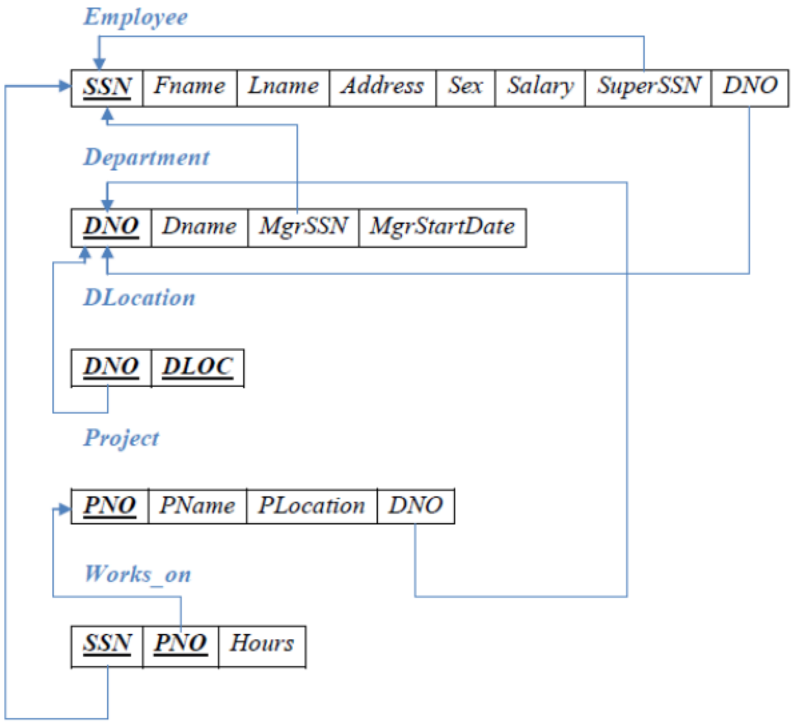
Third Semester

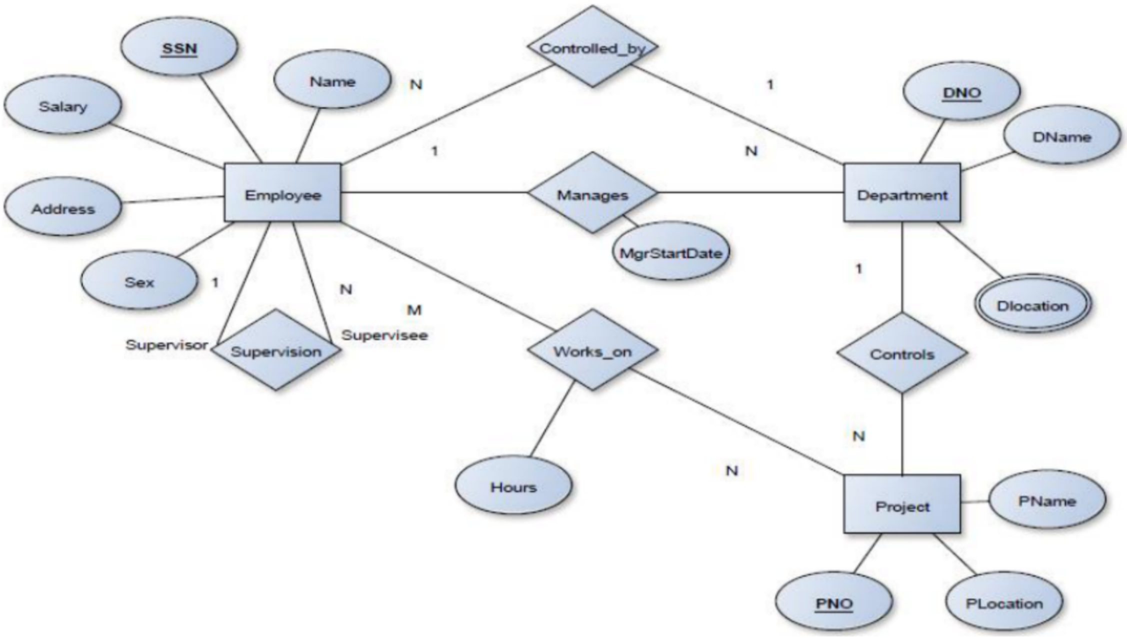
21-805-0307: DATABASE SYSTEMS LAB

Table of Contents

SCHEMA DIAGRAM



ER DIAGRAM



DDL COMMANDS

AIM

Develop SQL Queries to execute and verify the Data Definition Language commands and also implement Data Constraints.

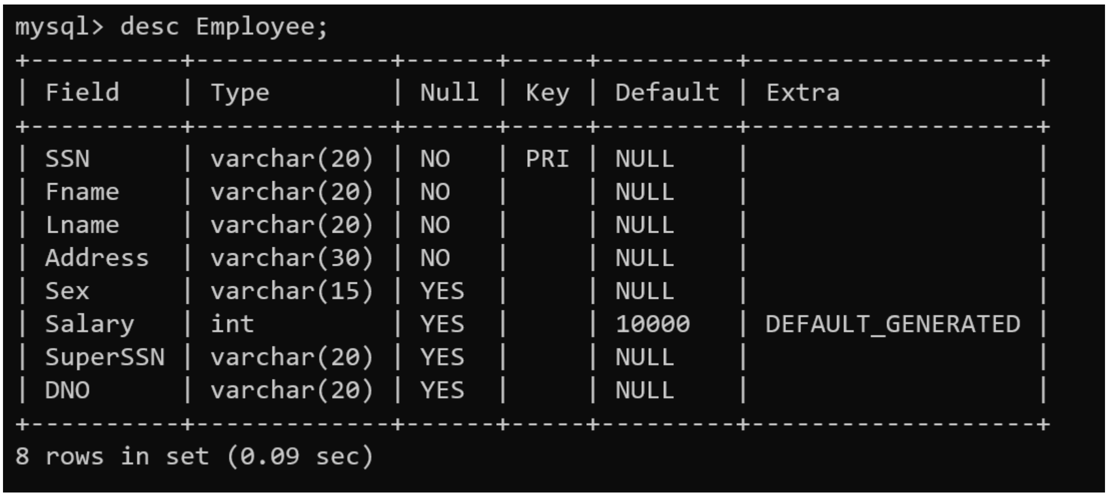
Question : 1

Create five tables using constraints like primary key, not null, check, default, null, unique, foreign key as per the above schema

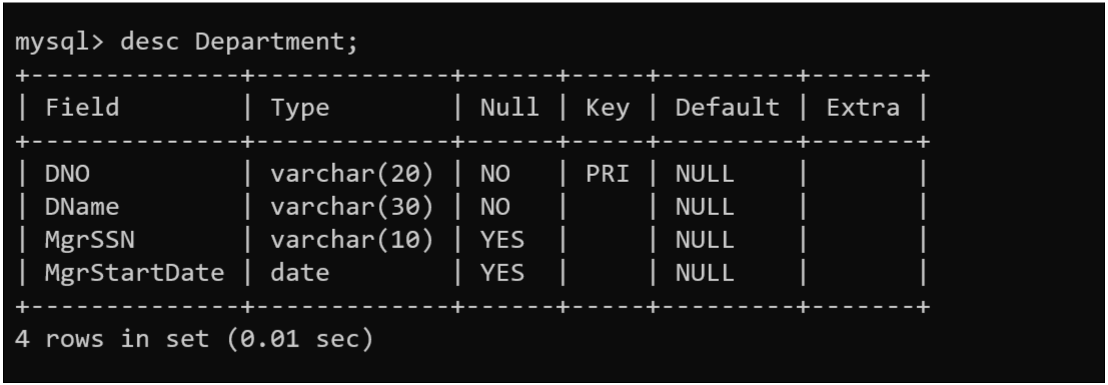
QUERY

```
mysql> create database Company;
Query OK, 1 row affected (0.05 sec)

mysql> use Company;
Database changed

mysql> create table Employee(SSN varchar(20) primary key,Fname varchar(20)
not null,Lname varchar(20) not null,Address varchar(30) not null,Sex varchar(15),
Salary int default(10000) check (Salary>5000),SuperSSN varchar(20),
DNO varchar(20));
Query OK, 0 rows affected (0.11 sec)

mysql> create table Department(DNO varchar(20) primary key,DName varchar(30)
not null,MgrSSN varchar(10), MgrStartDate date);
Query OK, 0 rows affected (0.03 sec)

mysql> create table DLocation(DNO varchar(20) primary key,DLOC varchar(30)
not null);
Query OK, 0 rows affected (0.02 sec)

mysql> create table Project(PNO varchar(20) primary key,PName varchar(30),
Plocation varchar(30), DNO varchar(20));
Query OK, 0 rows affected (0.04 sec)

mysql> create table Works_on(SSN varchar(20) primary key,PNO varchar(20),
Hours int,constraint PNO_Works_on foreign key(PNO) references Project(PNO));
Query OK, 0 rows affected (0.03 sec)
```

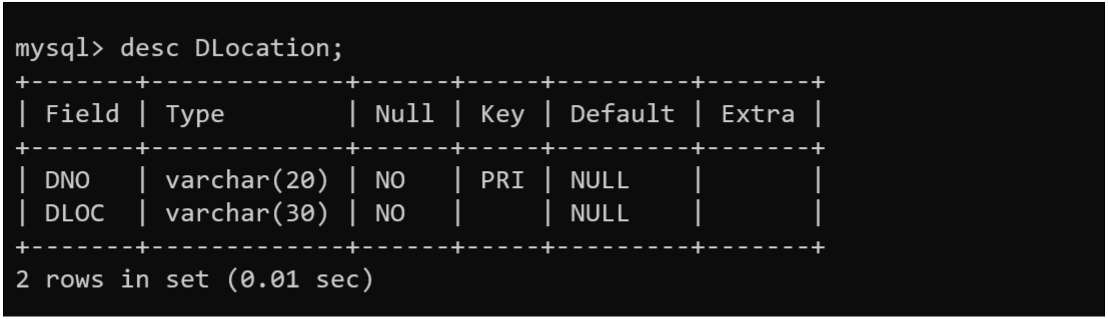
DATABASE TABLES



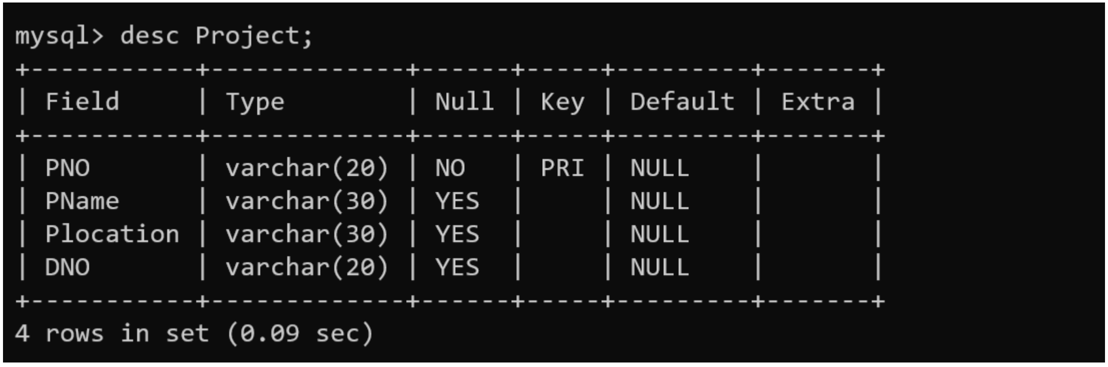
```
mysql> desc Employee;
+----------+-------------+------+-----+---------+-------------------+
| Field    | Type        | Null | Key | Default | Extra             |
+----------+-------------+------+-----+---------+-------------------+
| SSN      | varchar(20) | NO   | PRI | NULL    |                   |
| Fname    | varchar(20) | NO   |     | NULL    |                   |
| Lname    | varchar(20) | NO   |     | NULL    |                   |
| Address  | varchar(30) | NO   |     | NULL    |                   |
| Sex      | varchar(15) | YES  |     | NULL    |                   |
| Salary   | int         | YES  |     | 10000   | DEFAULT_GENERATED |
| SuperSSN | varchar(20) | YES  |     | NULL    |                   |
| DNO      | varchar(20) | YES  |     | NULL    |                   |
+----------+-------------+------+-----+---------+-------------------+
8 rows in set (0.09 sec)
```



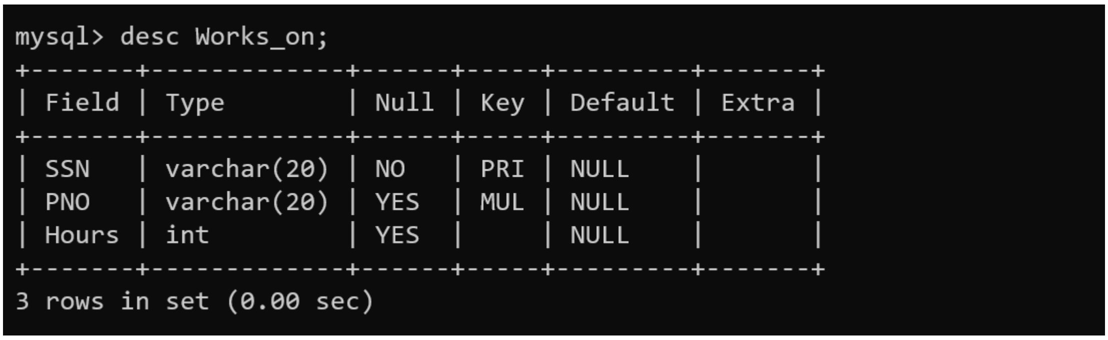
```
mysql> desc Department;
+--------------+-------------+------+-----+---------+-------+
| Field        | Type        | Null | Key | Default | Extra |
+--------------+-------------+------+-----+---------+-------+
| DNO          | varchar(20) | NO   | PRI | NULL    |       |
| DName        | varchar(30) | NO   |     | NULL    |       |
| MgrSSN       | varchar(10) | YES  |     | NULL    |       |
| MgrStartDate | date        | YES  |     | NULL    |       |
+--------------+-------------+------+-----+---------+-------+
4 rows in set (0.01 sec)
```



```
mysql> desc DLocation;
+-------+-------------+------+-----+---------+-------+
| Field | Type        | Null | Key | Default | Extra |
+-------+-------------+------+-----+---------+-------+
| DNO   | varchar(20) | NO   | PRI | NULL    |       |
| DLOC  | varchar(30) | NO   |     | NULL    |       |
+-------+-------------+------+-----+---------+-------+
2 rows in set (0.01 sec)
```



```
mysql> desc Project;
+-----------+-------------+------+-----+---------+-------+
| Field     | Type        | Null | Key | Default | Extra |
+-----------+-------------+------+-----+---------+-------+
| PNO       | varchar(20) | NO   | PRI | NULL    |       |
| PName     | varchar(30) | YES  |     | NULL    |       |
| Plocation | varchar(30) | YES  |     | NULL    |       |
| DNO       | varchar(20) | YES  |     | NULL    |       |
+-----------+-------------+------+-----+---------+-------+
4 rows in set (0.09 sec)
```



```
mysql> desc Works_on;
+-------+-------------+------+-----+---------+-------+
| Field | Type        | Null | Key | Default | Extra |
+-------+-------------+------+-----+---------+-------+
| SSN   | varchar(20) | NO   | PRI | NULL    |       |
| PNO   | varchar(20) | YES  | MUL | NULL    |       |
| Hours | int         | YES  |     | NULL    |       |
+-------+-------------+------+-----+---------+-------+
3 rows in set (0.00 sec)
```

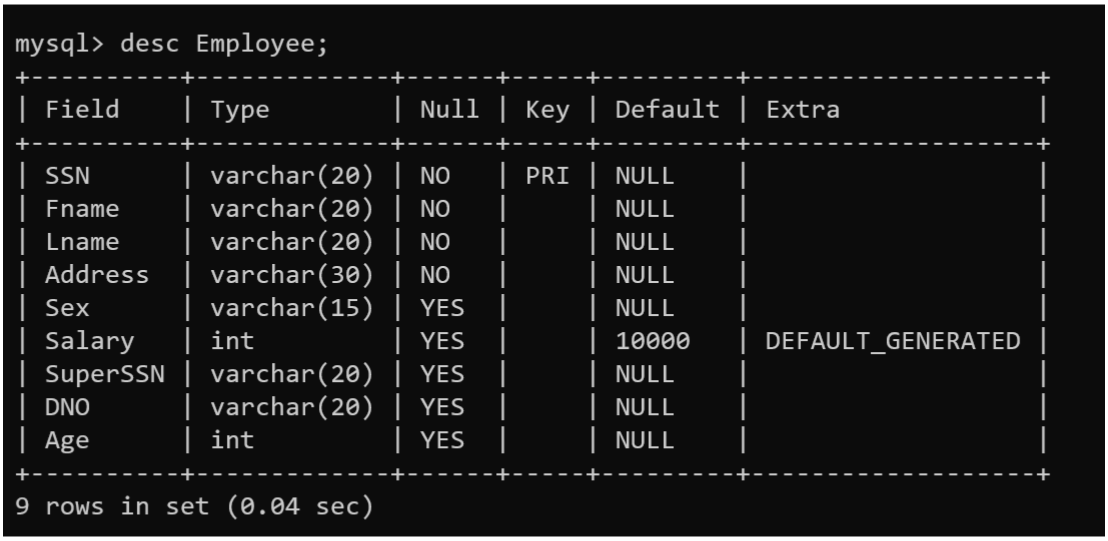
Question : 2

Add another column Age with datatype integer in employee table

QUERY

```
mysql> alter table Employee add(Age int);
Query OK, 0 rows affected (0.06 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

DATABASE TABLES



```
mysql> desc Employee;
+----------+-------------+------+-----+---------+-------------------+
| Field    | Type        | Null | Key | Default | Extra             |
+----------+-------------+------+-----+---------+-------------------+
| SSN      | varchar(20) | NO   | PRI | NULL    |                   |
| Fname    | varchar(20) | NO   |     | NULL    |                   |
| Lname    | varchar(20) | NO   |     | NULL    |                   |
| Address  | varchar(30) | NO   |     | NULL    |                   |
| Sex      | varchar(15) | YES  |     | NULL    |                   |
| Salary   | int         | YES  |     | 10000   | DEFAULT_GENERATED |
| SuperSSN | varchar(20) | YES  |     | NULL    |                   |
| DNO      | varchar(20) | YES  |     | NULL    |                   |
| Age      | int         | YES  |     | NULL    |                   |
+----------+-------------+------+-----+---------+-------------------+
9 rows in set (0.04 sec)
```

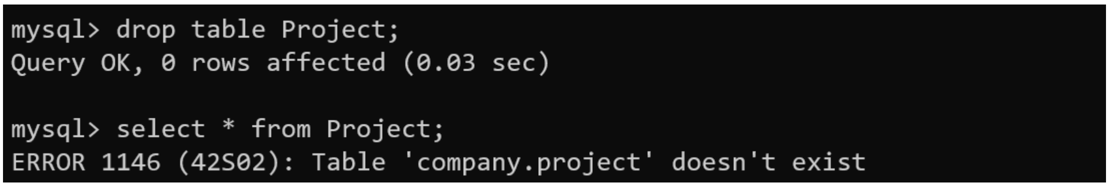
Question : 3

Drop a table named Project

QUERY

```
mysql>  drop table Project;
Query OK, 0 rows affected (0.03 sec)
```

DATABASE TABLES



```
mysql> drop table Project;
Query OK, 0 rows affected (0.03 sec)

mysql> select * from Project;
ERROR 1146 (42S02): Table 'company.project' doesn't exist
```

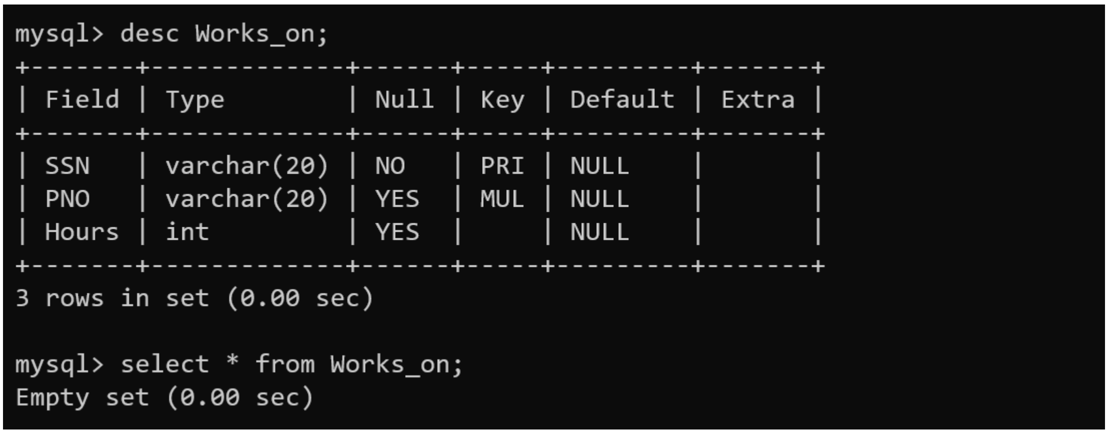
Question : 4

Truncate a table named WORKS_ON

QUERY

```
mysql> delete from Works_on;
Query OK, 0 rows affected (0.01 sec)
```

DATABASE TABLES



```
mysql> desc Works_on;
+-------+-------------+------+-----+---------+-------+
| Field | Type        | Null | Key | Default | Extra |
+-------+-------------+------+-----+---------+-------+
| SSN   | varchar(20) | NO   | PRI | NULL    |       |
| PNO   | varchar(20) | YES  | MUL | NULL    |       |
| Hours | int         | YES  |     | NULL    |       |
+-------+-------------+------+-----+---------+-------+
3 rows in set (0.00 sec)

mysql> select * from Works_on;
Empty set (0.00 sec)
```

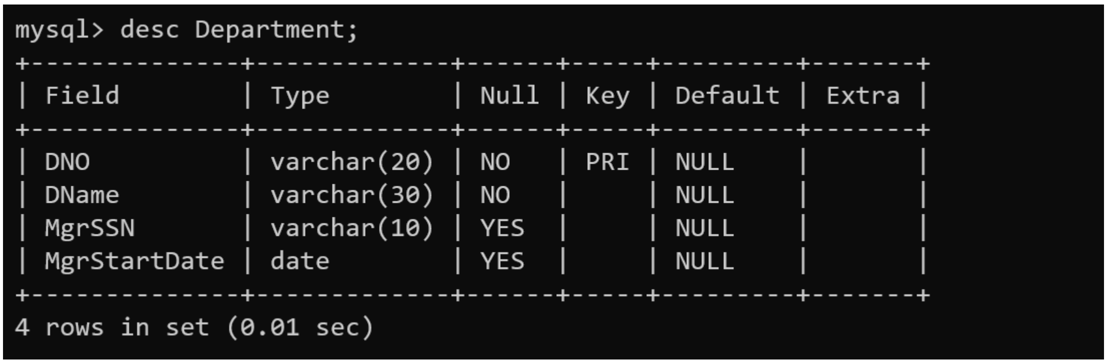
Question : 5

View the structure of the table Department

QUERY

```
mysql> desc Department;
```

DATABASE TABLES



```
mysql> desc Department;
+--------------+-------------+------+-----+---------+-------+
| Field        | Type        | Null | Key | Default | Extra |
+--------------+-------------+------+-----+---------+-------+
| DNO          | varchar(20) | NO   | PRI | NULL    |       |
| DName        | varchar(30) | NO   |     | NULL    |       |
| MgrSSN       | varchar(10) | YES  |     | NULL    |       |
| MgrStartDate | date        | YES  |     | NULL    |       |
+--------------+-------------+------+-----+---------+-------+
4 rows in set (0.01 sec)
```

DML COMMANDS

AIM

Develop SQL Queries to execute and verify the Data Manipulation Language commands.

Question : 1

Insert five records in the tables as per the above schema

QUERY

```
mysql> insert into Employee values("e1001","Archana","Suresh","13B,
Highway Gardens,Kozhikode","Female",60000,"SP1002","D001",28);
Query OK, 1 row affected (0.02 sec)

mysql> insert into Employee values("e1002","Justin","Varghese",
"Rose Villa,Kochi","Male",50000,"SP1001","D003",36);
Query OK, 1 row affected (0.00 sec)

mysql> insert into Employee values("e1003","Meera","Kumar","11B,Arcadia
Building,Mumbai","Female",70000,"SP1004","D005",31);
Query OK, 1 row affected (0.01 sec)

mysql> insert into Employee values("e1004","Kailas","Nath","V3,DD Homes,
Bangalore","Male",30000,"SP1003","D002",25);
Query OK, 1 row affected (0.00 sec)

mysql> insert into Employee values("e1005","Sara","Khalid","Ashok Nagar,
West Delhi","Female",45000,"SP1005","D004",27);
Query OK, 1 row affected (0.00 sec)

mysql>  insert into Department values("D001","Accounts","M1003","2015-09-01");
Query OK, 1 row affected (0.01 sec)

mysql> insert into Department values("D002","HR","M1002","2016-12-05");
Query OK, 1 row affected (0.00 sec)

mysql> insert into Department values("D003","Marketing","M1005","2012-04-04");
Query OK, 1 row affected (0.00 sec)
```

