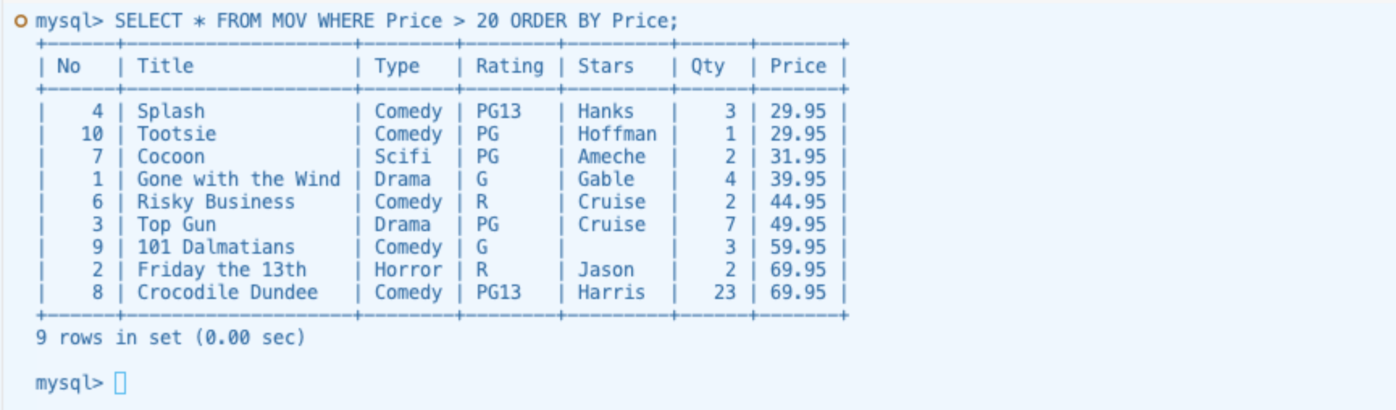
PRACTICAL EXERCISE-21

Q: Write SQL commands for the following on the basis of given table MOV:

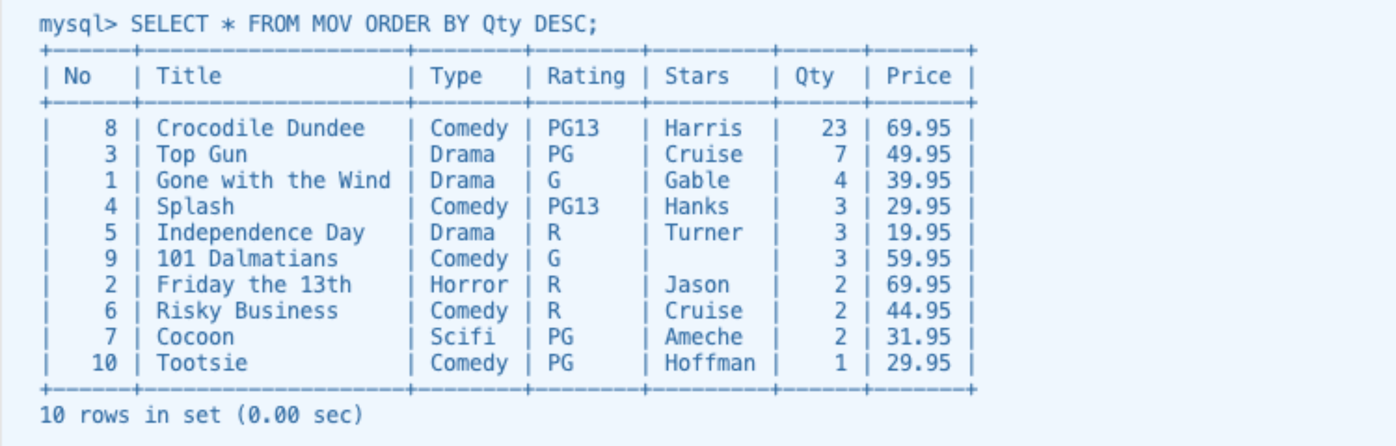
No	Title	Type	Rating	Stars	Qty	Price
1	Gone with the Wind	Drama	G	Gable	4	39.95
2	Friday the 13th	Horror	R	Jason	2	69.95
3	Top Gun	Drama	PG	Cruise	7	49.95
4	Splash	Comedy	PG13	Hanks	3	29.95
5	Independence Day	Drama	R	Turner	3	19.95
6	Risky Business	Comedy	R	Cruise	2	44.95
7	Cocoon	Scifi	PG	Ameche	2	31.95
8	Crocodile Dundee	Comedy	PG13	Harris	23	69.95
9	101 Dalmatians	Comedy	G		3	59.95
10	Tootsie	Comedy	PG	Hoffman	1	29.95

(a) Display a list of all movies with Price over 20 and sorted by Price.



