```sql
CREATE DATABASE bank;
use bank;

CREATE TABLE CUSTOMER(
   Cust_ID VARCHAR(50) PRIMARY KEY,
   Name VARCHAR(255) NOT NULL,
   Age INT CHECK(Age>=18),
   Cust_type VARCHAR(20) CHECK(Cust_type IN('Regular', 'Corporate', 'Priority'))
);

INSERT INTO CUSTOMER
( Cust_ID, Name, Age, Cust_type)
VALUES
('1', 'John', 25, 'Regular'),
('2', 'Alice', 30, 'Corporate'),
('3', 'Bob', 22, 'Regular'),
('4', 'Emma', 28, 'Priority'),
('5', 'David', 35, 'Corporate'),
('6', 'Sophia', 29, 'Regular'),
('7', 'James', 24, 'Regular'),
('8', 'Olivia', 33, 'Priority'),
('9', 'Michael', 60, 'Regular'),
('10', 'Mia', 26, 'Regular');

drop table CUSTOMER;




CREATE TABLE BANKS(
            Bank_ID INT PRIMARY KEY,
   B_name VARCHAR(255) NOT NULL,
   B_city VARCHAR(255) NOT NULL
);

INSERT INTO BANKS
(Bank_ID, B_name, B_city)
VALUES
(101, 'Central Bank', 'Kolkata'),
(102, 'Indian Bank', 'Kolkata'),
(103, 'City Bank', 'Mumbai'),
(104, 'National Bank', 'Delhi'),
(105, 'Metro Bank', 'Chennai'),
(106, 'Cooperative Bank', 'Kolkata'),
(107, 'Regional Bank', 'Hyderabad'),
(108, 'Urban Bank', 'Pune'),
(109, 'State Bank', 'Bangalore'),
(110, 'Rural Bank', 'Jaipur');

DROP TABLE BANKS;

CREATE TABLE BORROWS(
            B_ID INT(50) PRIMARY KEY,
   Cust_ID VARCHAR(50) NOT NULL,
   Bank_ID INT NOT NULL,
   Amount INT CHECK(Amount>0),
   Borrow_date DATE,
   Duration INT CHECK(Duration>=0),
   Type VARCHAR(20) CHECK(Type IN('Home', 'Car', 'Personal')),
   FOREIGN KEY(Cust_ID) references CUSTOMER(Cust_ID),
   FOREIGN KEY(Bank_ID) REFERENCES BANKS(Bank_ID)
);
```

INSERT INTO BORROWS

(B_ID, Cust_ID, Bank_ID, Amount, Borrow_date, Duration, Type)

VALUES

(1, '1', 101, 15.5, '2022-01-15', 20, 'Home'),

(2, '2', 102, 12.0, '2018-12-03', 15, 'Car'),

(3, '3', 101, 8.7, '2023-03-10', 10, 'Personal'),

(4, '4', 103, 18.2, '2021-05-20', 25, 'Home'),

(5, '5', 102, 10.8, '2022-11-30', 17, 'Car'),

(6, '6', 101, 7.5, '2023-04-05', 12, 'Home'),

(7, '7', 104, 22.1, '2019-08-15', 30, 'Personal'),

(8, '8', 105, 14.6, '2020-12-22', 20, 'Home'),

(9, '9', 106, 9.3, '2021-07-18', 15, 'Car'),

(10, '10', 107, 6.4, '2023-02-28', 8, 'Personal');

DROP TABLE BORROWS;

A)

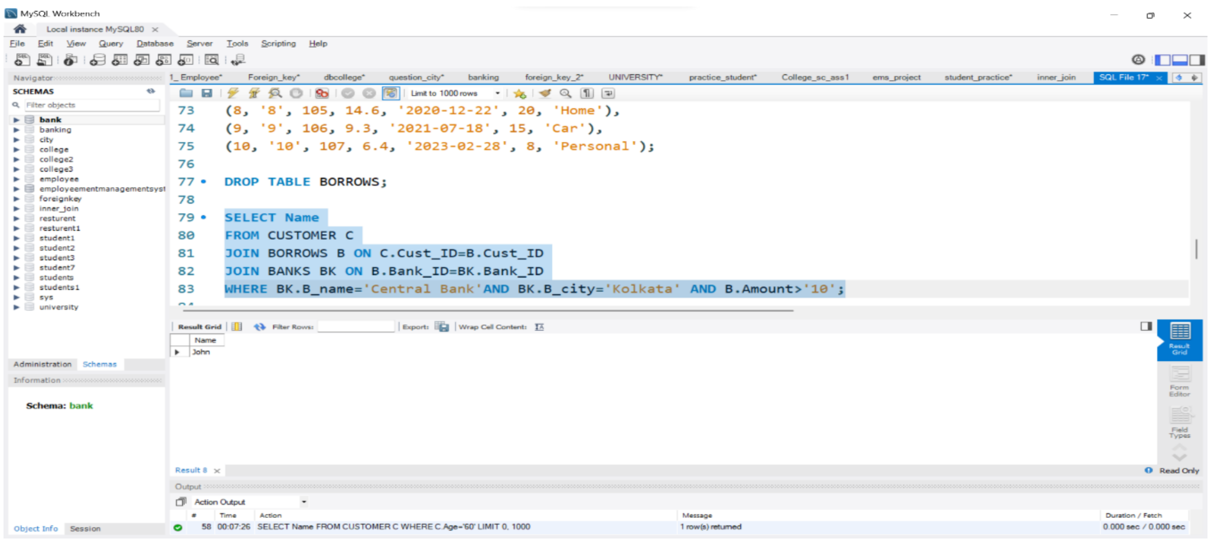
SELECT Name

FROM CUSTOMER C

JOIN BORROWS B ON C.Cust_ID=B.Cust_ID

JOIN BANKS BK ON B.Bank_ID=BK.Bank_ID

WHERE BK.B_name='Central Bank'AND BK.B_city='Kolkata' AND B.Amount>'10';



```
(8, '8', 105, 14.6, '2020-12-22', 20, 'Home'),
(9, '9', 106, 9.3, '2021-07-18', 15, 'Car'),
(10, '10', 107, 6.4, '2023-02-28', 8, 'Personal');

DROP TABLE BORROWS;

SELECT Name
FROM CUSTOMER C
JOIN BORROWS B ON C.Cust_ID=B.Cust_ID
JOIN BANKS BK ON B.Bank_ID=BK.Bank_ID
WHERE BK.B_name='Central Bank'AND BK.B_city='Kolkata' AND B.Amount>'10';
```

Name
John

#	Time	Action	Message	Duration / Fetch
58	00:07:26	SELECT Name FROM CUSTOMER C WHERE C.Age='60' LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec

B)

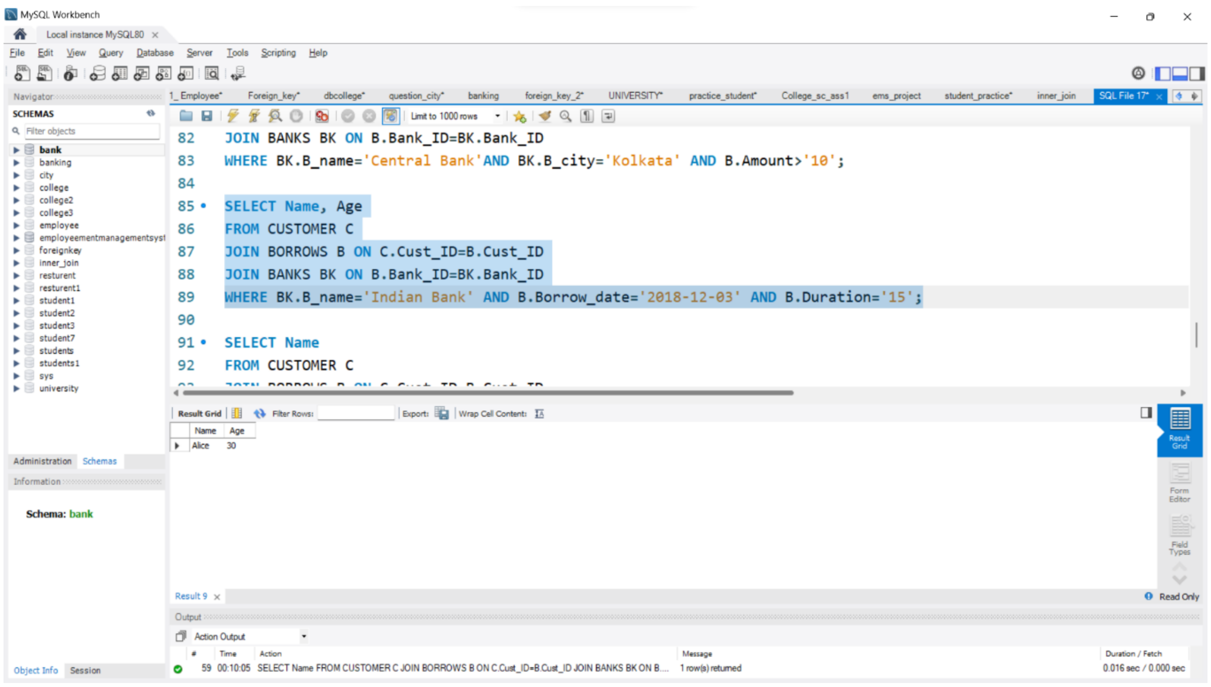
SELECT Name, Age

FROM CUSTOMER C

JOIN BORROWS B ON C.Cust_ID=B.Cust_ID

JOIN BANKS BK ON B.Bank_ID=BK.Bank_ID

WHERE BK.B_name='Indian Bank' AND B.Borrow_date='2018-12-03' AND B.Duration>='15';



```
JOIN BANKS BK ON B.Bank_ID=BK.Bank_ID
WHERE BK.B_name='Central Bank'AND BK.B_city='Kolkata' AND B.Amount>'10';

SELECT Name, Age
FROM CUSTOMER C
JOIN BORROWS B ON C.Cust_ID=B.Cust_ID
JOIN BANKS BK ON B.Bank_ID=BK.Bank_ID
WHERE BK.B_name='Indian Bank' AND B.Borrow_date='2018-12-03' AND B.Duration='15';

SELECT Name
FROM CUSTOMER C
```

Name	Age
Alice	30

```
#   Time      Action                                                                                Message            Duration / Fetch
59  00:10:05  SELECT Name FROM CUSTOMER C JOIN BORROWS B ON C.Cust_ID=B.Cust_ID JOIN BANKS BK ON B....  1 row(s) returned  0.016 sec / 0.000 sec
```

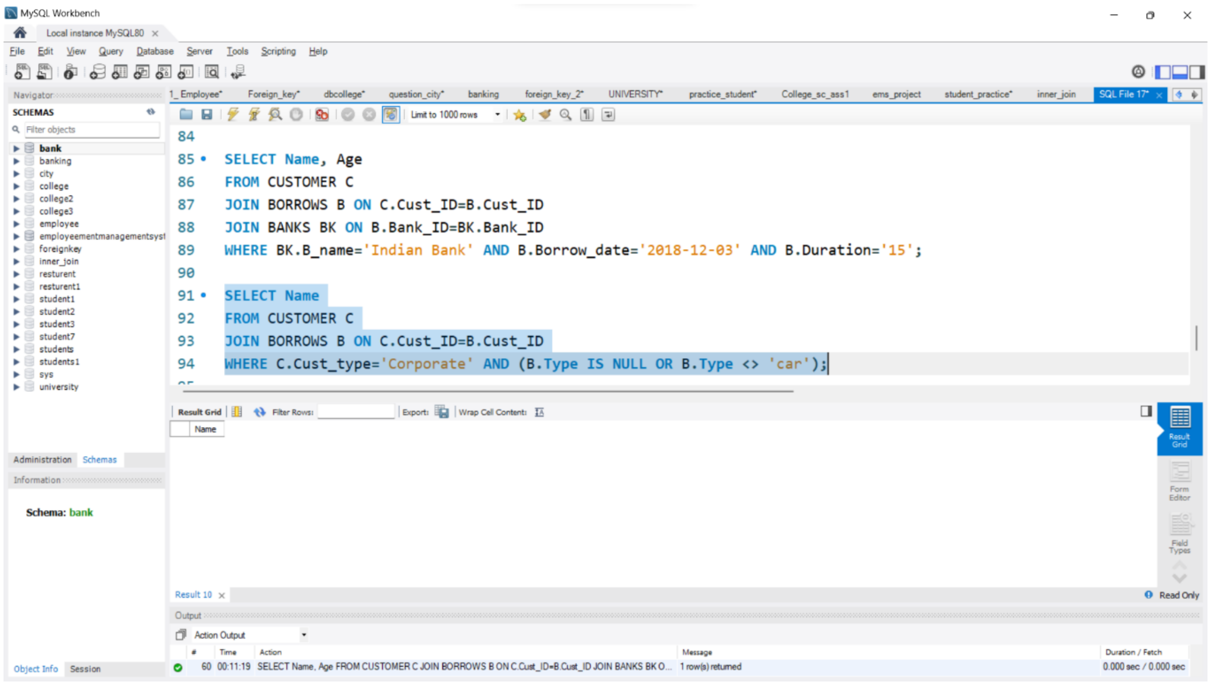
C)

SELECT Name

FROM CUSTOMER C

JOIN BORROWS B ON C.Cust_ID=B.Cust_ID

WHERE C.Cust_type='Corporate' AND (B.Type IS NULL OR B.Type <> 'car');



```
SELECT Name, Age
FROM CUSTOMER C
JOIN BORROWS B ON C.Cust_ID=B.Cust_ID
JOIN BANKS BK ON B.Bank_ID=BK.Bank_ID
WHERE BK.B_name='Indian Bank' AND B.Borrow_date='2018-12-03' AND B.Duration='15';

SELECT Name
FROM CUSTOMER C
JOIN BORROWS B ON C.Cust_ID=B.Cust_ID
WHERE C.Cust_type='Corporate' AND (B.Type IS NULL OR B.Type <> 'car');
```

```
Name
```

```
#   Time      Action                                                                          Message            Duration / Fetch
60  00:11:19  SELECT Name, Age FROM CUSTOMER C JOIN BORROWS B ON C.Cust_ID=B.Cust_ID JOIN BANKS BK O...  1 row(s) returned  0.000 sec / 0.000 sec
```