```
mysql> insert into Department values("D004","Sales","M1004","2019-08-20");
Query OK, 1 row affected (0.00 sec)

mysql> insert into Department values("D005","Management","M1001","2017-03-09");
Query OK, 1 row affected (0.00 sec)

mysql> insert into DLocation values("D001","Delhi");
Query OK, 1 row affected (0.01 sec)

mysql> insert into DLocation values("D002","Mumbai");
Query OK, 1 row affected (0.00 sec)

mysql> insert into DLocation values("D003","Bangalore");
Query OK, 1 row affected (0.00 sec)

mysql> insert into DLocation values("D004","Chennai");
Query OK, 1 row affected (0.00 sec)

mysql> insert into DLocation values("D005","Kochi");
Query OK, 1 row affected (0.00 sec)

mysql> insert into Project values("P002","Business_Management","Delhi","D005");
Query OK, 1 row affected (0.00 sec)

mysql> insert into Project values("P003","TV_Ads","Chennai","D003");
Query OK, 1 row affected (0.00 sec)

mysql> insert into Project values("P004","Expenditure_Check","Bangalore","D001");
Query OK, 1 row affected (0.01 sec)

mysql> insert into Project values("P005","Employee_Survey","Kochi","D002");
Query OK, 1 row affected (0.00 sec)

mysql> insert into Works_on values("S001","P002",9);
Query OK, 1 row affected (0.01 sec)

mysql> insert into Works_on values("S002","P005",12);
```

```
Query OK, 1 row affected (0.00 sec)

mysql> insert into Works_on values("S003","P001",4);
Query OK, 1 row affected (0.00 sec)

mysql> insert into Works_on values("S004","P003",8);
Query OK, 1 row affected (0.00 sec)

mysql> insert into Works_on values("S005","P004",10);
Query OK, 1 row affected (0.00 sec)
```

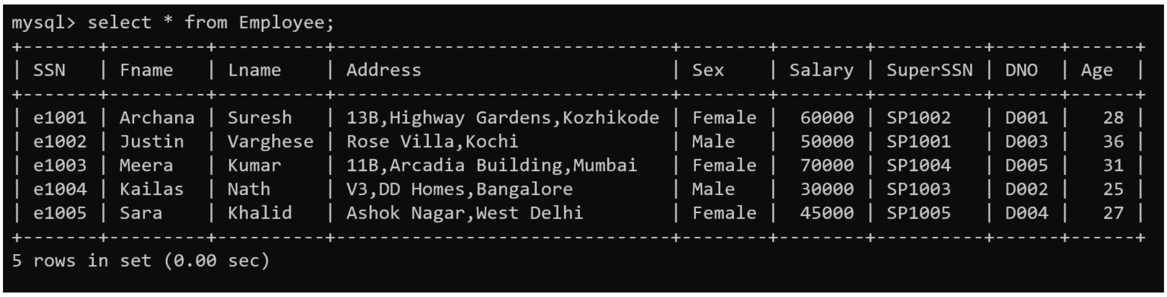
Question : 2

Display the entire content of the tables as per the above schema

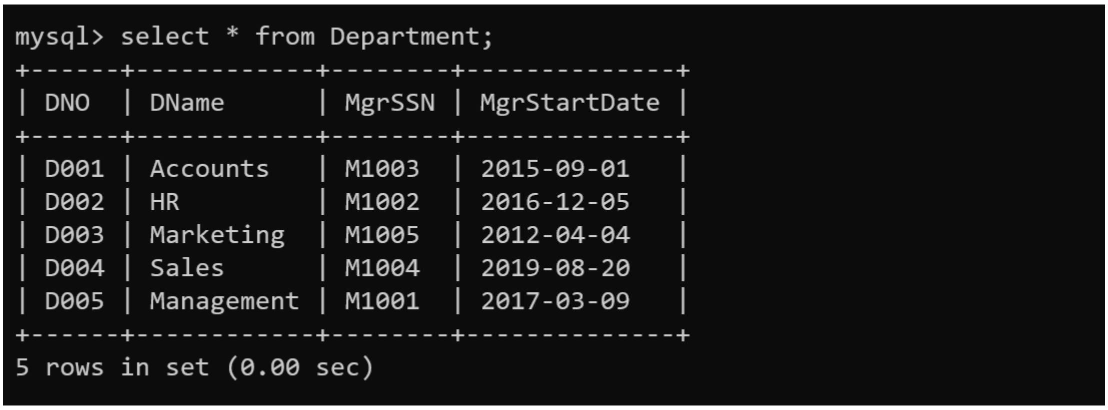
QUERY

```
mysql> select * from Employee;
mysql> select * from Department;
mysql> select * from DLocation;
mysql> select * from Project;
mysql> select * from Works_on;
```

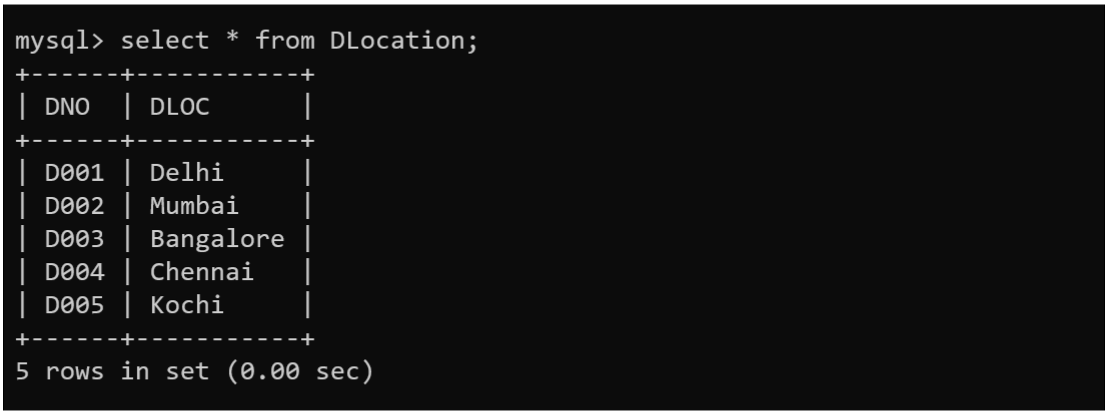
DATABASE TABLES



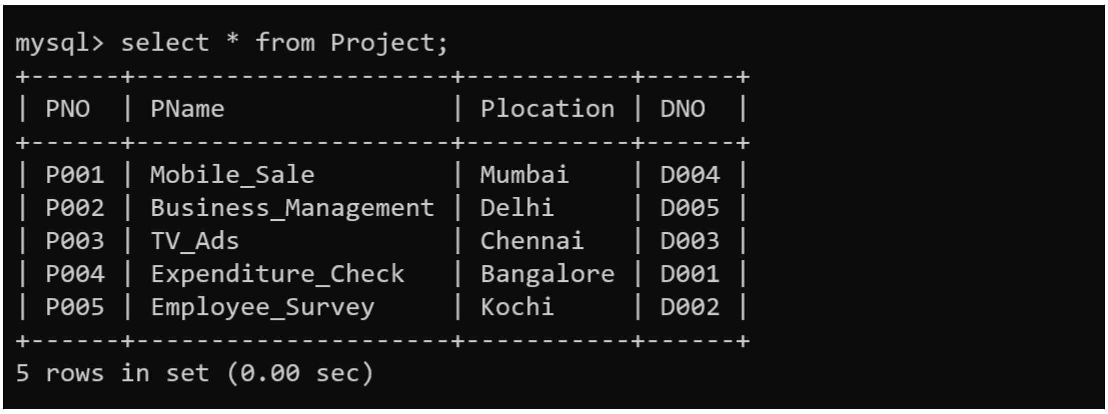
```
mysql> select * from Employee;
+-------+---------+----------+-------------------------------+--------+--------+----------+------+------+
| SSN   | Fname   | Lname    | Address                       | Sex    | Salary | SuperSSN | DNO  | Age  |
+-------+---------+----------+-------------------------------+--------+--------+----------+------+------+
| e1001 | Archana | Suresh   | 13B,Highway Gardens,Kozhikode | Female |  60000 | SP1002   | D001 |   28 |
| e1002 | Justin  | Varghese | Rose Villa,Kochi              | Male   |  50000 | SP1001   | D003 |   36 |
| e1003 | Meera   | Kumar    | 11B,Arcadia Building,Mumbai   | Female |  70000 | SP1004   | D005 |   31 |
| e1004 | Kailas  | Nath     | V3,DD Homes,Bangalore         | Male   |  30000 | SP1003   | D002 |   25 |
| e1005 | Sara    | Khalid   | Ashok Nagar,West Delhi        | Female |  45000 | SP1005   | D004 |   27 |
+-------+---------+----------+-------------------------------+--------+--------+----------+------+------+
5 rows in set (0.00 sec)
```



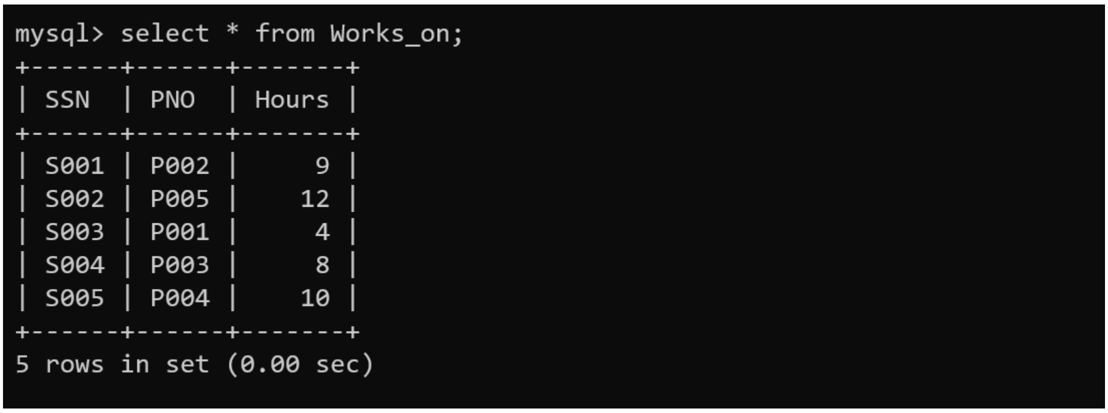
```
mysql> select * from Department;
+------+------------+--------+--------------+
| DNO  | DName      | MgrSSN | MgrStartDate |
+------+------------+--------+--------------+
| D001 | Accounts   | M1003  | 2015-09-01   |
| D002 | HR         | M1002  | 2016-12-05   |
| D003 | Marketing  | M1005  | 2012-04-04   |
| D004 | Sales      | M1004  | 2019-08-20   |
| D005 | Management | M1001  | 2017-03-09   |
+------+------------+--------+--------------+
5 rows in set (0.00 sec)
```



```
mysql> select * from DLocation;
+------+-----------+
| DNO  | DLOC      |
+------+-----------+
| D001 | Delhi     |
| D002 | Mumbai    |
| D003 | Bangalore |
| D004 | Chennai   |
| D005 | Kochi     |
+------+-----------+
5 rows in set (0.00 sec)
```



```
mysql> select * from Project;
+------+---------------------+-----------+------+
| PNO  | PName               | Plocation | DNO  |
+------+---------------------+-----------+------+
| P001 | Mobile_Sale         | Mumbai    | D004 |
| P002 | Business_Management | Delhi     | D005 |
| P003 | TV_Ads              | Chennai   | D003 |
| P004 | Expenditure_Check   | Bangalore | D001 |
| P005 | Employee_Survey     | Kochi     | D002 |
+------+---------------------+-----------+------+
5 rows in set (0.00 sec)
```



```
mysql> select * from Works_on;
+------+------+-------+
| SSN  | PNO  | Hours |
+------+------+-------+
| S001 | P002 |     9 |
| S002 | P005 |    12 |
| S003 | P001 |     4 |
| S004 | P003 |     8 |
| S005 | P004 |    10 |
+------+------+-------+
5 rows in set (0.00 sec)
```

Question : 3

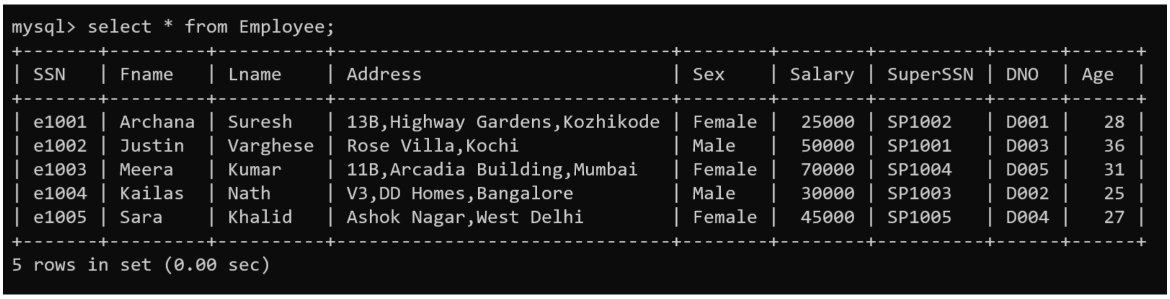
Modify the salary of the employee as 25000 whose SSN is e1001

QUERY

```
mysql> update Employee set Salary = 25000 where SSN = 'e1001';
```

```
Query OK, 1 row affected (0.03 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

DATABASE TABLES



```
mysql> select * from Employee;
+-------+---------+----------+------------------------------+--------+--------+----------+------+------+
| SSN   | Fname   | Lname    | Address                      | Sex    | Salary | SuperSSN | DNO  | Age  |
+-------+---------+----------+------------------------------+--------+--------+----------+------+------+
| e1001 | Archana | Suresh   | 13B,Highway Gardens,Kozhikode | Female |  25000 | SP1002   | D001 |   28 |
| e1002 | Justin  | Varghese | Rose Villa,Kochi             | Male   |  50000 | SP1001   | D003 |   36 |
| e1003 | Meera   | Kumar    | 11B,Arcadia Building,Mumbai  | Female |  70000 | SP1004   | D005 |   31 |
| e1004 | Kailas  | Nath     | V3,DD Homes,Bangalore        | Male   |  30000 | SP1003   | D002 |   25 |
| e1005 | Sara    | Khalid   | Ashok Nagar,West Delhi       | Female |  45000 | SP1005   | D004 |   27 |
+-------+---------+----------+------------------------------+--------+--------+----------+------+------+
5 rows in set (0.00 sec)
```

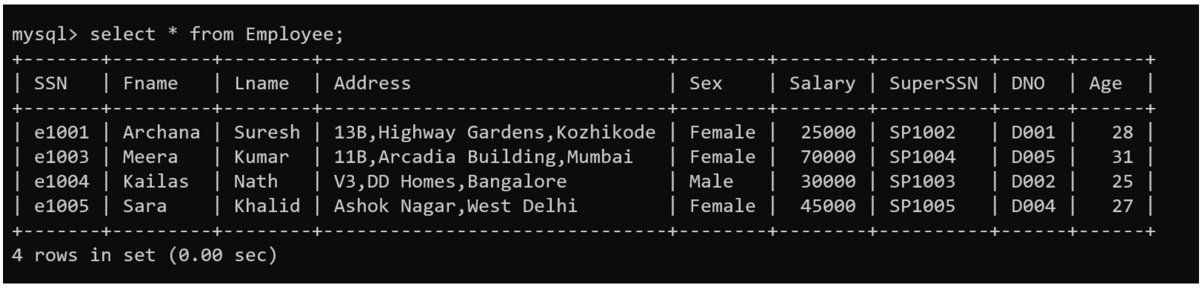
Question : 4

Delete the details of the employee whose SSN is ”e1002”

QUERY

```
mysql> delete from Employee where SSN = ’e1002’;
Query OK, 1 row affected (0.01 sec)
```

DATABASE TABLES



```
mysql> select * from Employee;
+-------+---------+--------+------------------------------+--------+--------+----------+------+------+
| SSN   | Fname   | Lname  | Address                      | Sex    | Salary | SuperSSN | DNO  | Age  |
+-------+---------+--------+------------------------------+--------+--------+----------+------+------+
| e1001 | Archana | Suresh | 13B,Highway Gardens,Kozhikode | Female |  25000 | SP1002   | D001 |   28 |
| e1003 | Meera   | Kumar  | 11B,Arcadia Building,Mumbai  | Female |  70000 | SP1004   | D005 |   31 |
| e1004 | Kailas  | Nath   | V3,DD Homes,Bangalore        | Male   |  30000 | SP1003   | D002 |   25 |
| e1005 | Sara    | Khalid | Ashok Nagar,West Delhi       | Female |  45000 | SP1005   | D004 |   27 |
+-------+---------+--------+------------------------------+--------+--------+----------+------+------+
4 rows in set (0.00 sec)
```

DCL COMMANDS

AIM

Develop SQL Queries to implement Data Control Language commands

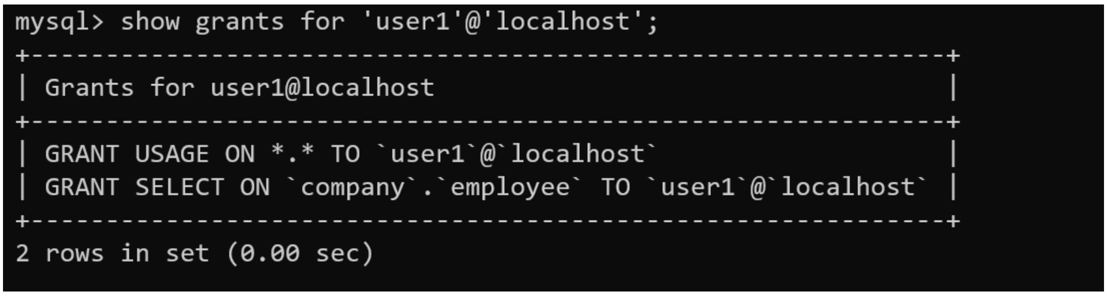
Question : 1

To grant a SELECT permission on employee table to user1

QUERY

```
mysql> create user ’user1’@’localhost’ identified by ’password’;
Query OK, 0 rows affected (0.26 sec)

mysql> grant select on Company.Employee to ’user1’@’localhost’;
Query OK, 0 rows affected (0.01 sec)
```

DATABASE TABLES



```
mysql> show grants for 'user1'@'localhost';
+---------------------------------------------------------+
| Grants for user1@localhost                              |
+---------------------------------------------------------+
| GRANT USAGE ON *.* TO `user1`@`localhost`               |
| GRANT SELECT ON `company`.`employee` TO `user1`@`localhost` |
+---------------------------------------------------------+
2 rows in set (0.00 sec)
```

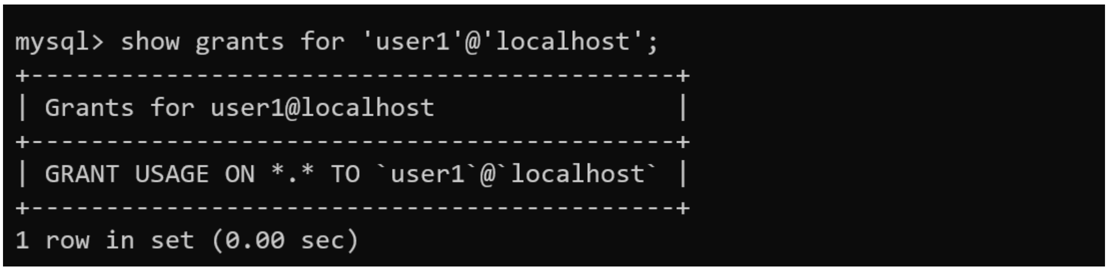
Question : 2

Revoking a privilege to all users in a table

QUERY

```
mysql> grant all on Company.Employee to ’user1’@’localhost’;
Query OK, 0 rows affected (0.01 sec)

mysql> Revoke all on Employee from ’user1’@’localhost’;
Query OK, 0 rows affected (0.01 sec)
```

DATABASE TABLES



```
mysql> show grants for 'user1'@'localhost';
+-------------------------------------------+
| Grants for user1@localhost                |
+-------------------------------------------+
| GRANT USAGE ON *.* TO `user1`@`localhost` |
+-------------------------------------------+
1 row in set (0.00 sec)
```

GROUP FUNCTION OR AGGREGATE FUNCTION

AIM

Develop SQL Queries to execute computation on table data with built-in functions

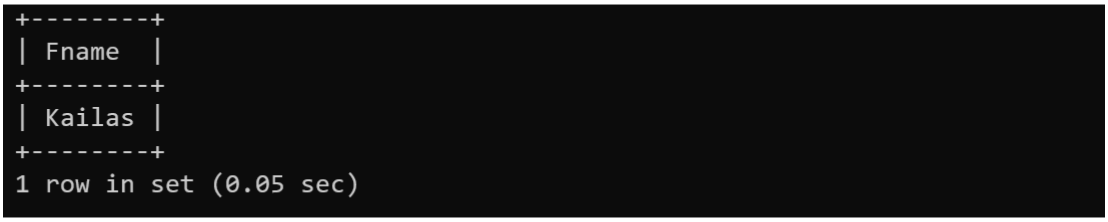
Question : 1

List the fname of all the employee having ‘a’ as the second last character in their name.

QUERY

```
mysql> select Fname from Employee where Fname like ’%a_’;
```

DATABASE TABLES



```
+--------+
| Fname  |
+--------+
| Kailas |
+--------+
1 row in set (0.05 sec)
```

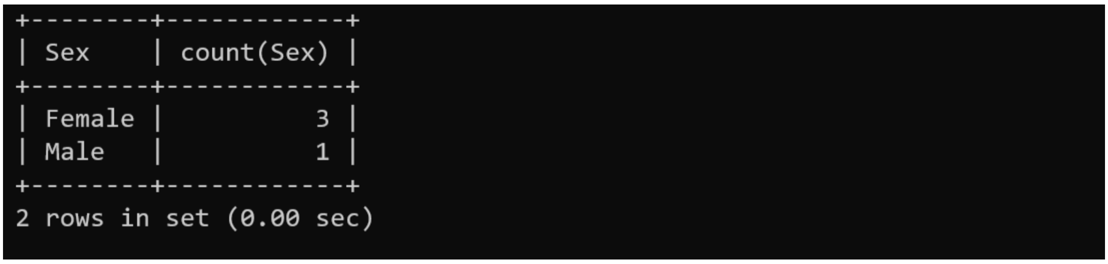
Question : 2

Count the total number of male and female employees in the Employee table.

QUERY

```
mysql> select Sex,count(Sex) from Employee group by Sex;
```

DATABASE TABLES



```
+--------+------------+
| Sex    | count(Sex) |
+--------+------------+
| Female |          3 |
| Male   |          1 |
+--------+------------+
2 rows in set (0.00 sec)
```

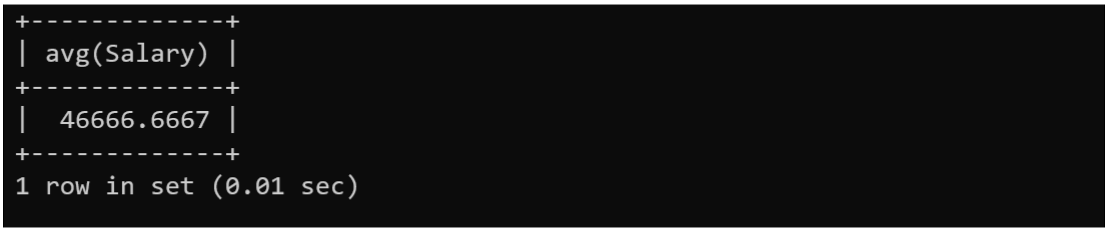
Question : 3

Calculate the average salary of the female employees.

QUERY

```
mysql> select avg(Salary) from Employee where Sex = ’Female’;
```

DATABASE TABLES



```
+-------------+
| avg(Salary) |
+-------------+
|  46666.6667 |
+-------------+
1 row in set (0.01 sec)
```

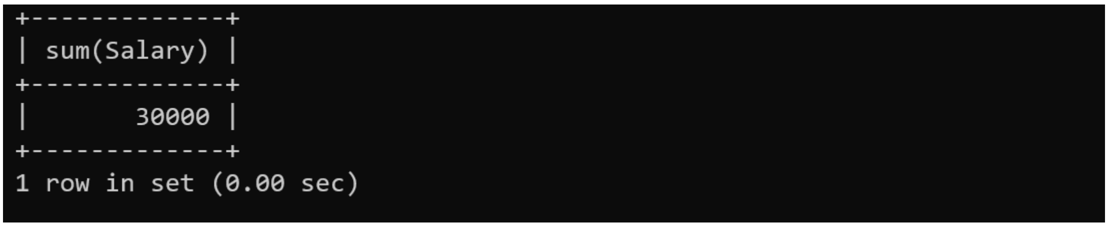
Question : 4

Calculate the sum of salaries of male employees.

QUERY

```
mysql> select sum(Salary) from Employee where Sex = ’Male’;
```

DATABASE TABLES



```
+-------------+
| sum(Salary) |
+-------------+
|       30000 |
+-------------+
1 row in set (0.00 sec)
```

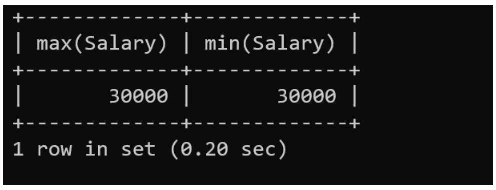
Question : 5

Display the maximum and minimum salaries of male employees.

QUERY

```
select max(Salary),min(Salary)  from Employee where Sex = ’Male’;
```

DATABASE TABLES



```
+-------------+-------------+
| max(Salary) | min(Salary) |
+-------------+-------------+
|       30000 |       30000 |
+-------------+-------------+
1 row in set (0.20 sec)
```

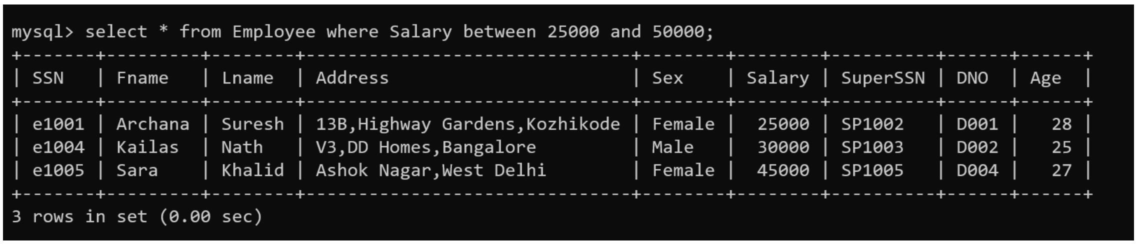
Question : 6

Display the details of all employees whose salary between 25000 and 50000

QUERY

```
 mysql> select * from Employee where Salary between 25000 and 50000;
```

DATABASE TABLES



```
mysql> select * from Employee where Salary between 25000 and 50000;
+-------+---------+--------+-------------------------------+--------+--------+----------+------+------+
| SSN   | Fname   | Lname  | Address                       | Sex    | Salary | SuperSSN | DNO  | Age  |
+-------+---------+--------+-------------------------------+--------+--------+----------+------+------+
| e1001 | Archana | Suresh | 13B,Highway Gardens,Kozhikode | Female |  25000 | SP1002   | D001 |   28 |
| e1004 | Kailas  | Nath   | V3,DD Homes,Bangalore         | Male   |  30000 | SP1003   | D002 |   25 |
| e1005 | Sara    | Khalid | Ashok Nagar,West Delhi        | Female |  45000 | SP1005   | D004 |   27 |
+-------+---------+--------+-------------------------------+--------+--------+----------+------+------+
3 rows in set (0.00 sec)
```

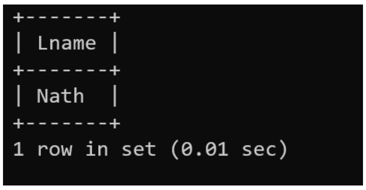
Question : 7

Display the lname of the employees whose salaries are 30000 or 40000 or 50000.