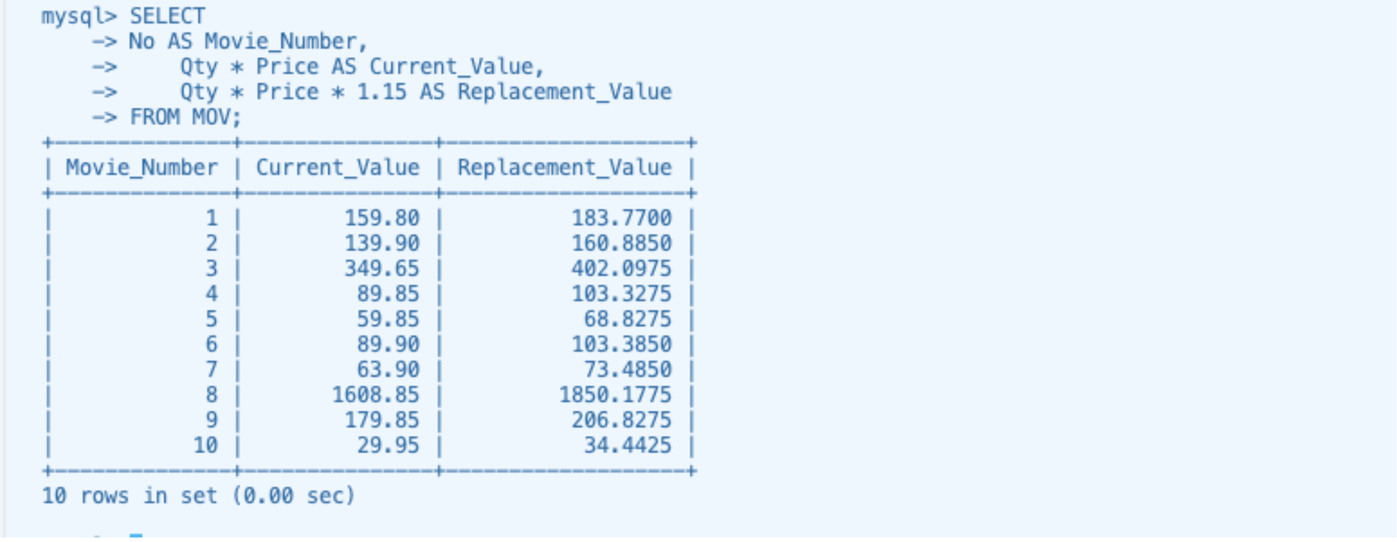
```
mysql> SELECT * FROM MOV WHERE Price > 20 ORDER BY Price;
+----+--------------------+--------+--------+---------+-----+-------+
| No | Title              | Type   | Rating | Stars   | Qty | Price |
+----+--------------------+--------+--------+---------+-----+-------+
|  4 | Splash             | Comedy | PG13   | Hanks   |   3 | 29.95 |
| 10 | Tootsie            | Comedy | PG     | Hoffman |   1 | 29.95 |
|  7 | Cocoon             | Scifi  | PG     | Ameche  |   2 | 31.95 |
|  1 | Gone with the Wind | Drama  | G      | Gable   |   4 | 39.95 |
|  6 | Risky Business     | Comedy | R      | Cruise  |   2 | 44.95 |
|  3 | Top Gun            | Drama  | PG     | Cruise  |   7 | 49.95 |
|  9 | 101 Dalmatians     | Comedy | G      |         |   3 | 59.95 |
|  2 | Friday the 13th    | Horror | R      | Jason   |   2 | 69.95 |
|  8 | Crocodile Dundee   | Comedy | PG13   | Harris  |  23 | 69.95 |
+----+--------------------+--------+--------+---------+-----+-------+
9 rows in set (0.00 sec)

mysql>
```

(b) Display all the movies sorted by QTY in decreasing order.



```
mysql> SELECT * FROM MOV ORDER BY Qty DESC;
+----+--------------------+--------+--------+---------+-----+-------+
| No | Title              | Type   | Rating | Stars   | Qty | Price |
+----+--------------------+--------+--------+---------+-----+-------+
|  8 | Crocodile Dundee   | Comedy | PG13   | Harris  |  23 | 69.95 |
|  3 | Top Gun            | Drama  | PG     | Cruise  |   7 | 49.95 |
|  1 | Gone with the Wind | Drama  | G      | Gable   |   4 | 39.95 |
|  4 | Splash             | Comedy | PG13   | Hanks   |   3 | 29.95 |
|  5 | Independence Day   | Drama  | R      | Turner  |   3 | 19.95 |
|  9 | 101 Dalmatians     | Comedy | G      |         |   3 | 59.95 |
|  2 | Friday the 13th    | Horror | R      | Jason   |   2 | 69.95 |
|  6 | Risky Business     | Comedy | R      | Cruise  |   2 | 44.95 |
|  7 | Cocoon             | Scifi  | PG     | Ameche  |   2 | 31.95 |
| 10 | Tootsie            | Comedy | PG     | Hoffman |   1 | 29.95 |
+----+--------------------+--------+--------+---------+-----+-------+
10 rows in set (0.00 sec)
```

