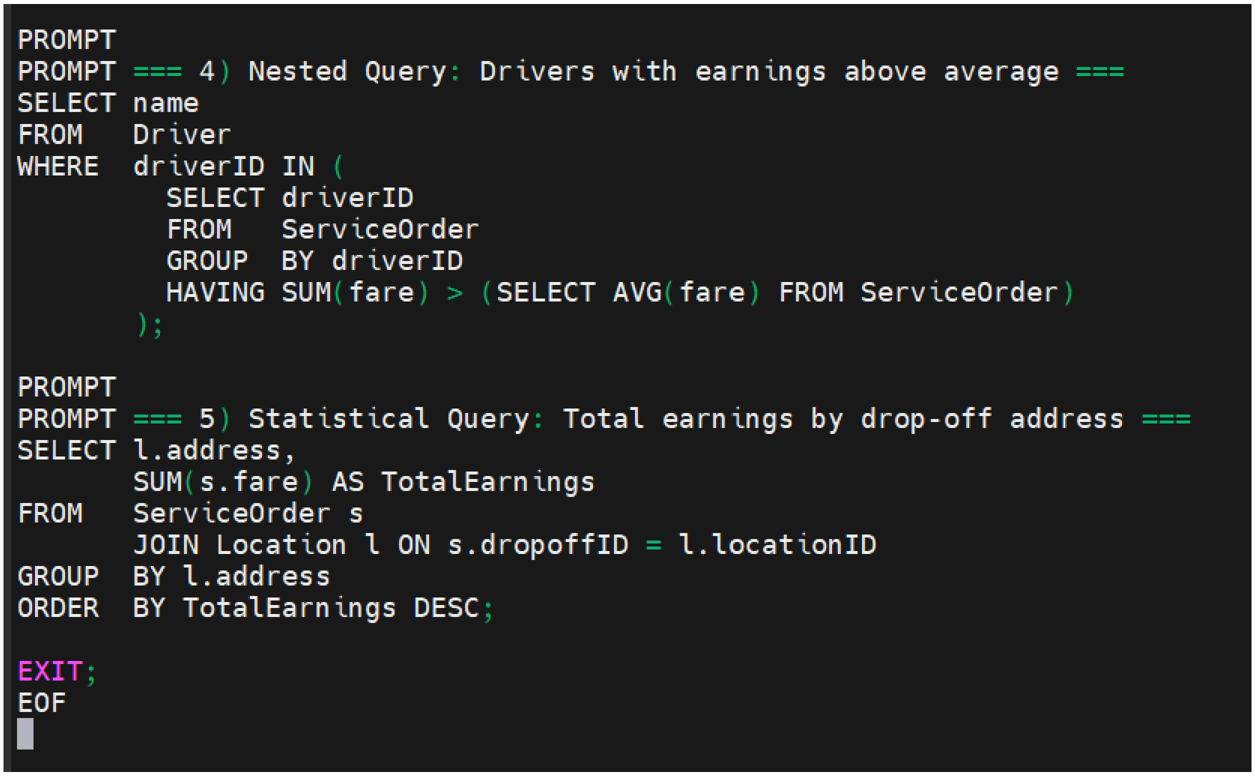


```
-- Service Orders
INSERT INTO ServiceOrder VALUES (1, 1, 1, 1, 1, 2, 18.50, 'Delivery', 'Completed');
INSERT INTO ServiceOrder VALUES (2, 2, 2, 2, 3, 4, 25.00, 'Delivery', 'Completed');
INSERT INTO ServiceOrder VALUES (3, 3, 3, 3, 2, 5, 32.75, 'Ride', 'Completed');
INSERT INTO ServiceOrder VALUES (4, 4, 4, 1, 4, 1, 20.00, 'Ride', 'Cancelled');
INSERT INTO ServiceOrder VALUES (5, 5, 1, 2, 5, 3, 15.50, 'Delivery', 'Completed');
INSERT INTO ServiceOrder VALUES (6, 3, 2, 1, 3, 1, 40.00, 'Ride', 'Completed');
INSERT INTO ServiceOrder VALUES (7, 2, 4, 3, 5, 4, 12.25, 'Delivery', 'Pending');
INSERT INTO ServiceOrder VALUES (8, 1, 3, 2, 2, 3, 21.50, 'Ride', 'Completed');

-- Payments
INSERT INTO Payment VALUES (1, 1, 18.50, 'Credit', 'Success');
INSERT INTO Payment VALUES (2, 2, 25.00, 'Credit', 'Success');
INSERT INTO Payment VALUES (3, 3, 32.75, 'Debit', 'Success');
