

**M.Sc. (Five Year Integrated) in Computer Science
(Artificial Intelligence & Data Science)**

Third Semester

Laboratory Record

21-805-0307: DATABASE SYSTEMS LAB

*Submitted in partial fulfillment
of the requirements for the award of degree in
Master of Science (Five Year Integrated)
in Computer Science (Artificial Intelligence & Data Science) of
Cochin University of Science and Technology (CUSAT)
Kochi*



Submitted by

**FARHA T A
(80521009)**

**DEPARTMENT OF COMPUTER SCIENCE
COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY (CUSAT)
KOCHI-682022**

JANUARY 2023

DEPARTMENT OF COMPUTER SCIENCE
COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY (CUSAT)
KOCHI, KERALA-682022



*This is to certify that the software laboratory record for **21-805-0307: Database Systems Lab** is a record of work carried out by **FARHA T A(80521009)**, in partial fulfillment of the requirements for the award of degree in **Master of Science (Five Year Integrated) in Computer Science (Artificial Intelligence & Data Science)** of Cochin University of Science and Technology (CUSAT), Kochi. The lab record has been approved as it satisfies the academic requirements in respect of the third semester laboratory prescribed for the Master of Science (Five Year Integrated) in Computer Science degree.*

Faculty Member in-charge

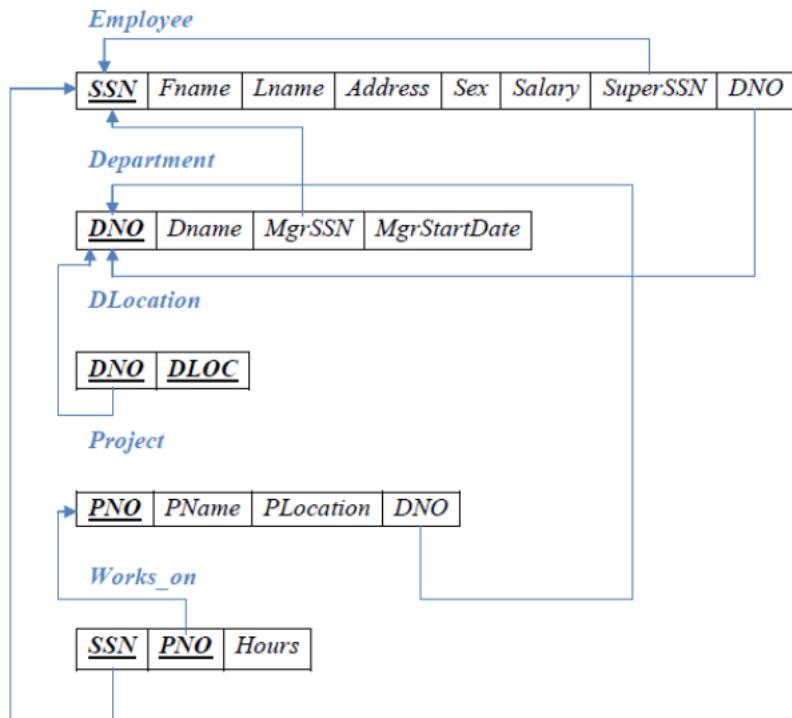
Sajmi Salam
Guest Faculty
Department of Computer Science
CUSAT