(c) Display a report listing a movie number, current value and replacement value for each movie in the above table. Calculate the replacement value for all movies as: QTY Price 1.15.



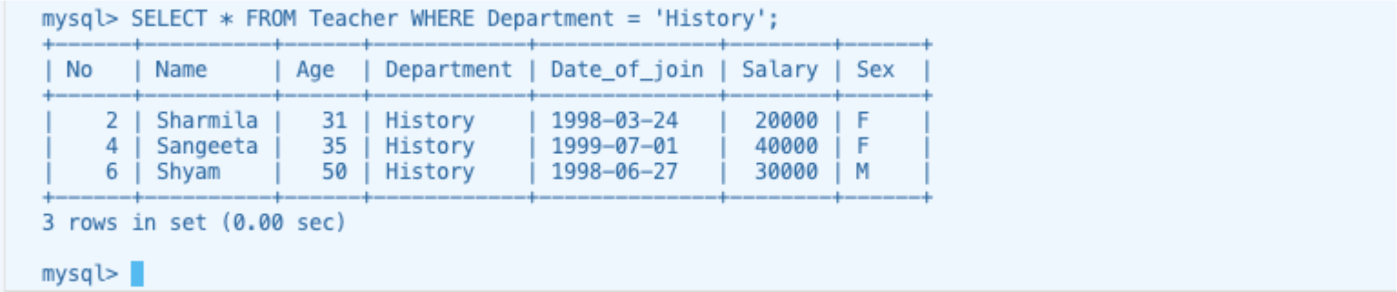
```
mysql> SELECT
    -> No AS Movie_Number,
    ->     Qty * Price AS Current_Value,
    ->     Qty * Price * 1.15 AS Replacement_Value
    -> FROM MOV;
+--------------+---------------+-------------------+
| Movie_Number | Current_Value | Replacement_Value |
+--------------+---------------+-------------------+
|            1 |        159.80 |          183.7700 |
|            2 |        139.90 |          160.8850 |
|            3 |        349.65 |          402.0975 |
|            4 |         89.85 |          103.3275 |
|            5 |         59.85 |           68.8275 |
|            6 |         89.90 |          103.3850 |
|            7 |         63.90 |           73.4850 |
|            8 |       1608.85 |         1850.1775 |
|            9 |        179.85 |          206.8275 |
|           10 |         29.95 |           34.4425 |
+--------------+---------------+-------------------+
10 rows in set (0.00 sec)
```

PRACTICAL EXERCISE-22

Q: Write SQL commands for the following on the basis of given table Teacher:

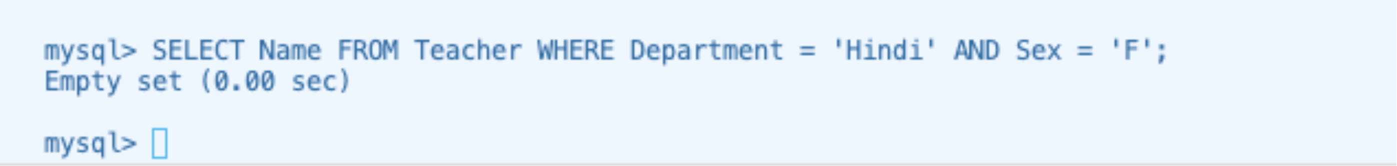
No.	Name	Age	Department	Date of join	Salary	Sex
1	Jugal	34	Computer	10/01/97	12000	M
2 .	Sharmila	31	History	24/03/98	20000	F
3 .	Sandeep	32	Maths	12/12/96	30000	M
4 .	Sangeeta	35	History	01/07/99	40000	F
5 .	Rakesh	42	Maths	05/09/97	25000	M
6 .	Shyam	50	History	27/06/98	30000	M
7 .	Shiv Om	44	Computer	25/02/97	21000	M
8 .	Shalakha	33	Maths	31/07/97	20000	F