INSERT INTO Payment VALUES (4, 4, 20.00, 'Credit', 'Failed');
INSERT INTO Payment VALUES (5, 5, 15.50, 'Debit', 'Success');
INSERT INTO Payment VALUES (6, 6, 40.00, 'Credit', 'Success');
INSERT INTO Payment VALUES (7, 7, 12.25, 'Credit', 'Pending');
INSERT INTO Payment VALUES (8, 8, 21.50, 'Credit', 'Success');

-- Ratings (customers who rated)
INSERT INTO Rating VALUES (1, 1, 5, 4.8, 'Fast delivery!');
INSERT INTO Rating VALUES (2, 2, 4, 4.9, 'Great driver!');
INSERT INTO Rating VALUES (3, 3, 5, 4.2, 'Comfortable ride');
INSERT INTO Rating VALUES (4, 5, 3, 4.7, 'Late pickup');
INSERT INTO Rating VALUES (5, 6, 5, 4.9, 'Excellent service');
-- Note: Orders 4, 7, and 8 are missing ratings to make Query #3 work

COMMIT;
EXIT;
EOF
```

queries.sh

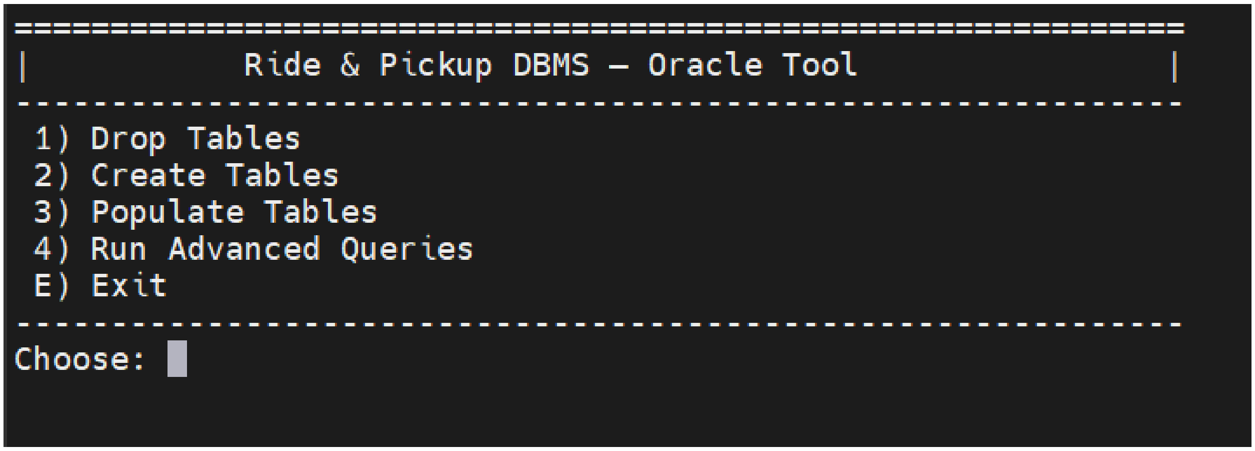


```
#!/bin/sh
echo -n "Enter Oracle password: "
read -s PASSWORD
echo
sqlplus64 "skirubak/$PASSWORD@(DESCRIPTION=
(ADDRESS=(PROTOCOL=TCP)(Host=oracle.scs.ryerson.ca)(Port=1521))