Dr. Philip Samuel
Professor and Head
Department of Computer Science
CUSAT

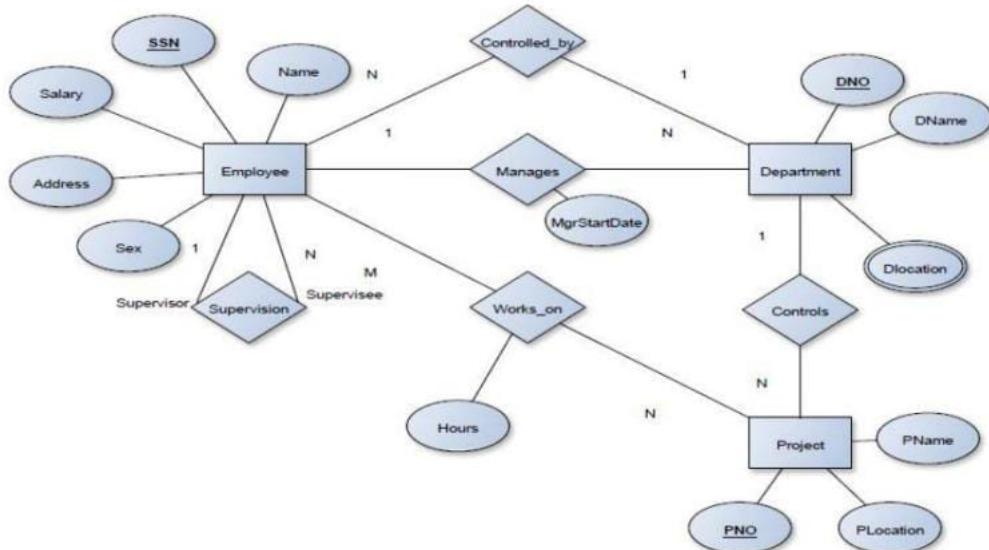
Table of Contents

Sl.No.	Program	Page.No.
1	Schema Diagram and ER Diagram	1
2	Queries to implement DDL Commands	2
3	Queries to implement DML Commands	6
4	Queries to implement DCL Commands.	11
5	Queries to implement Group Functions	12
6	Program to implement Nested Queries	15
7	Program to implement Views	18
8	Programs of Functions And Procedures	20
9	Implementation of Cursor	26
10	Implementation of Trigger	31
11	Queries to implement TCL Commands	35
12	Operations on NOSQL Systems	37
13	Simple Structure of GraphQL program	42
14	Programs demonstrating Java Database Connectivity	44
15	Project Report on Application Software	50

SCHEMA DIAGRAM



ER DIAGRAM



DDL COMMANDS

AIM

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

Questions : 1

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

QUERY

```
mysql> use COMPANY;  
Database changed
```

```
mysql> create table Employee(SSN varchar(20) primary key not null,Fname varchar(20),  
Lname varchar(20) null,Address varchar(20),Sex varchar(20) default 'Male',  
Salary int(20) check(Salary>5000),SuperSSN varchar(20) unique,DNO varchar(20));
```

```
Query OK, 0 rows affected, 1 warning (0.05 sec)
```

```
mysql> create table Department(DNO varchar(20) primary key,Dname varchar(20),  
MgrSSN varchar(20),MgrStartDate date not null);
```

```
Query OK, 0 rows affected (0.03 sec)
```

```
mysql> create table DLocation(DNO varchar(20) primary key,DLOC varchar(20));
```

```
Query OK, 0 rows affected (0.02 sec)
```

```
mysql> create table Project(PNO varchar(20) primary key,Pname varchar(20),  
Plocation varchar(20),DNO varchar(20),constraint dno_project foreign key(DNO)  
references Department(DNO));
```

```
Query OK, 0 rows affected (0.04 sec)
```

```
mysql> create table WORKS_ON(SSN varchar(20) primary key,PNO varchar(20),Hours int(20));
```

```
Query OK, 0 rows affected, 1 warning (0.02 sec)
```