(a) To show all information about the teacher of history department



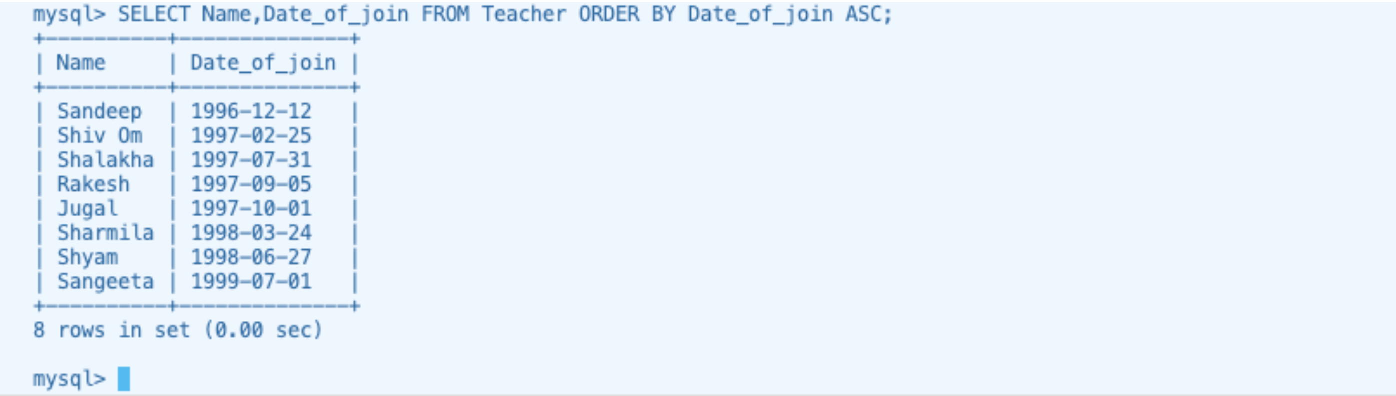
```
mysql> SELECT * FROM Teacher WHERE Department = 'History';
+------+----------+------+------------+--------------+--------+------+
| No   | Name     | Age  | Department | Date_of_join | Salary | Sex  |
+------+----------+------+------------+--------------+--------+------+
|    2 | Sharmila |   31 | History    | 1998-03-24   |  20000 | F    |
|    4 | Sangeeta |   35 | History    | 1999-07-01   |  40000 | F    |
|    6 | Shyam    |   50 | History    | 1998-06-27   |  30000 | M    |
+------+----------+------+------------+--------------+--------+------+
3 rows in set (0.00 sec)

mysql>
```

(b) To list the names of female teachers who are in Hindi department



```
mysql> SELECT Name FROM Teacher WHERE Department = 'Hindi' AND Sex = 'F';
Empty set (0.00 sec)

mysql>
```

(c) To list names of all teachers with their date of joining in ascending order.



```
mysql> SELECT Name,Date_of_join FROM Teacher ORDER BY Date_of_join ASC;
+----------+--------------+
| Name     | Date_of_join |
+----------+--------------+
| Sandeep  | 1996-12-12   |
| Shiv Om  | 1997-02-25   |
| Shalakha | 1997-07-31   |
| Rakesh   | 1997-09-05   |
| Jugal    | 1997-10-01   |
| Sharmila | 1998-03-24   |
| Shyam    | 1998-06-27   |
| Sangeeta | 1999-07-01   |
+----------+--------------+
8 rows in set (0.00 sec)

mysql> 
```

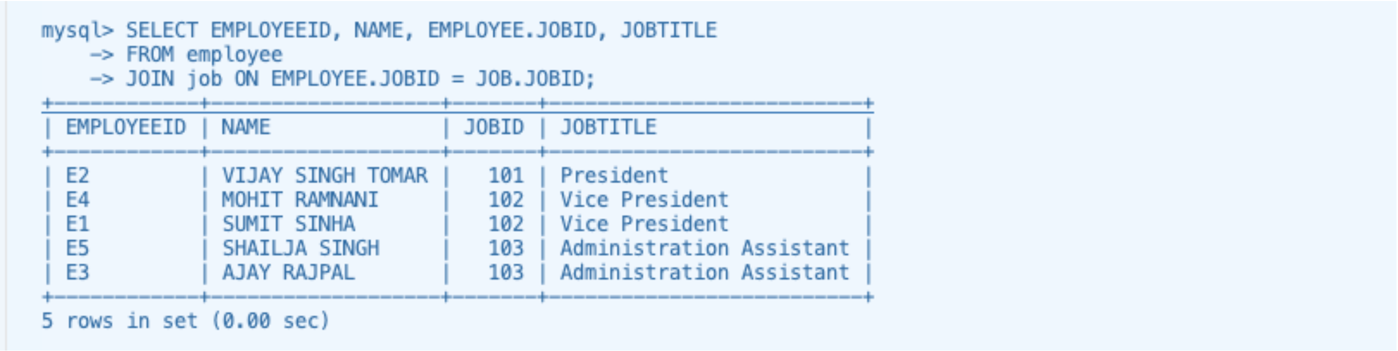
PRACTICAL EXERCISE-23

Q: Write the SQL queries for the given two tables job and employee:

JOBID	JOBTITLE	SALARY
101	President	200000
102	Vice President	125000
103	Administration Assistant	80000
104	Accounting Manager	70000
105	Accountant	65000
106	Sales Manager	80000

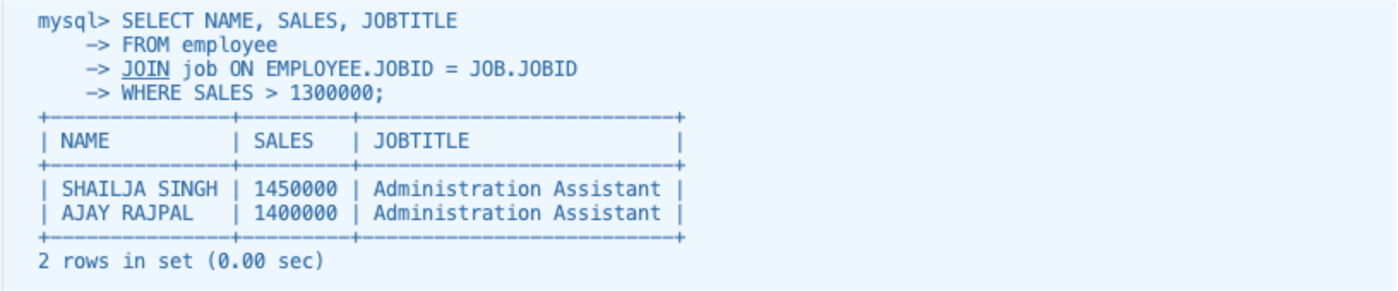
EMPLOYEEID	NAME	SALES	JOBID
E1	SUMIT SINHA	1100000	102
E2	VIJAY SINGH TOMAR	1300000	101
E3	AJAY RAJPAL	1400000	103
E4	MOHIT RAMNANI	1250000	102
E5	SHAILJA SINGH	1450000	103

(i) To display employee ids, names of employees, job ids with corresponding job titles.



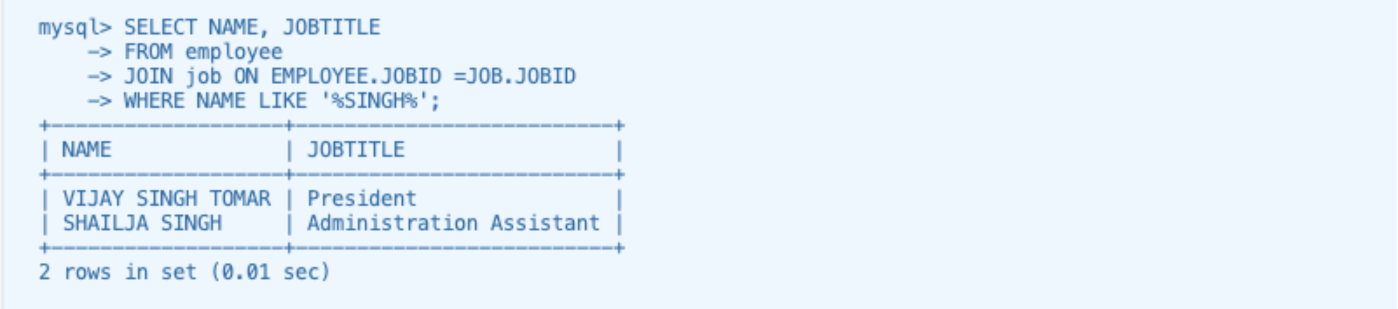
```
mysql> SELECT EMPLOYEEID, NAME, EMPLOYEE.JOBID, JOBTITLE
    -> FROM employee
    -> JOIN job ON EMPLOYEE.JOBID = JOB.JOBID;
+------------+-------------------+-------+--------------------------+
| EMPLOYEEID | NAME              | JOBID | JOBTITLE                 |
+------------+-------------------+-------+--------------------------+
| E2         | VIJAY SINGH TOMAR |   101 | President                |
| E4         | MOHIT RAMNANI     |   102 | Vice President           |
| E1         | SUMIT SINHA       |   102 | Vice President           |
| E5         | SHAILJA SINGH     |   103 | Administration Assistant |
| E3         | AJAY RAJPAL       |   103 | Administration Assistant |
+------------+-------------------+-------+--------------------------+
5 rows in set (0.00 sec)
```