(CONNECT_DATA=(SID=orcl)))" <<EOF

-- Formatting for cleaner query outputs
SET PAGESIZE 200
SET LINESIZE 200
COLUMN Customer FORMAT A20
COLUMN Driver   FORMAT A20
COLUMN address  FORMAT A30

PROMPT
PROMPT === 1) Join: List all completed orders with customer and driver names ===
SELECT s.orderID,
       c.name AS Customer,
       d.name AS Driver,
       s.status,
       s.fare
FROM   ServiceOrder s
       JOIN Customer c ON s.customerID = c.customerID
       JOIN Driver d   ON s.driverID   = d.driverID
WHERE  s.status = 'Completed';

PROMPT
PROMPT === 2) Aggregation: Average rating per driver ===
SELECT d.name,
       ROUND(AVG(r.driverRating),2) AS AvgRating
FROM   Driver d
       JOIN ServiceOrder s ON d.driverID = s.driverID
       JOIN Rating r       ON s.orderID  = r.orderID
GROUP  BY d.name;

PROMPT
PROMPT === 3) Set Operation: Customers who paid but have not rated ===
SELECT customerID
FROM   ServiceOrder
MINUS
SELECT s.customerID
FROM   ServiceOrder s
       JOIN Rating r ON s.orderID = r.orderID;
```

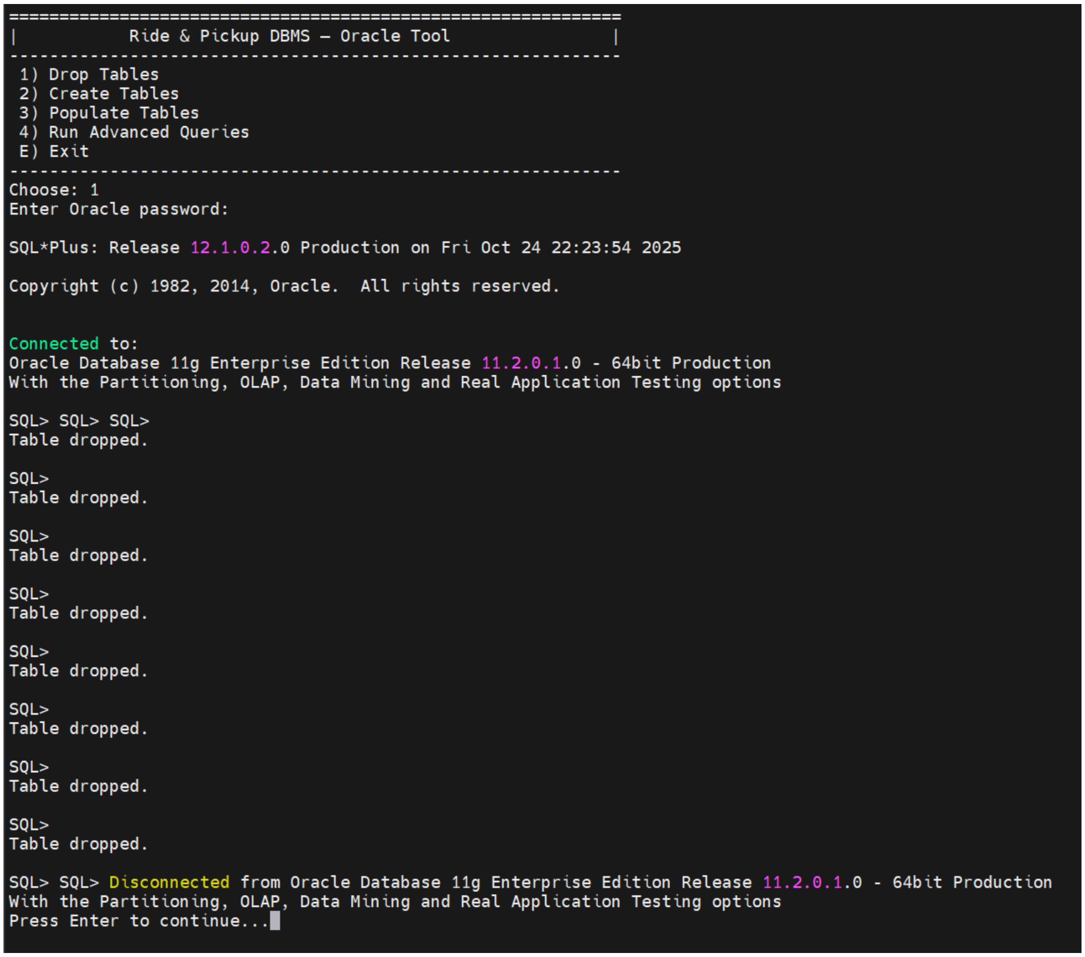


```
PROMPT
PROMPT === 4) Nested Query: Drivers with earnings above average ===
SELECT name
FROM   Driver
WHERE  driverID IN (
         SELECT driverID
         FROM   ServiceOrder
         GROUP  BY driverID
         HAVING SUM(fare) > (SELECT AVG(fare) FROM ServiceOrder)
       );

PROMPT
PROMPT === 5) Statistical Query: Total earnings by drop-off address ===
SELECT l.address,
       SUM(s.fare) AS TotalEarnings
FROM   ServiceOrder s
       JOIN Location l ON s.dropoffID = l.locationID
GROUP  BY l.address
ORDER  BY TotalEarnings DESC;

EXIT;
EOF
```

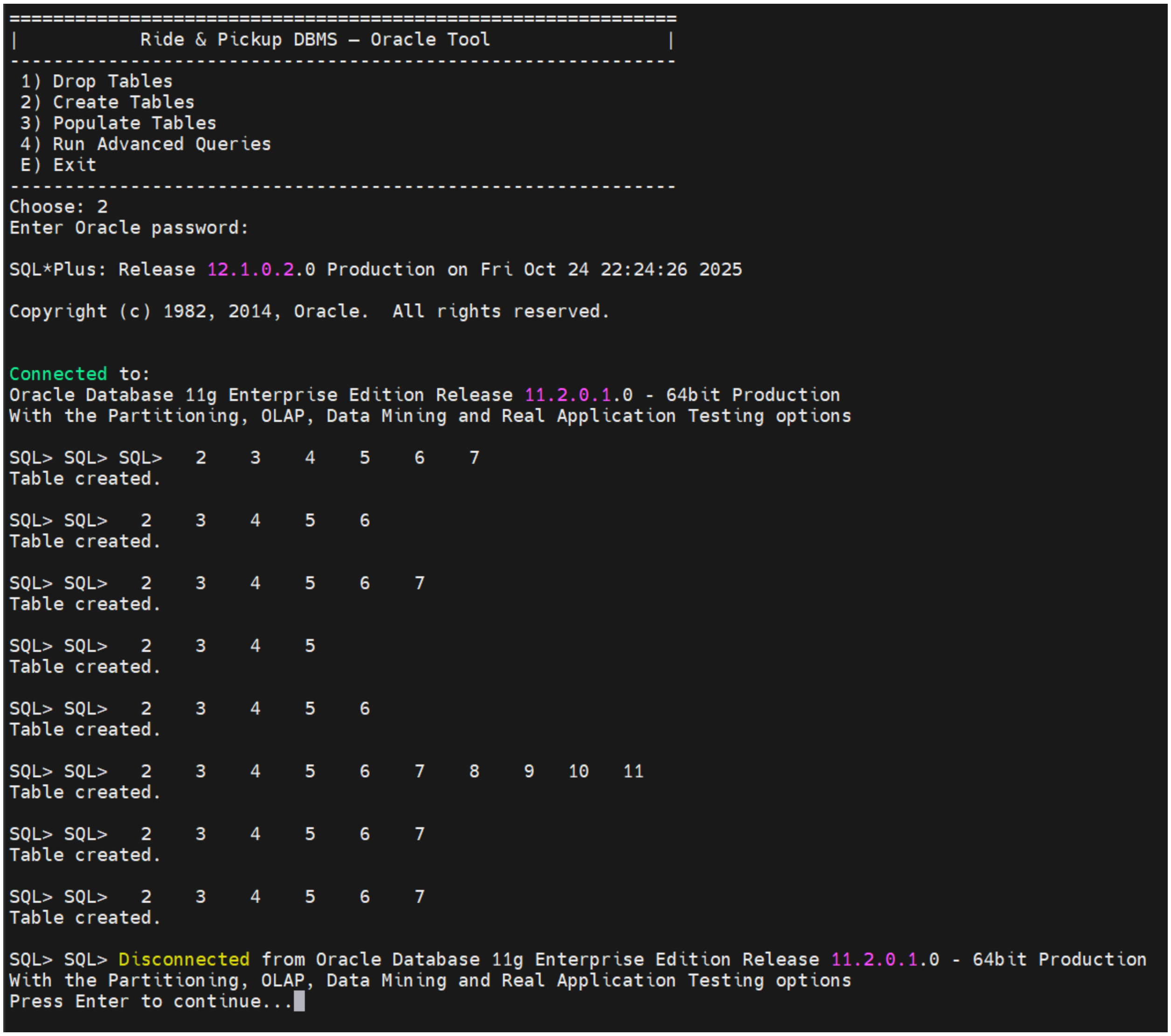
User interface output:



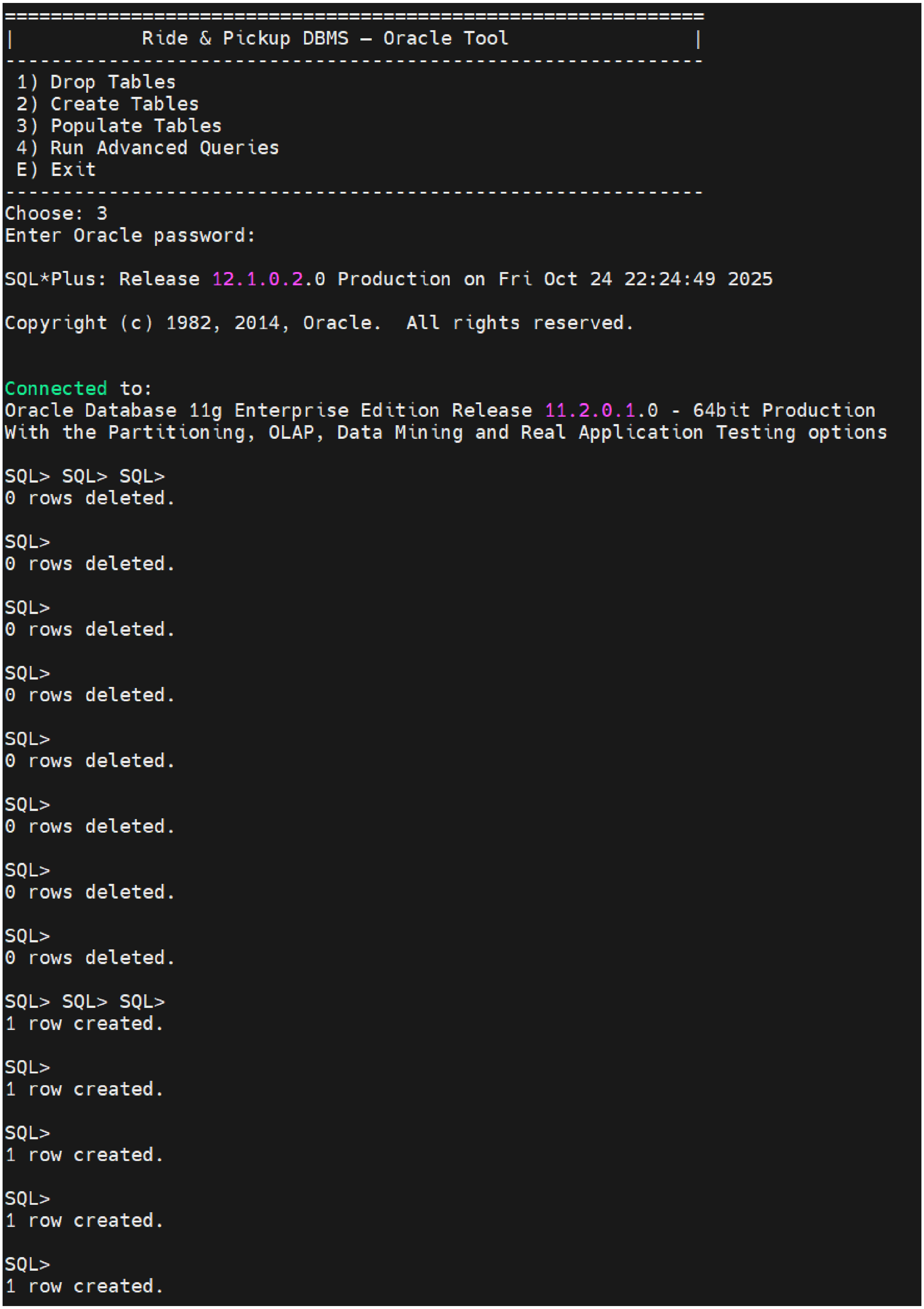
```
=============================================================
|           Ride & Pickup DBMS – Oracle Tool               |