DATABASE TABLES

```

mysql> desc Employee;
+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| SSN   | varchar(20) | NO   | PRI | NULL    |
| Fname | varchar(20) | YES  |     | NULL    |
| Lname | varchar(20) | YES  |     | NULL    |
| Address | varchar(20) | YES  |     | NULL    |
| Sex   | varchar(20) | YES  |     | Male    |
| Salary | int        | YES  |     | NULL    |
| SuperSSN | varchar(20) | YES  | UNI | NULL    |
| DNO   | varchar(20) | YES  |     | NULL    |
+-----+-----+-----+-----+-----+
mysql> desc Department;
+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| DNO   | varchar(20) | NO   | PRI | NULL    |
| Dname | varchar(20) | YES  |     | NULL    |
| MgrSSN | varchar(20) | YES  |     | NULL    |
| MgrStartDate | date | NO   |     | NULL    |
+-----+-----+-----+-----+-----+
mysql> desc DLocation;
+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+
| DNO   | varchar(20) | NO   | PRI | NULL    |
| DLOC  | varchar(20) | YES  |     | NULL    |
+-----+-----+-----+-----+
mysql> desc Project;
+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| PNO   | varchar(20) | NO   | PRI | NULL    |
| Pname | varchar(20) | YES  |     | NULL    |
| Plocation | varchar(20) | YES  |     | NULL    |
| DNO   | varchar(20) | YES  | MUL | NULL    |
+-----+-----+-----+-----+
mysql> desc WORKS_ON;
+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| SSN   | varchar(20) | NO   | PRI | NULL    |
| PNO   | varchar(20) | YES  |     | NULL    |
| Hours | int        | YES  |     | NULL    |
+-----+-----+-----+-----+

```

Questions : 2

Add another column Age with datatype integer in employee table

QUERY

```
mysql> alter table Employee add(Age integer);
```

```
Query OK, 0 rows affected (0.02 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

DATABASE TABLES

Field	Type	Null	Key	Default	Extra
SSN	varchar(20)	NO	PRI	NULL	
Fname	varchar(20)	YES		NULL	
Lname	varchar(20)	YES		NULL	
Address	varchar(20)	YES		NULL	
Sex	varchar(20)	YES		Male	
Salary	int	YES		NULL	
SuperSSN	varchar(20)	YES	UNI	NULL	
DNO	varchar(20)	YES		NULL	
Age	int	YES		NULL	

Questions : 3

Drop a table named Project

QUERY

```
mysql> drop table Project;
```

```
Query OK, 0 rows affected (0.02 sec)
```

DATABASE TABLES

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

Questions : 4

Truncate a table named WORKS_ON

QUERY

```
mysql> Truncate table WORKS_ON;
```

```
Query OK, 0 rows affected (0.03 sec)
```

DATABASE TABLES

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

mysql> select * from WORKS_ON;
Empty set (0.01 sec)
```

Questions : 5

View the structure of the table Department

QUERY

```
mysql> desc Department;
```

DATABASE TABLES

```
+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| DNO        | varchar(20) | NO   | PRI | NULL    |       |
| Dname      | varchar(20) | YES  |     | NULL    |       |
| MgrSSN     | varchar(20) | YES  |     | NULL    |       |
| MgrStartDate | date      | NO   |     | NULL    |       |
+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

DML COMMANDS

AIM

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

Questions : 1

Insert five records in the tables as per the above schema

QUERY

```
mysql> insert into Employee values('e001','Arun','Kumar','Kochi','Male',20000,'s1001','d01',25);
Query OK, 1 row affected (0.02 sec)

mysql> insert into Employee values('e002','Ann','Susan','Chennai','Female',25000,'s1002','d02',23);
Query OK, 1 row affected (0.01 sec)

mysql> insert into Employee values('e003','Anu','Priya','Chennai','Female',18000,'s1003','d03',23);
Query OK, 1 row affected (0.01 sec)

mysql> insert into Employee values('e004','Sidharth','Shukla','Pune','Male',17000,'s1004','d04',22);
Query OK, 1 row affected (0.01 sec)

mysql> insert into Employee values('e005','Ali','Khan','Mumbai','Male',30000,'s1005','d04',22);
Query OK, 1 row affected (0.01 sec)

mysql> insert into Department values("d01","Sales","m1001","2020-12-03");
Query OK, 1 row affected (0.01 sec)

mysql> insert into Department values("d02","Finance","m1002","2022-10-13");
Query OK, 1 row affected (0.01 sec)

mysql> insert into Department values("d03","Marketing","m1003","2021-11-04");
Query OK, 1 row affected (0.01 sec)

mysql> insert into Department values("d04","HR","m1004","2021-01-02");
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into Department values("d05","Designing","m1005","2022-02-12");
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into DLocation values("d01","Chennai");
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into DLocation values("d02","Pune");
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into DLocation values("d03","Banglore");
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into DLocation values("d04","Pune");
Query OK, 1 row affected (0.00 sec)
```