(ii) To display names of employees, sales and corresponding job titles who have achieved sales more than 1300000.



```
mysql> SELECT NAME, SALES, JOBTITLE
    -> FROM employee
    -> JOIN job ON EMPLOYEE.JOBID = JOB.JOBID
    -> WHERE SALES > 1300000;
+---------------+---------+--------------------------+
| NAME          | SALES   | JOBTITLE                 |
+---------------+---------+--------------------------+
| SHAILJA SINGH | 1450000 | Administration Assistant |
| AJAY RAJPAL   | 1400000 | Administration Assistant |
+---------------+---------+--------------------------+
2 rows in set (0.00 sec)
```

(iii) To display names and corresponding job titles of those employees who have 'SINGH' (anywhere) in their names.

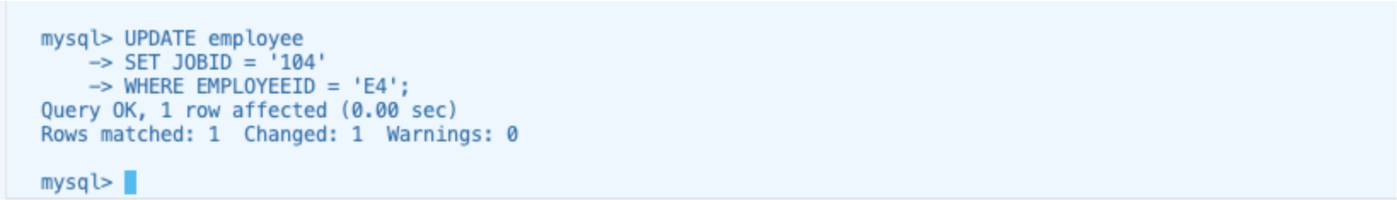


```
mysql> SELECT NAME, JOBTITLE
    -> FROM employee
    -> JOIN job ON EMPLOYEE.JOBID =JOB.JOBID
    -> WHERE NAME LIKE '%SINGH%';
+-------------------+--------------------------+
| NAME              | JOBTITLE                 |
+-------------------+--------------------------+
| VIJAY SINGH TOMAR | President                |
| SHAILJA SINGH     | Administration Assistant |
+-------------------+--------------------------+
2 rows in set (0.01 sec)
```

(iv) Identify foreign key in the table EMPLOYEE.

The foreign key in the table 'EMPLOYEE' is the column **JOBID** as it references the **JOBID** column in the 'job' table.

(e) Write SQL. command to change the JOBID to 104 of the EMPLOYEE with ID as E4 in the table 'EMPLOYEE’



```
mysql> UPDATE employee
    -> SET JOBID = '104'
    -> WHERE EMPLOYEEID = 'E4';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> 
```

PRACTICAL EXERCISE-24

Q: Consider the following tables Employee and Salary, Write SQL commands for the statements (i) to (iv) and give outputs for SQL. queries (v) to (vii)

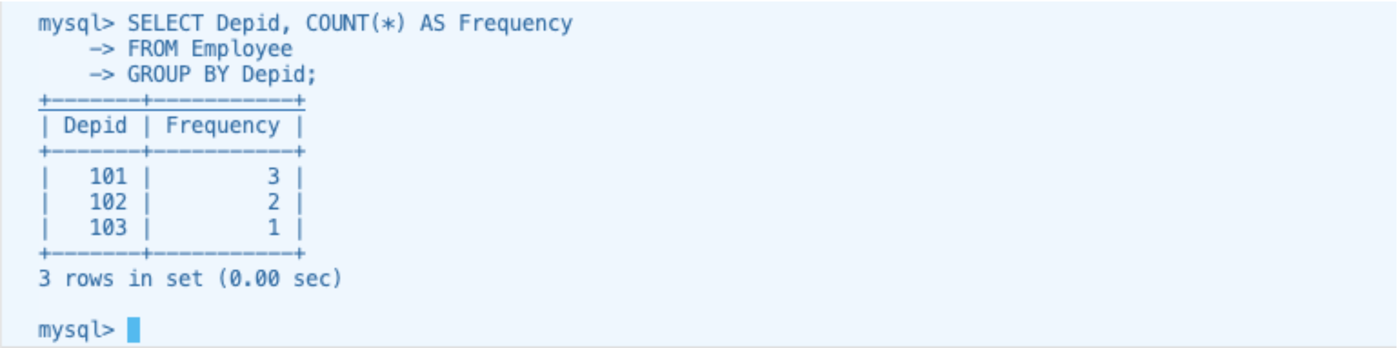
Table: Employee

Eid	Name	Depid	Qualification	Sec
1	Deepali Gupta	101	MCA	F
2	Rajat Tyagi	101	BCA	M
3	Hari Mohan	102	B.A.	M
4	Harry	102	M.A.	M
5	Sumit Mittal	103	B.Tech .	M
6	Jyoti	101	M.Tech .	F