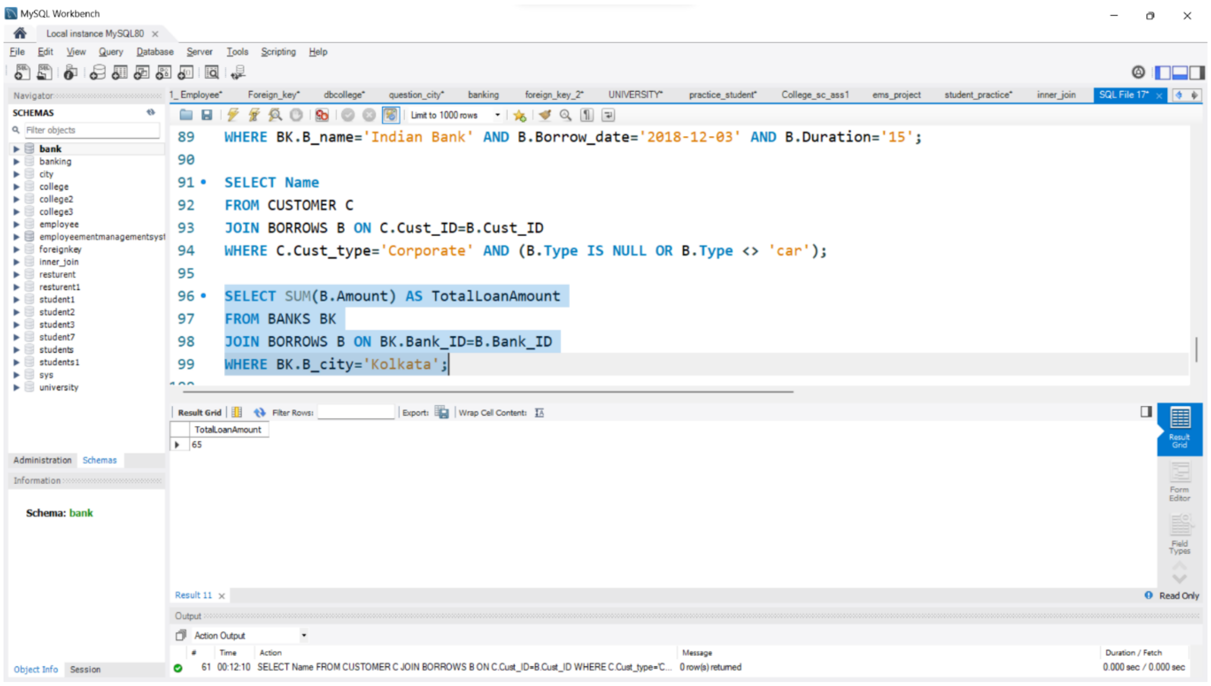
E)

SELECT SUM(B.Amount) AS TotalLoanAmount

FROM BANKS BK

JOIN BORROWS B ON BK.Bank_ID=B.Bank_ID

WHERE BK.B_city='Kolkata';



```
WHERE BK.B_name='Indian Bank' AND B.Borrow_date='2018-12-03' AND B.Duration='15';

SELECT Name
FROM CUSTOMER C
JOIN BORROWS B ON C.Cust_ID=B.Cust_ID
WHERE C.Cust_type='Corporate' AND (B.Type IS NULL OR B.Type <> 'car');

SELECT SUM(B.Amount) AS TotalLoanAmount
FROM BANKS BK
JOIN BORROWS B ON BK.Bank_ID=B.Bank_ID
WHERE BK.B_city='Kolkata';
```

```
TotalLoanAmount
65
```

```
#   Time      Action                                                                            Message            Duration / Fetch
61  00:12:10  SELECT Name FROM CUSTOMER C JOIN BORROWS B ON C.Cust_ID=B.Cust_ID WHERE C.Cust_type='C...  0 row(s) returned  0.000 sec / 0.000 sec
```

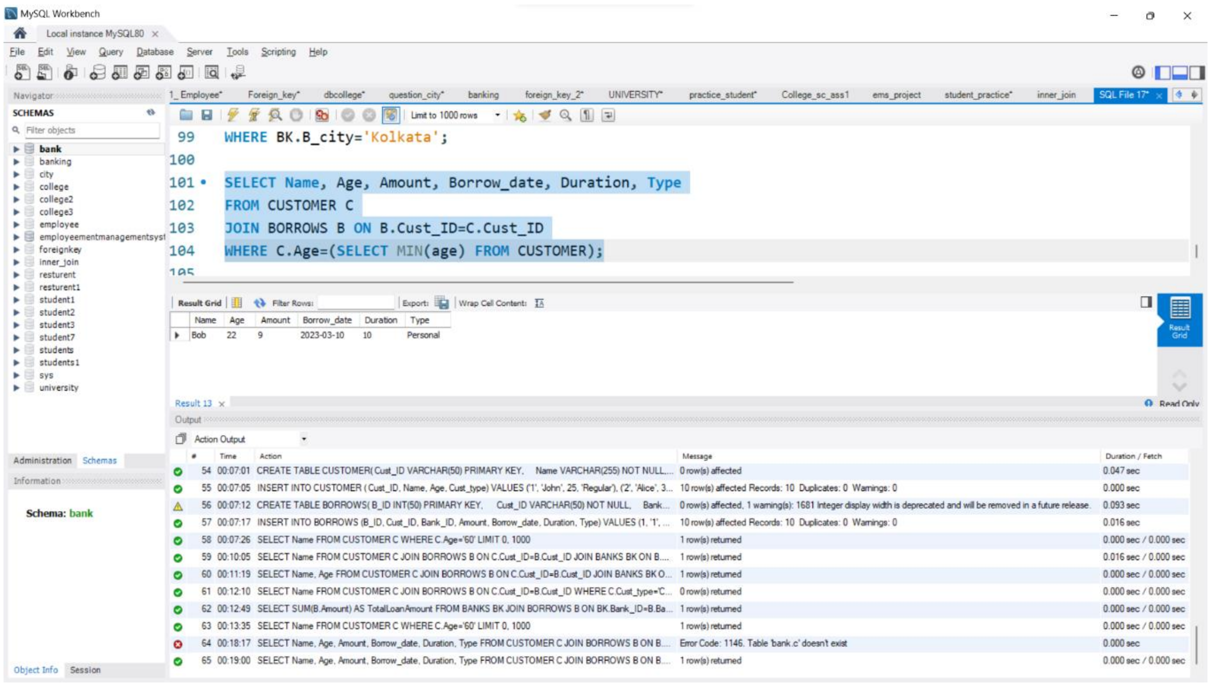
f)

SELECT Name, Age, Amount, Borrow_date, Duration, Type

FROM CUSTOMER C

JOIN BORROWS B ON B.Cust_ID=C.Cust_ID

WHERE C.Age=(SELECT MIN(age) FROM CUSTOMER);



```
WHERE BK.B_city='Kolkata';

SELECT Name, Age, Amount, Borrow_date, Duration, Type
FROM CUSTOMER C
JOIN BORROWS B ON B.Cust_ID=C.Cust_ID
WHERE C.Age=(SELECT MIN(age) FROM CUSTOMER);
```

Name	Age	Amount	Borrow_date	Duration	Type
Bob	22	9	2023-03-10	10	Personal

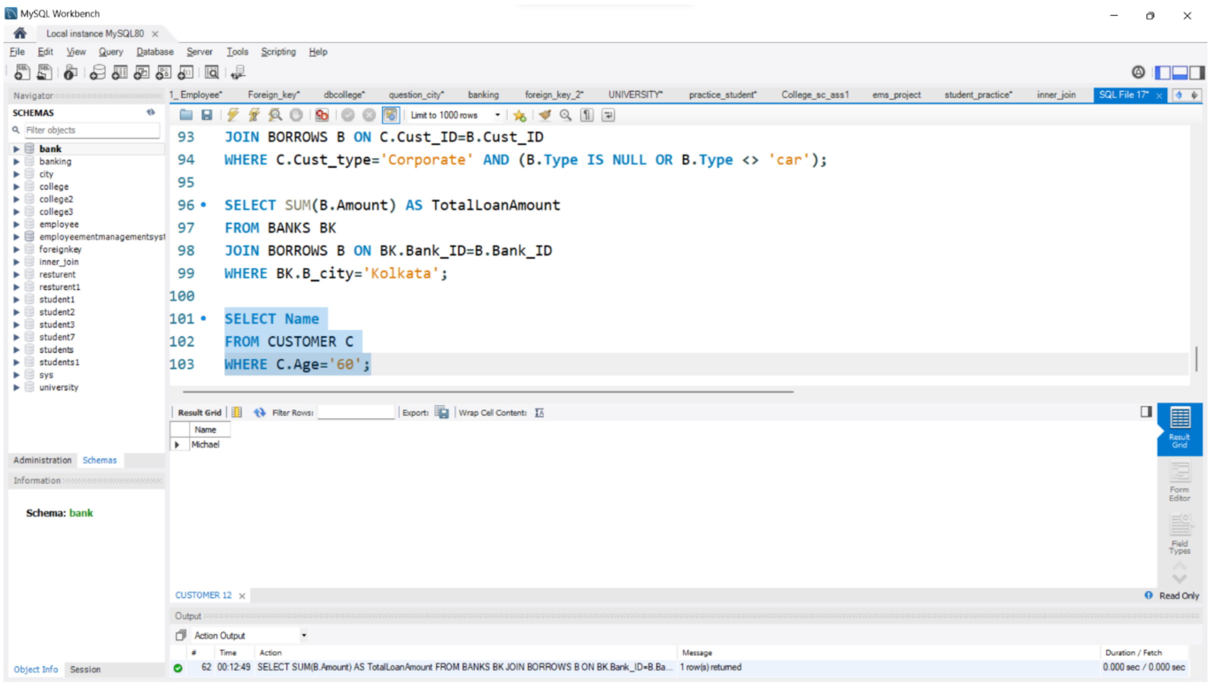
```
#   Time      Action                                                                                   Message                                                                                           Duration / Fetch
54  00:07:01  CREATE TABLE CUSTOMER( Cust_ID VARCHAR(50) PRIMARY KEY,    Name VARCHAR(255) NOT NULL,...  0 row(s) affected                                                                                 0.047 sec
55  00:07:05  INSERT INTO CUSTOMER ( Cust_ID, Name, Age, Cust_type) VALUES ('1', 'John', 25, 'Regular'), ('2', 'Alice', 3...  10 row(s) affected Records: 10  Duplicates: 0  Warnings: 0                          0.000 sec
56  00:07:12  CREATE TABLE BORROWS( B_ID INT(50) PRIMARY KEY,    Cust_ID VARCHAR(50) NOT NULL,    Bank...  0 row(s) affected, 1 warning(s): 1681 Integer display width is deprecated and will be removed in a future release.  0.093 sec
57  00:07:17  INSERT INTO BORROWS (B_ID, Cust_ID, Bank_ID, Amount, Borrow_date, Duration, Type) VALUES (1, '1', ...  10 row(s) affected Records: 10  Duplicates: 0  Warnings: 0                 0.016 sec
58  00:07:26  SELECT Name FROM CUSTOMER C WHERE C.Age='60' LIMIT 0, 1000                                 1 row(s) returned                                                                                 0.000 sec / 0.000 sec
59  00:10:05  SELECT Name FROM CUSTOMER C JOIN BORROWS B ON C.Cust_ID=B.Cust_ID JOIN BANKS BK ON B....  1 row(s) returned                                                                                 0.016 sec / 0.000 sec
60  00:11:19  SELECT Name, Age FROM CUSTOMER C JOIN BORROWS B ON C.Cust_ID=B.Cust_ID JOIN BANKS BK O...  1 row(s) returned                                                                                 0.000 sec / 0.000 sec
61  00:12:10  SELECT Name FROM CUSTOMER C JOIN BORROWS B ON C.Cust_ID=B.Cust_ID WHERE C.Cust_type='C...  0 row(s) returned                                                                                 0.000 sec / 0.000 sec
62  00:12:49  SELECT SUM(B.Amount) AS TotalLoanAmount FROM BANKS BK JOIN BORROWS B ON BK.Bank_ID=B.Ba...  1 row(s) returned                                                                                 0.000 sec / 0.000 sec
63  00:13:35  SELECT Name FROM CUSTOMER C WHERE C.Age='60' LIMIT 0, 1000                                 1 row(s) returned                                                                                 0.000 sec / 0.000 sec
64  00:18:17  SELECT Name, Age, Amount, Borrow_date, Duration, Type FROM CUSTOMER C JOIN BORROWS B ON B....  Error Code: 1146. Table 'bank.c' doesn't exist                                               0.000 sec
65  00:19:00  SELECT Name, Age, Amount, Borrow_date, Duration, Type FROM CUSTOMER C JOIN BORROWS B ON B....  1 row(s) returned                                                                                 0.000 sec / 0.000 sec
```

G)

SELECT Name

FROM CUSTOMER C

WHERE C.Age='60';



```
JOIN BORROWS B ON C.Cust_ID=B.Cust_ID
WHERE C.Cust_type='Corporate' AND (B.Type IS NULL OR B.Type <> 'car');

SELECT SUM(B.Amount) AS TotalLoanAmount
FROM BANKS BK
JOIN BORROWS B ON BK.Bank_ID=B.Bank_ID
WHERE BK.B_city='Kolkata';

SELECT Name
FROM CUSTOMER C
WHERE C.Age='60';
```