```
mysql> insert into DLocation values("d05","Chennai");
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into Project values("p1001","Project Zen","Chennai","d01");
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into Project values("p1002","Project Breeze","Mumbai","d02");
Query OK, 1 row affected (0.00 sec)
```

```
mysql> insert into Project values("p1003","Project Mecha","Mumbai","d03");
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into Project values("p1004","Project Program","Banglore","d04");
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into Project values("p1005","Project Dynamite","Chennai","d05");
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into WORKS_ON values("e1001","p1001",4);
Query OK, 1 row affected (0.02 sec)
```

```
mysql> insert into WORKS_ON values("e1002","p1002",5);
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into WORKS_ON values("e1003","p1003",3);
```

```
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into WORKS_ON values("e1004","p1004",5);
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into WORKS_ON values("e1005","p1005",6);
Query OK, 1 row affected (0.01 sec)
```

Questions : 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  |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
| e001 | Arun   | Kumar  | Kochi   | Male  | 20000 | s1001  | d01  | 25   |
| e002 | Ann    | Susan  | Chennai | Female | 25000 | s1002  | d02  | 23   |
| e003 | Anu    | Priya  | Chennai | Female | 18000 | s1003  | d03  | 23   |
| e004 | Sidharth | Shukla | Pune   | Male  | 17000 | s1004  | d04  | 22   |
| e005 | Ali    | Khan   | Mumbai  | Male  | 30000 | s1005  | d04  | 22   |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)

mysql> select * from Department;
+----+-----+-----+-----+
| DNO | Dname | MgrSSN | MgrStartDate |
+----+-----+-----+-----+
| d01 | Sales | m1001 | 2020-12-03 |
| d02 | Finance | m1002 | 2022-10-13 |
| d03 | Marketing | m1003 | 2021-11-04 |
| d04 | HR | m1004 | 2021-01-02 |
| d05 | Designing | m1005 | 2022-02-12 |
+----+-----+-----+-----+
5 rows in set (0.00 sec)
```

```
mysql> select * from DLocation;
+----+-----+
| DNO | DLOC      |
+----+-----+
| d01 | Chennai   |
| d02 | Pune       |
| d03 | Banglore   |
| d04 | Pune       |
| d05 | Chennai   |
+----+
5 rows in set (0.00 sec)

mysql> select * from Project;
+----+-----+-----+-----+
| PNO  | Pname        | Plocation | DNO   |
+----+-----+-----+-----+
| p1001 | Project Zen  | Chennai    | d01   |
| p1002 | Project Breeze | Mumbai    | d02   |
| p1003 | Project Mecha | Mumbai    | d03   |
| p1004 | Project Program | Banglore  | d04   |
| p1005 | Project Dynamite | Chennai  | d05   |
+----+
5 rows in set (0.00 sec)

mysql> select * from WORKS_ON;
+----+-----+-----+
| SSN  | PNO   | Hours |
+----+-----+-----+
| e1001 | p1001 | 4     |
| e1002 | p1002 | 5     |
| e1003 | p1003 | 3     |
| e1004 | p1004 | 5     |
| e1005 | p1005 | 6     |
+----+
5 rows in set (0.00 sec)
```

Questions : 3

Modify the salary of the employee as 25000 whose SSN is e001

QUERY

```
mysql> update Employee set Salary=25000 where SSN = 'e001';
Query OK, 1 row affected (0.02 sec)

Rows matched: 1  Changed: 1  Warnings: 0
```