Table: Salary

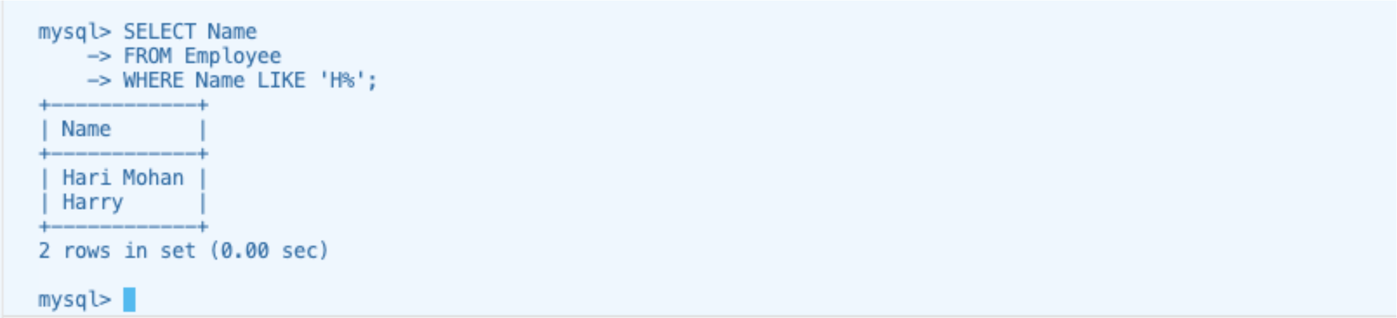
Eid	Basic	D.A.	HRA	Bonus
1	6000	2000	2300	200
2	2000	300	300	30
3	1000	300	300	40
4	1500	390	490	30
5	8000	900	900	80
6	10000	300	490	89

(i) To display the frequency of employees department wise.



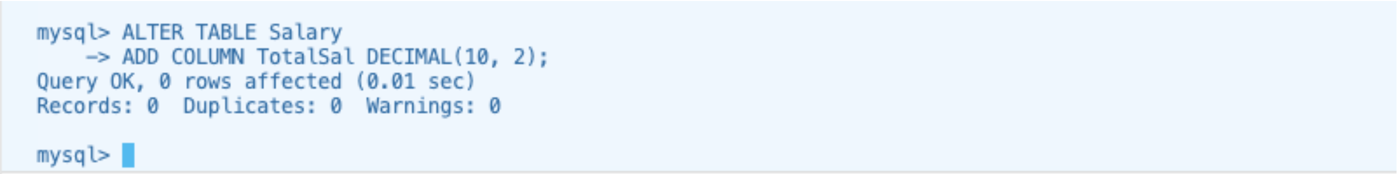
```
mysql> SELECT Depid, COUNT(*) AS Frequency
    -> FROM Employee
    -> GROUP BY Depid;
+-------+-----------+
| Depid | Frequency |
+-------+-----------+
|   101 |         3 |
|   102 |         2 |
|   103 |         1 |
+-------+-----------+
3 rows in set (0.00 sec)

mysql> 
```

(ii) To list the names of those employees only whose name starts with 'H'



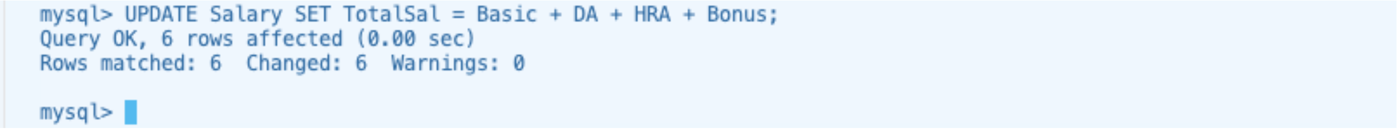
```
mysql> SELECT Name
    -> FROM Employee
    -> WHERE Name LIKE 'H%';
+------------+
| Name       |
+------------+
| Hari Mohan |
| Harry      |
+------------+
2 rows in set (0.00 sec)

mysql> 
```

(iii) To add a new column in salary table. The column name is Total Sal.