QUERY

```
mysql> select Lname from Employee where Salary in(30000,40000,50000);
```

DATABASE TABLES



```
+-------+
| Lname |
+-------+
| Nath  |
+-------+
1 row in set (0.01 sec)
```

NESTED QUERIES

AIM

Develop SQL Queries to implement Nested Queries/ Sub Queries and Joins

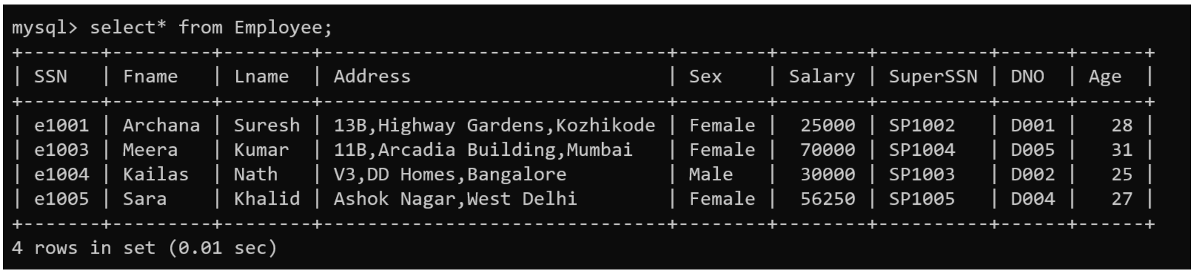
Question : 1

Update the salary by 0.25 times for all the employees whose Plocation is ‘Chennai’.

QUERY

```
mysql> update Employee,DLocation set Salary = Salary + Salary*0.25 where
Employee.DNO = DLocation.DNO and DLOC = 'Chennai';
Query OK, 1 row affected (0.16 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

DATABASE TABLES



```
mysql> select* from Employee;
+-------+---------+--------+------------------------------+--------+--------+----------+------+------+
| SSN   | Fname   | Lname  | Address                      | Sex    | Salary | SuperSSN | DNO  | Age  |
+-------+---------+--------+------------------------------+--------+--------+----------+------+------+
| e1001 | Archana | Suresh | 13B,Highway Gardens,Kozhikode | Female |  25000 | SP1002   | D001 |   28 |
| e1003 | Meera   | Kumar  | 11B,Arcadia Building,Mumbai  | Female |  70000 | SP1004   | D005 |   31 |
| e1004 | Kailas  | Nath   | V3,DD Homes,Bangalore        | Male   |  30000 | SP1003   | D002 |   25 |
| e1005 | Sara    | Khalid | Ashok Nagar,West Delhi       | Female |  56250 | SP1005   | D004 |   27 |
+-------+---------+--------+------------------------------+--------+--------+----------+------+------+
4 rows in set (0.01 sec)
```

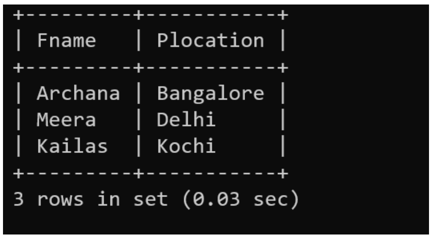
Question : 2

To display the name and project location of employees whose working hour is greater than 5

QUERY

```
mysql> select Fname,Plocation from Employee,Project,Works_on where
Employee.DNO = Project.DNO and Project.PNO = Works_on.PNO and Hours>5;
```

DATABASE TABLES



```
+---------+-----------+
| Fname   | Plocation |
+---------+-----------+
| Archana | Bangalore |
| Meera   | Delhi     |
| Kailas  | Kochi     |
+---------+-----------+
3 rows in set (0.03 sec)
```

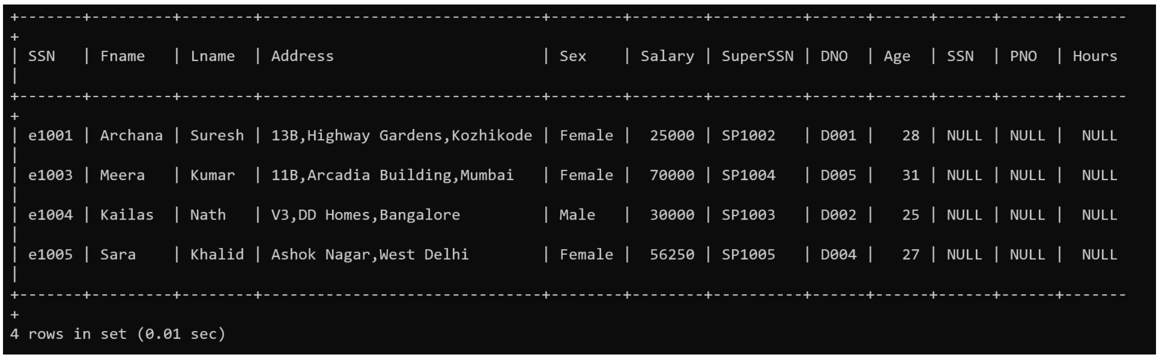
Question : 3

Left join employee table and works_on table

QUERY

```
mysql> select * from Employee left join Works_on on Employee.SSN = Works_on.SSN;
```

DATABASE TABLES



```
+-------+---------+--------+-------------------------------+--------+--------+----------+------+------+------+------+-------
+
| SSN   | Fname   | Lname  | Address                       | Sex    | Salary | SuperSSN | DNO  | Age  | SSN  | PNO  | Hours
|
+-------+---------+--------+-------------------------------+--------+--------+----------+------+------+------+------+-------
+
| e1001 | Archana | Suresh | 13B,Highway Gardens,Kozhikode | Female |  25000 | SP1002   | D001 |   28 | NULL | NULL |  NULL
|
| e1003 | Meera   | Kumar  | 11B,Arcadia Building,Mumbai   | Female |  70000 | SP1004   | D005 |   31 | NULL | NULL |  NULL
|
| e1004 | Kailas  | Nath   | V3,DD Homes,Bangalore         | Male   |  30000 | SP1003   | D002 |   25 | NULL | NULL |  NULL
|
| e1005 | Sara    | Khalid | Ashok Nagar,West Delhi        | Female |  56250 | SP1005   | D004 |   27 | NULL | NULL |  NULL
|
+-------+---------+--------+-------------------------------+--------+--------+----------+------+------+------+------+-------
+
4 rows in set (0.01 sec)
```

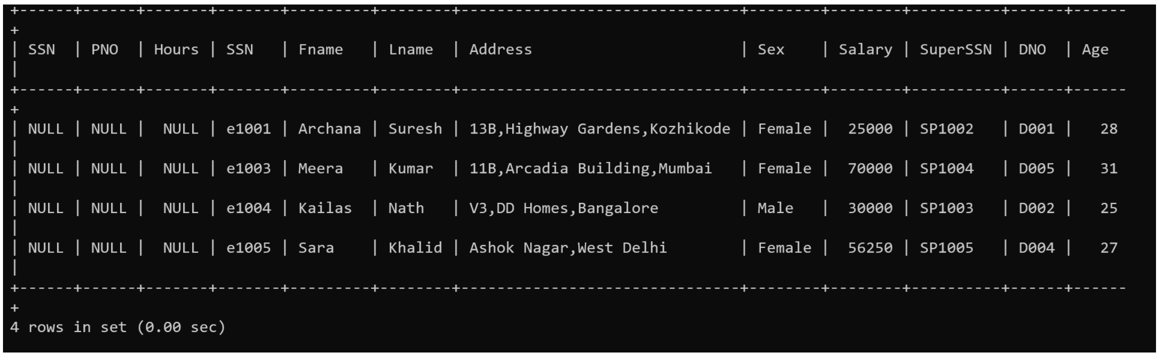
Question : 4

Right join works_on table and employee table

QUERY

```
mysql> select * from Works_on right join Employee on Employee.SSN = Works_on.SSN;
```

DATABASE TABLES



```
+------+------+-------+-------+---------+--------+-------------------------------+--------+--------+----------+------+------
+
| SSN  | PNO  | Hours | SSN   | Fname   | Lname  | Address                       | Sex    | Salary | SuperSSN | DNO  | Age
|
+------+------+-------+-------+---------+--------+-------------------------------+--------+--------+----------+------+------
+
| NULL | NULL |  NULL | e1001 | Archana | Suresh | 13B,Highway Gardens,Kozhikode | Female |  25000 | SP1002   | D001 |   28
|
| NULL | NULL |  NULL | e1003 | Meera   | Kumar  | 11B,Arcadia Building,Mumbai   | Female |  70000 | SP1004   | D005 |   31
|
| NULL | NULL |  NULL | e1004 | Kailas  | Nath   | V3,DD Homes,Bangalore         | Male   |  30000 | SP1003   | D002 |   25
|
| NULL | NULL |  NULL | e1005 | Sara    | Khalid | Ashok Nagar,West Delhi        | Female |  56250 | SP1005   | D004 |   27
|
+------+------+-------+-------+---------+--------+-------------------------------+--------+--------+----------+------+------
+
4 rows in set (0.00 sec)
```

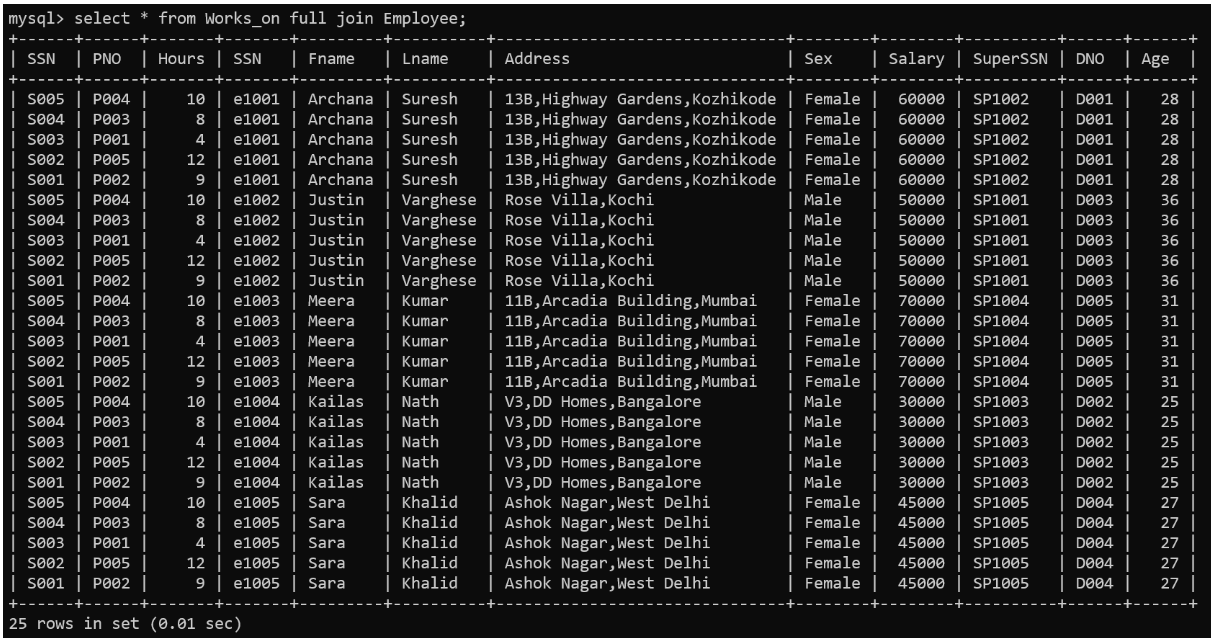
Question : 5

Full join works_on table and employee table

QUERY

```
mysql> select * from WORKS_ON full join EMPLOYEE ;
```

DATABASE TABLES



```
mysql> select * from Works_on full join Employee;
+------+------+-------+-------+---------+----------+-------------------------------+--------+--------+----------+------+------+
| SSN  | PNO  | Hours | SSN   | Fname   | Lname    | Address                       | Sex    | Salary | SuperSSN | DNO  | Age  |
+------+------+-------+-------+---------+----------+-------------------------------+--------+--------+----------+------+------+
| S005 | P004 |    10 | e1001 | Archana | Suresh   | 13B,Highway Gardens,Kozhikode | Female |  60000 | SP1002   | D001 |   28 |
| S004 | P003 |     8 | e1001 | Archana | Suresh   | 13B,Highway Gardens,Kozhikode | Female |  60000 | SP1002   | D001 |   28 |
| S003 | P001 |     4 | e1001 | Archana | Suresh   | 13B,Highway Gardens,Kozhikode | Female |  60000 | SP1002   | D001 |   28 |
| S002 | P005 |    12 | e1001 | Archana | Suresh   | 13B,Highway Gardens,Kozhikode | Female |  60000 | SP1002   | D001 |   28 |
| S001 | P002 |     9 | e1001 | Archana | Suresh   | 13B,Highway Gardens,Kozhikode | Female |  60000 | SP1002   | D001 |   28 |
| S005 | P004 |    10 | e1002 | Justin  | Varghese | Rose Villa,Kochi              | Male   |  50000 | SP1001   | D003 |   36 |
| S004 | P003 |     8 | e1002 | Justin  | Varghese | Rose Villa,Kochi              | Male   |  50000 | SP1001   | D003 |   36 |
| S003 | P001 |     4 | e1002 | Justin  | Varghese | Rose Villa,Kochi              | Male   |  50000 | SP1001   | D003 |   36 |
| S002 | P005 |    12 | e1002 | Justin  | Varghese | Rose Villa,Kochi              | Male   |  50000 | SP1001   | D003 |   36 |
| S001 | P002 |     9 | e1002 | Justin  | Varghese | Rose Villa,Kochi              | Male   |  50000 | SP1001   | D003 |   36 |
| S005 | P004 |    10 | e1003 | Meera   | Kumar    | 11B,Arcadia Building,Mumbai   | Female |  70000 | SP1004   | D005 |   31 |
| S004 | P003 |     8 | e1003 | Meera   | Kumar    | 11B,Arcadia Building,Mumbai   | Female |  70000 | SP1004   | D005 |   31 |
| S003 | P001 |     4 | e1003 | Meera   | Kumar    | 11B,Arcadia Building,Mumbai   | Female |  70000 | SP1004   | D005 |   31 |
| S002 | P005 |    12 | e1003 | Meera   | Kumar    | 11B,Arcadia Building,Mumbai   | Female |  70000 | SP1004   | D005 |   31 |
| S001 | P002 |     9 | e1003 | Meera   | Kumar    | 11B,Arcadia Building,Mumbai   | Female |  70000 | SP1004   | D005 |   31 |
| S005 | P004 |    10 | e1004 | Kailas  | Nath     | V3,DD Homes,Bangalore         | Male   |  30000 | SP1003   | D002 |   25 |
| S004 | P003 |     8 | e1004 | Kailas  | Nath     | V3,DD Homes,Bangalore         | Male   |  30000 | SP1003   | D002 |   25 |
| S003 | P001 |     4 | e1004 | Kailas  | Nath     | V3,DD Homes,Bangalore         | Male   |  30000 | SP1003   | D002 |   25 |
| S002 | P005 |    12 | e1004 | Kailas  | Nath     | V3,DD Homes,Bangalore         | Male   |  30000 | SP1003   | D002 |   25 |
| S001 | P002 |     9 | e1004 | Kailas  | Nath     | V3,DD Homes,Bangalore         | Male   |  30000 | SP1003   | D002 |   25 |
| S005 | P004 |    10 | e1005 | Sara    | Khalid   | Ashok Nagar,West Delhi        | Female |  45000 | SP1005   | D004 |   27 |
| S004 | P003 |     8 | e1005 | Sara    | Khalid   | Ashok Nagar,West Delhi        | Female |  45000 | SP1005   | D004 |   27 |
| S003 | P001 |     4 | e1005 | Sara    | Khalid   | Ashok Nagar,West Delhi        | Female |  45000 | SP1005   | D004 |   27 |
| S002 | P005 |    12 | e1005 | Sara    | Khalid   | Ashok Nagar,West Delhi        | Female |  45000 | SP1005   | D004 |   27 |
| S001 | P002 |     9 | e1005 | Sara    | Khalid   | Ashok Nagar,West Delhi        | Female |  45000 | SP1005   | D004 |   27 |
+------+------+-------+-------+---------+----------+-------------------------------+--------+--------+----------+------+------+
25 rows in set (0.01 sec)
```

VIEWS

AIM

Develop SQL Queries for creating and dropping Views

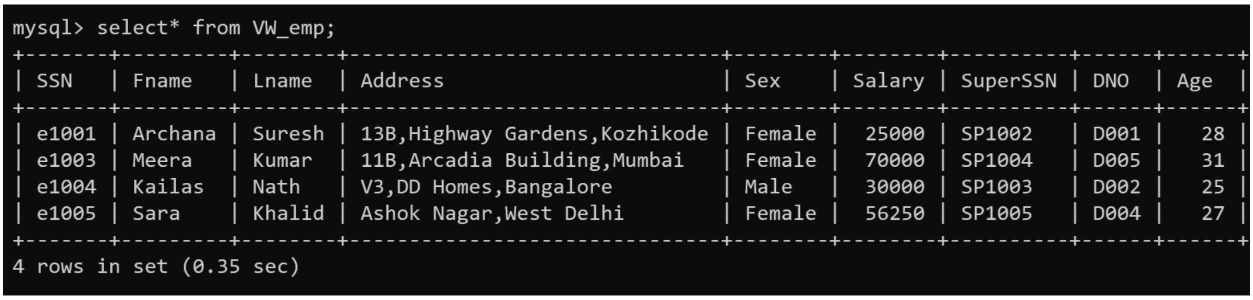
Question : 1

Create a view VW_emp on employee table

QUERY

```
mysql> create view VW_emp as select * from Employee ;
Query OK, 0 rows affected (0.16 sec)
```

DATABASE TABLES



```
mysql> select* from VW_emp;
+-------+---------+--------+-------------------------------+--------+--------+----------+------+------+
| SSN   | Fname   | Lname  | Address                       | Sex    | Salary | SuperSSN | DNO  | Age  |
+-------+---------+--------+-------------------------------+--------+--------+----------+------+------+
| e1001 | Archana | Suresh | 13B,Highway Gardens,Kozhikode | Female |  25000 | SP1002   | D001 |   28 |
| e1003 | Meera   | Kumar  | 11B,Arcadia Building,Mumbai   | Female |  70000 | SP1004   | D005 |   31 |
| e1004 | Kailas  | Nath   | V3,DD Homes,Bangalore         | Male   |  30000 | SP1003   | D002 |   25 |
| e1005 | Sara    | Khalid | Ashok Nagar,West Delhi        | Female |  56250 | SP1005   | D004 |   27 |
+-------+---------+--------+-------------------------------+--------+--------+----------+------+------+
4 rows in set (0.35 sec)
```

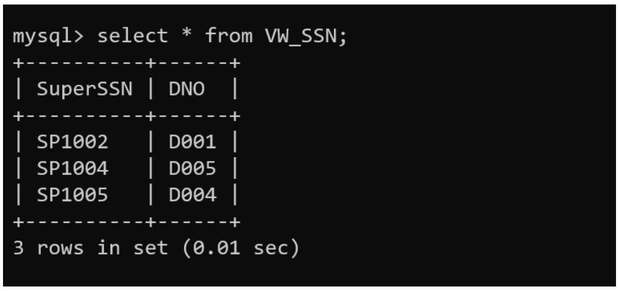
Question : 2

Create another view VW_SSN contains SuperSSN and Dno of female employees

QUERY

```
mysql> create view VW_SSN as select SuperSSN,DNO from Employee where
Sex = 'Female';
Query OK, 0 rows affected (0.02 sec)
```

DATABASE TABLES



```
mysql> select * from VW_SSN;
+----------+------+
| SuperSSN | DNO  |
+----------+------+
| SP1002   | D001 |
| SP1004   | D005 |
| SP1005   | D004 |
+----------+------+
3 rows in set (0.01 sec)
```

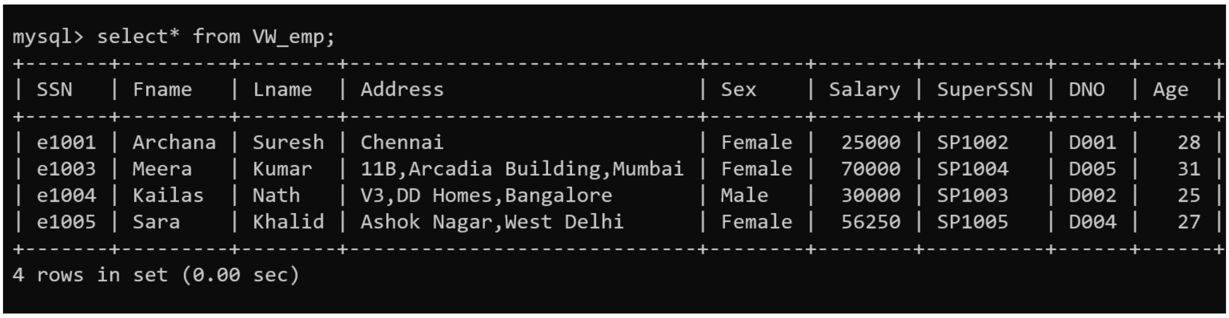
Question : 3

Update the address of employee to Chennai whose id is e1001 in view VW_emp

QUERY

```
mysql> update VW_emp set Address="Chennai" where SSN ='e1001';
Query OK, 1 row affected (0.02 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

DATABASE TABLES



```
mysql> select* from VW_emp;
+-------+---------+--------+-----------------------------+--------+--------+----------+------+------+
| SSN   | Fname   | Lname  | Address                     | Sex    | Salary | SuperSSN | DNO  | Age  |
+-------+---------+--------+-----------------------------+--------+--------+----------+------+------+
| e1001 | Archana | Suresh | Chennai                     | Female |  25000 | SP1002   | D001 |   28 |
| e1003 | Meera   | Kumar  | 11B,Arcadia Building,Mumbai | Female |  70000 | SP1004   | D005 |   31 |
| e1004 | Kailas  | Nath   | V3,DD Homes,Bangalore       | Male   |  30000 | SP1003   | D002 |   25 |
| e1005 | Sara    | Khalid | Ashok Nagar,West Delhi      | Female |  56250 | SP1005   | D004 |   27 |
+-------+---------+--------+-----------------------------+--------+--------+----------+------+------+
4 rows in set (0.00 sec)
```

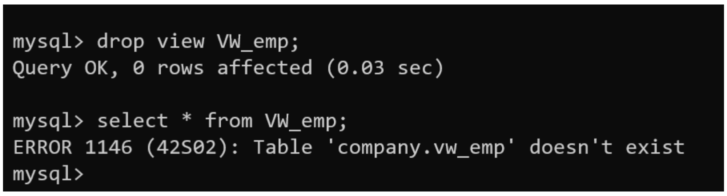
Question : 4

Delete the view VW_emp

QUERY

```
mysql> drop view VW_emp;
Query OK, 0 rows affected (0.00 sec)
```

DATABASE TABLES



```
mysql> drop view VW_emp;
Query OK, 0 rows affected (0.03 sec)

mysql> select * from VW_emp;
ERROR 1146 (42S02): Table 'company.vw_emp' doesn't exist
mysql>
```

FUNCTIONS AND PROCEDURES

AIM

Develop PL/SQL program to familiarize with Function and Procedure

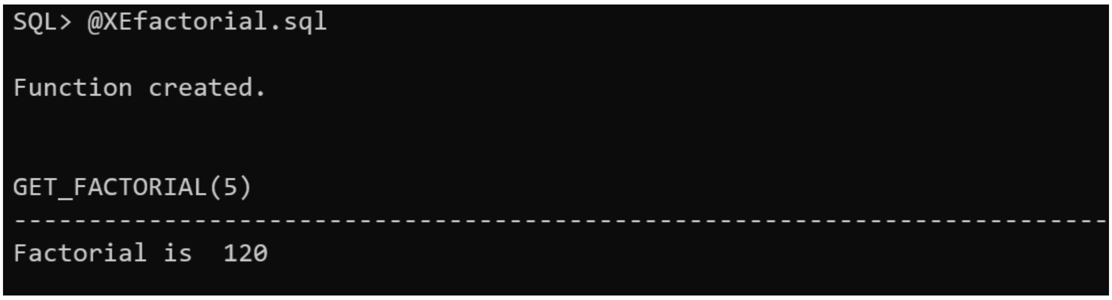
Question : 1

Write a PL/SQL function to find factorial of a number

QUERY

```
SQL*Plus: Release 11.2.0.2.0 Production on Fri Dec 16 14:29:44 2022

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

SQL> connect
Enter user-name: system
Enter password:
Connected.

SQL> set serveroutput on
SQL> edit@factorial.sql

create or replace function get_factorial(N int)
return varchar
is
fact int := 1;
begin
for i in 1..N loop
fact := fact*i;
end loop;
return ’Factorial is  ’  || fact ;
end;
/
select get_factorial(5) from dual;

SQL> @XEfactorial.sql

Function created.
```

DATABASE TABLES



```
SQL> @XEfactorial.sql

Function created.


GET_FACTORIAL(5)
--------------------------------------------------------------------------
Factorial is  120
```

Question : 2

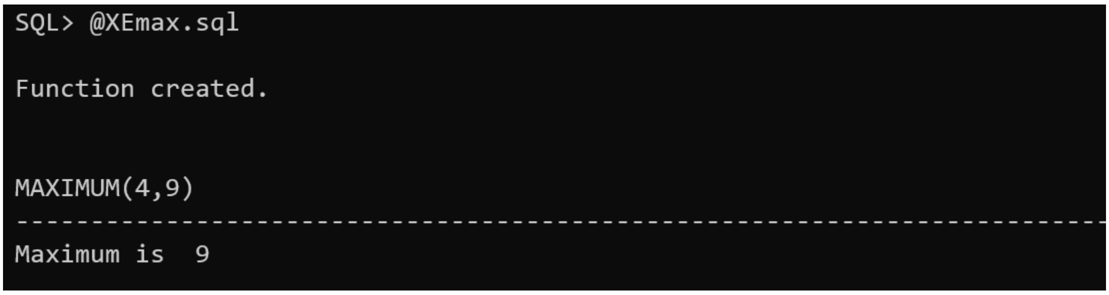
Write a PL/SQL function to find maximum of two numbers