DATABASE TABLES

```
mysql> select * from Employee;
+----+-----+-----+-----+-----+-----+-----+-----+-----+
| SSN | Fname | Lname | Address | Sex   | Salary | SuperSSN | DNO   | Age  |
+----+-----+-----+-----+-----+-----+-----+-----+
| e001 | Arun  | Kumar | Kochi   | Male  | 25000 | s1001  | d01   | 25   |
| e002 | Ann   | Susan | Chennai | Female | 25000 | s1002  | d02   | 23   |
| e003 | Anu   | Priya | Chennai | Female | 18000 | s1003  | d03   | 23   |
| e004 | Sidharth | Shukla | Pune   | Male  | 17000 | s1004  | d04   | 22   |
| e005 | Ali   | Khan  | Mumbai  | Male  | 30000 | s1005  | d04   | 22   |
+----+-----+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

Questions : 4

Delete the details of the employee whose SSN is "e002"

QUERY

```
mysql> delete from Employee where SSN="e002";
Query OK, 1 row affected (0.01 sec)
```

DATABASE TABLES

```
mysql> select * from Employee;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| SSN | Fname | Lname | Address | Sex | Salary | SuperSSN | DNO | Age |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| e001 | Arun   | Kumar  | Kochi   | Male | 25000 | s1001  | d01  | 25  |
| e003 | Anu    | Priya  | Chennai | Female | 18000 | s1003  | d03  | 23  |
| e004 | Sidharth | Shukla | Pune   | Male | 17000 | s1004  | d04  | 22  |
| e005 | Ali    | Khan   | Mumbai  | Male | 30000 | s1005  | d04  | 22  |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

DCL COMMANDS

AIM

Develop SQL Queries to implement Data Control Language commands

Questions : 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 (0.01 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 (0.01 sec)
```

Questions : 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 (0.00 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

Questions : 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

```
mysql> Select Fname from Employee where Fname like '%a_';
+-----+
| Fname |
+-----+
| Hanan |
+-----+
1 row in set (0.00 sec)
```

Questions : 2

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

QUERY

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

DATABASE TABLES

```
mysql> Select Sex,count(*) from Employee group by Sex;
+-----+-----+
| Sex | count(*) |
+-----+-----+
| Male |      3 |
| Female |     1 |
+-----+-----+
2 rows in set (0.01 sec)
```

Questions : 3

Calculate the average salary of the female employees.

QUERY

```
mysql> select avg(Salary) from EMPLOYEE where Sex="Female";
```

DATABASE TABLES

```
mysql> Select avg(Salary) from Employee where Sex="Female";
+-----+
| avg(Salary) |
+-----+
| 18000.0000 |
+-----+
1 row in set (0.01 sec)
```

Questions : 4

Calculate the sum of salaries of male employees.

QUERY

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

DATABASE TABLES

```
mysql> select sum(Salary) from Employee where Sex="Male";
+-----+
| sum(Salary) |
+-----+
|      72000 |
+-----+
1 row in set (0.00 sec)
```

Questions : 5

Display the maximum and minimum salaries of male employees.

QUERY

```
mysql> select max(Salary),min(Salary) from Employee where Sex="Male";
```

DATABASE TABLES

```
mysql> select max(Salary),min(Salary) from Employee where Sex="Male";
+-----+-----+
| max(Salary) | min(Salary) |
+-----+-----+
|      30000 |      17000 |
+-----+-----+
1 row in set (0.01 sec)
```

Questions : 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 |
+-----+-----+-----+-----+-----+-----+-----+
| e001 | Arun | Kumar | Kochi | Male | 25000 | s1001 | d01 | 25 |
| e005 | Hanan | Khan | Mumbai | Male | 30000 | s1005 | d04 | 22 |
+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.01 sec)
```

Questions : 7

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

QUERY

```
mysql> select Lname from Employee where Salary=30000 or Salary=40000
or Salary = 50000;
```

DATABASE TABLES

```
mysql> select Lname from Employee where Salary=30000 or Salary=40000 or Salary = 50000;
+-----+
| Lname |
+-----+
| Khan |
+-----+
1 row in set (0.00 sec)
```

NESTED QUERIES

AIM

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

Questions : 1

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

QUERY

```
mysql> Update Employee,Project set Salary=Salary+Salary*0.25
where Employee.DNo=Project.DNo and PLocation="Chennai";
Query OK, 1 row affected (0.02 sec)
Rows matched: 1    Changed: 1    Warnings: 0
```