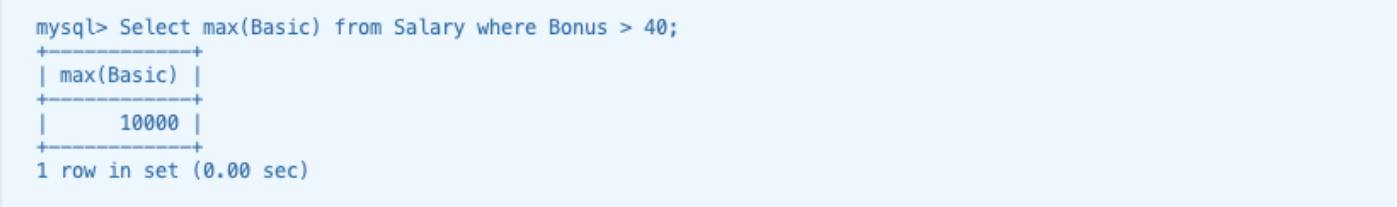
```
mysql> ALTER TABLE Salary
    -> ADD COLUMN TotalSal DECIMAL(10, 2);
Query OK, 0 rows affected (0.01 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> 
```

(iv) To store the corresponding values in the Total Sal column.



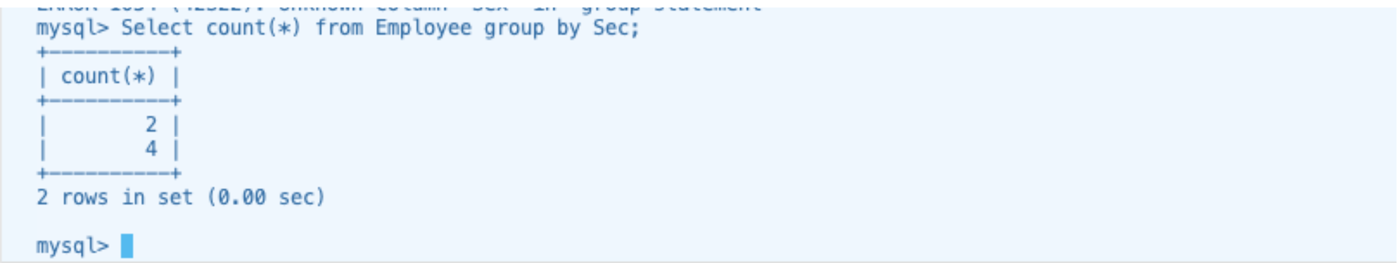
```
mysql> UPDATE Salary SET TotalSal = Basic + DA + HRA + Bonus;
Query OK, 6 rows affected (0.00 sec)
Rows matched: 6  Changed: 6  Warnings: 0

mysql> 
```

(v) Select max(Basic) from Salary where Bonus > 40:



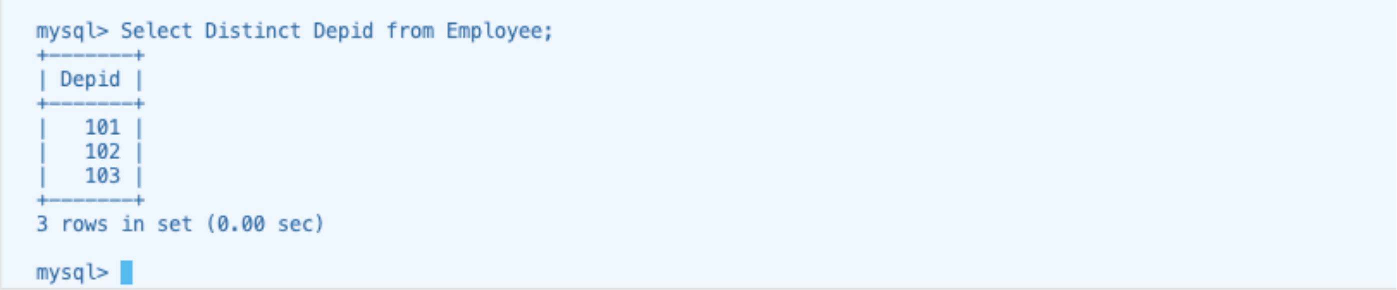
```
mysql> Select max(Basic) from Salary where Bonus > 40;
+------------+
| max(Basic) |
+------------+
|      10000 |
+------------+
1 row in set (0.00 sec)
```

(vi) Select count(*) from Employee group by Sex;



```
ERROR 1054 (42S22): Unknown column 'Sex' in 'group statement'
mysql> Select count(*) from Employee group by Sec;
+----------+
| count(*) |
+----------+
|        2 |
|        4 |
+----------+
2 rows in set (0.00 sec)

mysql> 
```

(vii) Select Distinct Depid from Employee:



```
mysql> Select Distinct Depid from Employee;
+-------+
| Depid |
+-------+
|   101 |
|   102 |
|   103 |
+-------+
3 rows in set (0.00 sec)

mysql> 
```