-------------------------------------------------------------
 1) Drop Tables
 2) Create Tables
 3) Populate Tables
 4) Run Advanced Queries
 E) Exit
-------------------------------------------------------------
Choose: 
```



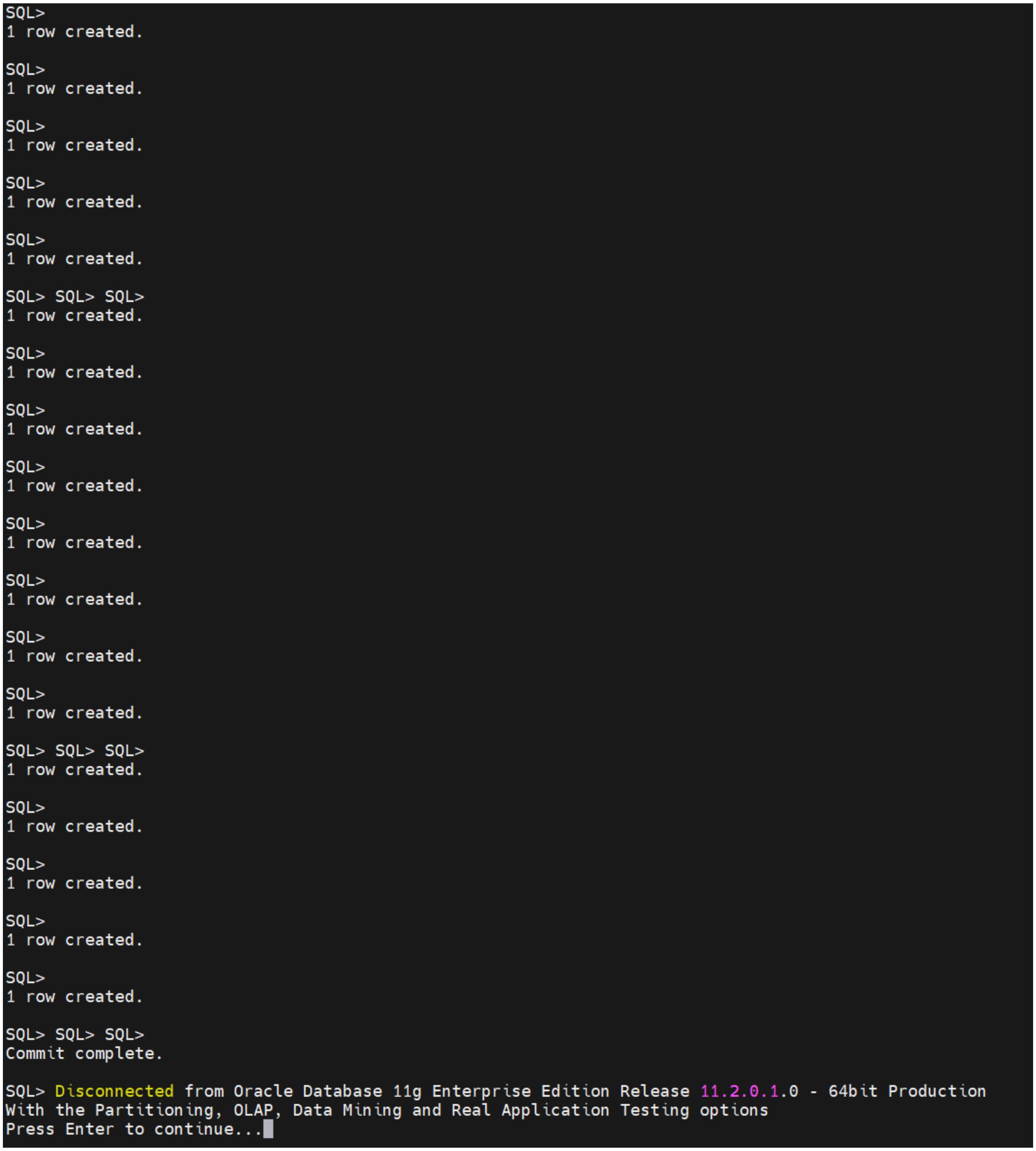
```
=============================================================
|           Ride & Pickup DBMS – Oracle Tool               |
-------------------------------------------------------------
 1) Drop Tables
 2) Create Tables
 3) Populate Tables
 4) Run Advanced Queries
 E) Exit
-------------------------------------------------------------
Choose: 1
Enter Oracle password:

SQL*Plus: Release 12.1.0.2.0 Production on Fri Oct 24 22:23:54 2025

Copyright (c) 1982, 2014, Oracle.  All rights reserved.


Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL> SQL> SQL>
Table dropped.

SQL>
Table dropped.

SQL>
Table dropped.

SQL>
Table dropped.

SQL>
Table dropped.