Name
Michael

```
#   Time      Action                                                                                   Message            Duration / Fetch
62  00:12:49  SELECT SUM(B.Amount) AS TotalLoanAmount FROM BANKS BK JOIN BORROWS B ON BK.Bank_ID=B.Ba...  1 row(s) returned  0.000 sec / 0.000 sec
```

Q2)

```
CREATE DATABASE SELL;
USE SELL;

CREATE TABLE CUSTOMERS(
cid INT PRIMARY KEY,
 cname VARCHAR(50) NOT NULL,
 city VARCHAR(50) NOT NULL,
 ph_no VARCHAR(15) UNIQUE
 );

 INSERT INTO CUSTOMERS
 (cid, cname, city, ph_no)
 VALUES
(1, 'John Doe', 'New York', '123-456-7890'),
(2, 'Alice Smith', 'Los Angeles', '555-123-4567'),
(3, 'Bob Johnson', 'Chicago', '888-555-1234'),
(4, 'Sarah Brown', 'Miami', '777-111-2222'),
(5, 'Emily White', 'Houston', '999-888-7777'),
(6, 'Michael Davis', 'Chicago', '333-444-5555'),
(7, 'Jennifer Lee', 'San Francisco', '444-333-2222'),
(8, 'David Wilson', 'Dallas', '666-555-4444'),
(9, 'Karen Miller', 'Boston', '222-333-4444'),
(10, 'James Harris', 'Chicago', '111-222-3333');

DROP TABLE CUSTOMERS;

CREATE TABLE ITEMS(
ino INT PRIMARY KEY,
iname VARCHAR(255) NOT NULL,
price INT NOT NULL,
type VARCHAR(25) CHECK(type IN('Groceries', 'Stationeries', 'Electronics'))
);

INSERT INTO ITEMS
(ino, iname, price, type)
VALUES
(101, 'Milk', 2.99, 'Electronics'),
(102, 'Notebook', 1.49, 'Stationeries'),
(103, 'Mobile Phone', 499.99, 'Electronics'),
(104, 'Bread', 1.79, 'Groceries'),
(105, 'Pen', 0.99, 'Stationeries'),
(106, 'Laptop', 799.99, 'Electronics'),
(107, 'Apples', 0.75, 'Groceries'),
(108, 'Stapler', 2.49, 'Stationeries'),
(109, 'Headphones', 49.99, 'Electronics'),
(110, 'Cereal', 3.49, 'Electronics');

CREATE TABLE ORDERS(
 order_no INT PRIMARY KEY,
 ino INT,
 cid INT,
 ord_date DATE,
 qty INT NOT NULL,
 FOREIGN KEY(ino) REFERENCES ITEMS(ino),
 FOREIGN KEY(cid) REFERENCES CUSTOMERS(cid)
 );

INSERT INTO ORDERS
(order_no, ino, cid, ord_date, qty)
VALUES
```

(1001, 101, 1, '2023-01-15', 2),

(1002, 102, 2, '2023-02-20', 5),

(1003, 103, 3, '2023-03-10', 1),

(1004, 104, 4, '2023-04-05', 3),

(1005, 105, 5, '2023-05-15', 10),

(1006, 106, 6, '2023-06-20', 2),

(1007, 107, 7, '2023-07-25', 6),

(1008, 108, 8, '2023-08-10', 3),

(1009, 109, 9, '2023-09-05', 4),

(1010, 110, 10, '2023-10-01', 2);

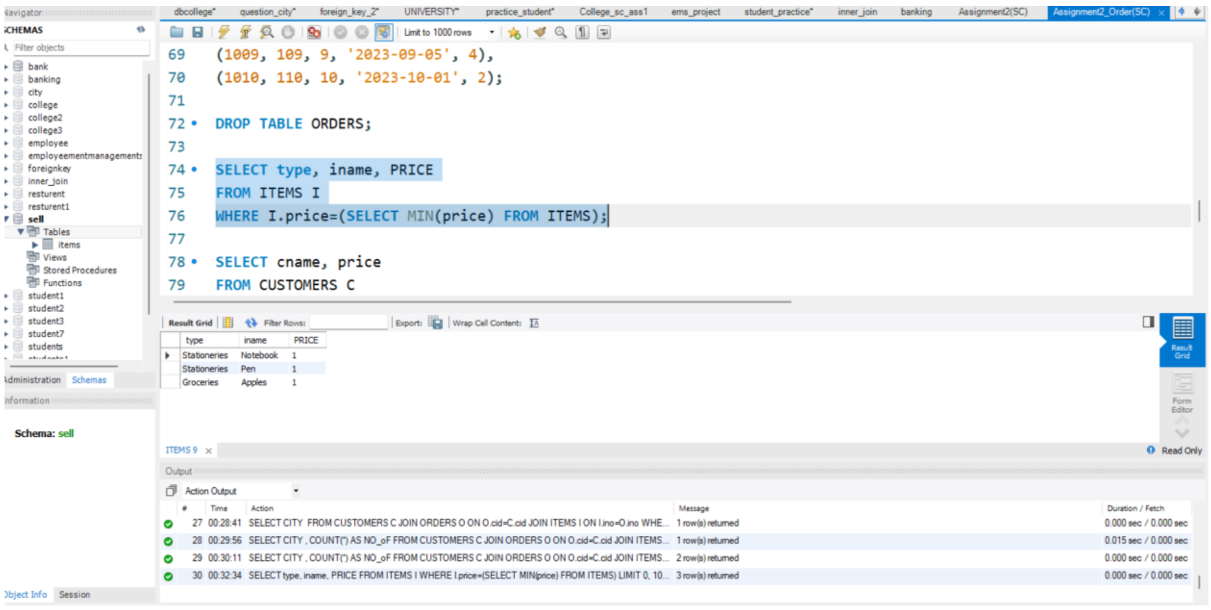
DROP TABLE ORDERS;

a.

SELECT type, iname, PRICE

FROM ITEMS I

WHERE I.price=(SELECT MIN(price) FROM ITEMS);



```
(1009, 109, 9, '2023-09-05', 4),
(1010, 110, 10, '2023-10-01', 2);

DROP TABLE ORDERS;

SELECT type, iname, PRICE
FROM ITEMS I
WHERE I.price=(SELECT MIN(price) FROM ITEMS);

SELECT cname, price
FROM CUSTOMERS C
```

type	iname	PRICE
Stationeries	Notebook	1
Stationeries	Pen	1
Groceries	Apples	1

```
#   Time      Action                                                                                   Message            Duration / Fetch
27  00:28:41  SELECT CITY  FROM CUSTOMERS C JOIN ORDERS O ON O.cid=C.cid JOIN ITEMS I ON I.ino=O.ino WHE...  1 row(s) returned  0.000 sec / 0.000 sec
28  00:29:56  SELECT CITY , COUNT(*) AS NO_oF FROM CUSTOMERS C JOIN ORDERS O ON O.cid=C.cid JOIN ITEMS...   1 row(s) returned  0.015 sec / 0.000 sec
29  00:30:11  SELECT CITY , COUNT(*) AS NO_oF FROM CUSTOMERS C JOIN ORDERS O ON O.cid=C.cid JOIN ITEMS...   2 row(s) returned  0.000 sec / 0.000 sec
30  00:32:34  SELECT type, iname, PRICE FROM ITEMS I WHERE I.price=(SELECT MIN(price) FROM ITEMS) LIMIT 0, 10...  3 row(s) returned  0.000 sec / 0.000 sec
```

b)

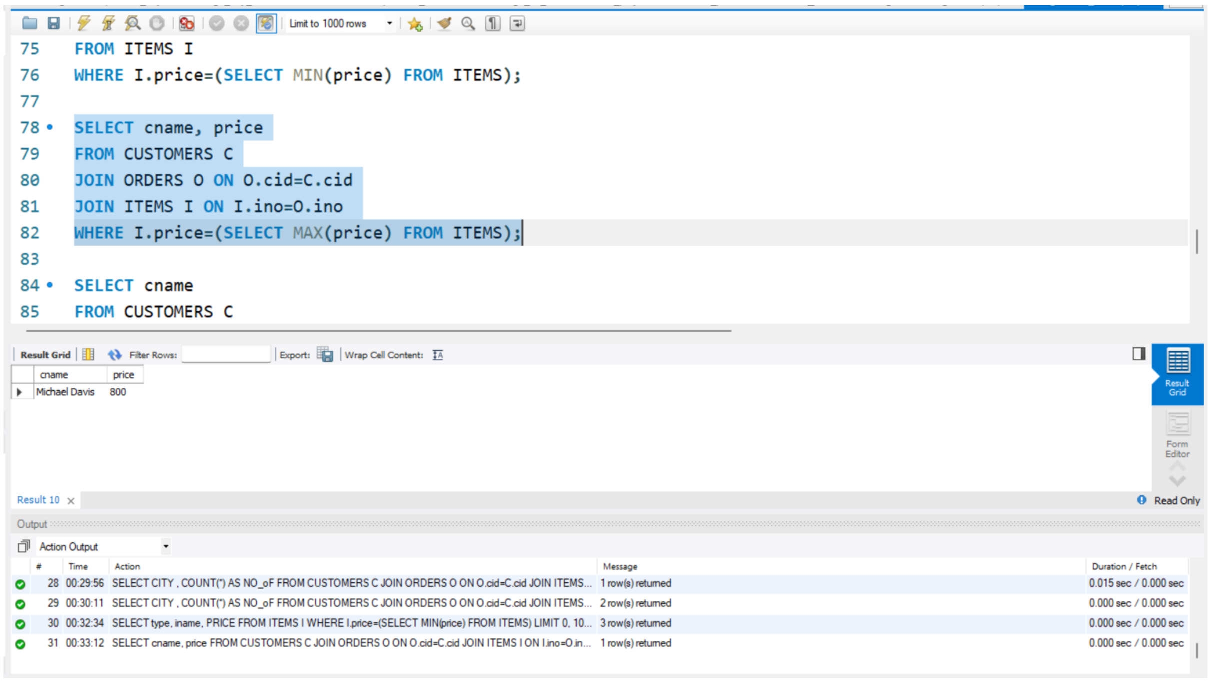
SELECT cname, price

FROM CUSTOMERS C

JOIN ORDERS O ON O.cid=C.cid

JOIN ITEMS I ON I.ino=O.ino

WHERE I.price=(SELECT MAX(price) FROM ITEMS);



```
FROM ITEMS I
WHERE I.price=(SELECT MIN(price) FROM ITEMS);

SELECT cname, price
FROM CUSTOMERS C
JOIN ORDERS O ON O.cid=C.cid
JOIN ITEMS I ON I.ino=O.ino
WHERE I.price=(SELECT MAX(price) FROM ITEMS);

SELECT cname
FROM CUSTOMERS C
```

cname	price
Michael Davis	800

```
#   Time      Action                                                                              Message              Duration / Fetch
28  00:29:56  SELECT CITY , COUNT(*) AS NO_oF FROM CUSTOMERS C JOIN ORDERS O ON O.cid=C.cid JOIN ITEMS...  1 row(s) returned  0.015 sec / 0.000 sec
29  00:30:11  SELECT CITY , COUNT(*) AS NO_oF FROM CUSTOMERS C JOIN ORDERS O ON O.cid=C.cid JOIN ITEMS...  2 row(s) returned  0.000 sec / 0.000 sec
30  00:32:34  SELECT type, iname, PRICE FROM ITEMS I WHERE I.price=(SELECT MIN(price) FROM ITEMS) LIMIT 0, 10...  3 row(s) returned  0.000 sec / 0.000 sec
31  00:33:12  SELECT cname, price FROM CUSTOMERS C JOIN ORDERS O ON O.cid=C.cid JOIN ITEMS I ON I.ino=O.in...  1 row(s) returned  0.000 sec / 0.000 sec
```

C)

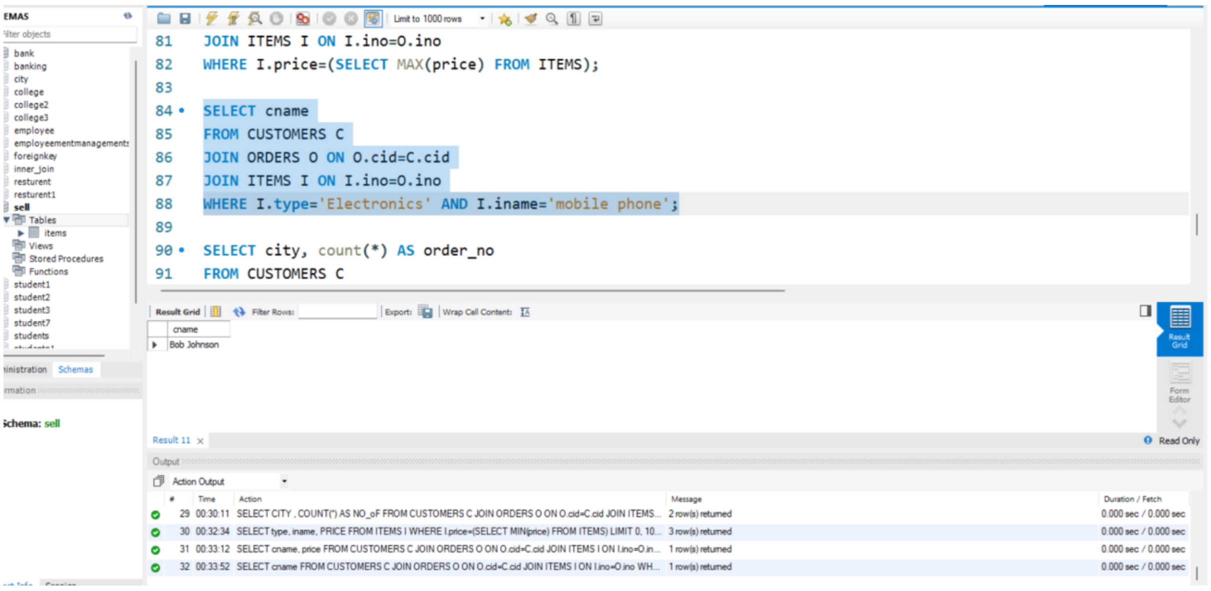
SELECT cname

FROM CUSTOMERS C

JOIN ORDERS O ON O.cid=C.cid

JOIN ITEMS I ON I.ino=O.ino

WHERE I.type='Electronics' AND I.iname='mobile phone';