QUERY

```
SQL*Plus: Release 11.2.0.2.0 Production on Fri Dec 16 14:29:44 2022

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

SQL> connect
Enter user-name: system
Enter password:
Connected.

SQL> set serveroutput on
SQL> edit@max.sql

create or replace function maximum(n1 int, n2 int)
return varchar
is
m int := 0;
begin
if n1>n2 then
m := n1;
else
m := n2;
end if;
return  ’Maximum is  ’ ||m;
end;
```

```
/
select maximum(4,9) from dual;

SQL> @XEmax.sql

Function created.
```

DATABASE TABLES



```
SQL> @XEmax.sql

Function created.


MAXIMUM(4,9)
--------------------------------------------------------------------------------
Maximum is  9
```

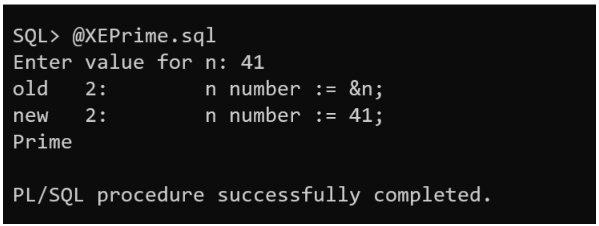
Question : 3

Write a PL/SQL procedure to print the prime

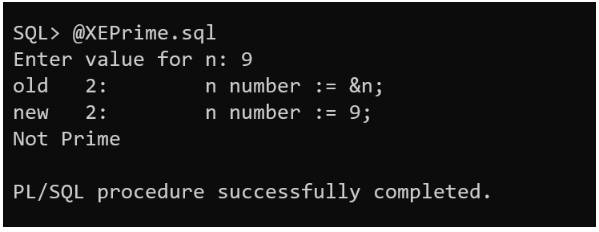
QUERY

```
SQL*Plus: Release 21.0.0.0.0 - Production on Sun Jan 15 23:37:24 2023
Version 21.3.0.0.0

Copyright (c) 1982, 2021, Oracle.  All rights reserved.

Enter user-name: system
Enter password:
Last Successful login time: Sun Jan 15 2023 16:42:04 +05:30

Connected to:
Oracle Database 21c Express Edition Release 21.0.0.0.0 - Production
Version 21.3.0.0.0

SQL> set serveroutput on
SQL> edit@Prime.sql
SQL> connect
Enter user-name: system
Enter password:
Connected.


SQL> set serveroutput on

SQL> edit@Prime.sql

declare
n number := &n;
i number := 1;
flag number := 0;

begin
while i < n/2
loop
i := i+1;
if (mod(n,i)= 0) then
flag := 1;
end if;
end loop;
if (flag = 1) then
dbms_output.put_line(’Not Prime’);
else
dbms_output.put_line(’Prime’);
end if;
end;
/
SQL> @XEprime.sql
```

DATABASE TABLES



```
SQL> @XEPrime.sql
Enter value for n: 41
old   2:        n number := &n;
new   2:        n number := 41;
Prime

PL/SQL procedure successfully completed.
```



```
SQL> @XEPrime.sql
Enter value for n: 9
old   2:        n number := &n;
new   2:        n number := 9;
Not Prime

PL/SQL procedure successfully completed.
```

Question : 4

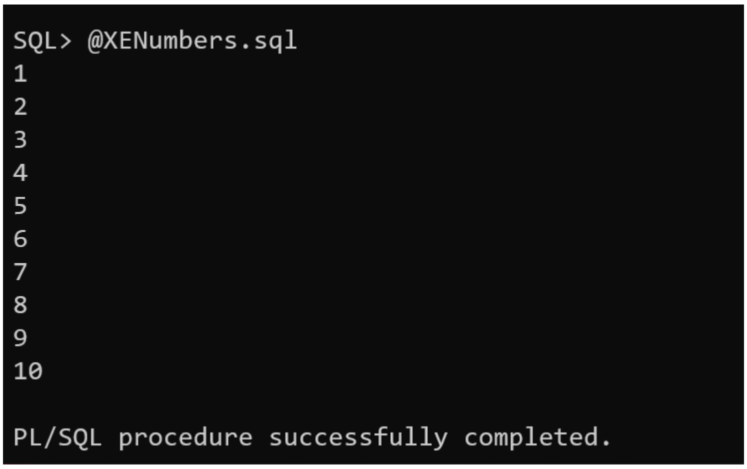
Write a PL/SQL procedure to display numbers from 1 to 10 using while loop

QUERY

```
SQL*Plus: Release 21.0.0.0.0 - Production on Sun Jan 15 23:44:54 2023
Version 21.3.0.0.0

Copyright (c) 1982, 2021, Oracle.  All rights reserved.

Enter user-name: system
Enter password:
Last Successful login time: Sun Jan 15 2023 23:44:19 +05:30

Connected to:
Oracle Database 21c Express Edition Release 21.0.0.0.0 - Production
Version 21.3.0.0.0

SQL> set serveroutput on
SQL> edit@Numbers.sql

DECLARE
     i INTEGER := 1;
BEGIN
     WHILE i <= 10 LOOP
         DBMS_OUTPUT.PUT_LINE(i);
     i := i+1;
```

```
    END LOOP;
END;
/

SQL> @XENumbers.sql

Function created.
```

DATABASE TABLES



```
SQL> @XENumbers.sql
1
2
3
4
5
6
7
8
9
10

PL/SQL procedure successfully completed.
```

CURSOR

AIM

Develop PL/SQL program to implement Cursor

Question : 1

Write a PL/SQL cursor program to update the salary of each employee of department number D001 in the Employee table as per the schema

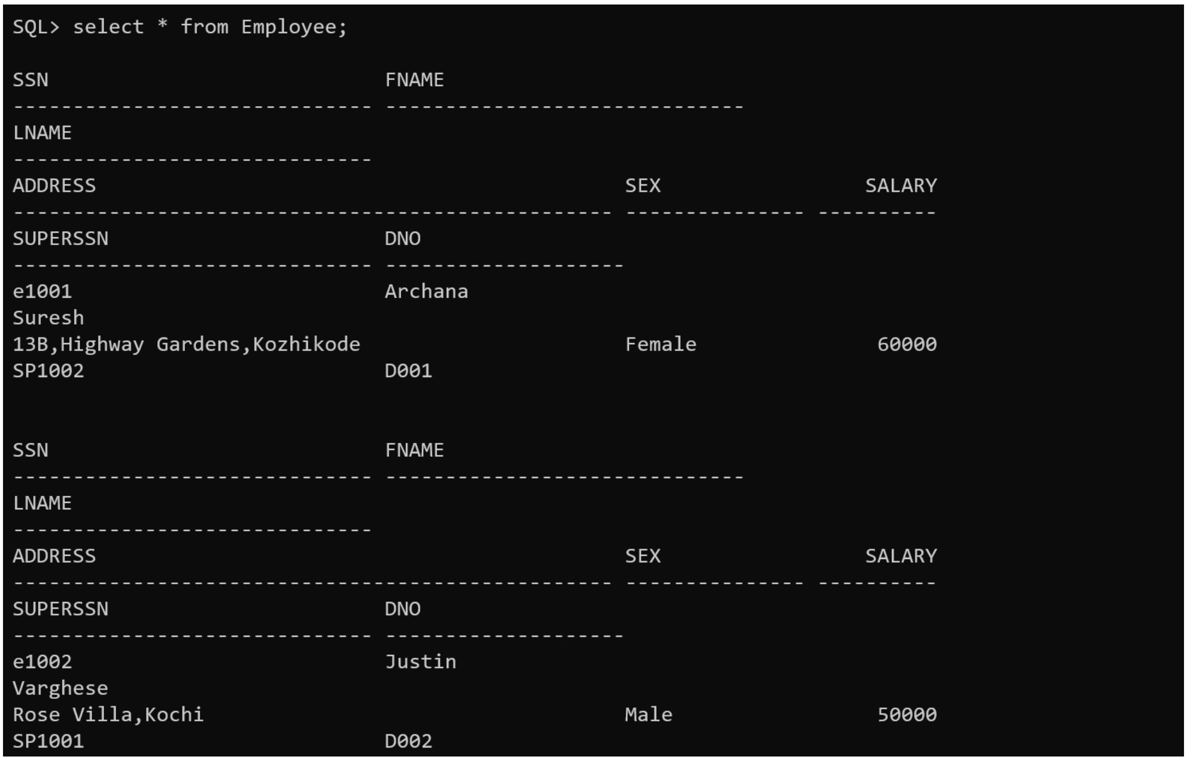
QUERY

```
SQL> create table Employee(SSN varchar(30),Fname varchar(30),Lname varchar(30),
Address varchar(50),Sex varchar(15),Salary number(30),SuperSSN varchar(30),
DNO varchar(20));

Table created.

SQL> create table Department(DNO varchar(20),Dname varchar(30),MgrSSN varchar(30),
MgrStartDate varchar(20));

Table created.

SQL> insert into Employee values('e1001','Archana','Suresh','13B,Highway Gardens,
Kozhikode','Female',60000,'SP1002','D001');

1 row created.

SQL> insert into Employee values('e1002','Justin','Varghese','Rose Villa,
Kochi','Male',50000,'SP1001','D002');

1 row created.

SQL> insert into Employee values('e1003','Meera','Kumar','11B,Arcadia Building,
Mumbai','Female',70000,'SP1004','D001');

1 row created.
```

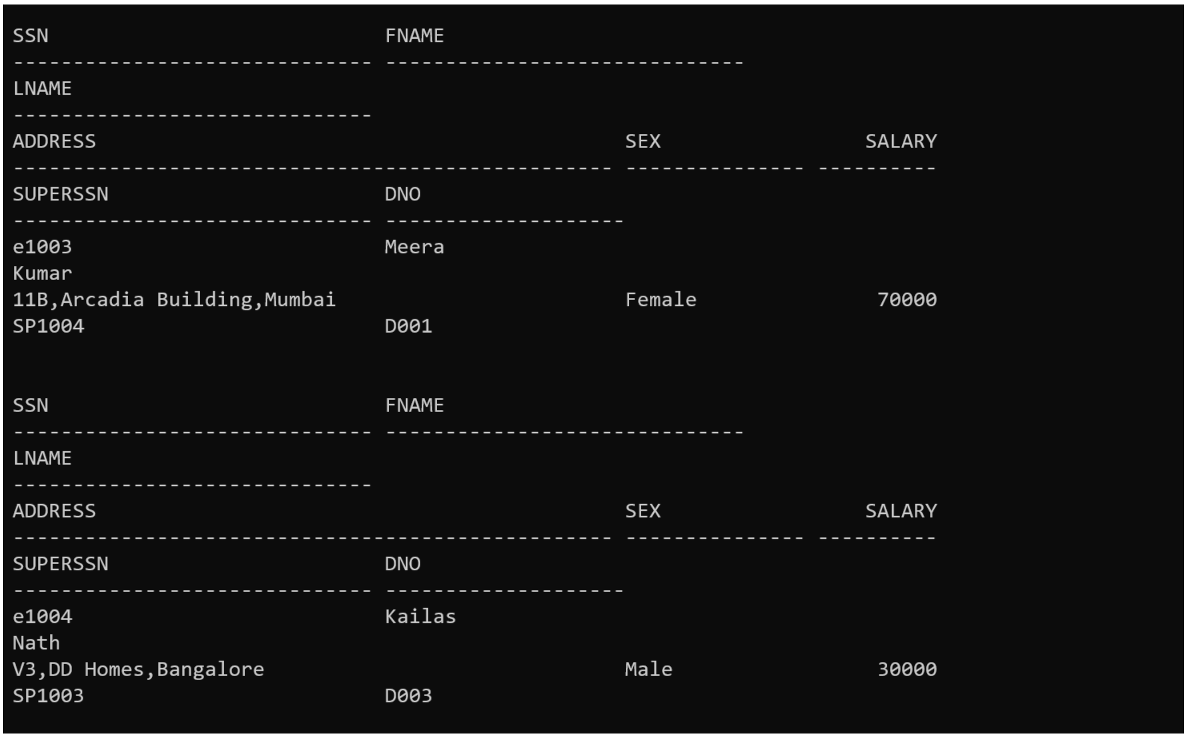
```
SQL> insert into Employee values(’e1004’,’Kailas’,’Nath’,’V3,DD Homes,Bangalore’,
’Male’,30000,’SP1003’,’D003’);

1 row created.

SQL> insert into Employee values(’e1005 ’,’Sara’,’Khaild’,’Ashok Nagar,
West Delhi’,’Female’,45000,’SP1005’,’D004’);

1 row created.

SQL> insert into Employee values(’e1006 ’,’Rahul’,’Ashok’,’LV Road,Bengaluru’,
’Male’,55000,’SP1005’,’D005’);

1 row created.

SQL> create table Department(DNO varchar(20),Dname varchar(30),MgrSSN varchar(30),
MgrStartDate varchar(20));

Table created.

SQL> insert into Department values(’D001’,’Accounts’,’M1003’,’2015-09-01’);
1 row created.

SQL> insert into Department values(’D002’,’HR’,’M1002’,’2016-12-05’);

1 row created.

SQL> insert into Department values(’D003’,’Marketing’,’M1005’,’2012-04-04’);

1 row created.

SQL> insert into Department values(’D004’,’Sales’,’M1004’,’2019-08-20’);

1 row created.

SQL> insert into Department values(’D005’,’Management’,’M1001’,’2017-03-09’);

1 row created.
```

```
SQL> declare cursor employee_cur is
  2  select SSN,Salary from Employee where DNO = 'D001'
  3  for update;
  4  incr_sal number;
  5  begin
  6  for employee_rec in employee_cur loop
  7  if employee_rec.Salary < 50000 then
  8  incr_sal := .15;
  9  else
 10  incr_sal := .10;
 11  end if;
 12  update Employee set Salary = Salary + Salary * incr_sal where current of
 employee_cur;
 13  end loop;
 14  end;
 15  /

PL/SQL procedure successfully completed.
```

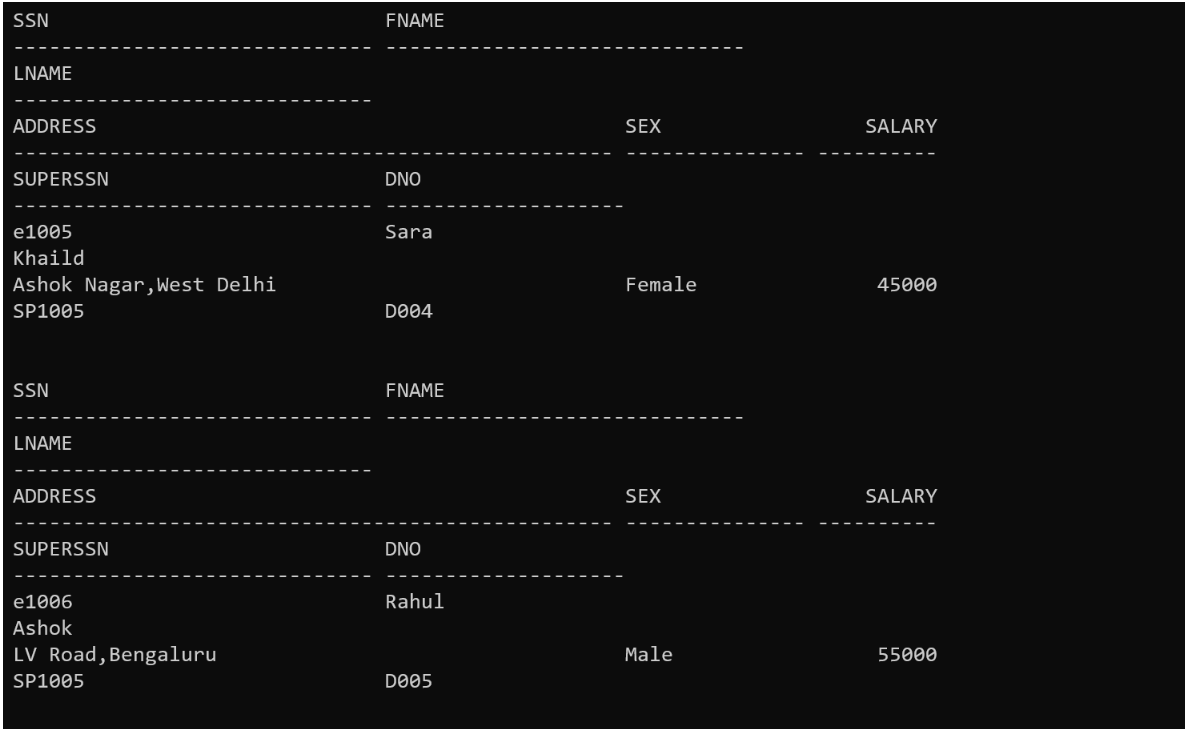
DATABASE TABLES



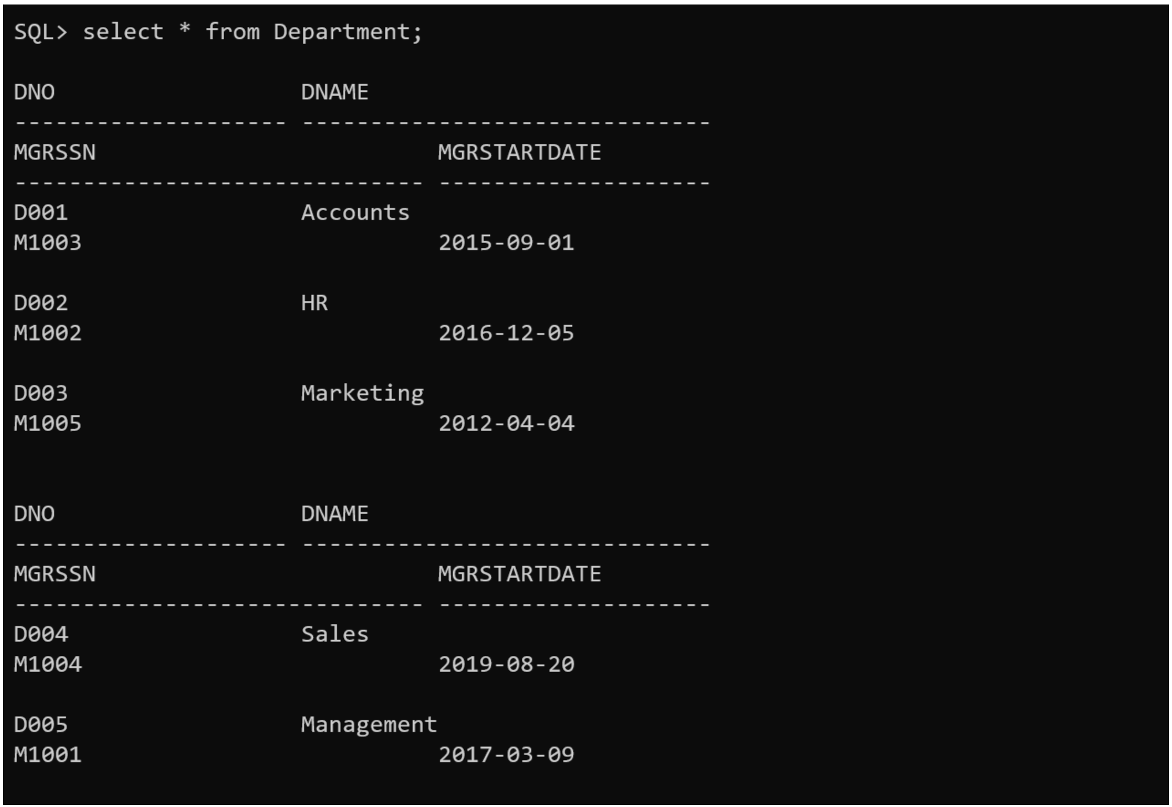
```
SQL> select * from Employee;

SSN                            FNAME
------------------------------ ------------------------------
LNAME
------------------------------
ADDRESS                                            SEX                 SALARY
-------------------------------------------------- --------------- ----------
SUPERSSN                       DNO
------------------------------ --------------------
e1001                          Archana
Suresh
13B,Highway Gardens,Kozhikode                      Female               60000
SP1002                         D001


SSN                            FNAME
------------------------------ ------------------------------
LNAME
------------------------------
ADDRESS                                            SEX                 SALARY
-------------------------------------------------- --------------- ----------
SUPERSSN                       DNO
------------------------------ --------------------
e1002                          Justin
Varghese
Rose Villa,Kochi                                   Male                 50000
SP1001                         D002
```



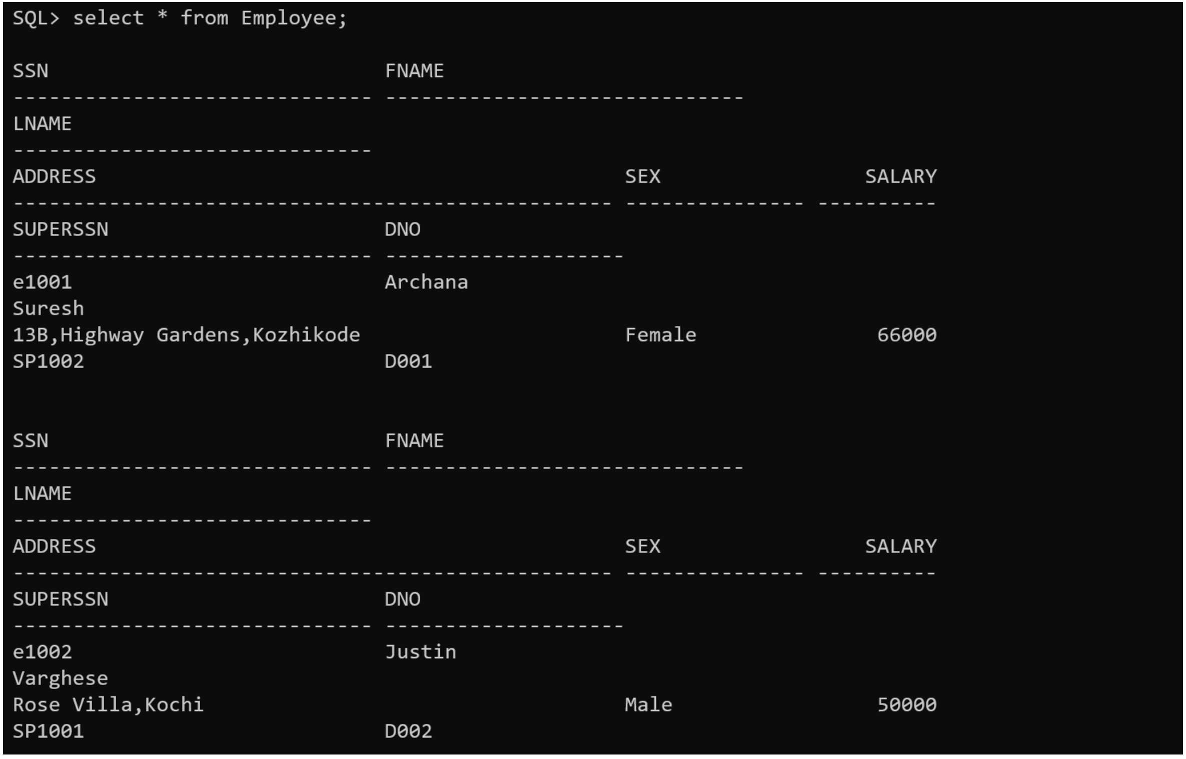
```
SSN                            FNAME
------------------------------ ------------------------------
LNAME
------------------------------
ADDRESS                                            SEX                 SALARY
-------------------------------------------------- --------------- ----------
SUPERSSN                       DNO
------------------------------ --------------------
e1003                          Meera
Kumar
11B,Arcadia Building,Mumbai                        Female               70000
SP1004                         D001


SSN                            FNAME
------------------------------ ------------------------------
LNAME
------------------------------
ADDRESS                                            SEX                 SALARY
-------------------------------------------------- --------------- ----------
SUPERSSN                       DNO
------------------------------ --------------------
e1004                          Kailas
Nath
V3,DD Homes,Bangalore                              Male                 30000
SP1003                         D003
```



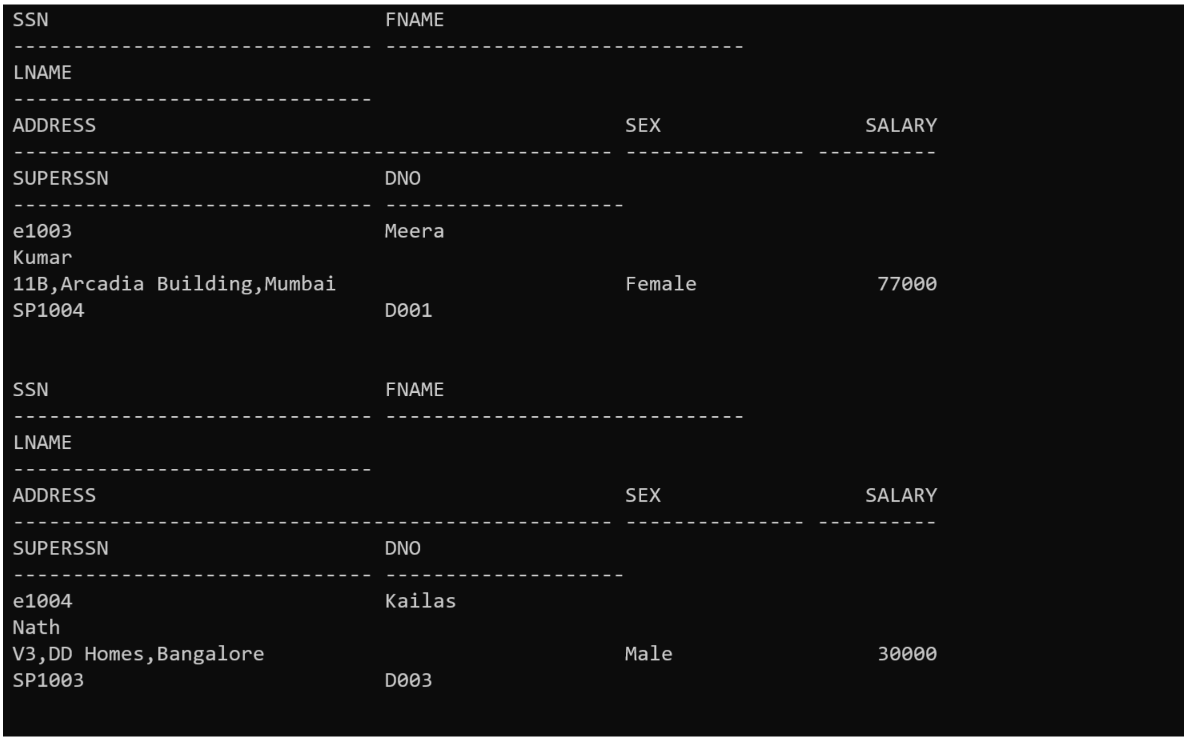
```
SSN                            FNAME
------------------------------ ------------------------------
LNAME
------------------------------
ADDRESS                                            SEX                 SALARY
-------------------------------------------------- --------------- ----------
SUPERSSN                       DNO
------------------------------ --------------------
e1005                          Sara
Khaild
Ashok Nagar,West Delhi                             Female               45000
SP1005                         D004


SSN                            FNAME
------------------------------ ------------------------------
LNAME
------------------------------
ADDRESS                                            SEX                 SALARY
-------------------------------------------------- --------------- ----------
SUPERSSN                       DNO
------------------------------ --------------------
e1006                          Rahul
Ashok
LV Road,Bengaluru                                  Male                 55000
SP1005                         D005
```