SQL>
Table dropped.

SQL>
Table dropped.

SQL>
Table dropped.

SQL> SQL> Disconnected from Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
Press Enter to continue...
```



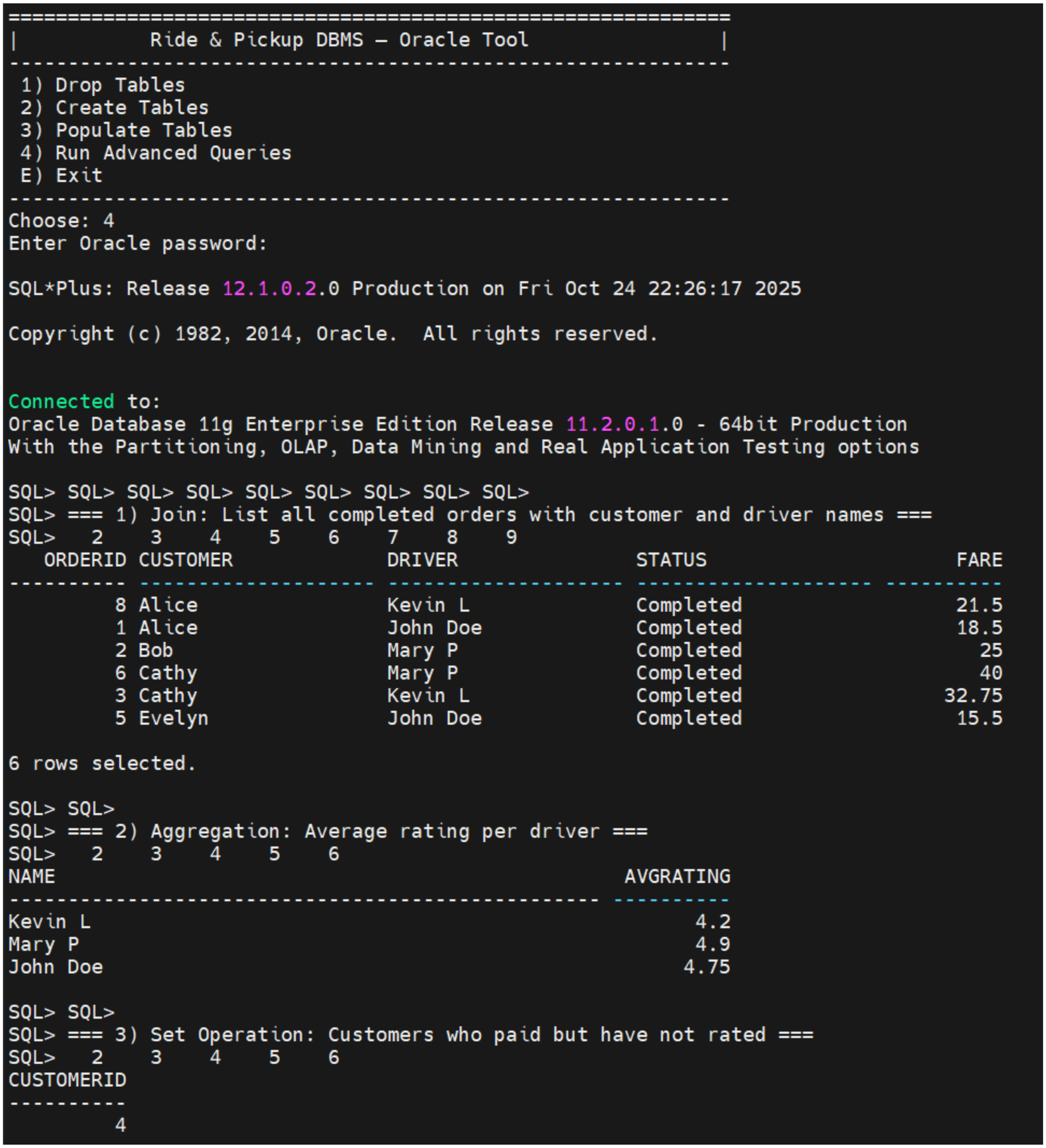
```
=============================================================
|           Ride & Pickup DBMS — Oracle Tool                |
-------------------------------------------------------------
 1) Drop Tables
 2) Create Tables
 3) Populate Tables
 4) Run Advanced Queries
 E) Exit
-------------------------------------------------------------
Choose: 2
Enter Oracle password:

SQL*Plus: Release 12.1.0.2.0 Production on Fri Oct 24 22:24:26 2025

Copyright (c) 1982, 2014, Oracle.  All rights reserved.


Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL> SQL> SQL>   2    3    4    5    6    7
Table created.

SQL> SQL>   2    3    4    5    6
Table created.

SQL> SQL>   2    3    4    5    6    7
Table created.