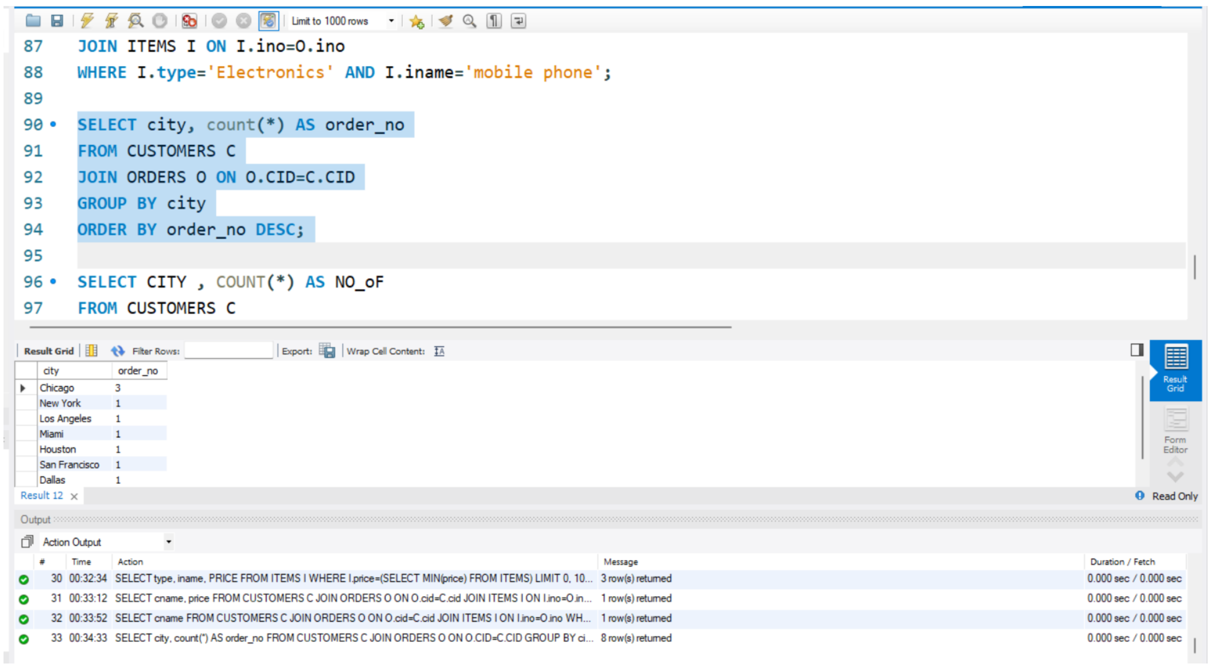
```
JOIN ITEMS I ON I.ino=O.ino
WHERE I.price=(SELECT MAX(price) FROM ITEMS);

SELECT cname
FROM CUSTOMERS C
JOIN ORDERS O ON O.cid=C.cid
JOIN ITEMS I ON I.ino=O.ino
WHERE I.type='Electronics' AND I.iname='mobile phone';

SELECT city, count(*) AS order_no
FROM CUSTOMERS C
```

cname
Bob Johnson

```
#   Time      Action                                                                              Message              Duration / Fetch
29  00:30:11  SELECT CITY , COUNT(*) AS NO_oF FROM CUSTOMERS C JOIN ORDERS O ON O.cid=C.cid JOIN ITEMS...  2 row(s) returned  0.000 sec / 0.000 sec
30  00:32:34  SELECT type, iname, PRICE FROM ITEMS I WHERE I.price=(SELECT MIN(price) FROM ITEMS) LIMIT 0, 10...  3 row(s) returned  0.000 sec / 0.000 sec
31  00:33:12  SELECT cname, price FROM CUSTOMERS C JOIN ORDERS O ON O.cid=C.cid JOIN ITEMS I ON I.ino=O.in...  1 row(s) returned  0.000 sec / 0.000 sec
32  00:33:52  SELECT cname FROM CUSTOMERS C JOIN ORDERS O ON O.cid=C.cid JOIN ITEMS I ON I.ino=O.ino WH...  1 row(s) returned  0.000 sec / 0.000 sec
```

E)

```
SELECT city, count(*) AS order_no

FROM CUSTOMERS C

JOIN ORDERS O ON O.CID=C.CID

GROUP BY city

ORDER BY order_no DESC;
```

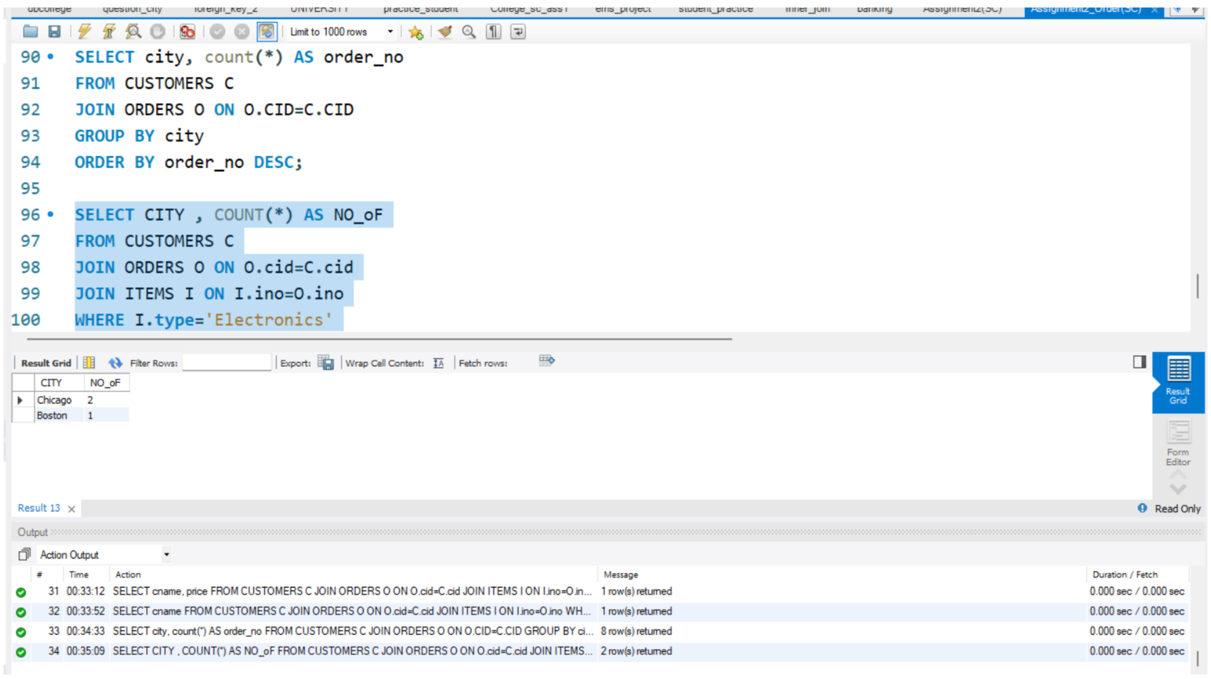


```
JOIN ITEMS I ON I.ino=O.ino
WHERE I.type='Electronics' AND I.iname='mobile phone';

SELECT city, count(*) AS order_no
FROM CUSTOMERS C
JOIN ORDERS O ON O.CID=C.CID
GROUP BY city
ORDER BY order_no DESC;

SELECT CITY , COUNT(*) AS NO_oF
FROM CUSTOMERS C
```

city	order_no
Chicago	3
New York	1
Los Angeles	1
Miami	1
Houston	1
San Francisco	1
Dallas	1

```
#   Time      Action                                                                              Message            Duration / Fetch
30  00:32:34  SELECT type, iname, PRICE FROM ITEMS I WHERE I.price=(SELECT MIN(price) FROM ITEMS) LIMIT 0, 10...  3 row(s) returned  0.000 sec / 0.000 sec
31  00:33:12  SELECT cname, price FROM CUSTOMERS C JOIN ORDERS O ON O.cid=C.cid JOIN ITEMS I ON I.ino=O.in...  1 row(s) returned  0.000 sec / 0.000 sec
32  00:33:52  SELECT cname FROM CUSTOMERS C JOIN ORDERS O ON O.cid=C.cid JOIN ITEMS I ON I.ino=O.ino WH...      1 row(s) returned  0.000 sec / 0.000 sec
33  00:34:33  SELECT city, count(*) AS order_no FROM CUSTOMERS C JOIN ORDERS O ON O.CID=C.CID GROUP BY ci...  8 row(s) returned  0.000 sec / 0.000 sec
```

```
SELECT CITY , COUNT(*) AS NO_oF

FROM CUSTOMERS C

JOIN ORDERS O ON O.cid=C.cid

JOIN ITEMS I ON I.ino=O.ino

WHERE I.type='Electronics'

GROUP BY C.city

ORDER BY COUNT(*) DESC LIMIT 2;
```



```
SELECT city, count(*) AS order_no
FROM CUSTOMERS C
JOIN ORDERS O ON O.CID=C.CID
GROUP BY city
ORDER BY order_no DESC;

SELECT CITY , COUNT(*) AS NO_oF
FROM CUSTOMERS C
JOIN ORDERS O ON O.cid=C.cid
JOIN ITEMS I ON I.ino=O.ino
WHERE I.type='Electronics'
```

	CITY	NO_oF
▶	Chicago	2
	Boston	1

```
#   Time      Action                                                                                          Message            Duration / Fetch
31  00:33:12  SELECT cname, price FROM CUSTOMERS C JOIN ORDERS O ON O.cid=C.cid JOIN ITEMS I ON I.ino=O.in...  1 row(s) returned  0.000 sec / 0.000 sec
32  00:33:52  SELECT cname FROM CUSTOMERS C JOIN ORDERS O ON O.cid=C.cid JOIN ITEMS I ON I.ino=O.ino WH...     1 row(s) returned  0.000 sec / 0.000 sec
33  00:34:33  SELECT city, count(*) AS order_no FROM CUSTOMERS C JOIN ORDERS O ON O.CID=C.CID GROUP BY ci...   8 row(s) returned  0.000 sec / 0.000 sec
34  00:35:09  SELECT CITY , COUNT(*) AS NO_oF FROM CUSTOMERS C JOIN ORDERS O ON O.cid=C.cid JOIN ITEMS...      2 row(s) returned  0.000 sec / 0.000 sec
```

3. Consider the following Hotel Management Database

HOTEL(Hid, Name, City)

ROOM(Rid, Hid, tariff)

BOOKING(Booking_no, Guest_name, Hid, Rid, start_date, end_date)

Create the above relations through SQL commands specifying integrity constraints. Write SQL commands for the following queries. (Insert sufficient records in each table so that the queries yield some results)

a. Give the names of guests who have stayed in the same hotel at least thrice.

b. List the costliest rooms in each hotel.

c. List the total earnings from bookings in the last month for each hotel in Kolkata.

d. Insert a new booking in the database.

e. Change the start_date and end_date of the new booking.

f. Delete the record for the rooms with no bookings in the last 6 months.

```
CREATE DATABASE HOTEL_MANAGEMENT_SYSTEM;

USE HOTEL_MANAGEMENT_SYSTEM;

CREATE TABLE HOTEL (

    Hid INT PRIMARY KEY,

    Name VARCHAR(255) NOT NULL,
```