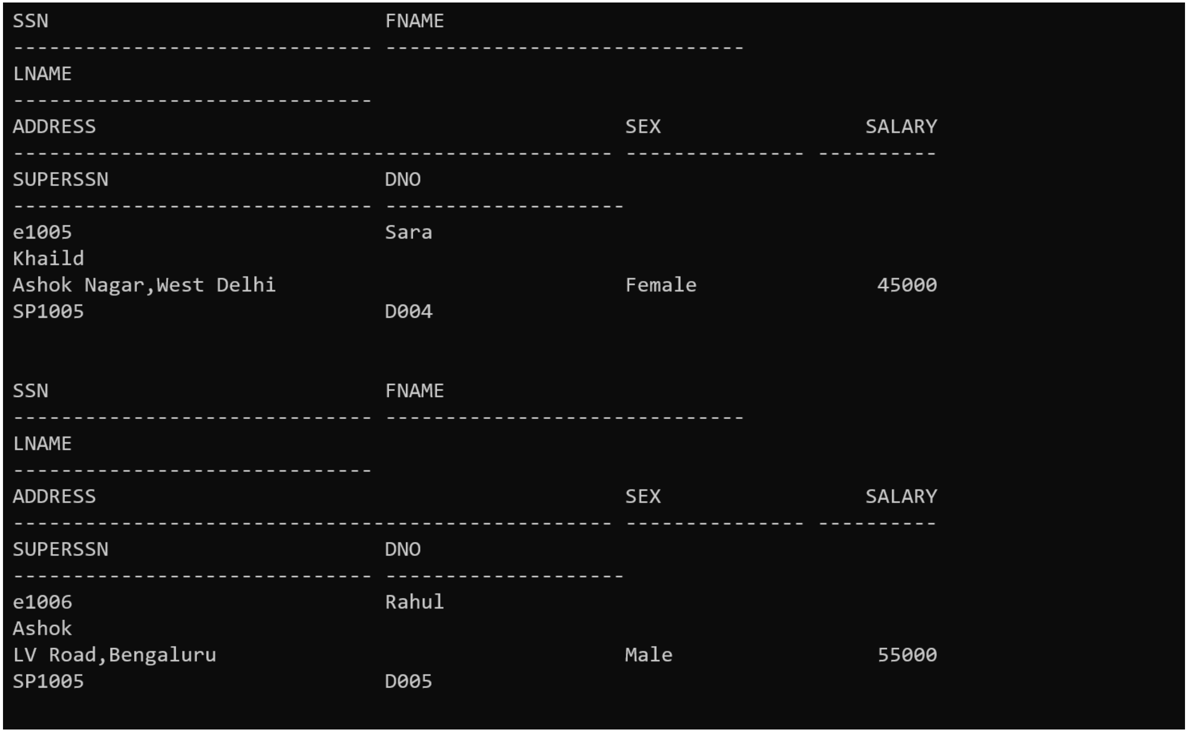
```
SQL> select * from Department;

DNO                  DNAME
-------------------- ------------------------------
MGRSSN                         MGRSTARTDATE
------------------------------ --------------------
D001                 Accounts
M1003                          2015-09-01

D002                 HR
M1002                          2016-12-05

D003                 Marketing
M1005                          2012-04-04


DNO                  DNAME
-------------------- ------------------------------
MGRSSN                         MGRSTARTDATE
------------------------------ --------------------
D004                 Sales
M1004                          2019-08-20

D005                 Management
M1001                          2017-03-09
```



```
SQL> select * from Employee;

SSN                            FNAME
------------------------------ ------------------------------
LNAME
------------------------------
ADDRESS                                            SEX                 SALARY
-------------------------------------------------- --------------- ----------
SUPERSSN                       DNO
------------------------------ --------------------
e1001                          Archana
Suresh
13B,Highway Gardens,Kozhikode                      Female               66000
SP1002                         D001


SSN                            FNAME
------------------------------ ------------------------------
LNAME
------------------------------
ADDRESS                                            SEX                 SALARY
-------------------------------------------------- --------------- ----------
SUPERSSN                       DNO
------------------------------ --------------------
e1002                          Justin
Varghese
Rose Villa,Kochi                                   Male                 50000
SP1001                         D002
```



```
SSN                            FNAME
------------------------------ ------------------------------
LNAME
------------------------------
ADDRESS                                            SEX                 SALARY
-------------------------------------------------- --------------- ----------
SUPERSSN                       DNO
------------------------------ --------------------
e1003                          Meera
Kumar
11B,Arcadia Building,Mumbai                        Female               77000
SP1004                         D001


SSN                            FNAME
------------------------------ ------------------------------
LNAME
------------------------------
ADDRESS                                            SEX                 SALARY
-------------------------------------------------- --------------- ----------
SUPERSSN                       DNO
------------------------------ --------------------
e1004                          Kailas
Nath
V3,DD Homes,Bangalore                              Male                 30000
SP1003                         D003
```



```
SSN                            FNAME
------------------------------ ------------------------------
LNAME
------------------------------
ADDRESS                                            SEX                 SALARY
-------------------------------------------------- --------------- ----------
SUPERSSN                       DNO
------------------------------ --------------------
e1005                          Sara
Khaild
Ashok Nagar,West Delhi                             Female               45000
SP1005                         D004


SSN                            FNAME
------------------------------ ------------------------------
LNAME
------------------------------
ADDRESS                                            SEX                 SALARY
-------------------------------------------------- --------------- ----------
SUPERSSN                       DNO
------------------------------ --------------------
e1006                          Rahul
Ashok
LV Road,Bengaluru                                  Male                 55000
SP1005                         D005
```

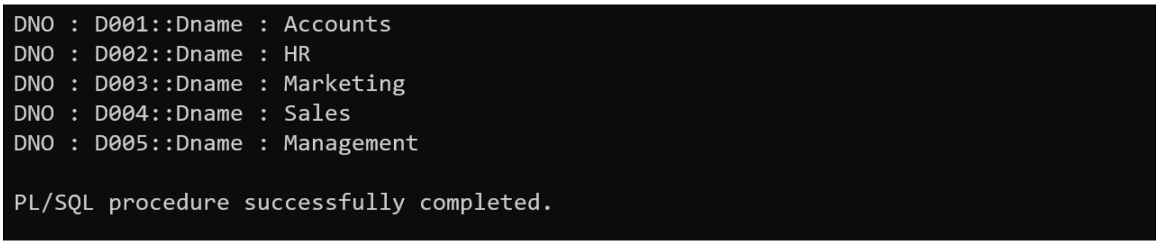
Question : 2

Write a PL/SQL cursor program to retrieve Dno and DName from Department table as per the schema

QUERY

```
SQL> declare cursor department_cur is
  2  select DNO,Dname from Department;
  3  data1 Department.DNO%type;
  4  data2 Department.Dname%type;
  5  begin
  6  open department_cur;
  7  loop
  8  fetch department_cur into data1,data2;
  9  exit when department_cur%notfound;
 10  dbms_output.put_line('DNO : '||data1||'::Dname : '||data2);
 11  end loop;
 12  close department_cur;
 13  end;
 14  /
```

DATABASE TABLES



```
DNO : D001::Dname : Accounts
DNO : D002::Dname : HR
DNO : D003::Dname : Marketing
DNO : D004::Dname : Sales
DNO : D005::Dname : Management

PL/SQL procedure successfully completed.
```

TRIGGER

AIM

Develop PL/SQL program to implement Trigger

Question : 1

Write PL/SQL trigger program to display the salary differences between the old values and new values in the table employee as per the schema

QUERY

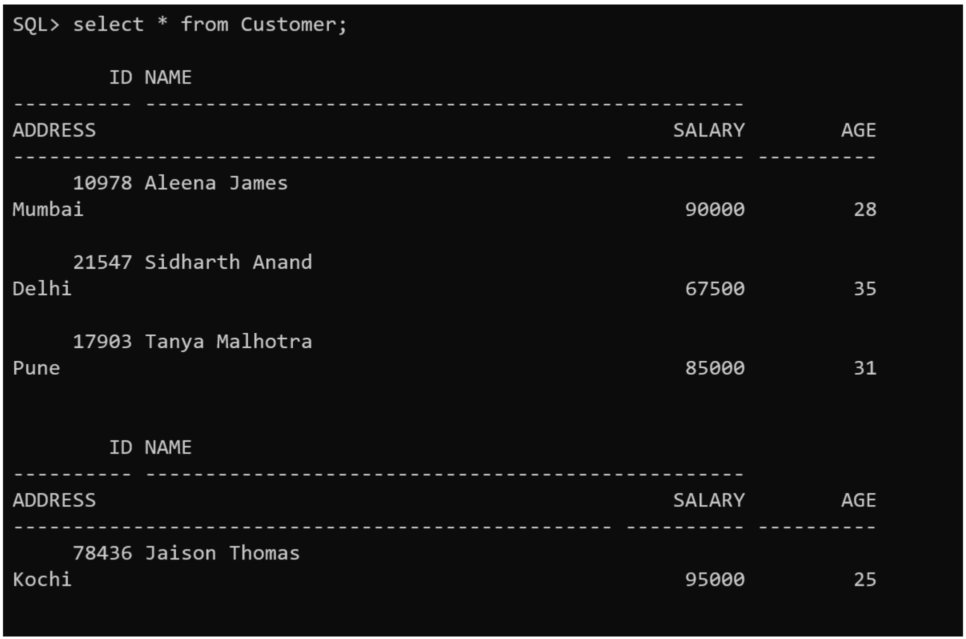
```
SQL> create table Customer(ID Number(38),NAME Varchar(50),ADDRESS Varchar(50),
SALARY Number(38),AGE Number(38));

Table created.

SQL> desc Customer;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 ID                                                 NUMBER(38)
 NAME                                               VARCHAR2(50)
 ADDRESS                                            VARCHAR2(50)
 SALARY                                             NUMBER(38)
 AGE                                                NUMBER(38)

SQL> insert into Customer values(10978,’Aleena James’,’Mumbai’,90000,28);
1 row created.

SQL> insert into Customer values(21547,’Sidharth Anand’,’Delhi’,67500,35);
1 row created.

SQL> insert into Customer values(17903,’Tanya Malhotra’,’Pune’,85000,31);
1 row created.

SQL> insert into Customer values(78436,’Jaison Thomas’,’Kochi’,95000,25);
1 row created.
```

```
SQL> select * from Customer;

        ID NAME
---------- --------------------------------------------------
ADDRESS                                                SALARY        AGE
-------------------------------------------------- ---------- ----------
     10978 Aleena James
Mumbai                                                  90000         28


     21547 Sidharth Anand
Delhi                                                   67500         35


     17903 Tanya Malhotra
Pune                                                    85000         31



        ID NAME
---------- --------------------------------------------------
ADDRESS                                                SALARY        AGE
-------------------------------------------------- ---------- ----------
     78436 Jaison Thomas
Kochi                                                   95000         25

SQL> edit @salary_difference.sql
CREATE OR REPLACE TRIGGER display_salary_changes
BEFORE DELETE OR INSERT OR UPDATE ON customer
FOR EACH ROW
WHEN (NEW.ID > 0 )
DECLARE
sal_diff number;
BEGIN
sal_diff :=  :NEW.salary  -  :OLD.salary;
dbms_output.put_line('Old salary :  '|| :OLD.salary);
dbms_output.put_line('New salary :  '|| :NEW.salary);
dbms_output.put_line('Salary difference :  '|| sal_diff);
END;
```

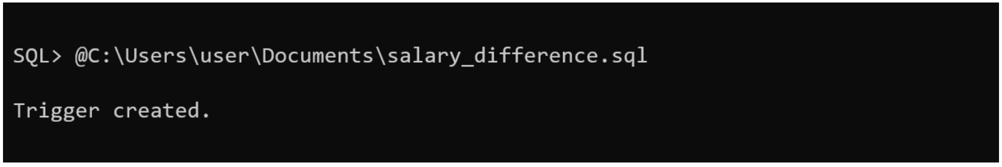
```
/
```

```
SQL> edit @m.sql

DECLARE
BEGIN
UPDATE Customer
SET SALARY = SALARY + 5000;
END;
/
```

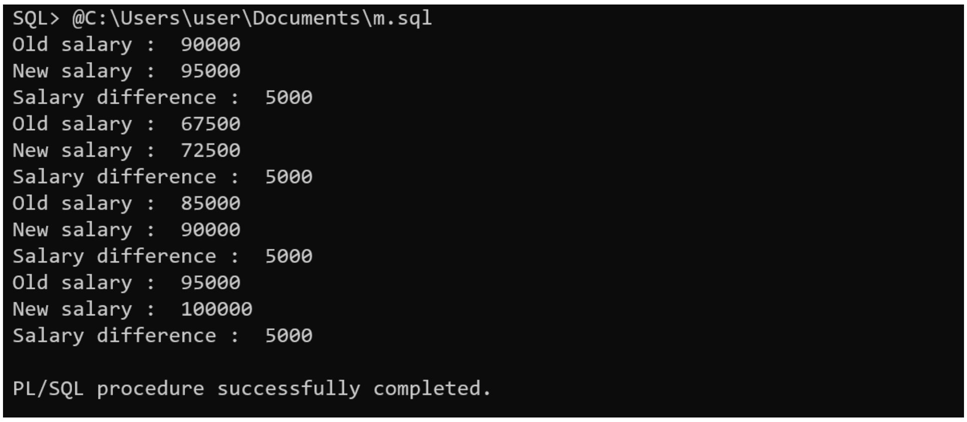
DATABASE TABLES



```
SQL> select * from Customer;

        ID NAME
---------- --------------------------------------------------
ADDRESS                                                SALARY        AGE
-------------------------------------------------- ---------- ----------
     10978 Aleena James
Mumbai                                                  90000         28

     21547 Sidharth Anand
Delhi                                                   67500         35

     17903 Tanya Malhotra
Pune                                                    85000         31


        ID NAME
---------- --------------------------------------------------
ADDRESS                                                SALARY        AGE
-------------------------------------------------- ---------- ----------
     78436 Jaison Thomas
Kochi                                                   95000         25
```



```
SQL> @C:\Users\user\Documents\salary_difference.sql

Trigger created.
```



```
SQL> @C:\Users\user\Documents\m.sql
Old salary :  90000
New salary :  95000
Salary difference :  5000
Old salary :  67500
New salary :  72500
Salary difference :  5000
Old salary :  85000
New salary :  90000
Salary difference :  5000
Old salary :  95000
New salary :  100000
Salary difference :  5000

PL/SQL procedure successfully completed.
```

Question : 2

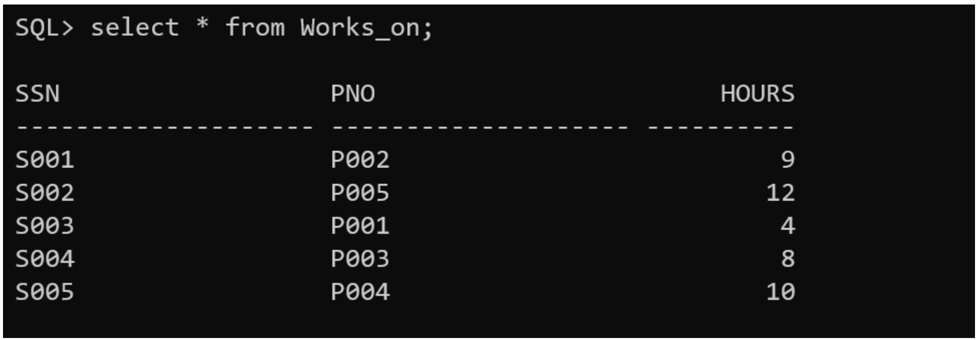
Write PL/SQL trigger program to display the hour differences between the old values and new values in the table Works_on as per the schema

QUERY

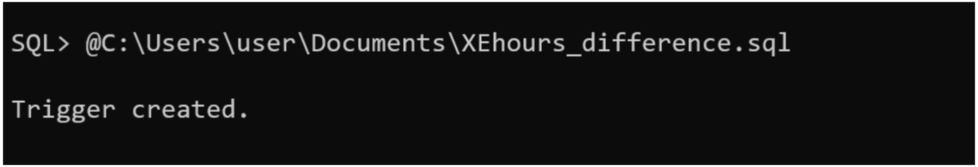
```
SQL> create table Works_on(SSN varchar(20),PNO varchar(20),Hours int);

Table created.

SQL> insert into Works_on values(’S001’,’P002’,9);

1 row created.

SQL> insert into Works_on values(’S002’,’P005’,12);

1 row created.

SQL> insert into Works_on values(’S003’,’P001’,4);

1 row created.

SQL> insert into Works_on values(’S004’,’P003’,8);

1 row created.

SQL> insert into Works_on values(’S005’,’P004’,10);

1 row created.


SQL> select * from Works_on;

SSN                  PNO                       HOURS
-------------------- -------------------- ----------
S001                 P002                          9
S002                 P005                         12
S003                 P001                          4
```

```
S004                 P003                            8
S005                 P004                           10

SQL> set serveroutput on
SQL> edit@hours_difference.sql



CREATE OR REPLACE TRIGGER
display_hour_updates
BEFORE DELETE OR INSERT OR UPDATE
ON Works_on
FOR EACH ROW
WHEN(NEW.HOURS > 0)
DECLARE
     hour_diff number;
BEGIN
     hour_diff := :NEW.HOURS - :OLD.HOURS;
     dbms_output.put_line(’Old Hours : ’|| : OLD.HOURS);
     dbms_output.put_line(’New Hours : ’|| : NEW.HOURS);
     dbms_output.put_line(’Hour difference : ’|| hour_diff);
END;
/

SQL> edit@H.sql

DECLARE
BEGIN
        UPDATE Works_on
        SET Hours = Hours + 5;
END;
/
```

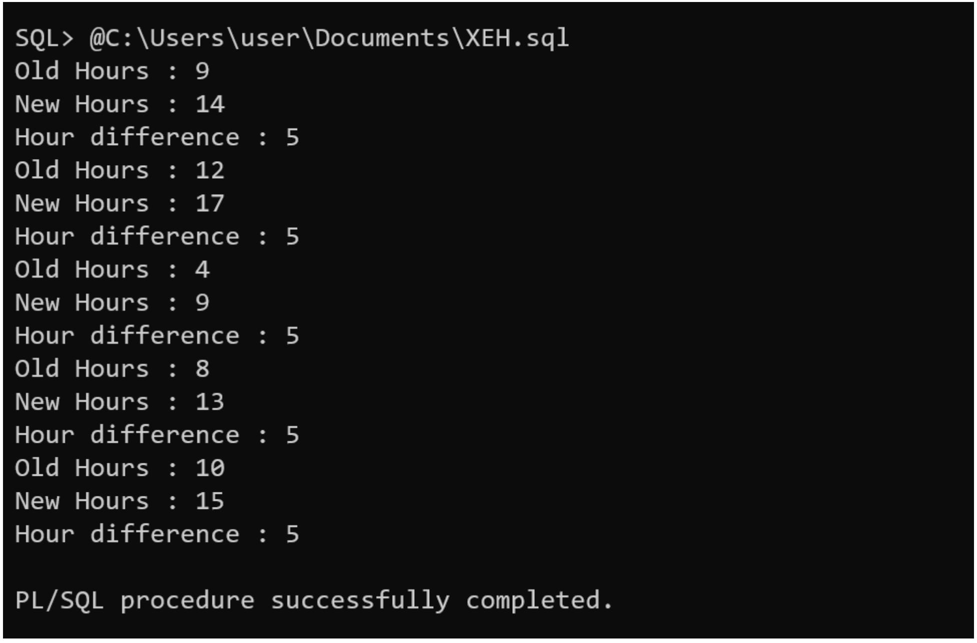
DATABASE TABLES



```
SQL> select * from Works_on;

SSN                  PNO                       HOURS
-------------------- -------------------- ----------
S001                 P002                          9
S002                 P005                         12
S003                 P001                          4
S004                 P003                          8
S005                 P004                         10
```



```
SQL> @C:\Users\user\Documents\XEhours_difference.sql

Trigger created.
```



```
SQL> @C:\Users\user\Documents\XEH.sql
Old Hours : 9
New Hours : 14
Hour difference : 5
Old Hours : 12
New Hours : 17
Hour difference : 5
Old Hours : 4
New Hours : 9
Hour difference : 5
Old Hours : 8
New Hours : 13
Hour difference : 5
Old Hours : 10
New Hours : 15
Hour difference : 5

PL/SQL procedure successfully completed.
```

TCL

AIM

Develop SQL Queries to understand the concept of Transaction Control Language

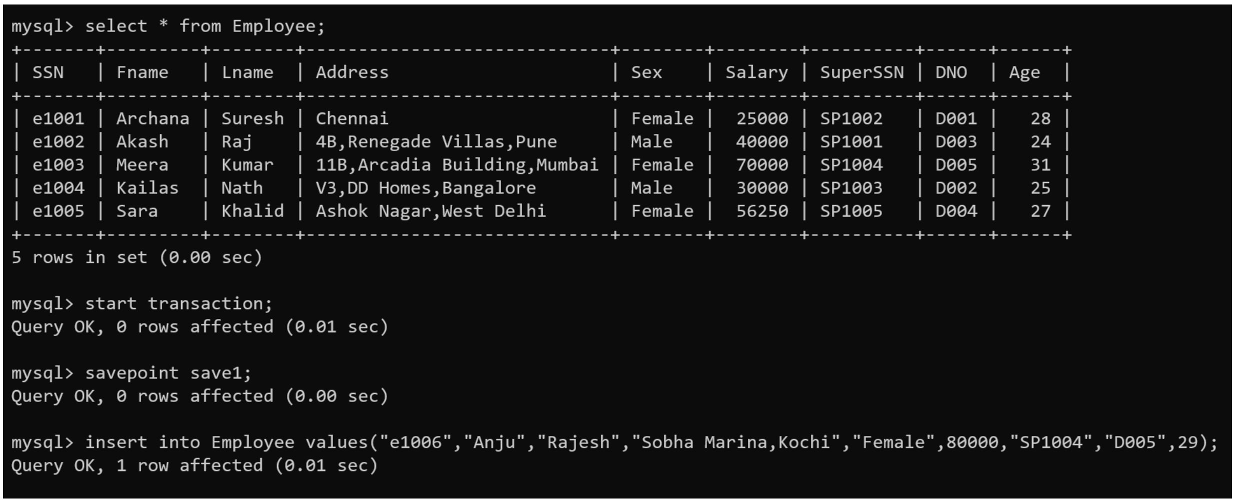
Question : 1

Creating Check points in the program

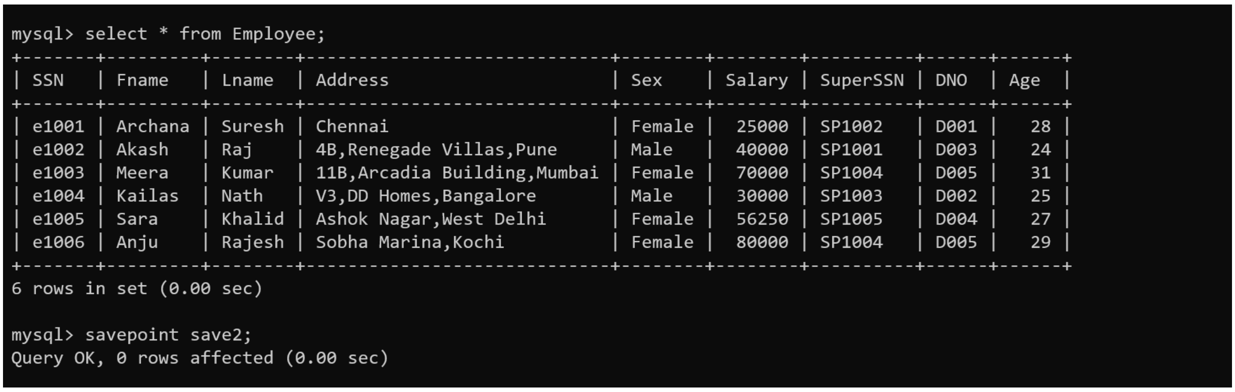
QUERY

```
mysql> start transaction;
Query OK, 0 rows affected (0.01 sec)

mysql> savepoint save1;
Query OK, 0 rows affected (0.00 sec)

mysql> insert into Employee values("e1006","Anju","Rajesh","Sobha Marina,
Kochi","Female",80000,"SP1004","D005",29);
Query OK, 1 row affected (0.01 sec)

mysql> savepoint save2;
Query OK, 0 rows affected (0.00 sec)
```