DATABASE TABLES

```
mysql> select * from employee;
+----+-----+-----+-----+-----+-----+-----+-----+-----+
| SSN | Fname | Lname | Address | Sex   | Salary | SuperSSN | DNo  | Age  |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
| e001 | Arun   | Kumar  | Chennai | Male  | 31250 | s1001   | d01  | 25  |
| e003 | Anu    | Priya  | Chennai | Female | 22500 | s1003   | d03  | 23  |
| e004 | Sidharth | Shukla | Pune   | Male  | 17000 | s1004   | d04  | 22  |
| e005 | Hanan  | Khan   | Mumbai  | Male  | 30000 | s1005   | d04  | 22  |
+----+-----+-----+-----+-----+-----+-----+-----+-----+
4 rows in set (0.01 sec)
```

Questions : 2

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

QUERY

```
mysql> select Employee.Fname,PROJECT.PLocation FROM Employee,WORKS_ON,Project
where WORKS_ON.Hours > 5 and Employee.SSN = WORKS_ON.SSN
and WORKS_ON.PNo=project.PNo;
```

DATABASE TABLES

```
+-----+
| Fname | PLocation |
+-----+
| Hanan | Chennai  |
+-----+
1 row in set (0.00 sec)
```

Questions : 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

```
mysql> select * from EMPLOYEE left join WORKS_ON on EMPLOYEE.SSN = WORKS_ON.SSN;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| SSN | Fname | Lname | Address | Sex | Salary | SuperSSN | DNO | Age | SSN | PNO | Hours |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| e001 | Arun   | Kumar  | Kochi   | Male | 25000 | s1001  | d01  | 25  | e001 | p1001 | 4   |
| e003 | Anu    | Priya  | Chennai | Female | 22500 | s1003  | d03  | 23  | e003 | p1003 | 3   |
| e004 | Sidharth | Shukla | Pune   | Male | 17000 | s1004  | d04  | 22  | e004 | p1004 | 5   |
| e005 | Hanan  | Khan   | Mumbai  | Male | 30000 | s1005  | d04  | 22  | e005 | p1005 | 6   |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
4 rows in set (0.01 sec)
```

Questions : 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

```
mysql> select * from WORKS_ON RIGHT join EMPLOYEE on EMPLOYEE.SSN = WORKS_ON.SSN;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| SSN | PNO | Hours | SSN | Fname | Lname | Address | Sex | Salary | SuperSSN | DNO | Age |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| e001 | p1001 |     4 | e001 | Arun   | Kumar  | Kochi   | Male | 25000 | s1001  | d01  | 25  |
| e003 | p1003 |     3 | e003 | Anu    | Priya  | Chennai | Female | 22500 | s1003  | d03  | 23  |
| e004 | p1004 |     5 | e004 | Sidharth | Shukla | Pune   | Male | 17000 | s1004  | d04  | 22  |
| e005 | p1005 |     6 | e005 | Hanan  | Khan   | Mumbai  | Male | 30000 | s1005  | d04  | 22  |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

Questions : 5

Full join works_on table and employee table

QUERY

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

DATABASE TABLES

SSN	PNO	Hours	SSN	Fname	Lname	Address	Sex	Salary	SuperSSN	DNO	Age
e001	p1001	4	e005	Hanan	Khan	Mumbai	Male	30000	s1005	d04	22
e001	p1001	4	e004	Sidharth	Shukla	Pune	Male	17000	s1004	d04	22
e001	p1001	4	e003	Anu	Priya	Chennai	Female	22500	s1003	d03	23
e001	p1001	4	e001	Arun	Kumar	Kochi	Male	25000	s1001	d01	25
e002	p1002	5	e005	Hanan	Khan	Mumbai	Male	30000	s1005	d04	22
e002	p1002	5	e004	Sidharth	Shukla	Pune	Male	17000	s1004	d04	22
e002	p1002	5	e003	Anu	Priya	Chennai	Female	22500	s1003	d03	23
e002	p1002	5	e001	Arun	Kumar	Kochi	Male	25000	s1001	d01	25
e003	p1003	3	e005	Hanan	Khan	Mumbai	Male	30000	s1005	d04	22
e003	p1003	3	e004	Sidharth	Shukla	Pune	Male	17000	s1004	d04	22
e003	p1003	3	e003	Anu	Priya	Chennai	Female	22500	s1003	d03	23
e003	p1003	3	e001	Arun	Kumar	Kochi	Male	25000	s1001	d01	25
e004	p1004	5	e005	Hanan	Khan	Mumbai	Male	30000	s1005	d04	22
e004	p1004	5	e004	Sidharth	Shukla	Pune	Male	17000	s1004	d04	22
e004	p1004	5	e003	Anu	Priya	Chennai	Female	22500	s1003	d03	23
e004	p1004	5	e001	Arun	Kumar	Kochi	Male	25000	s1001	d01	25
e005	p1005	6	e005	Hanan	Khan	Mumbai	Male	30000	s1005	d04	22
e005	p1005	6	e004	Sidharth	Shukla	Pune	Male	17000	s1004	d04	22
e005	p1005	6	e003	Anu	Priya	Chennai	Female	22500	s1003	d03	23
e005	p1005	6	e001	Arun	Kumar	Kochi	Male	25000	s1001	d01	25

20 rows in set (0.00 sec)

VIEWS

AIM

Develop SQL Queries for creating and dropping Views

Questions : 1

Create a view VW_emp on employee table

QUERY