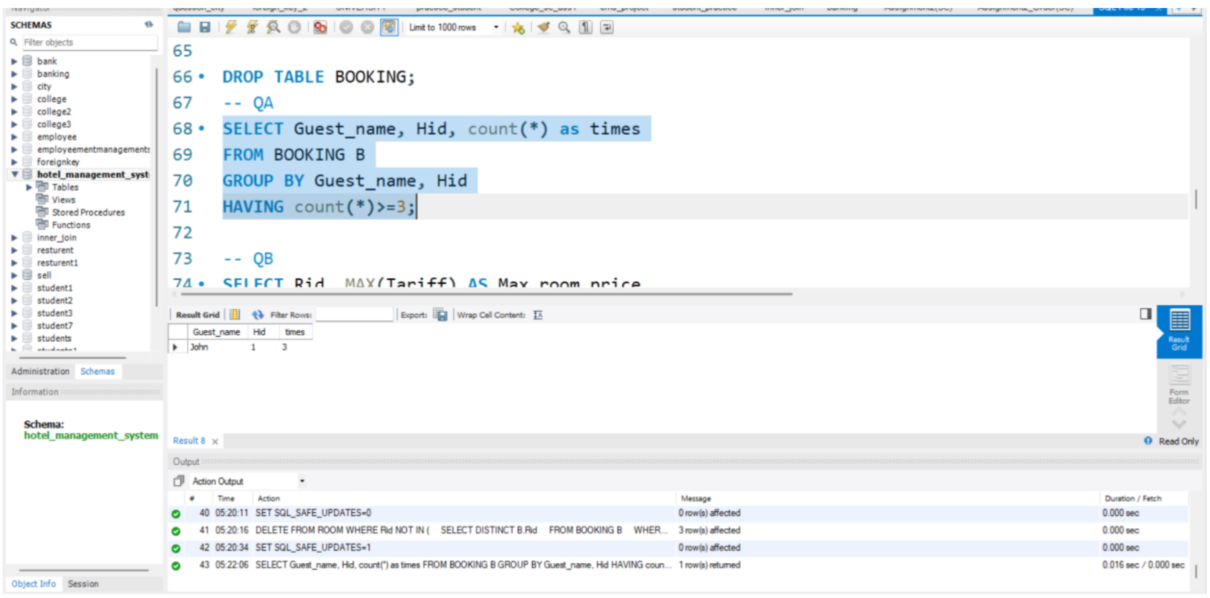
```sql
    City VARCHAR(255) NOT NULL
);



CREATE TABLE ROOM (
    Rid INT PRIMARY KEY,
    Hid INT,
    Tariff DECIMAL(10, 2) NOT NULL,
    FOREIGN KEY (Hid) REFERENCES HOTEL(Hid)
);

-- Create BOOKING table with foreign key references to HOTEL and ROOM
CREATE TABLE BOOKING (
    Booking_no INT PRIMARY KEY,
    Guest_name VARCHAR(255) NOT NULL,
    Hid INT,
    Rid INT,
    Start_date DATE NOT NULL,
    End_date DATE NOT NULL,
    FOREIGN KEY (Hid) REFERENCES HOTEL(Hid),
    FOREIGN KEY (Rid) REFERENCES ROOM(Rid)
);
INSERT INTO HOTEL (Hid, Name, City) VALUES
(1, 'Hotel A', 'Kolkata'),
(2, 'Hotel B', 'Mumbai'),
(3, 'Hotel C', 'Delhi'),
(4, 'Hotel D', 'Bangalore'),
(5, 'Hotel E', 'Chennai'),
(6, 'Hotel F', 'Hyderabad'),
(7, 'Hotel G', 'Pune'),
(8, 'Hotel H', 'Jaipur'),
(9, 'Hotel I', 'Lucknow'),
(10, 'Hotel J', 'Ahmedabad');

INSERT INTO ROOM
(Rid, Hid, Tariff) VALUES
(101, 1, 1500),
(102, 1, 2000),
(201, 2, 1800),
(202, 2, 2200),
(301, 3, 1200),
(302, 3, 1600),
(401, 4, 2500),
(402, 4, 3000),
(501, 5, 1700),
(502, 5, 1900);

INSERT INTO BOOKING
(Booking_no, Guest_name, Hid, Rid, Start_date, End_date) VALUES
(1, 'John', 1, 101, '2023-09-01', '2023-09-05'),
(2, 'Alice', 1, 101, '2023-08-10', '2023-08-15'),
(3, 'John', 1, 202, '2023-09-20', '2023-09-25'),
(4, 'Carol', 2, 201, '2023-08-05', '2023-08-10'),
(5, 'David', 3, 302, '2023-09-05', '2023-09-10'),
(6, 'Eve', 3, 302, '2023-10-01', '2023-10-05'),
(7, 'Frank', 1, 102, '2023-10-15', '2023-10-20'),
(8, 'Grace', 2, 202, '2023-11-01', '2023-11-05'),
(9, 'John',1, 402, '2023-10-10', '2023-10-20'),
(10, 'Isabel', 5, 502, '2023-11-05', '2023-11-10');

DROP TABLE BOOKING;
-- QA
```

SELECT Guest_name, Hid, count(*) as times

FROM BOOKING B

GROUP BY Guest_name, Hid

HAVING count(*)>=3;



```
DROP TABLE BOOKING;
-- QA
SELECT Guest_name, Hid, count(*) as times
FROM BOOKING B
GROUP BY Guest_name, Hid
HAVING count(*)>=3;

-- QB
SELECT Rid, MAX(Tariff) AS Max_room_price
```

Guest_name	Hid	times
John	1	3

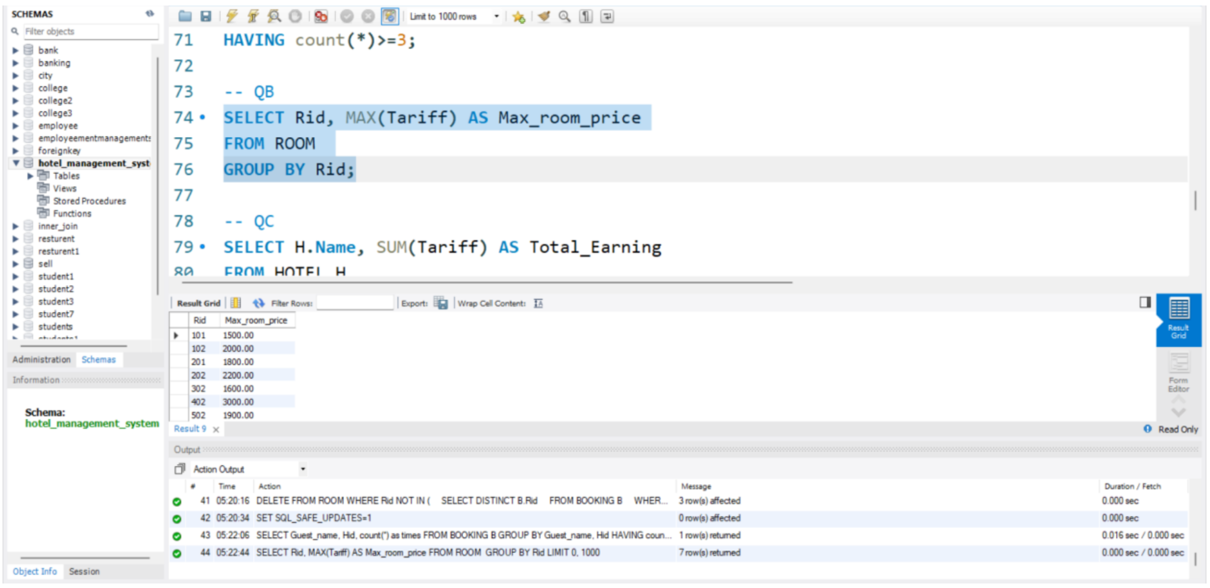
```
#   Time      Action                                                                                  Message            Duration / Fetch
40  05:20:11  SET SQL_SAFE_UPDATES=0                                                                  0 row(s) affected  0.000 sec
41  05:20:16  DELETE FROM ROOM WHERE Rid NOT IN (    SELECT DISTINCT B.Rid    FROM BOOKING B    WHER...  3 row(s) affected  0.000 sec
42  05:20:34  SET SQL_SAFE_UPDATES=1                                                                  0 row(s) affected  0.000 sec
43  05:22:06  SELECT Guest_name, Hid, count(*) as times FROM BOOKING B GROUP BY Guest_name, Hid HAVING coun...  1 row(s) returned  0.016 sec / 0.000 sec
```

-- QB

SELECT Rid, MAX(Tariff) AS Max_room_price

FROM ROOM

GROUP BY Rid;



```
HAVING count(*)>=3;

-- QB
SELECT Rid, MAX(Tariff) AS Max_room_price
FROM ROOM
GROUP BY Rid;

-- QC
SELECT H.Name, SUM(Tariff) AS Total_Earning
FROM HOTEL H
```

Rid	Max_room_price
101	1500.00
102	2000.00
201	1800.00
202	2200.00
302	1600.00
402	3000.00
502	1900.00

```
#   Time      Action                                                                                  Message            Duration / Fetch
41  05:20:16  DELETE FROM ROOM WHERE Rid NOT IN (    SELECT DISTINCT B.Rid    FROM BOOKING B    WHER...  3 row(s) affected  0.000 sec
42  05:20:34  SET SQL_SAFE_UPDATES=1                                                                  0 row(s) affected  0.000 sec
43  05:22:06  SELECT Guest_name, Hid, count(*) as times FROM BOOKING B GROUP BY Guest_name, Hid HAVING coun...  1 row(s) returned  0.016 sec / 0.000 sec
44  05:22:44  SELECT Rid, MAX(Tariff) AS Max_room_price FROM ROOM  GROUP BY Rid LIMIT 0, 1000           7 row(s) returned  0.000 sec / 0.000 sec
```

-- QC

SELECT H.Name, SUM(Tariff) AS Total_Earning

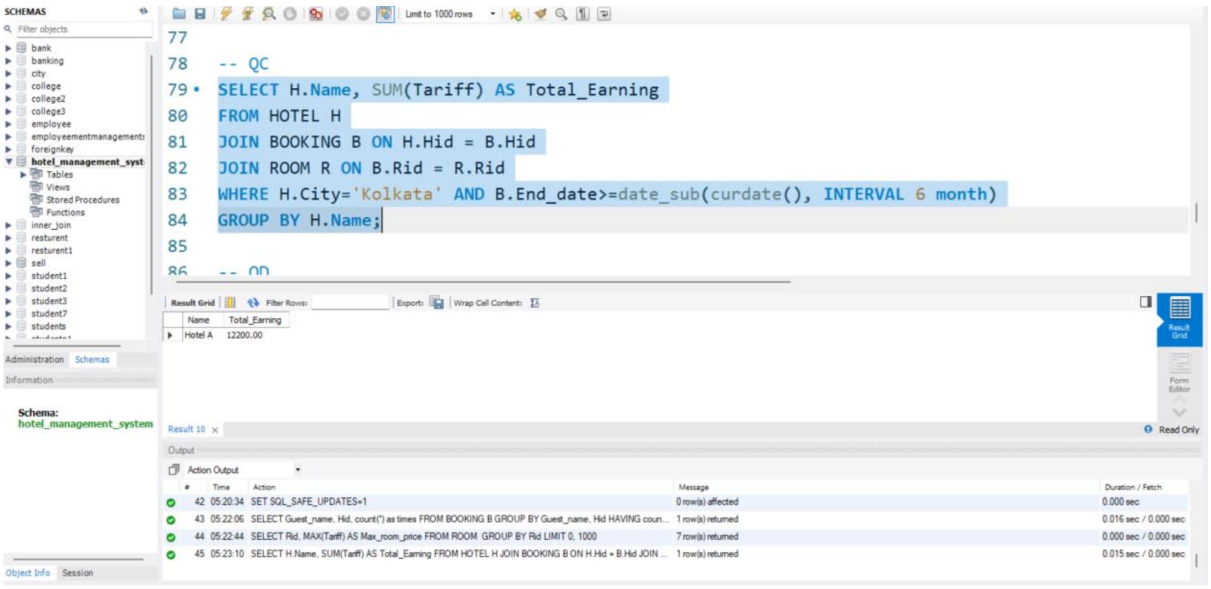
FROM HOTEL H

JOIN BOOKING B ON H.Hid = B.Hid

JOIN ROOM R ON B.Rid = R.Rid

WHERE H.City='Kolkata' AND B.End_date>=date_sub(curdate(), INTERVAL 6 month)

GROUP BY H.Name;



```sql
-- QC
SELECT H.Name, SUM(Tariff) AS Total_Earning
FROM HOTEL H
JOIN BOOKING B ON H.Hid = B.Hid
JOIN ROOM R ON B.Rid = R.Rid
WHERE H.City='Kolkata' AND B.End_date>=date_sub(curdate(), INTERVAL 6 month)
GROUP BY H.Name;

-- QD
```

Name	Total_Earning
Hotel A	12200.00

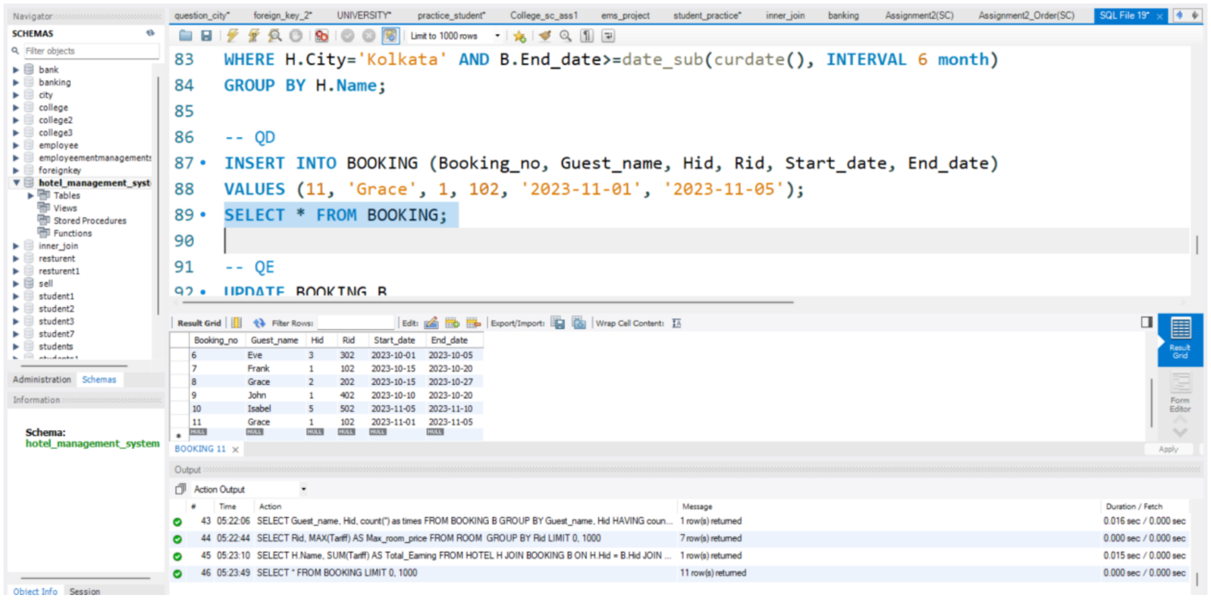
```
#   Time      Action                                                                                    Message             Duration / Fetch
42  05:20:34  SET SQL_SAFE_UPDATES=1                                                                    0 row(s) affected   0.000 sec
43  05:22:06  SELECT Guest_name, Hid, count(*) as times FROM BOOKING B GROUP BY Guest_name, Hid HAVING coun...  1 row(s) returned   0.016 sec / 0.000 sec
44  05:22:44  SELECT Rid, MAX(Tariff) AS Max_room_price FROM ROOM  GROUP BY Rid LIMIT 0, 1000                7 row(s) returned   0.000 sec / 0.000 sec
45  05:23:10  SELECT H.Name, SUM(Tariff) AS Total_Earning FROM HOTEL H JOIN BOOKING B ON H.Hid = B.Hid JOIN ...  1 row(s) returned   0.015 sec / 0.000 sec
```