DATABASE TABLES



```
mysql> select * from Employee;
+-------+---------+--------+-----------------------------+--------+--------+----------+------+------+
| SSN   | Fname   | Lname  | Address                     | Sex    | Salary | SuperSSN | DNO  | Age  |
+-------+---------+--------+-----------------------------+--------+--------+----------+------+------+
| e1001 | Archana | Suresh | Chennai                     | Female |  25000 | SP1002   | D001 |   28 |
| e1002 | Akash   | Raj    | 4B,Renegade Villas,Pune     | Male   |  40000 | SP1001   | D003 |   24 |
| e1003 | Meera   | Kumar  | 11B,Arcadia Building,Mumbai | Female |  70000 | SP1004   | D005 |   31 |
| e1004 | Kailas  | Nath   | V3,DD Homes,Bangalore       | Male   |  30000 | SP1003   | D002 |   25 |
| e1005 | Sara    | Khalid | Ashok Nagar,West Delhi      | Female |  56250 | SP1005   | D004 |   27 |
+-------+---------+--------+-----------------------------+--------+--------+----------+------+------+
5 rows in set (0.00 sec)

mysql> start transaction;
Query OK, 0 rows affected (0.01 sec)

mysql> savepoint save1;
Query OK, 0 rows affected (0.00 sec)

mysql> insert into Employee values("e1006","Anju","Rajesh","Sobha Marina,Kochi","Female",80000,"SP1004","D005",29);
Query OK, 1 row affected (0.01 sec)
```



```
mysql> select * from Employee;
+-------+---------+--------+----------------------------+--------+--------+---------+------+------+
| SSN   | Fname   | Lname  | Address                    | Sex    | Salary | SuperSSN | DNO  | Age  |
+-------+---------+--------+----------------------------+--------+--------+---------+------+------+
| e1001 | Archana | Suresh | Chennai                    | Female |  25000 | SP1002   | D001 |   28 |
| e1002 | Akash   | Raj    | 4B,Renegade Villas,Pune    | Male   |  40000 | SP1001   | D003 |   24 |
| e1003 | Meera   | Kumar  | 11B,Arcadia Building,Mumbai | Female |  70000 | SP1004   | D005 |   31 |
| e1004 | Kailas  | Nath   | V3,DD Homes,Bangalore      | Male   |  30000 | SP1003   | D002 |   25 |
| e1005 | Sara    | Khalid | Ashok Nagar,West Delhi     | Female |  56250 | SP1005   | D004 |   27 |
| e1006 | Anju    | Rajesh | Sobha Marina,Kochi         | Female |  80000 | SP1004   | D005 |   29 |
+-------+---------+--------+----------------------------+--------+--------+---------+------+------+
6 rows in set (0.00 sec)

mysql> savepoint save2;
Query OK, 0 rows affected (0.00 sec)
```

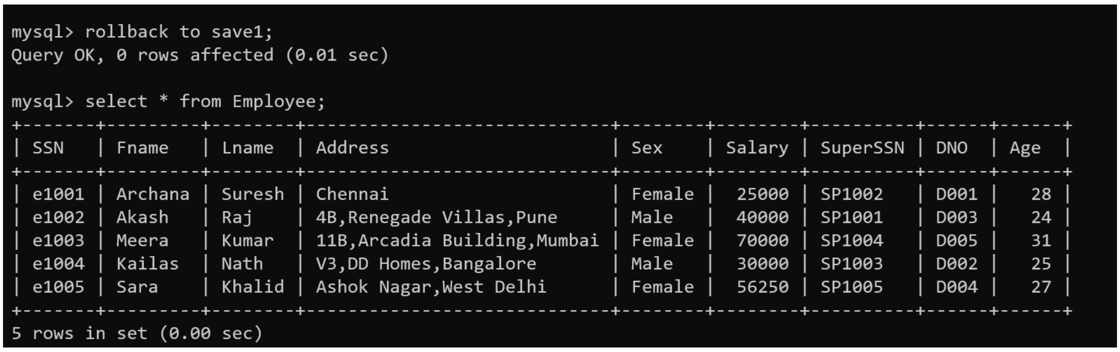
Question : 2

Rollback to a previously created Checkpoint in the program

QUERY

```
mysql> rollback to save1;
Query OK, 0 rows affected (0.01 sec)
```

DATABASE TABLES



```
mysql> rollback to save1;
Query OK, 0 rows affected (0.01 sec)

mysql> select * from Employee;
+-------+---------+--------+----------------------------+--------+--------+---------+------+------+
| SSN   | Fname   | Lname  | Address                    | Sex    | Salary | SuperSSN | DNO  | Age  |
+-------+---------+--------+----------------------------+--------+--------+---------+------+------+
| e1001 | Archana | Suresh | Chennai                    | Female |  25000 | SP1002   | D001 |   28 |
| e1002 | Akash   | Raj    | 4B,Renegade Villas,Pune    | Male   |  40000 | SP1001   | D003 |   24 |
| e1003 | Meera   | Kumar  | 11B,Arcadia Building,Mumbai | Female |  70000 | SP1004   | D005 |   31 |
| e1004 | Kailas  | Nath   | V3,DD Homes,Bangalore      | Male   |  30000 | SP1003   | D002 |   25 |
| e1005 | Sara    | Khalid | Ashok Nagar,West Delhi     | Female |  56250 | SP1005   | D004 |   27 |
+-------+---------+--------+----------------------------+--------+--------+---------+------+------+
5 rows in set (0.00 sec)
```

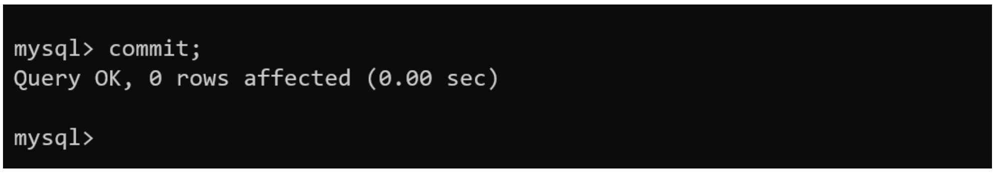
Question : 3

Commit the program

QUERY

```
mysql> commit;
Query OK, 0 rows affected (0.00 sec)
```

DATABASE TABLES



```
mysql> commit;
Query OK, 0 rows affected (0.00 sec)

mysql>
```

MongoDB

AIM

Develop program to perform operations in MongoDB

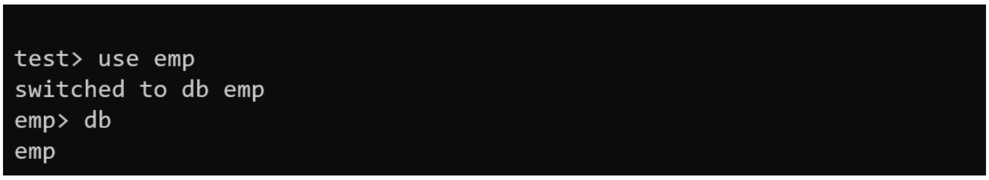
Question : 1

Create a database emp

QUERY

```
test> use emp
```

DATABASE TABLES



```
test> use emp
switched to db emp
emp> db
emp
```

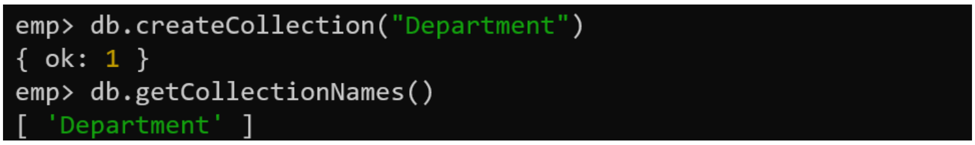
Question : 2

Create new Collection

QUERY

```
emp> db.createCollection("Department")
{ ok: 1 }
```

DATABASE TABLES



```
emp> db.createCollection("Department")
{ ok: 1 }
emp> db.getCollectionNames()
[ 'Department' ]
```

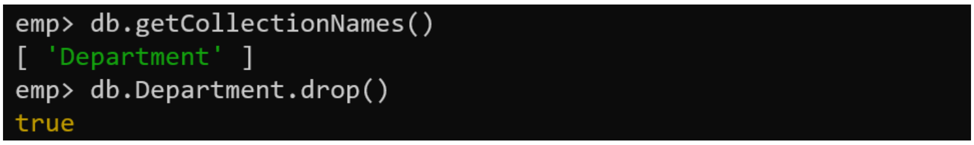
Question : 3

Check the collection list created and drop collection

QUERY

```
emp> db.getCollectionNames()
emp> db.Department.drop()
```

DATABASE TABLES



```
emp> db.getCollectionNames()
[ 'Department' ]
emp> db.Department.drop()
true
```

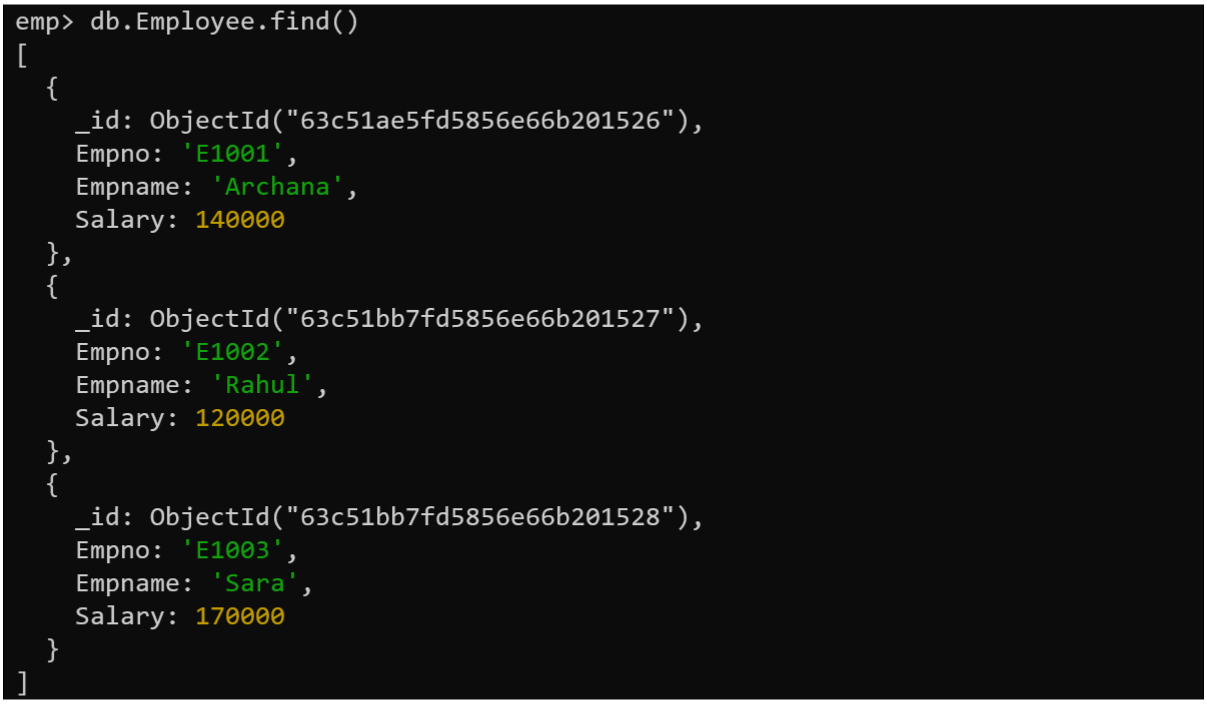
Question : 4

Insert document in selected Collection

QUERY

```
emp> db.Employee.insertOne({"Empno" : "E1001" , "Empname" : "Archana" ,
"Salary" : 140000})
{
  acknowledged: true,
  insertedId: ObjectId("63c51ae5fd5856e66b201526")
}
emp> try{ db.Employee.insertMany([{"Empno" : "E1002" , "Empname" : "Rahul" ,
"Salary" : 120000},{"Empno" : "E1003" , "Empname" : "Sara" , "Salary" : 170000}]);
... }
... catch(e){
... print(e);
... }
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId("63c51bb7fd5856e66b201527"),
    '1': ObjectId("63c51bb7fd5856e66b201528")
  }
}
```

DATABASE TABLES



```
emp> db.Employee.find()
[
  {
    _id: ObjectId("63c51ae5fd5856e66b201526"),
    Empno: 'E1001',
    Empname: 'Archana',
    Salary: 140000
  },
  {
    _id: ObjectId("63c51bb7fd5856e66b201527"),
    Empno: 'E1002',
    Empname: 'Rahul',
    Salary: 120000
  },
  {
    _id: ObjectId("63c51bb7fd5856e66b201528"),
    Empno: 'E1003',
    Empname: 'Sara',
    Salary: 170000
  }
]
```

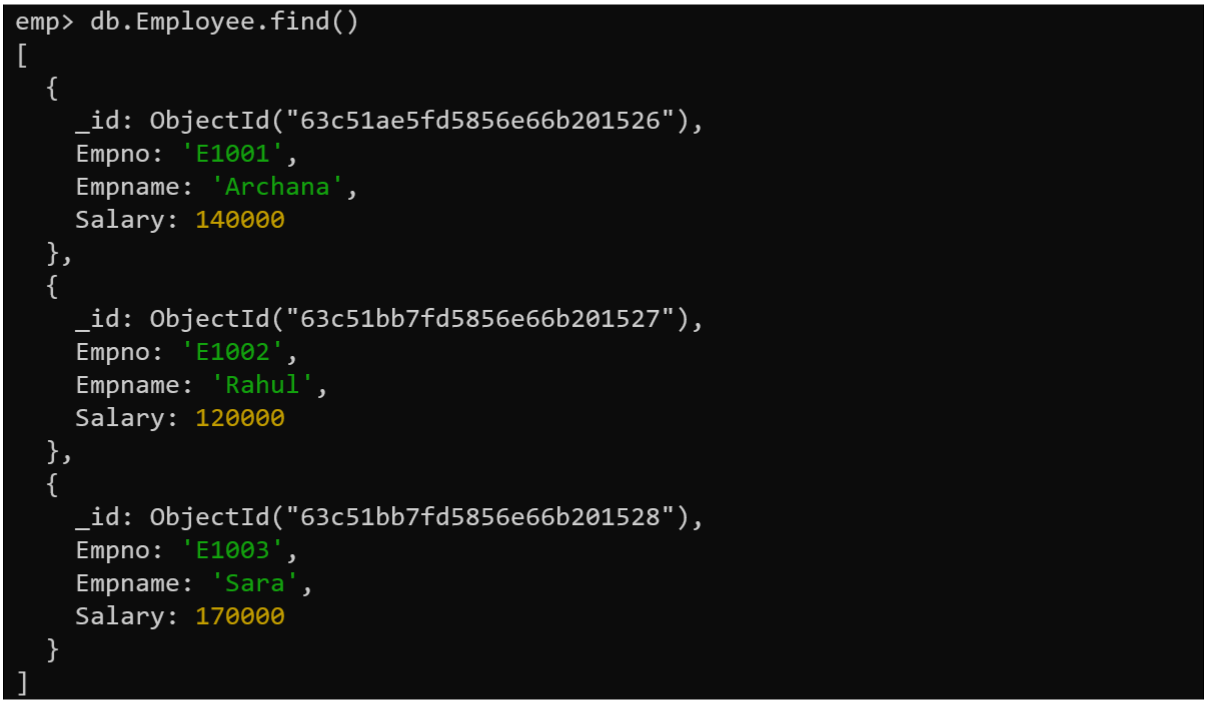
Question : 5

To get the list documents in Collection

QUERY

```
emp> db.Employee.find()
```

DATABASE TABLES



```
emp> db.Employee.find()
[
  {
    _id: ObjectId("63c51ae5fd5856e66b201526"),
    Empno: 'E1001',
    Empname: 'Archana',
    Salary: 140000
  },
  {
    _id: ObjectId("63c51bb7fd5856e66b201527"),
    Empno: 'E1002',
    Empname: 'Rahul',
    Salary: 120000
  },
  {
    _id: ObjectId("63c51bb7fd5856e66b201528"),
    Empno: 'E1003',
    Empname: 'Sara',
    Salary: 170000
  }
]
```

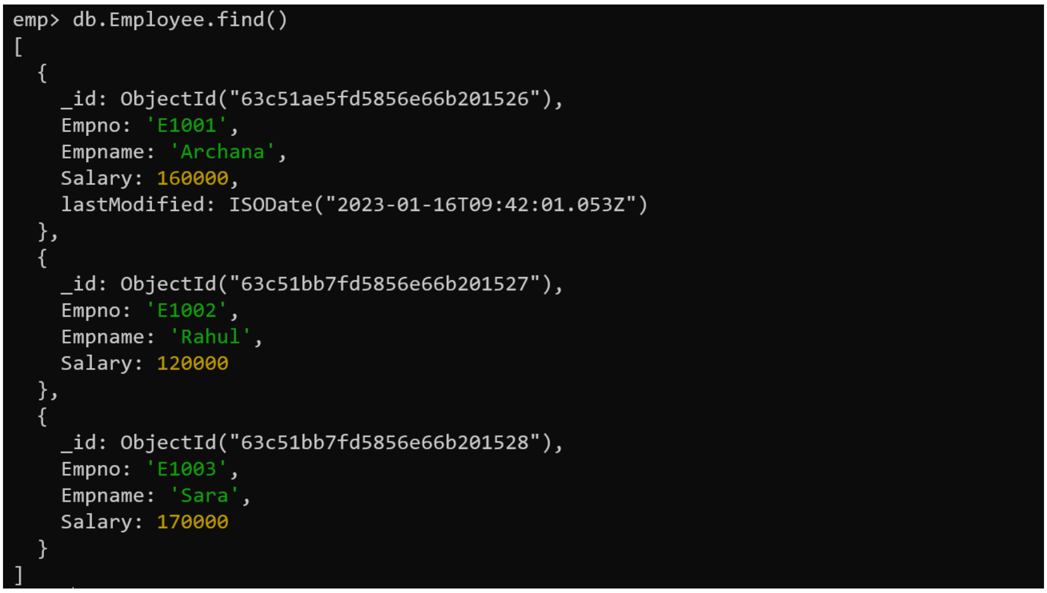
Question : 6

Update the document in Collection

QUERY

```
emp> db.Employee.updateOne({"Empno" : "E1001"},
... {
... $set : {"Salary" : 160000},
... $currentDate : {lastModified : true}
... }
... )
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

DATABASE TABLES



```
emp> db.Employee.find()
[
  {
    _id: ObjectId("63c51ae5fd5856e66b201526"),
    Empno: 'E1001',
    Empname: 'Archana',
    Salary: 160000,
    lastModified: ISODate("2023-01-16T09:42:01.053Z")
  },
  {
    _id: ObjectId("63c51bb7fd5856e66b201527"),
    Empno: 'E1002',
    Empname: 'Rahul',
    Salary: 120000
  },
  {
    _id: ObjectId("63c51bb7fd5856e66b201528"),
    Empno: 'E1003',
    Empname: 'Sara',
    Salary: 170000
  }
]
```

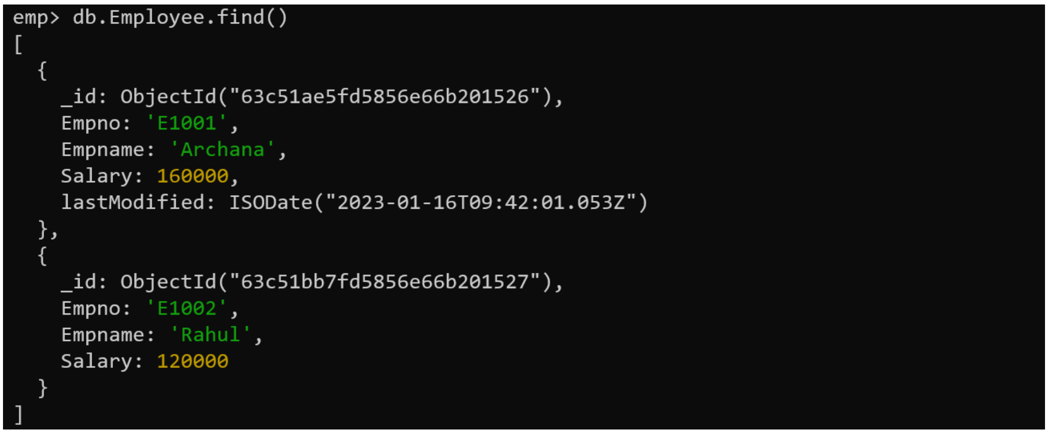
Question : 7

Delete the document in selected Collection

QUERY

```
emp> db.Employee.deleteOne({"Empname" : "Sara"});
{ acknowledged: true, deletedCount: 1 }
```

DATABASE TABLES



```
emp> db.Employee.find()
[
  {
    _id: ObjectId("63c51ae5fd5856e66b201526"),
    Empno: 'E1001',
    Empname: 'Archana',
    Salary: 160000,
    lastModified: ISODate("2023-01-16T09:42:01.053Z")
  },
  {
    _id: ObjectId("63c51bb7fd5856e66b201527"),
    Empno: 'E1002',
    Empname: 'Rahul',
    Salary: 120000
  }
]
```

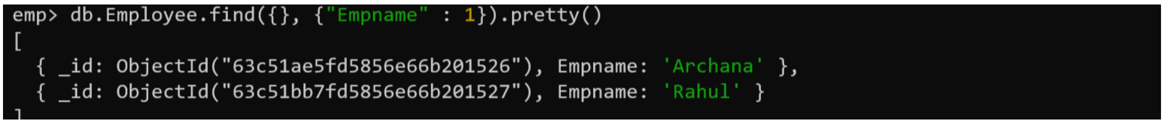
Question : 8

Projection using find() method

QUERY

```
emp> db.Employee.find({}, {"Empname" : 1}).pretty()
```

DATABASE TABLES



```
emp> db.Employee.find({}, {"Empname" : 1}).pretty()
[
  { _id: ObjectId("63c51ae5fd5856e66b201526"), Empname: 'Archana' },
  { _id: ObjectId("63c51bb7fd5856e66b201527"), Empname: 'Rahul' }
]
```

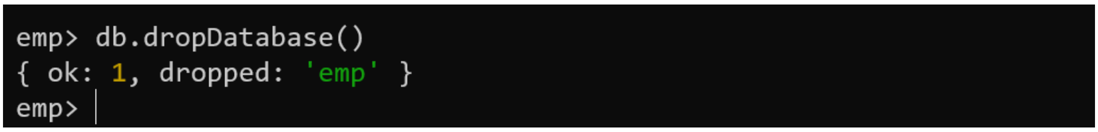
Question : 9

Drop database emp

QUERY

```
emp> db.dropDatabase()
```

DATABASE TABLES



```
emp> db.dropDatabase()
{ ok: 1, dropped: 'emp' }
emp>
```

GraphQL

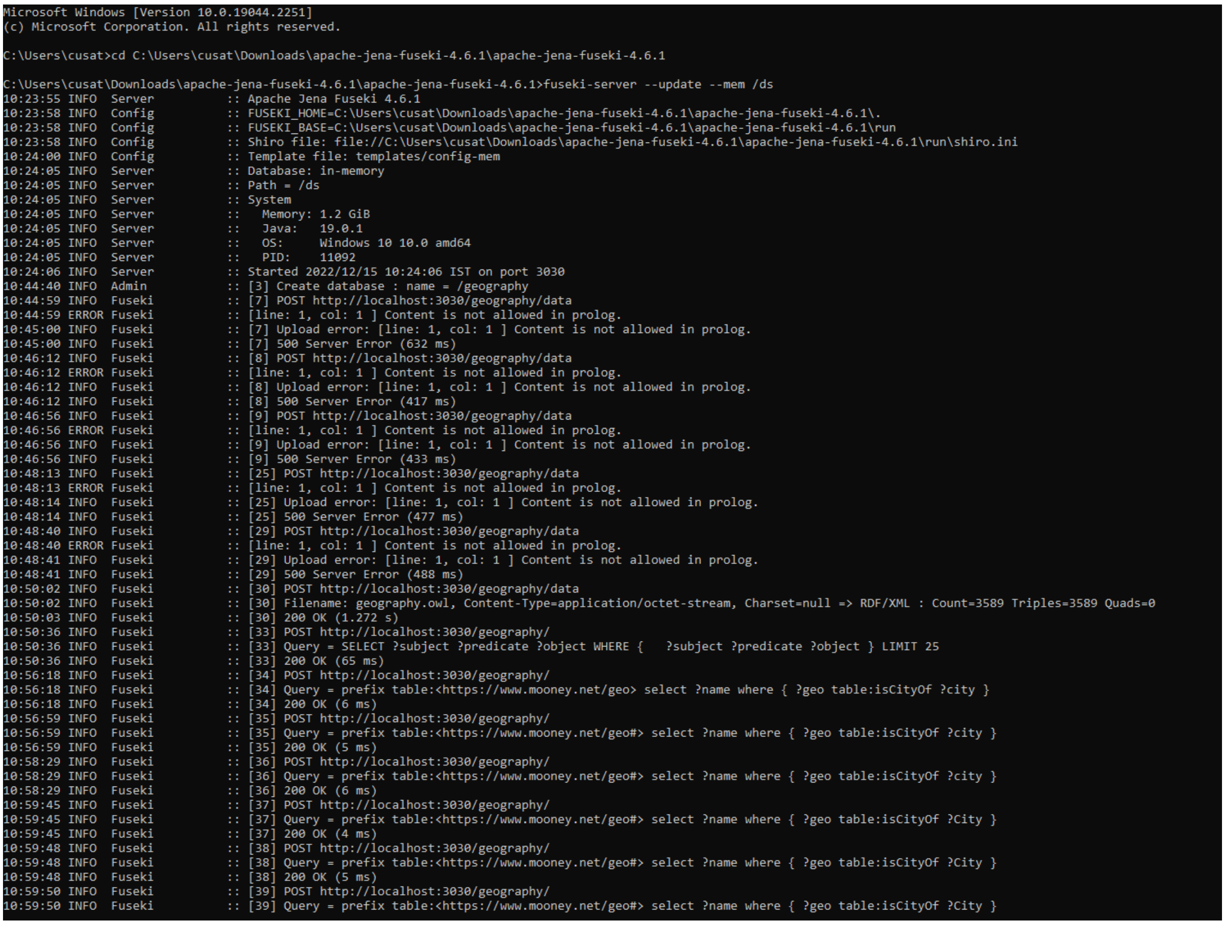
AIM

Develop PL/SQL program to implement GraphQL

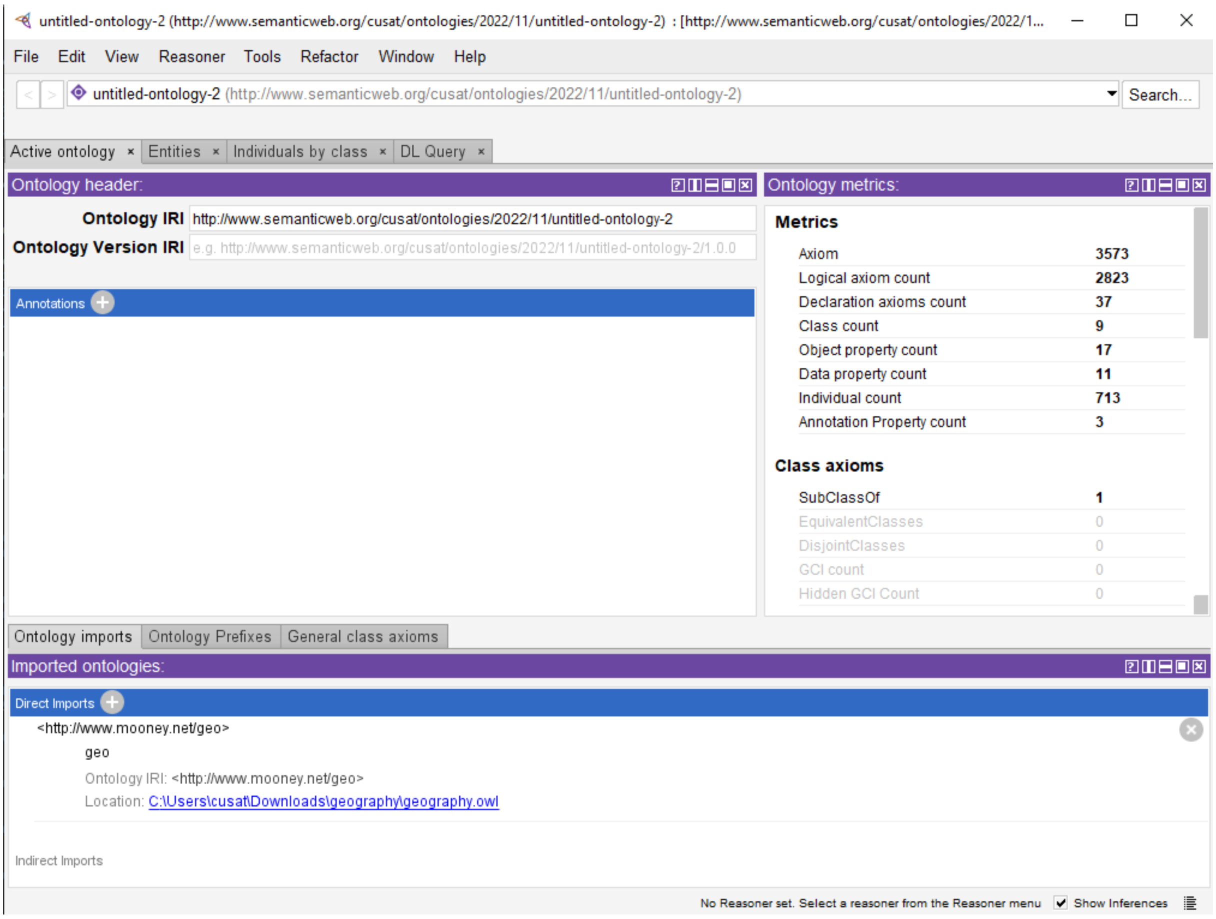
Question : 1

Develop a GraphQL program to perform different operations in created ontology

DATABASE TABLES



```
Microsoft Windows [Version 10.0.19044.2251]
(c) Microsoft Corporation. All rights reserved.