```
mysql> create view VW_emp as select*from EMPLOYEE;
Query OK, 0 rows affected (0.00 sec)
```

DATABASE TABLES

```
mysql> select * from VW_emp;
+-----+-----+-----+-----+-----+-----+-----+-----+
| SSN | Fname | Lname | Address | Sex   | Salary | SuperSSN | DNO  | Age  |
+-----+-----+-----+-----+-----+-----+-----+-----+
| e001 | Arun  | Kumar | Kochi   | Male  | 25000 | s1001  | d01  | 25  |
| e003 | Anu   | Priya | Chennai | Female | 22500 | s1003  | d03  | 23  |
| e004 | Sidharth | Shukla | Pune    | Male  | 17000 | s1004  | d04  | 22  |
| e005 | Hanan | Khan  | Mumbai  | Male  | 30000 | s1005  | d04  | 22  |
+-----+-----+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

Questions : 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 (0.01 sec)
```

DATABASE TABLES

```
mysql> select * from VW_SSN;
+-----+-----+
| SuperSSN | DNO  |
+-----+-----+
| s1003    | d03  |
+-----+-----+
1 row in set (0.00 sec)
```

Questions : 3

Update the address of employee to Chennai whose id is e100 in view VW_emp

QUERY

```
mysql> UPDATE VW_emp SET Address="Chennai" WHERE SSN='e1001';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

DATABASE TABLES

```
mysql> select * from VW_emp;
+-----+-----+-----+-----+-----+-----+-----+-----+
| SSN | Fname | Lname | Address | Sex | Salary | SuperSSN | DNO | Age |
+-----+-----+-----+-----+-----+-----+-----+-----+
| e001 | Arun | Kumar | Chennai | Male | 25000 | s1001 | d01 | 25 |
| e003 | Anu | Priya | Chennai | Female | 22500 | s1003 | d03 | 23 |
| e004 | Sidharth | Shukla | Pune | Male | 17000 | s1004 | d04 | 22 |
| e005 | Hanan | Khan | Mumbai | Male | 30000 | s1005 | d04 | 22 |
+-----+-----+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

Questions : 4

Delete the view VW_emp

QUERY

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

DATABASE TABLES

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

FUNCTIONS AND PROCEDURES

AIM

Develop PL/SQL program to familiarize with Function and Procedure

Questions : 1

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

QUERY

```
SQL> connect
Enter user-name: farha
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
```

Questions : 2

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

QUERY