-- QD

INSERT INTO BOOKING (Booking_no, Guest_name, Hid, Rid, Start_date, End_date)

VALUES (11, 'Grace', 1, 102, '2023-11-01', '2023-11-05');

SELECT * FROM BOOKING;

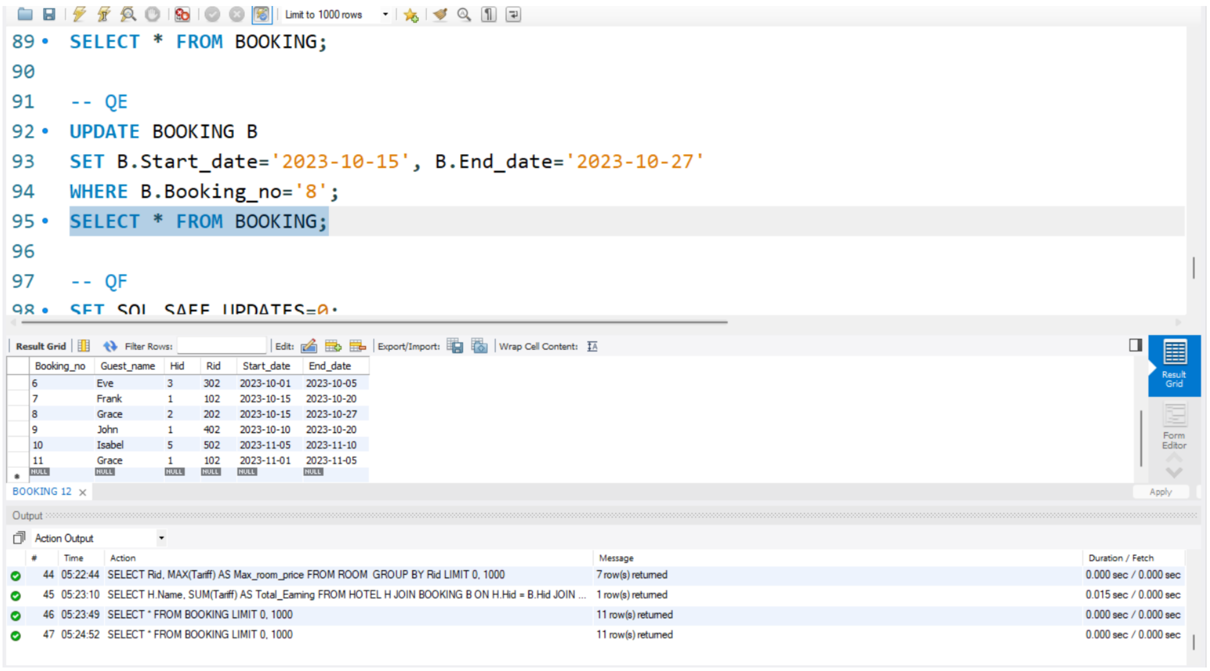


```sql
WHERE H.City='Kolkata' AND B.End_date>=date_sub(curdate(), INTERVAL 6 month)
GROUP BY H.Name;

-- QD
INSERT INTO BOOKING (Booking_no, Guest_name, Hid, Rid, Start_date, End_date)
VALUES (11, 'Grace', 1, 102, '2023-11-01', '2023-11-05');
SELECT * FROM BOOKING;

-- QE
UPDATE BOOKING B
```

Booking_no	Guest_name	Hid	Rid	Start_date	End_date
6	Eve	3	302	2023-10-01	2023-10-05
7	Frank	1	102	2023-10-15	2023-10-20
8	Grace	2	202	2023-10-15	2023-10-27
9	John	1	402	2023-10-10	2023-10-20
10	Isabel	5	502	2023-11-05	2023-11-10
11	Grace	1	102	2023-11-01	2023-11-05
NULL	NULL	NULL	NULL	NULL	NULL

```
#   Time      Action                                                                                    Message              Duration / Fetch
43  05:22:06  SELECT Guest_name, Hid, count(*) as times FROM BOOKING B GROUP BY Guest_name, Hid HAVING coun...  1 row(s) returned    0.016 sec / 0.000 sec
44  05:22:44  SELECT Rid, MAX(Tariff) AS Max_room_price FROM ROOM  GROUP BY Rid LIMIT 0, 1000                7 row(s) returned    0.000 sec / 0.000 sec
45  05:23:10  SELECT H.Name, SUM(Tariff) AS Total_Earning FROM HOTEL H JOIN BOOKING B ON H.Hid = B.Hid JOIN ...  1 row(s) returned    0.015 sec / 0.000 sec
46  05:23:49  SELECT * FROM BOOKING LIMIT 0, 1000                                                       11 row(s) returned   0.000 sec / 0.000 sec
```

```
-- QE
UPDATE BOOKING B
SET B.Start_date='2023-10-15', B.End_date='2023-10-27'
WHERE B.Booking_no='8';
SELECT * FROM BOOKING;
```



```
SELECT * FROM BOOKING;

-- QE
UPDATE BOOKING B
SET B.Start_date='2023-10-15', B.End_date='2023-10-27'
WHERE B.Booking_no='8';
SELECT * FROM BOOKING;

-- QF
SET SQL_SAFE_UPDATES=0;
```

Booking_no	Guest_name	Hid	Rid	Start_date	End_date
6	Eve	3	302	2023-10-01	2023-10-05
7	Frank	1	102	2023-10-15	2023-10-20
8	Grace	2	202	2023-10-15	2023-10-27
9	John	1	402	2023-10-10	2023-10-20
10	Isabel	5	502	2023-11-05	2023-11-10
11	Grace	1	102	2023-11-01	2023-11-05
NULL	NULL	NULL	NULL	NULL	NULL

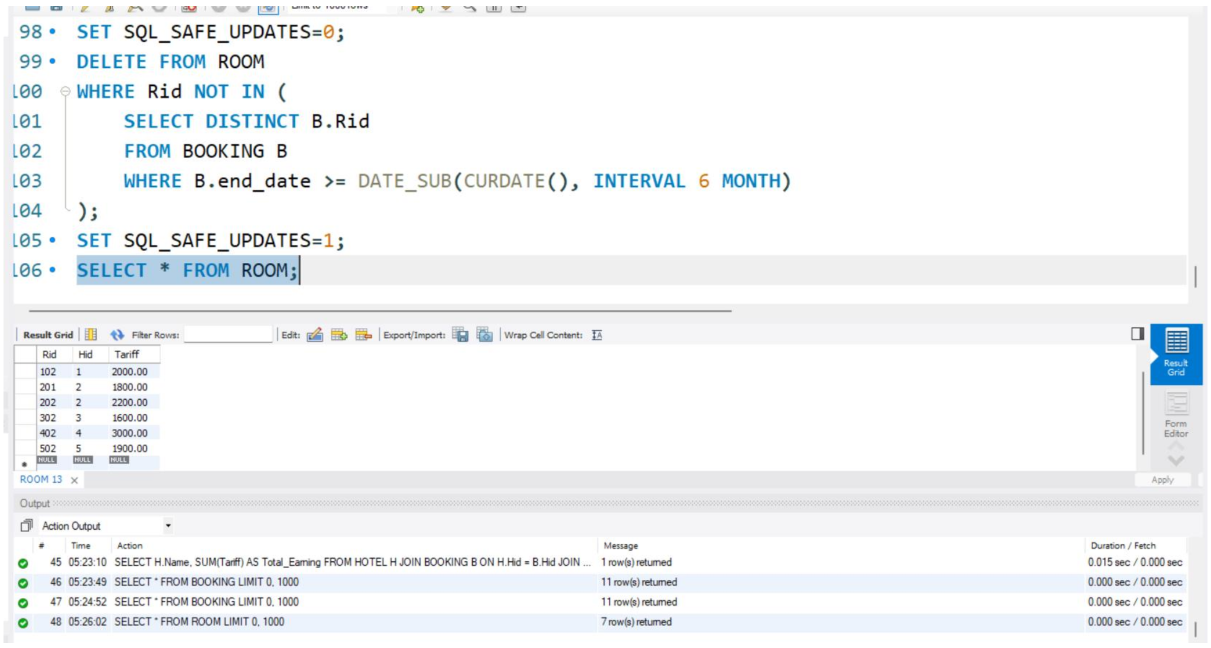
BOOKING 12

Output

Action Output

```
#   Time      Action                                                                          Message            Duration / Fetch
44  05:22:44  SELECT Rid, MAX(Tariff) AS Max_room_price FROM ROOM  GROUP BY Rid LIMIT 0, 1000  7 row(s) returned  0.000 sec / 0.000 sec
45  05:23:10  SELECT H.Name, SUM(Tariff) AS Total_Earning FROM HOTEL H JOIN BOOKING B ON H.Hid = B.Hid JOIN ...  1 row(s) returned  0.015 sec / 0.000 sec
46  05:23:49  SELECT * FROM BOOKING LIMIT 0, 1000                                              11 row(s) returned  0.000 sec / 0.000 sec
47  05:24:52  SELECT * FROM BOOKING LIMIT 0, 1000                                              11 row(s) returned  0.000 sec / 0.000 sec
```

```
-- QF
SET SQL_SAFE_UPDATES=0;
DELETE FROM ROOM
WHERE Rid NOT IN (
    SELECT DISTINCT B.Rid
    FROM BOOKING B
    WHERE B.end_date >= DATE_SUB(CURDATE(), INTERVAL 6 MONTH)
);
SET SQL_SAFE_UPDATES=1;
SELECT * FROM ROOM;
```



```sql
SET SQL_SAFE_UPDATES=0;
DELETE FROM ROOM
WHERE Rid NOT IN (
    SELECT DISTINCT B.Rid
    FROM BOOKING B
    WHERE B.end_date >= DATE_SUB(CURDATE(), INTERVAL 6 MONTH)
);
SET SQL_SAFE_UPDATES=1;
SELECT * FROM ROOM;
```

Rid	Hid	Tariff
102	1	2000.00
201	2	1800.00
202	2	2200.00
302	3	1600.00
402	4	3000.00
502	5	1900.00
NULL	NULL	NULL

```
#   Time      Action                                                                               Message           Duration / Fetch
45  05:23:10  SELECT H.Name, SUM(Tariff) AS Total_Earning FROM HOTEL H JOIN BOOKING B ON H.Hid = B.Hid JOIN ...  1 row(s) returned   0.015 sec / 0.000 sec
46  05:23:49  SELECT * FROM BOOKING LIMIT 0, 1000                                                  11 row(s) returned  0.000 sec / 0.000 sec
47  05:24:52  SELECT * FROM BOOKING LIMIT 0, 1000                                                  11 row(s) returned  0.000 sec / 0.000 sec
48  05:26:02  SELECT * FROM ROOM LIMIT 0, 1000                                                     7 row(s) returned   0.000 sec / 0.000 sec
```

4. Create the below relations through SQL commands specifying integrity constraints.

EMPLOYEE (Fname, Lname, Essn, Bdate, Address, Sex, Salary, Mgr_ssn, Dno)

DEPARTMENT(Dname, Dnumber, Mgr_ssn, Mgr_start_date)

DEPT_LOCATIONS (Dnumber, Dlocation)

PROJECT(Pname, Pnumber, Plocation, Dnumber)

WORKS_ON (Essn, Pnumber, Hours)

DEPENDENT (Essn, Dependent_name, Sex, Bdate, Relationship)

Specify the following queries in SQL:

a. Retrieve the names of all employees who work in the department that has

the employee with the highest salary among all employees.

b. Retrieve the names of all employees whose supervisor’s supervisor has

‘888665555’ for Ssn.

c. Retrieve the names of employees who make at least $10,000 more than

the employee who is paid the least in the company.

d. For each department whose average employee salary is more than

$30,000, retrieve the department name and the number of employees

working for that department.

e. Retrieve the number of male employees in each department making more than

$30,000.

f. For each department that has more than five employees, retrieve the department number and the number of its employees who are making more than $40,000.

g. Find the sum of the salaries of all employees, the maximum salary, the minimum salary, and the average salary.