C:\Users\cusat>cd C:\Users\cusat\Downloads\apache-jena-fuseki-4.6.1\apache-jena-fuseki-4.6.1

C:\Users\cusat\Downloads\apache-jena-fuseki-4.6.1\apache-jena-fuseki-4.6.1>fuseki-server --update --mem /ds
10:23:55 INFO  Server          :: Apache Jena Fuseki 4.6.1
10:23:58 INFO  Config          :: FUSEKI_HOME=C:\Users\cusat\Downloads\apache-jena-fuseki-4.6.1\apache-jena-fuseki-4.6.1\.
10:23:58 INFO  Config          :: FUSEKI_BASE=C:\Users\cusat\Downloads\apache-jena-fuseki-4.6.1\apache-jena-fuseki-4.6.1\run
10:23:58 INFO  Config          :: Shiro file: file://C:\Users\cusat\Downloads\apache-jena-fuseki-4.6.1\apache-jena-fuseki-4.6.1\run\shiro.ini
10:24:00 INFO  Config          :: Template file: templates/config-mem
10:24:05 INFO  Server          :: Database: in-memory
10:24:05 INFO  Server          :: Path = /ds
10:24:05 INFO  Server          :: System
10:24:05 INFO  Server          ::   Memory: 1.2 GiB
10:24:05 INFO  Server          ::   Java:   19.0.1
10:24:05 INFO  Server          ::   OS:     Windows 10 10.0 amd64
10:24:05 INFO  Server          ::   PID:    11092
10:24:06 INFO  Server          :: Started 2022/12/15 10:24:06 IST on port 3030
10:44:40 INFO  Admin           :: [3] Create database : name = /geography
10:44:59 INFO  Fuseki          :: [7] POST http://localhost:3030/geography/data
10:44:59 ERROR Fuseki          :: [line: 1, col: 1 ] Content is not allowed in prolog.
10:45:00 INFO  Fuseki          :: [7] Upload error: [line: 1, col: 1 ] Content is not allowed in prolog.
10:45:00 INFO  Fuseki          :: [7] 500 Server Error (632 ms)
10:46:12 INFO  Fuseki          :: [8] POST http://localhost:3030/geography/data
10:46:12 ERROR Fuseki          :: [line: 1, col: 1 ] Content is not allowed in prolog.
10:46:12 INFO  Fuseki          :: [8] Upload error: [line: 1, col: 1 ] Content is not allowed in prolog.
10:46:12 INFO  Fuseki          :: [8] 500 Server Error (417 ms)
10:46:56 INFO  Fuseki          :: [9] POST http://localhost:3030/geography/data
10:46:56 ERROR Fuseki          :: [line: 1, col: 1 ] Content is not allowed in prolog.
10:46:56 INFO  Fuseki          :: [9] Upload error: [line: 1, col: 1 ] Content is not allowed in prolog.
10:46:56 INFO  Fuseki          :: [9] 500 Server Error (433 ms)
10:48:13 INFO  Fuseki          :: [25] POST http://localhost:3030/geography/data
10:48:13 ERROR Fuseki          :: [line: 1, col: 1 ] Content is not allowed in prolog.
10:48:14 INFO  Fuseki          :: [25] Upload error: [line: 1, col: 1 ] Content is not allowed in prolog.
10:48:14 INFO  Fuseki          :: [25] 500 Server Error (477 ms)
10:48:40 INFO  Fuseki          :: [29] POST http://localhost:3030/geography/data
10:48:40 ERROR Fuseki          :: [line: 1, col: 1 ] Content is not allowed in prolog.
10:48:41 INFO  Fuseki          :: [29] Upload error: [line: 1, col: 1 ] Content is not allowed in prolog.
10:48:41 INFO  Fuseki          :: [29] 500 Server Error (488 ms)
10:50:02 INFO  Fuseki          :: [30] POST http://localhost:3030/geography/data
10:50:02 INFO  Fuseki          :: [30] Filename: geography.owl, Content-Type=application/octet-stream, Charset=null => RDF/XML : Count=3589 Triples=3589 Quads=0
10:50:03 INFO  Fuseki          :: [30] 200 OK (1.272 s)
10:50:36 INFO  Fuseki          :: [33] POST http://localhost:3030/geography/
10:50:36 INFO  Fuseki          :: [33] Query = SELECT ?subject ?predicate ?object WHERE {   ?subject ?predicate ?object } LIMIT 25
10:50:36 INFO  Fuseki          :: [33] 200 OK (65 ms)
10:56:18 INFO  Fuseki          :: [34] POST http://localhost:3030/geography/
10:56:18 INFO  Fuseki          :: [34] Query = prefix table:<https://www.mooney.net/geo> select ?name where { ?geo table:isCityOf ?city }
10:56:18 INFO  Fuseki          :: [34] 200 OK (6 ms)
10:56:59 INFO  Fuseki          :: [35] POST http://localhost:3030/geography/
10:56:59 INFO  Fuseki          :: [35] Query = prefix table:<https://www.mooney.net/geo#> select ?name where { ?geo table:isCityOf ?city }
10:56:59 INFO  Fuseki          :: [35] 200 OK (5 ms)
10:58:29 INFO  Fuseki          :: [36] POST http://localhost:3030/geography/
10:58:29 INFO  Fuseki          :: [36] Query = prefix table:<https://www.mooney.net/geo#> select ?name where { ?geo table:isCityOf ?city }
10:58:29 INFO  Fuseki          :: [36] 200 OK (6 ms)
10:59:45 INFO  Fuseki          :: [37] POST http://localhost:3030/geography/
10:59:45 INFO  Fuseki          :: [37] Query = prefix table:<https://www.mooney.net/geo#> select ?name where { ?geo table:isCityOf ?City }
10:59:45 INFO  Fuseki          :: [37] 200 OK (4 ms)
10:59:48 INFO  Fuseki          :: [38] POST http://localhost:3030/geography/
10:59:48 INFO  Fuseki          :: [38] Query = prefix table:<https://www.mooney.net/geo#> select ?name where { ?geo table:isCityOf ?City }
10:59:48 INFO  Fuseki          :: [38] 200 OK (5 ms)
10:59:50 INFO  Fuseki          :: [39] POST http://localhost:3030/geography/
10:59:50 INFO  Fuseki          :: [39] Query = prefix table:<https://www.mooney.net/geo#> select ?name where { ?geo table:isCityOf ?City }
```



untitled-ontology-2 (http://www.semanticweb.org/cusat/ontologies/2022/11/untitled-ontology-2) : [http://www.semanticweb.org/cusat/ontologies/2022/1...
File Edit View Reasoner Tools Refactor Window Help
untitled-ontology-2 (http://www.semanticweb.org/cusat/ontologies/2022/11/untitled-ontology-2)
Search...
Active ontology × Entities × Individuals by class × DL Query ×
Ontology header:
Ontology metrics:
Annotations
Ontology imports Ontology Prefixes General class axioms
Imported ontologies:
Direct Imports
Indirect Imports

Ontology IRI: http://www.semanticweb.org/cusat/ontologies/2022/11/untitled-ontology-2

Ontology Version IRI: e.g. http://www.semanticweb.org/cusat/ontologies/2022/11/untitled-ontology-2/1.0.0

Metrics

Axiom	3573
Logical axiom count	2823
Declaration axioms count	37
Class count	9
Object property count	17
Data property count	11
Individual count	713
Annotation Property count	3

Class axioms

SubClassOf	1
EquivalentClasses	0
DisjointClasses	0
GCI count	0
Hidden GCI Count	0

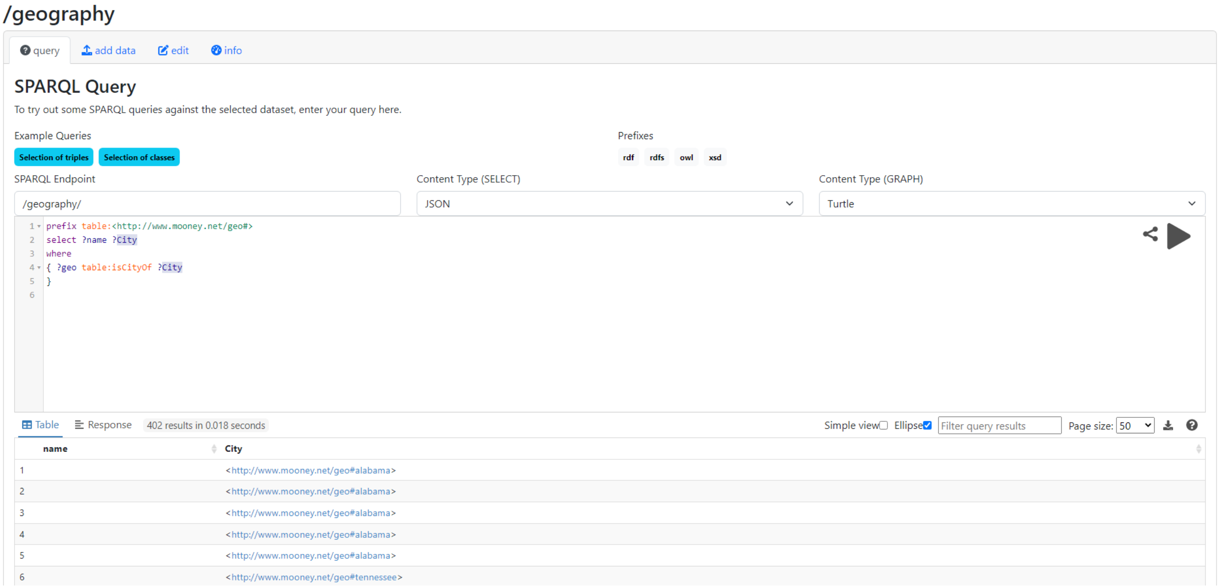
<http://www.mooney.net/geo>

geo

Ontology IRI: <http://www.mooney.net/geo>

Location: C:\Users\cusat\Downloads\geography\geography.owl

No Reasoner set. Select a reasoner from the Reasoner menu ☑ Show Inferences



SPARQL Query

To try out some SPARQL queries against the selected dataset, enter your query here.

Example Queries

Prefixes

SPARQL Endpoint: /geography/

Content Type (SELECT): JSON

Content Type (GRAPH): Turtle

```
prefix table:<http://www.mooney.net/geo#>
select ?name ?City
where
{ ?geo table:isCityOf ?City
}
```

402 results in 0.018 seconds

Simple view□ Ellipse☑ Filter query results Page size: 50

	name	City
1		<http://www.mooney.net/geo#alabama>
2		<http://www.mooney.net/geo#alabama>
3		<http://www.mooney.net/geo#alabama>
4		<http://www.mooney.net/geo#alabama>
5		<http://www.mooney.net/geo#alabama>
6		<http://www.mooney.net/geo#tennessee>

JAVA DATABASE CONNECTIVITY

AIM

Develop program to implement Java Database Connectivity

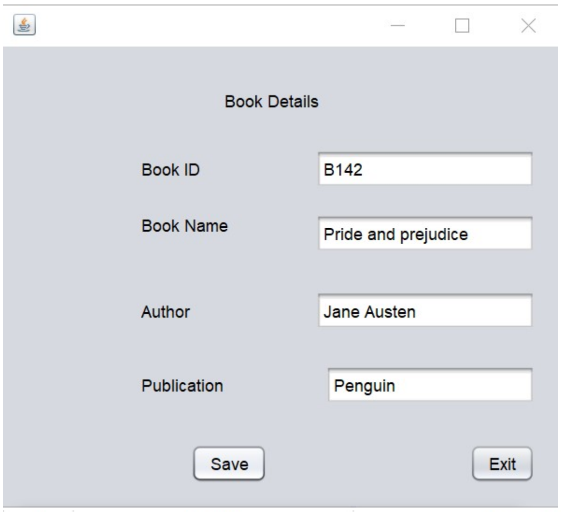
Question : 1

Write a program which connects to an online book database and insert the details of the books in to the database

QUERY

```
private void saveActionPerformed(java.awt.event.ActionEvent evt) {
        String a =t1.getText();
        String b =t2.getText();
        String c =t3.getText();
        String d =t4.getText();
        try{
          Class.forName("com.mysql.jdbc.Driver");
            Connection con=(Connection)DriverManager.getConnection("jdbc:
            mysql://localhost:3306/labcycle","root","root");
            java.sql.Statement stmt=(java.sql.Statement) con.createStatement();
            String query = "insert into library values('"+a+"','"+b+"',
            '"+c+"','"+d+"')";
            stmt.executeUpdate(query);
            JOptionPane.showMessageDialog(null,"Your book details have been
            saved successfully!!");
        }
        catch(Exception e){

        JOptionPane.showMessageDialog(this,e.getMessage());
            }
}
```

DATABASE TABLES



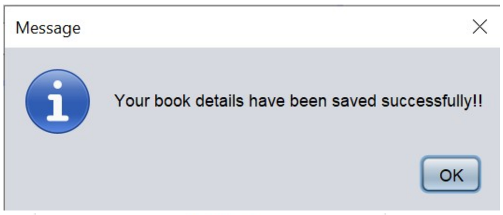
Book Details

Book ID: B142

Book Name: Pride and prejudice

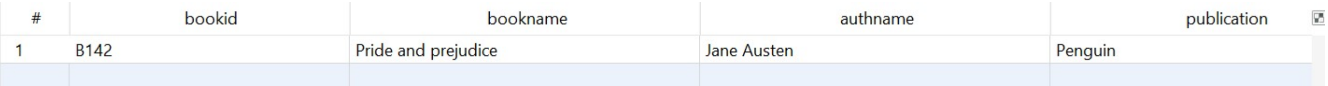
Author: Jane Austen

Publication: Penguin



Message

Your book details have been saved successfully!!



#	bookid	bookname	authname	publication
1	B142	Pride and prejudice	Jane Austen	Penguin

Question : 2

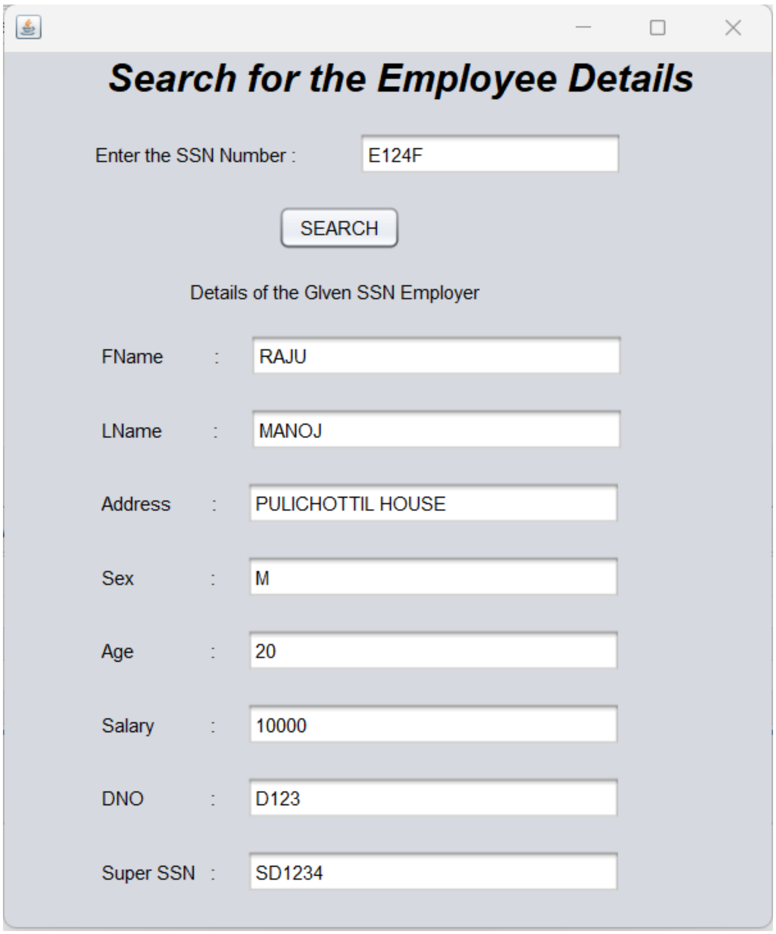
Write a program which connects to an online Employee database and retrieve the details of the employees in the database as per the schema

QUERY

```
private void jButton1MouseClicked(java.awt.event.MouseEvent evt) {
      String ssnno=t1.getText();

      PreparedStatement ps;
      ResultSet rs;
  try{
      Class.forName("com.mysql.jdbc.Driver");
      Connection con=(Connection)DriverManager.getConnection("jdbc:
      mysql://localhost:3306/labcycle_2","root","root");
      //Statement stmt=(Statement) con.createStatement();
      String querysql="select * from employee where SSNNo = ’"+ssnno+"’";
      ps=con.prepareStatement(querysql);
      //ps.setString(1,search.getText());
      rs=ps.executeQuery();
      //stmt.executeUpdate(querysql);
      if(rs.next()){
          ssnno=rs.getString("SSNNo");
          t1.setText(ssnno);
```

```
        String fname=rs.getString("FName");
        t2.setText(fname);
        String lname=rs.getString("LName");
        t3.setText(lname);
        String add=rs.getString("Address");
        t4.setText(add);
        String sex=rs.getString("Sex");
        t5.setText(sex);
        String age=rs.getString("Age");
        t6.setText(age);
        String salary=rs.getString("Salary");
        t7.setText(salary);
         String dno=rs.getString("DNO");
        t8.setText(dno);
        String superssn=rs.getString("SuperSSN");
        t9.setText(superssn);
    }
    JOptionPane.showMessageDialog(null,"Search successfull");
}
catch(Exception e){
    JOptionPane.showMessageDialog(null,e.getMessage());
  }        // TODO add your handling code here:
}
```

DATABASE TABLES

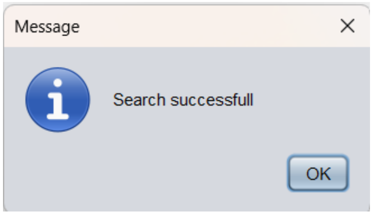


Search for the Employee Details

Enter the SSN Number : E124F

Details of the GIven SSN Employer

FName	:	RAJU
LName	:	MANOJ
Address	:	PULICHOTTIL HOUSE
Sex	:	M
Age	:	20
Salary	:	10000
DNO	:	D123
Super SSN	:	SD1234



Message

Search successfull

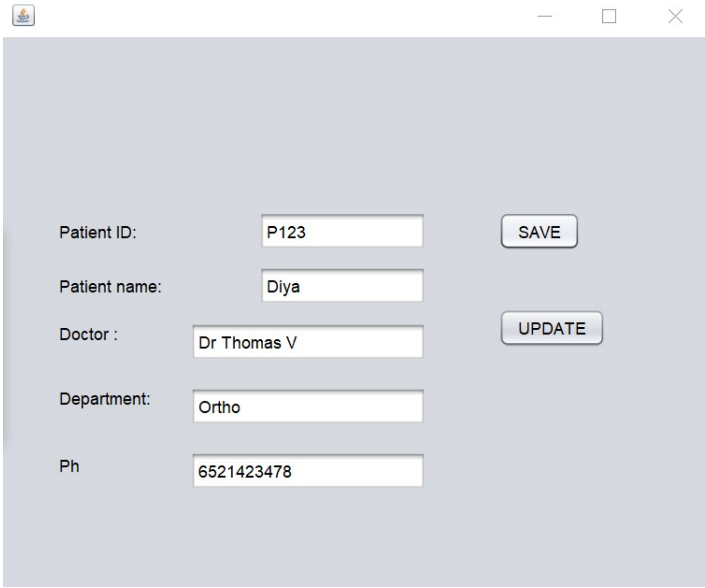
Question : 3

Write a program which connects to an online hospital database and update the details of the patients in the database

QUERY

```
private void updateMouseClicked(java.awt.event.MouseEvent evt) {
       // TODO add your handling code here:
       String patientid=t1.getText();
        String patientname=t2.getText();
        String doctoconsult=t3.getText();
        String department=t4.getText();
        String phno=t5.getText();
      try{
       PreparedStatement ps;
       Class.forName("com.mysql.jdbc.Driver");
       Connection con=(Connection)DriverManager.getConnection("jdbc:mysql://
    localhost:3306/labcycle","root","root");
    Statement stmt=con.createStatement();
    String sql="update hospital set pid='"+patientid+"',pname=
    '"+patientname+"',dtoc='"+doctoconsult+"',dep='"+department+"',
    phno='"+phno+"' where pid='"+patientid+"'";
    ps=con.prepareStatement(sql);
    ps.execute();
    stmt.executeUpdate(sql);
    JOptionPane.showMessageDialog(null,"Data successfully updated!!");

    }
    catch(Exception e){
       JOptionPane.showMessageDialog(this, e.getMessage());
    }

}
```

DATABASE TABLES



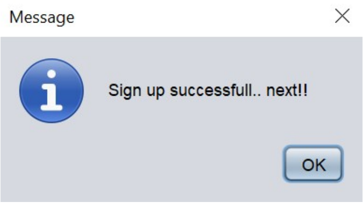
Patient ID: P123

Patient name: Diya

Doctor : Dr Thomas V

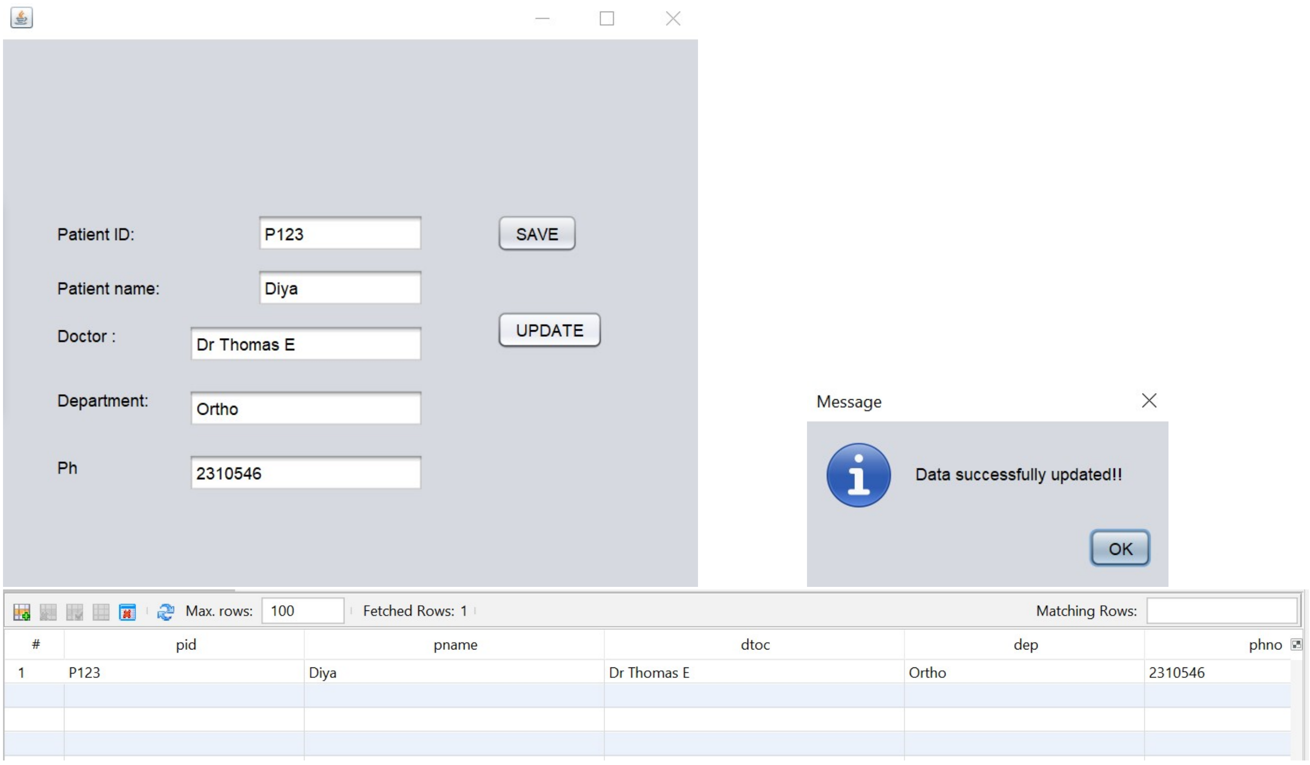
Department: Ortho

Ph 6521423478



Message

Sign up successfull.. next!!



Patient ID:	P123
Patient name:	Diya
Doctor :	Dr Thomas E
Department:	Ortho
Ph	2310546

Data successfully updated!!

Max. rows: 100 Fetched Rows: 1 Matching Rows:

#	pid	pname	dtoc	dep	phno
1	P123	Diya	Dr Thomas E	Ortho	2310546

Question : 4

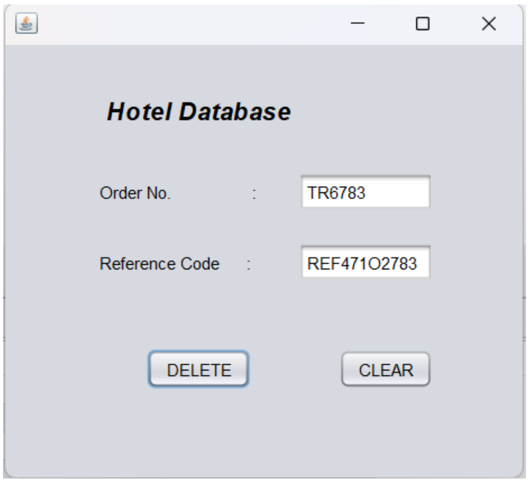
Write a program which connects to an online Hotel database and delete the details of the orders from the database

QUERY