SQL> SQL>   2    3    4    5
Table created.

SQL> SQL>   2    3    4    5    6
Table created.

SQL> SQL>   2    3    4    5    6    7    8    9   10   11
Table created.

SQL> SQL>   2    3    4    5    6    7
Table created.

SQL> SQL>   2    3    4    5    6    7
Table created.

SQL> SQL> Disconnected from Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
Press Enter to continue...
```



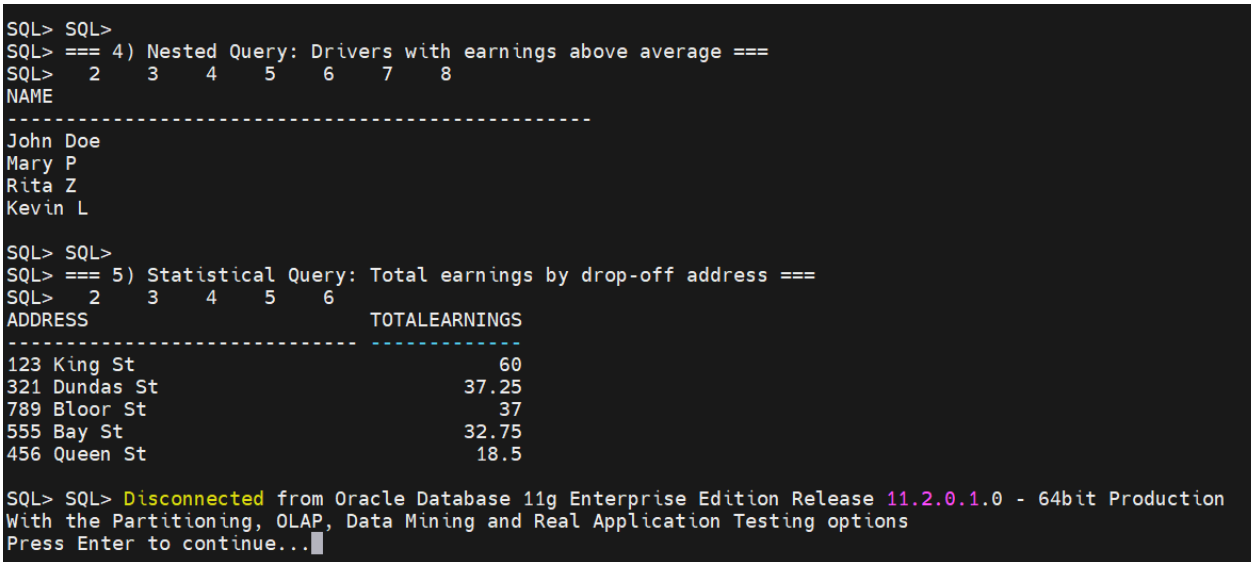
```
=============================================================
|           Ride & Pickup DBMS – Oracle Tool               |
-------------------------------------------------------------
 1) Drop Tables
 2) Create Tables
 3) Populate Tables
 4) Run Advanced Queries
 E) Exit
-------------------------------------------------------------
Choose: 3
Enter Oracle password:

SQL*Plus: Release 12.1.0.2.0 Production on Fri Oct 24 22:24:49 2025

Copyright (c) 1982, 2014, Oracle.  All rights reserved.


Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL> SQL> SQL>
0 rows deleted.

SQL>
0 rows deleted.

SQL>
0 rows deleted.

SQL>
0 rows deleted.

SQL>
0 rows deleted.

SQL>
0 rows deleted.

SQL>
0 rows deleted.

SQL>
0 rows deleted.

SQL> SQL> SQL>
1 row created.

SQL>
1 row created.

SQL>
1 row created.

SQL>
1 row created.

SQL>
1 row created.
```



```
SQL>
1 row created.

SQL>
1 row created.

SQL>
1 row created.

SQL>
1 row created.

SQL>
1 row created.

SQL> SQL> SQL>
1 row created.

SQL>
1 row created.

SQL>
1 row created.

SQL>
1 row created.

SQL>
1 row created.

SQL>
1 row created.

SQL>
1 row created.

SQL>
1 row created.

SQL> SQL> SQL>
1 row created.

SQL>
1 row created.

SQL>
1 row created.

SQL>
1 row created.

SQL>
1 row created.

SQL> SQL> SQL>
Commit complete.

SQL> Disconnected from Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
Press Enter to continue...
```



```
=============================================================
|           Ride & Pickup DBMS — Oracle Tool                |
-------------------------------------------------------------
 1) Drop Tables
 2) Create Tables
 3) Populate Tables
 4) Run Advanced Queries
 E) Exit
-------------------------------------------------------------
Choose: 4
Enter Oracle password:

SQL*Plus: Release 12.1.0.2.0 Production on Fri Oct 24 22:26:17 2025

Copyright (c) 1982, 2014, Oracle.  All rights reserved.


Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL> SQL> SQL> SQL> SQL> SQL> SQL> SQL> SQL>
SQL> === 1) Join: List all completed orders with customer and driver names ===
SQL>   2    3    4    5    6    7    8    9
   ORDERID CUSTOMER             DRIVER               STATUS                     FARE
---------- -------------------- -------------------- -------------------- ----------
         8 Alice                Kevin L              Completed                  21.5
         1 Alice                John Doe             Completed                  18.5
         2 Bob                  Mary P               Completed                    25
         6 Cathy                Mary P               Completed                    40
         3 Cathy                Kevin L              Completed                 32.75
         5 Evelyn               John Doe             Completed                  15.5

6 rows selected.

SQL> SQL>
SQL> === 2) Aggregation: Average rating per driver ===
SQL>   2    3    4    5    6
NAME                                                AVGRATING
-------------------------------------------------- ----------
Kevin L                                                   4.2
Mary P                                                    4.9
John Doe                                                 4.75

SQL> SQL>
SQL> === 3) Set Operation: Customers who paid but have not rated ===
SQL>   2    3    4    5    6
CUSTOMERID
----------
         4
```



```
SQL> SQL>
SQL> === 4) Nested Query: Drivers with earnings above average ===
SQL>   2    3    4    5    6    7    8
NAME
--------------------------------------------------
John Doe
Mary P
Rita Z
Kevin L

SQL> SQL>
SQL> === 5) Statistical Query: Total earnings by drop-off address ===
SQL>   2    3    4    5    6
ADDRESS                        TOTALEARNINGS
------------------------------ -------------
123 King St                               60
321 Dundas St                          37.25
789 Bloor St                              37
555 Bay St                             32.75
456 Queen St                            18.5

SQL> SQL> Disconnected from Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
Press Enter to continue...
```