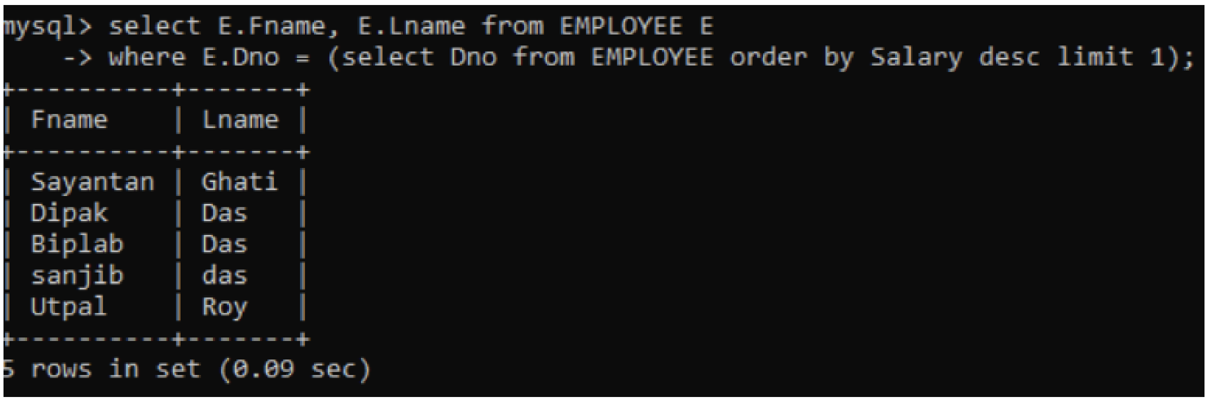
h. List the names of managers who have at least one dependent.

```
create table EMPLOYEE(
fname varchar(50) not null,
lname varchar(50) not null,
Essn varchar(50) primary key,
Bdate varchar(50) not null,
Address varchar(50),
sex varchar(50),
salary varchar(50),
Mgr_ssn varchar(50),
Dno varchar(50)
);

create table DEPARTMENT(
Dname varchar(50),
Dnumber varchar(50) primary key,
Mgr_ssn varchar(50),
Mgr_start_date date,
foreign key (Mgr_ssn) references EMPLOYEE(Essn)
);

create table DEPT_LOCATION(
Dnumber varchar(50),
Dlocation varchar(50),
primary key (Dnumber,Dlocation),
foreign key (Dnumber) references DEPARTMENT(Dnumber)
);

create table PROJECT(
Pname varchar(50),
Pnumber varchar(50) primary key,
Plocation varchar(50),
Dnumber varchar(50),
foreign key (Dnumber) references DEPARTMENT(Dnumber)
);

create table WORKS_ON(
Essn varchar(50) primary key,
Pnumber varchar(50),
Hours decimal(15,2),
foreign key (Pnumber) references PROJECT(Pnumber)
);

create table DEPENDENT(
Essn varchar(50) primary key,
Dependent_name varchar(50),
Sex varchar(50),
Bdate date,
Relationship varchar(50),
foreign key (Essn) references EMPLOYEE(Essn)
);
INSERT INTO EMPLOYEE (Fname, Lname, Essn, Bdate, Address, Sex, Salary, Mgr_ssn, Dno)
VALUES
('John', 'Doe', '111223333', '1980-05-15', '123 Main St', 'M', 50000, NULL, 1),
('Jane', 'Smith', '222334444', '1982-09-20', '456 Elm St', 'F', 55000, '111223333', 1),
('Michael', 'Johnson', '333445555', '1978-02-10', '789 Oak St', 'M', 60000, '111223333', 2),
('Emily', 'Brown', '444556666', '1985-07-07', '101 Pine St', 'F', 52000, '222334444', 2),
('David', 'Lee', '555667777', '1987-11-30', '202 Cedar St', 'M', 48000, '222334444', 1),
('Sarah', 'White', '666778888', '1983-04-25', '303 Maple St', 'F', 65000, '111223333', 3),
('Ryan', 'Hall', '777889999', '1981-08-18', '404 Birch St', 'M', 70000, '333445555', 3),
('Amanda', 'Clark', '888990000', '1979-01-05', '505 Walnut St', 'F', 58000, '333445555', 3),
('Christopher', 'Adams', '999001111', '1984-12-12', '606 Cherry St', 'M', 62000, '333445555', 2),
('Megan', 'Moore', '123456789', '1986-06-28', '707 Sycamore St', 'F', 54000, '222334444', 2);
update EMPLOYEE set Fname='Sayan', Lname='Das', Address='777 XYZ' where Essn = '111223333';

INSERT INTO EMPLOYEE (Fname, Lname, Essn, Bdate, Address, Sex, Salary, Mgr_ssn, Dno)
VALUES
('sanjib', 'das', '456987333', '1980-03-25', '789 centre more', 'M', '80000', '888665555', '2');
select * from EMPLOYEE;

INSERT INTO DEPARTMENT (Dname, Dnumber, Mgr_ssn, Mgr_start_date)
VALUES
('HR', 1, '111223333', '2000-01-01'),
('IT', 2, '222334444', '2001-02-01'),
('Finance', 3, '333445555', '2002-03-01');

INSERT INTO DEPT_LOCATION (Dnumber, Dlocation)
VALUES
(1, 'New York'),
(2, 'San Francisco'),
(3, 'Chicago');
update DEPT_LOCATION set Dlocation = 'Chennai' where Dnumber = '3';
INSERT INTO PROJECT (Pname, Pnumber, Plocation, Dnumber)
VALUES
('ProjectA', 1, 'New York', 1),
('ProjectB', 2, 'San Francisco', 2),
('ProjectC', 3, 'Chicago', 3);
update PROJECT set Plocation = 'Mumbai' where Pnumber = '1';
update PROJECT set Plocation = 'Bangalore' where Pnumber = '2';
update PROJECT set Plocation = 'Chennai' where Pnumber = '3';

INSERT INTO WORKS_ON (Essn, Pnumber, Hours)
VALUES
('111223333', 1, 40),
('222334444', 2, 35),
('333445555', 3, 42),
('444556666', 1, 38),
('555667777', 2, 37),
('666778888', 3, 41),
('777889999', 1, 39),
('888990000', 2, 36),
('999001111', 3, 40),
('123456789', 1, 37);

INSERT INTO DEPENDENT (Essn, Dependent_name, Sex, Bdate, Relationship)
VALUES
('111223333', 'Child1', 'F', '2005-03-10', 'Daughter'),
('444556666', 'Child2', 'M', '2008-08-15', 'Son'),
('222334444', 'Child3', 'F', '2006-05-20', 'Daughter'),
('333445555', 'Child4', 'M', '2007-12-25', 'Son'),
('555667777', 'Spouse', 'F', '1982-04-18', 'Wife'),
('666778888', 'Child5', 'M', '2009-11-05', 'Son'),
('777889999', 'Child6', 'F', '2010-09-02', 'Daughter'),
('888990000', 'Child7', 'M', '2012-07-12', 'Son'),
('999001111', 'Child8', 'F', '2014-01-30', 'Daughter'),
('123456789', 'Child9', 'M', '2016-06-14', 'Son');
```

a)

```
SELECT E.Fname, E.Lname
FROM EMPLOYEE E
```

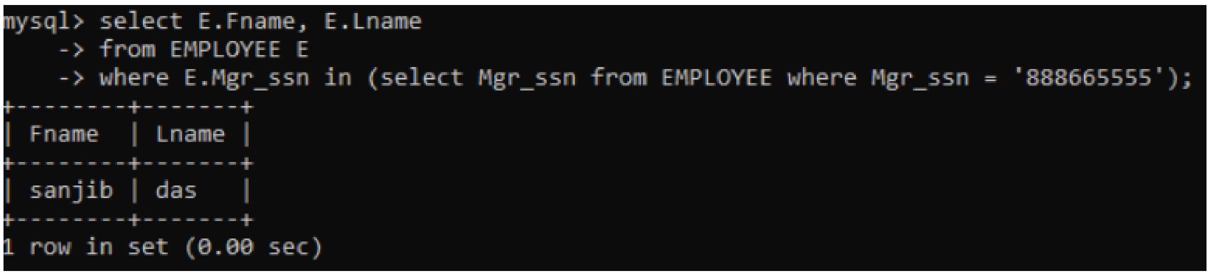
```
WHERE E.Dno = (
   SELECT D.Dnumber
   FROM DEPARTMENT D
   WHERE D.Mgr_ssn = (
      SELECT Essn
      FROM EMPLOYEE
      WHERE Salary = (SELECT MAX(Salary) FROM EMPLOYEE)
   )
);
```



```
mysql> select E.Fname, E.Lname from EMPLOYEE E
    -> where E.Dno = (select Dno from EMPLOYEE order by Salary desc limit 1);
+----------+-------+
| Fname    | Lname |
+----------+-------+
| Sayantan | Ghati |
| Dipak    | Das   |
| Biplab   | Das   |
| sanjib   | das   |
| Utpal    | Roy   |
+----------+-------+
5 rows in set (0.09 sec)
```

B)

```
SELECT E.Fname, E.Lname
FROM EMPLOYEE E
WHERE E.Mgr_ssn IN (
   SELECT E1.Essn
   FROM EMPLOYEE E1
   WHERE E1.Mgr_ssn = '888665555'
);
```



```
mysql> select E.Fname, E.Lname
    -> from EMPLOYEE E
    -> where E.Mgr_ssn in (select Mgr_ssn from EMPLOYEE where Mgr_ssn = '888665555');
+--------+-------+
| Fname  | Lname |
+--------+-------+
| sanjib | das   |
+--------+-------+
1 row in set (0.00 sec)
```

C)

SELECT Fname, Lname

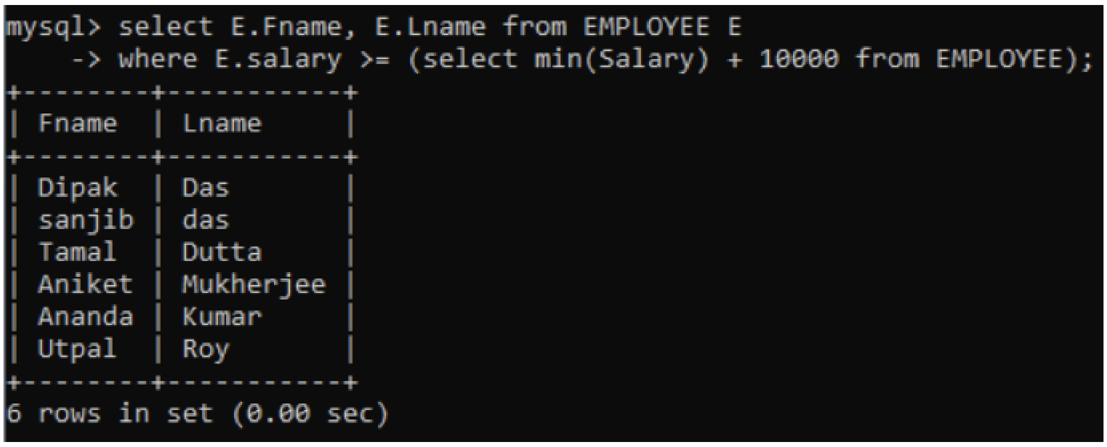
FROM EMPLOYEE

WHERE Salary >= (

  SELECT MIN(Salary) + 10000

  FROM EMPLOYEE

);



```
mysql> select E.Fname, E.Lname from EMPLOYEE E
    -> where E.salary >= (select min(Salary) + 10000 from EMPLOYEE);
+--------+-----------+
| Fname  | Lname     |
+--------+-----------+
| Dipak  | Das       |
| sanjib | das       |
| Tamal  | Dutta     |
| Aniket | Mukherjee |
| Ananda | Kumar     |
| Utpal  | Roy       |
+--------+-----------+
6 rows in set (0.00 sec)
```

D)

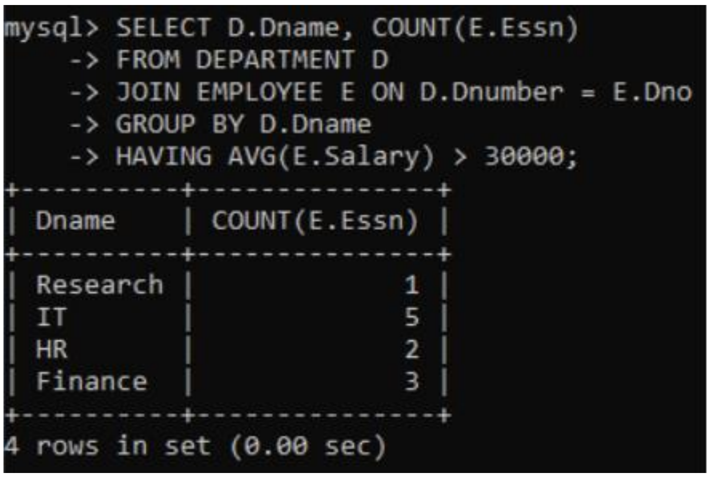
SELECT D.Dname, COUNT(E.Essn) AS NumEmployees

FROM DEPARTMENT D

JOIN EMPLOYEE E ON D.Dnumber = E.Dno

GROUP BY D.Dname

HAVING AVG(E.Salary) > 30000;



```
mysql> SELECT D.Dname, COUNT(E.Essn)
    -> FROM DEPARTMENT D
    -> JOIN EMPLOYEE E ON D.Dnumber = E.Dno
    -> GROUP BY D.Dname
    -> HAVING AVG(E.Salary) > 30000;
+----------+---------------+
| Dname    | COUNT(E.Essn) |
+----------+---------------+
| Research |             1 |
| IT       |             5 |
| HR       |             2 |
| Finance  |             3 |
+----------+---------------+
4 rows in set (0.00 sec)
```

E)

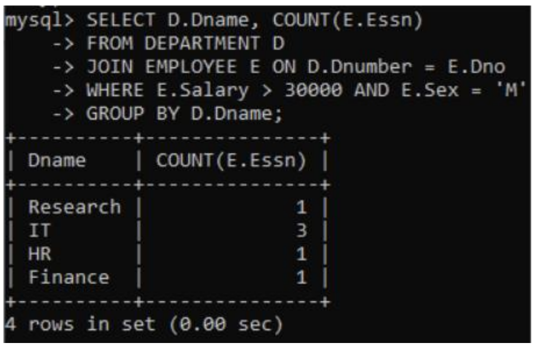
SELECT D.Dname, COUNT(E.Essn) AS NumMaleEmployees