```
private void jButton1MouseClicked(java.awt.event.MouseEvent evt) {
        String orderno = t1.getText();
        String refno = t2.getText();
        PreparedStatement ps;
        ResultSet rs;
           try{
           Class.forName("com.mysql.cj.jdbc.Driver");
           Connection con = DriverManager.getConnection("jdbc:mysql://
           localhost:3306/labcycle_2","root","root");
           //Statement stm = con.createStatement();

           String sql = "SELECT OrderNO FROM hotel WHERE OrderNO=’"+orderno+"’";
           ps = con.prepareStatement(sql);
           rs = ps.executeQuery();
           Statement stm = con.createStatement();
           if (rs.next())
```

```
            {
            String sql2 = "DELETE FROM hotel WHERE OrderNO =’"+orderno+"’";
            stm.executeUpdate(sql2);
            JOptionPane.showMessageDialog(null, "Deleted the Order \n
            Reference No : "
            + refno+" \n Record Successfully!!");
            }
            else
            {
            JOptionPane.showMessageDialog(null, "Order No : " + orderno+" \n
            Does not exsist.");
            }

        }
        catch(Exception e){
            JOptionPane.showMessageDialog(this, e.getMessage());
        }        // TODO add your handling code here:
    }
```

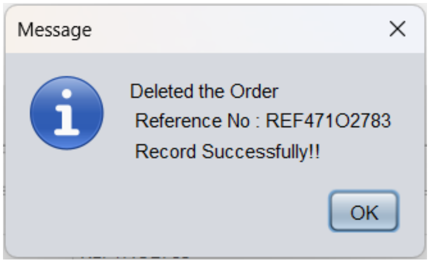
DATABASE TABLES



Hotel Database

Order No. : TR6783

Reference Code : REF471O2783



Message

Deleted the Order
Reference No : REF471O2783
Record Successfully!!

PROJECT

AIM

Develop an Application software using Java and MySQL for an Information Management Purpose.

PROJECT DESCRIPTION

The main aim of the project is to create a fully functional airline ticket reservation system which will hold the details of flight schedules and passengers.The project uses Java NetBeans as Front end and MySQL as Back end.Database and Backend will be added to current static system which will make it dynamic.The project creates a connection between MySQL and Java to store and manipulate information.This system provides proper management of data in a centralized and organized manner.The main advantage of this system is that data can be easily added,updated,searched and deleted without changing the source code.

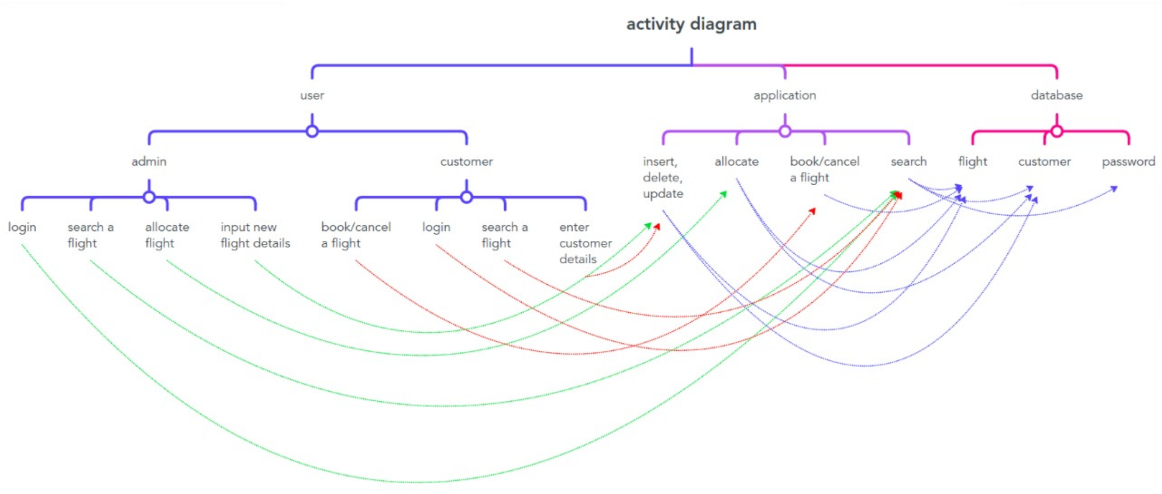
USERS AND FUNCTIONALITIES

1. System has 2 users :
   - Administrator(admin)
   - Customer
2. Functionalities for admin :
   - Add,Update and delete customers
   - Add,update and delete flights
3. Functionalities for users :
   - Book a flight (by inputting their details)
   - View the flight ticket
   - Edit their details
   - Cancel a flight

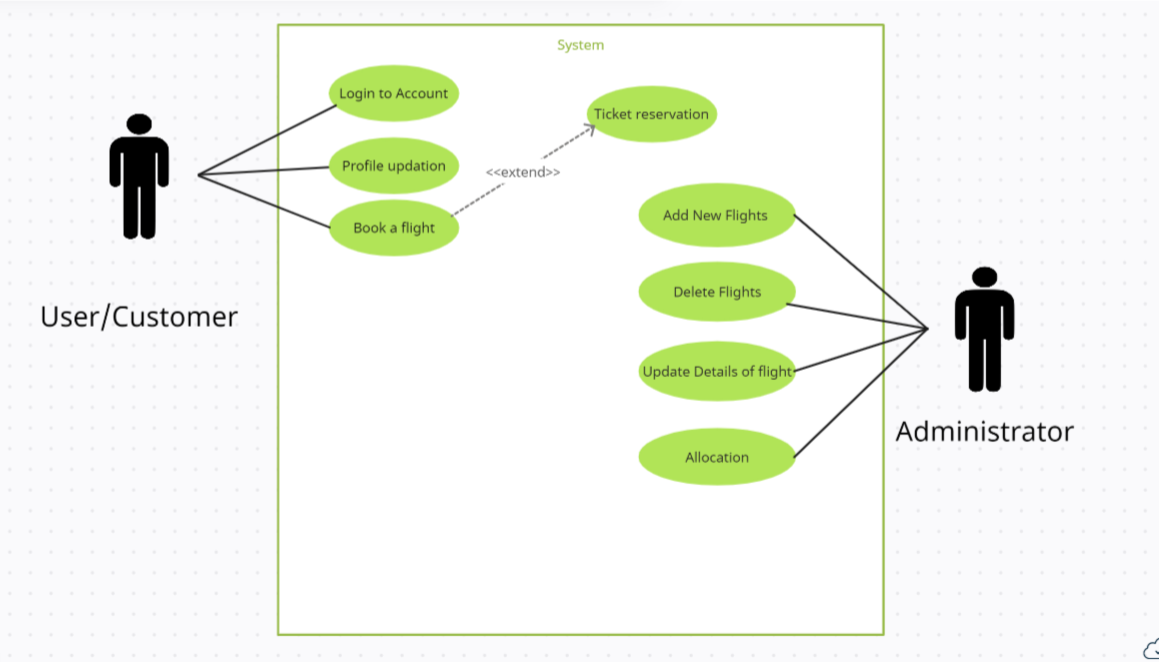
RESEARCH DESIGN

UML Diagrams

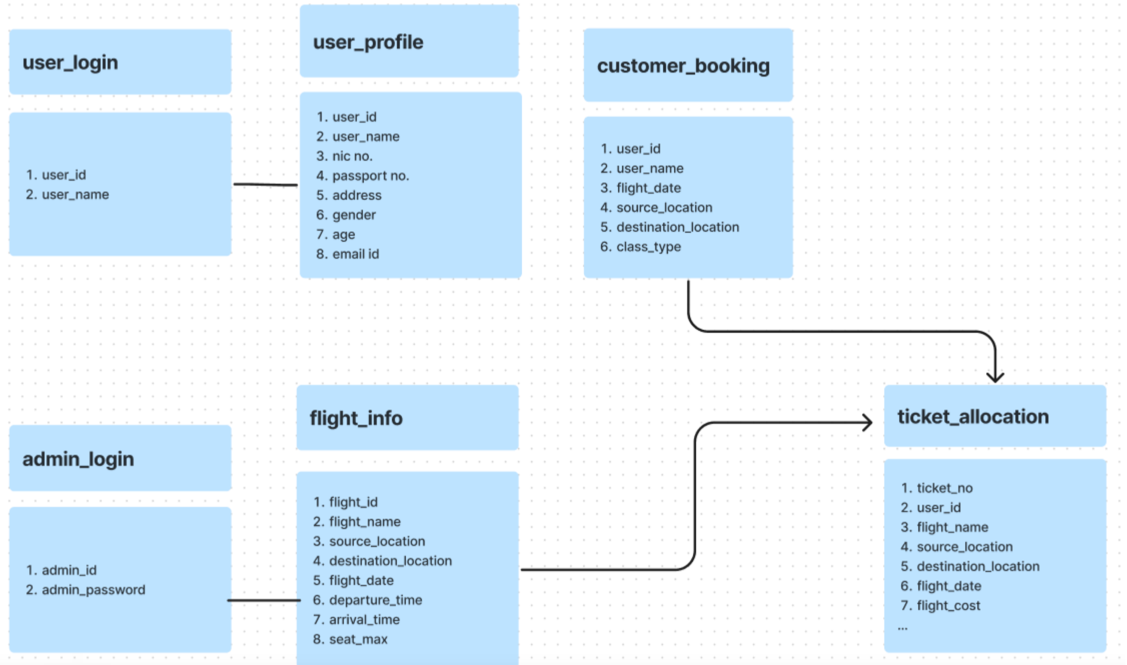
1. Activity diagram



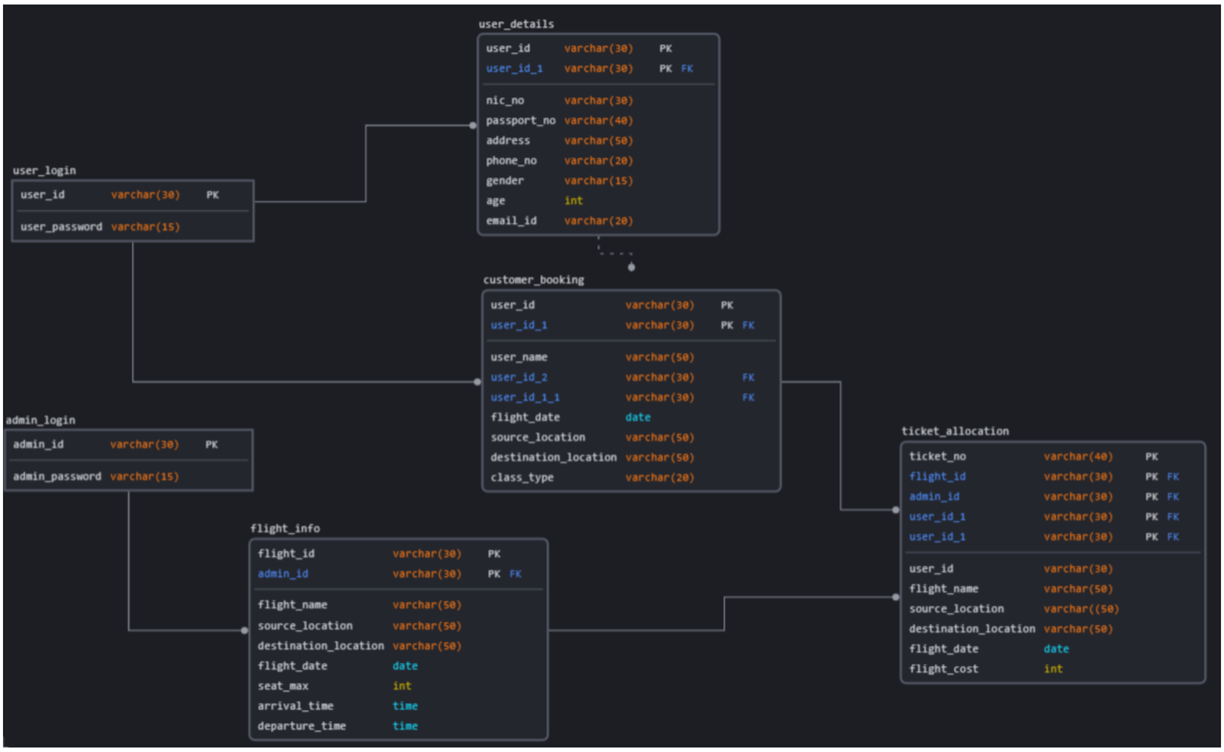
2. Use case diagram



3. Class diagram



Database Design - schema



user_details

user_id	varchar(30)	PK	
user_id_1	varchar(30)	PK	FK
nic_no	varchar(30)		
passport_no	varchar(40)		
address	varchar(50)		
phone_no	varchar(20)		
gender	varchar(15)		
age	int		
email_id	varchar(20)		

user_login

user_id	varchar(30)	PK
user_password	varchar(15)	

customer_booking

user_id	varchar(30)	PK	
user_id_1	varchar(30)	PK	FK
user_name	varchar(50)		
user_id_2	varchar(30)		FK
user_id_1_1	varchar(30)		FK
flight_date	date		
source_location	varchar(50)		
destination_location	varchar(50)		
class_type	varchar(20)		

admin_login

admin_id	varchar(30)	PK
admin_password	varchar(15)	

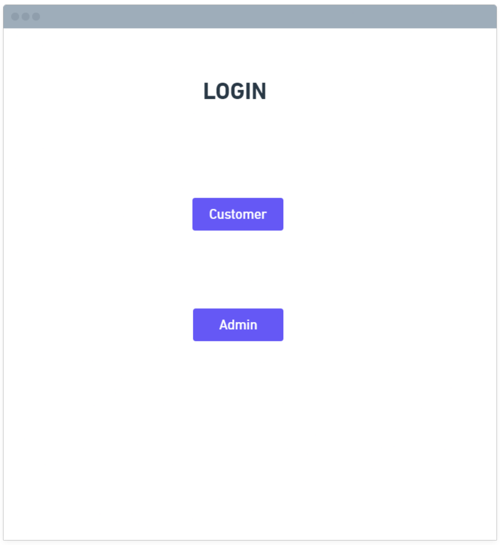
ticket_allocation

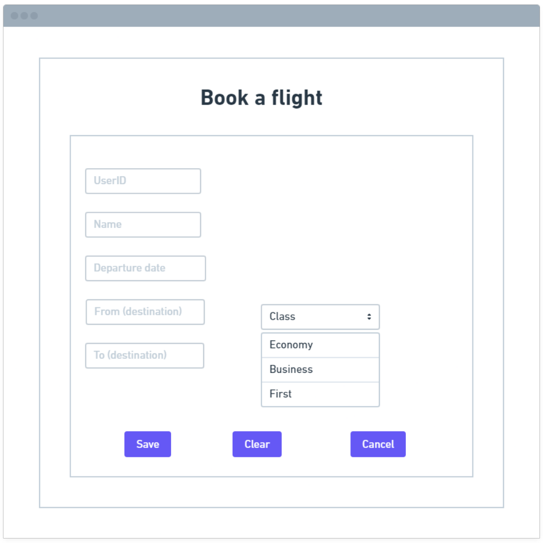
ticket_no	varchar(40)	PK	
flight_id	varchar(30)	PK	FK
admin_id	varchar(30)	PK	FK
user_id_1	varchar(30)	PK	FK
user_id_1	varchar(30)	PK	FK
user_id	varchar(30)		
flight_name	varchar(50)		
source_location	varchar((50)		
destination_location	varchar(50)		
flight_date	date		
flight_cost	int		

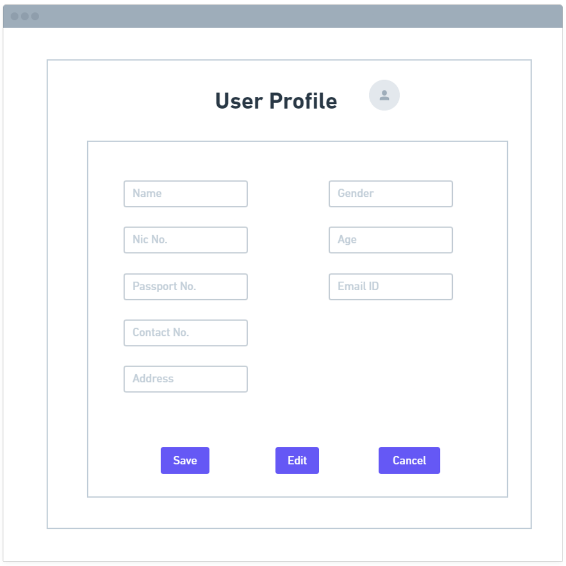
flight_info

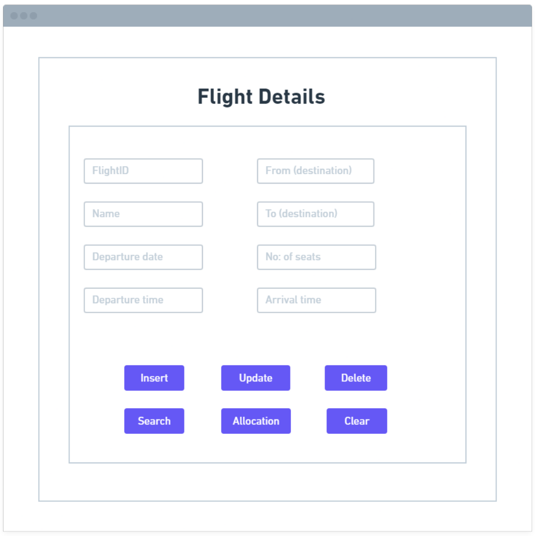
flight_id	varchar(30)	PK	
admin_id	varchar(30)	PK	FK
flight_name	varchar(50)		
source_location	varchar(50)		
destination_location	varchar(50)		
flight_date	date		
seat_max	int		
arrival_time	time		
departure_time	time		

Wireframes for GUI









SOFTWARE TOOLS

MySQL : MySQL is a popular open-source relational database management system that can be used to store and manage the data for the BBMS.

Java Development Kit (JDK): The JDK is a software development kit that includes the necessary tools to develop Java applications. It includes the Java Runtime Environment (JRE) and the Java compiler.

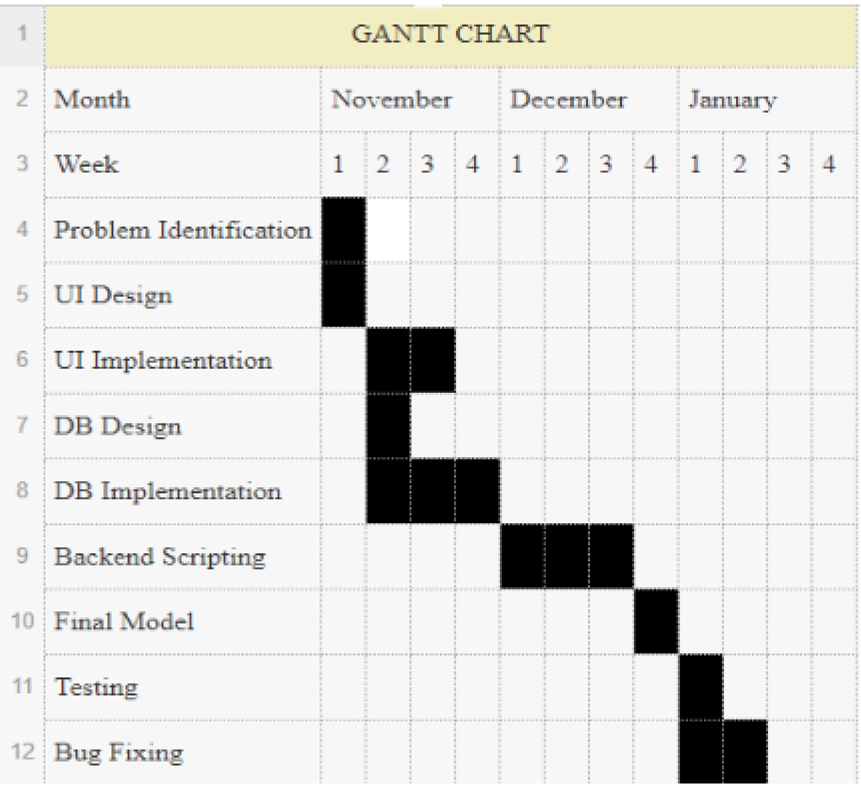
Integrated Development Environment (IDE): An IDE is a software application that provides a comprehensive environment for software development. Some popular IDEs for Java development include Eclipse, IntelliJ IDEA, and NetBeans.

Java Persistence API (JPA): JPA is a Java specification for accessing, persisting, and managing data between Java objects/classes and a relational database. It can be used to interact with the MySQL database in the BBMS.

JavaServer Faces (JSF): JSF is a Java-based web application framework that can be used to develop the user interface of the BBMS.

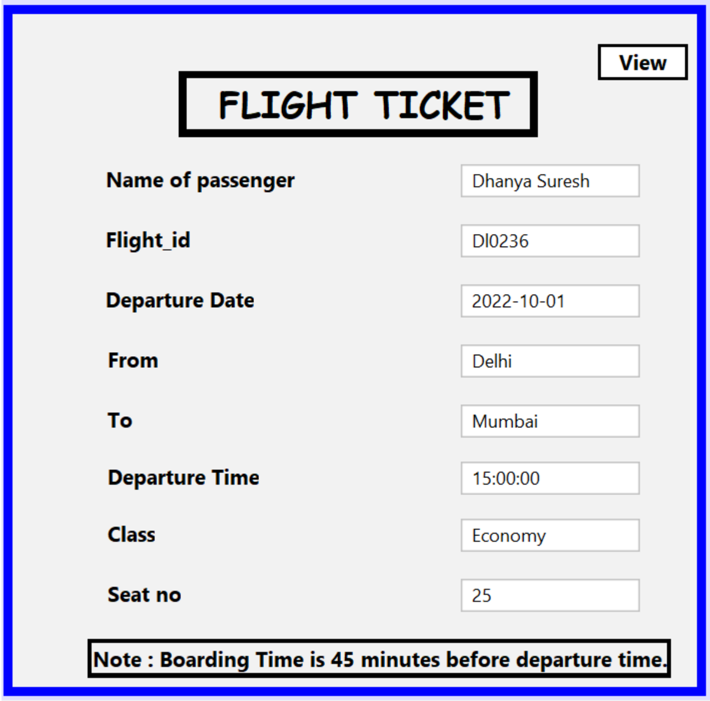
MySQL Connector/J: This is a JDBC driver that allows Java applications to connect to a MySQL database.

IMPLEMENTATION



1	GANTT CHART												
2	Month	November				December				January			
3	Week	1	2	3	4	1	2	3	4	1	2	3	4
4	Problem Identification	■											
5	UI Design	■											
6	UI Implementation		■	■									
7	DB Design		■										
8	DB Implementation		■	■	■								
9	Backend Scripting					■	■	■					
10	Final Model								■				
11	Testing									■			
12	Bug Fixing									■	■		

RESULT AND OUTPUT



FLIGHT TICKET

Name of passenger: Dhanya Suresh

Flight_id: DI0236

Departure Date: 2022-10-01

From: Delhi

To: Mumbai

Departure Time: 15:00:00

Class: Economy

Seat no: 25

Note : Boarding Time is 45 minutes before departure time.

CRITICAL EVALUATION

Test case no	Description	Input	Expected output	Actual Output	Result
1	Sign In whether the input username and password are right	Right username and password	Signin Success fully	Signin Sucess full	passed
2	Sign in whether the inputs username and password are wrong	Wrong username and password	signin Unsucess fully	Signin Unsucess full	Passed
3	Check whether the tables are created	Input are given	Tables created	table creates	passed
4	Check Whether Updation, Deletion, Search, are done according to the given inputs	Inputs are Given	Updation Deletion, Search Addition are are done	Done sucessfully	Passed

CONCLUSION

The system is typically used by two types of users: customers, and administrators (admins), and it can be designed as a web-based or a desktop-based application. It can also be accessed by authorized users via a web browser or a client application. The built system effectively stores,deletes,updates,inserts and retrieves the data in a secure and efficient manner.

REFERENCES

1. https://www.geeksforgeeks.org/java-database-connectivity-with-mysql/

2. https://stackoverflow.com/questions/65655200/how-to-fix-clean-and-build-project-netbeans-error