FROM DEPARTMENT D

JOIN EMPLOYEE E ON D.Dnumber = E.Dno

WHERE E.Sex = 'M' AND E.Salary > 30000

GROUP BY D.Dname;



```
mysql> SELECT D.Dname, COUNT(E.Essn)
    -> FROM DEPARTMENT D
    -> JOIN EMPLOYEE E ON D.Dnumber = E.Dno
    -> WHERE E.Salary > 30000 AND E.Sex = 'M'
    -> GROUP BY D.Dname;
+----------+--------------+
| Dname    | COUNT(E.Essn) |
+----------+--------------+
| Research |            1 |
| IT       |            3 |
| HR       |            1 |
| Finance  |            1 |
+----------+--------------+
4 rows in set (0.00 sec)
```

F)

SELECT D.Dnumber, COUNT(E.Essn) AS NumEmployees

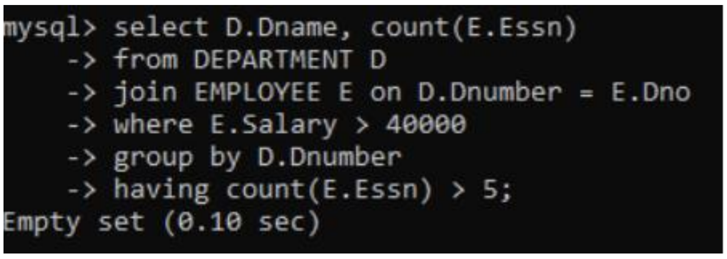
FROM DEPARTMENT D

JOIN EMPLOYEE E ON D.Dnumber = E.Dno

WHERE E.Salary > 40000

GROUP BY D.Dnumber

HAVING COUNT(E.Essn) > 5;

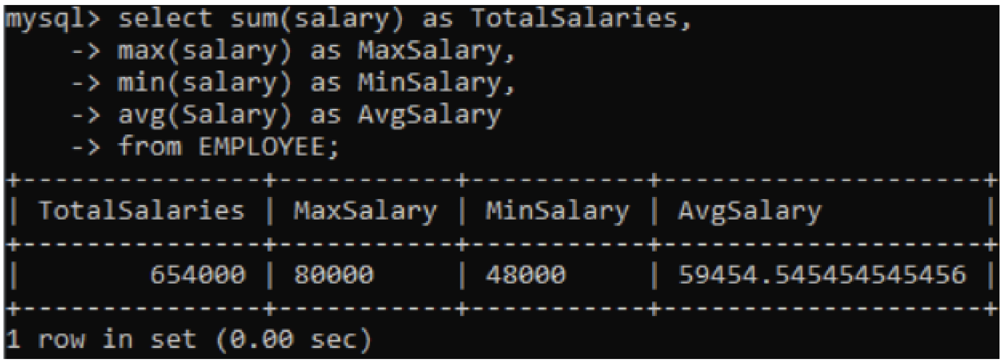


```
mysql> select D.Dname, count(E.Essn)
    -> from DEPARTMENT D
    -> join EMPLOYEE E on D.Dnumber = E.Dno
    -> where E.Salary > 40000
    -> group by D.Dnumber
    -> having count(E.Essn) > 5;
Empty set (0.10 sec)
```

G)

SELECT SUM(Salary) AS TotalSalary, MAX(Salary) AS MaxSalary, MIN(Salary) AS MinSalary, AVG(Salary) AS AvgSalary

FROM EMPLOYEE;



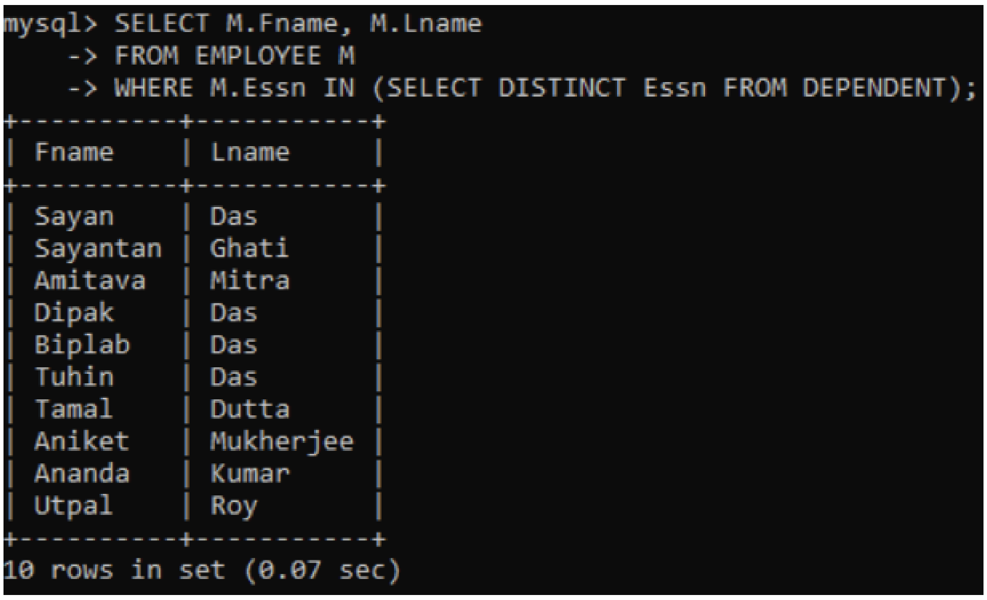
```
mysql> select sum(salary) as TotalSalaries,
    -> max(salary) as MaxSalary,
    -> min(salary) as MinSalary,
    -> avg(Salary) as AvgSalary
    -> from EMPLOYEE;
+---------------+-----------+-----------+--------------------+
| TotalSalaries | MaxSalary | MinSalary | AvgSalary          |
+---------------+-----------+-----------+--------------------+
|        654000 | 80000     | 48000     | 59454.545454545456 |
+---------------+-----------+-----------+--------------------+
1 row in set (0.00 sec)
```

H)

SELECT DISTINCT M.Fname, M.Lname

FROM EMPLOYEE M

JOIN DEPENDENT D ON M.Essn = D.Essn;



```
mysql> SELECT M.Fname, M.Lname
    -> FROM EMPLOYEE M
    -> WHERE M.Essn IN (SELECT DISTINCT Essn FROM DEPENDENT);
+----------+-----------+
| Fname    | Lname     |
+----------+-----------+
| Sayan    | Das       |
| Sayantan | Ghati     |
| Amitava  | Mitra     |
| Dipak    | Das       |
| Biplab   | Das       |
| Tuhin    | Das       |
| Tamal    | Dutta     |
| Aniket   | Mukherjee |
| Ananda   | Kumar     |
| Utpal    | Roy       |
+----------+-----------+
10 rows in set (0.07 sec)
```

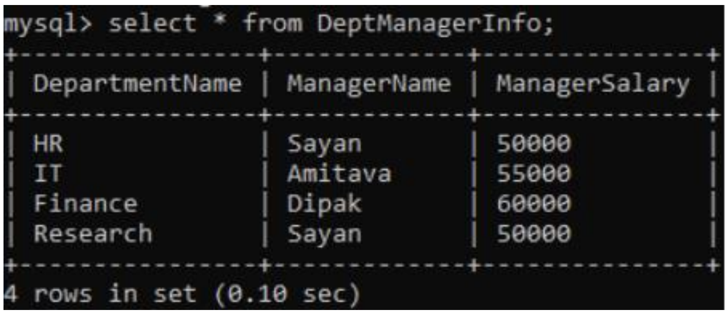
5.A)

CREATE VIEW DepartmentManagers AS

SELECT D.Dname AS DepartmentName, M.Fname || ' ' || M.Lname AS ManagerName, M.Salary AS ManagerSalary

FROM DEPARTMENT D

INNER JOIN EMPLOYEE M ON D.Mgr_ssn = M.Essn;



```
mysql> select * from DeptManagerInfo;
+----------------+-------------+---------------+
| DepartmentName | ManagerName | ManagerSalary |
+----------------+-------------+---------------+
| HR             | Sayan       | 50000         |
| IT             | Amitava     | 55000         |
| Finance        | Dipak       | 60000         |
| Research       | Sayan       | 50000         |
+----------------+-------------+---------------+
4 rows in set (0.10 sec)
```

B)

CREATE VIEW ResearchEmployeeInfo AS

SELECT E.Fname || ' ' || E.Lname AS EmployeeName,

S.Fname || ' ' || S.Lname AS SupervisorName,

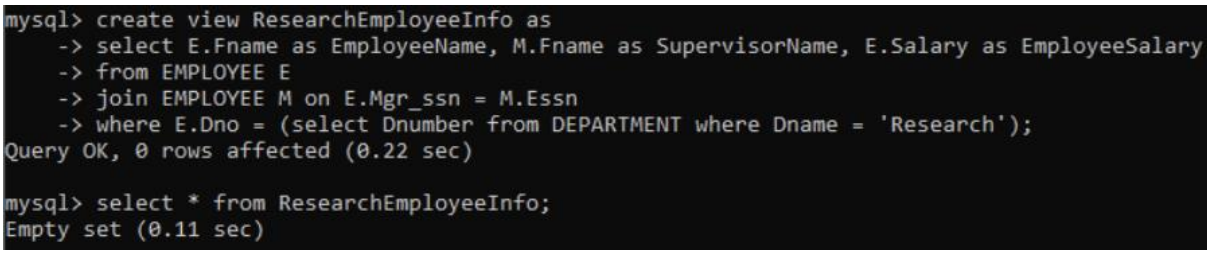
E.Salary AS EmployeeSalary

FROM EMPLOYEE E

INNER JOIN EMPLOYEE S ON E.Mgr_ssn = S.Essn

INNER JOIN DEPARTMENT D ON E.Dno = D.Dnumber

WHERE D.Dname = 'Research';



```
mysql> create view ResearchEmployeeInfo as
    -> select E.Fname as EmployeeName, M.Fname as SupervisorName, E.Salary as EmployeeSalary
    -> from EMPLOYEE E
    -> join EMPLOYEE M on E.Mgr_ssn = M.Essn
    -> where E.Dno = (select Dnumber from DEPARTMENT where Dname = 'Research');
Query OK, 0 rows affected (0.22 sec)

mysql> select * from ResearchEmployeeInfo;
Empty set (0.11 sec)
```

C)

CREATE VIEW ProjectInfo AS

SELECT P.Pname AS ProjectName, D.Dname AS ControllingDepartment,

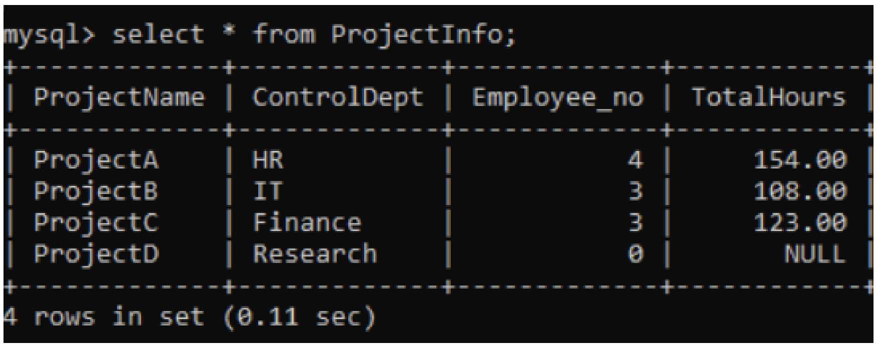
COUNT(W.Essn) AS NumberOfEmployees, SUM(W.Hours) AS TotalHoursPerWeek

FROM PROJECT P

INNER JOIN DEPARTMENT D ON P.Dnumber = D.Dnumber

LEFT JOIN WORKS_ON W ON P.Pnumber = W.Pnumber

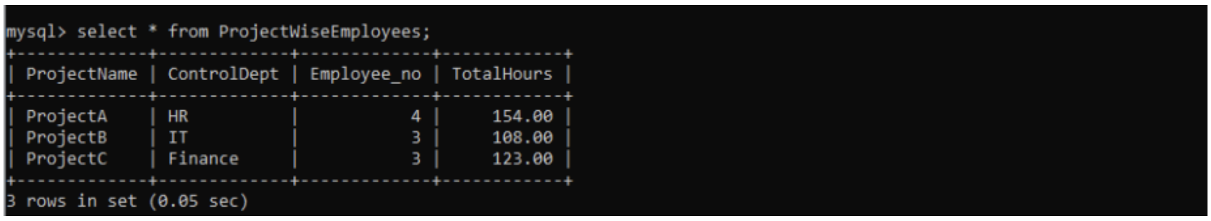
GROUP BY P.Pname, D.Dname;



```
mysql> select * from ProjectInfo;
+-------------+------------+-------------+------------+
| ProjectName | ControlDept | Employee_no | TotalHours |
+-------------+------------+-------------+------------+
| ProjectA    | HR         |           4 |     154.00 |
| ProjectB    | IT         |           3 |     108.00 |
| ProjectC    | Finance    |           3 |     123.00 |
| ProjectD    | Research   |           0 |       NULL |
+-------------+------------+-------------+------------+
4 rows in set (0.11 sec)
```

D)

```
CREATE VIEW ProjectsWithManyEmployees AS

SELECT P.Pname AS ProjectName, D.Dname AS ControllingDepartment,

    COUNT(W.Essn) AS NumberOfEmployees, SUM(W.Hours) AS TotalHoursPerWeek

FROM PROJECT P

INNER JOIN DEPARTMENT D ON P.Dnumber = D.Dnumber

INNER JOIN WORKS_ON W ON P.Pnumber = W.Pnumber

GROUP BY P.Pname, D.Dname

HAVING COUNT(W.Essn) > 1;
```



```
mysql> select * from ProjectWiseEmployees;
+-------------+------------+-------------+------------+
| ProjectName | ControlDept | Employee_no | TotalHours |
+-------------+------------+-------------+------------+
| ProjectA    | HR         |           4 |     154.00 |
| ProjectB    | IT         |           3 |     108.00 |
| ProjectC    | Finance    |           3 |     123.00 |
+-------------+------------+-------------+------------+
3 rows in set (0.05 sec)
```