```
SQL> connect
Enter user-name: farha
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
```

Questions : 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: farha

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: farha

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.
```

Questions : 4

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

QUERY

```
SQL> connect
Enter user-name: farha
Enter password:
Connected.
```

```
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

Questions : 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','Khaid', '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	SEX	SALARY
LNAME			
ADDRESS			
SUPERSSN	DNO		
e1001	Archana		
Suresh			
13B,Highway Gardens,Kozhikode		Female	60000
SP1002	D001		
SSN	FNAME	SEX	SALARY
LNAME			
ADDRESS			
SUPERSSN	DNO		
e1002	Justin		
Varghese			
Rose Villa,Kochi		Male	50000
SP1001	D002		
SSN	FNAME	SEX	SALARY
LNAME			
ADDRESS			
SUPERSSN	DNO		
e1003	Meera		
Kumar			
11B,Arcadia Building,Mumbai		Female	70000
SP1004	D001		
SSN	FNAME	SEX	SALARY
LNAME			
ADDRESS			
SUPERSSN	DNO		
e1004	Kailas		
Nath			
V3,DD Homes,Bangalore		Male	30000
SP1003	D003		

```
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 Khaild Ashok Nagar,West Delhi SP1005	Sara	Female	45000
SSN	FNAME		
LNAME			
ADDRESS		SEX	SALARY
SUPERSSN	DNO		
e1006 Ashok LV Road,Bengaluru SP1005	Rahul	Male	55000

Questions : 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

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

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\XEHhours_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

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)

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)

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)

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)

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)

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)

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)

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 program to perform GraphQL.

OUTPUT

/geography

query add data edit info

SPARQL Query

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

Example Queries Selection of triples Selection of classes

SPARQL Endpoint /geography/ Content Type (SELECT) JSON Content Type (GRAPH) Turtle

```

1: prefix table:<http://www.mooney.net/geo#>
2: select ?name ?City
3: where
4: { ?geo table:isCityOf ?City
5: }
6:

```

Table Response 402 results in 0.018 seconds

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>

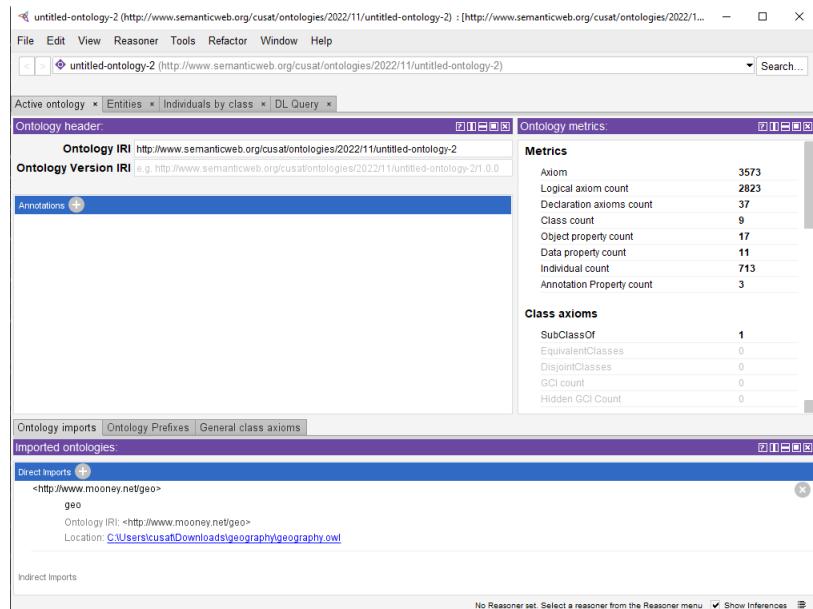
Simple view Ellipse Filter query results Page size 50

Microsoft Windows [Version 10.0.19044.251]
(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>fuseki-server --update --mem /ds
[02:23:55] INFO Server :: Apache Jena Fuseki 4.6.1
[02:23:58] INFO Config :: FUSEKI_HOME=C:\Users\cusat\Downloads\apache-jena-fuseki-4.6.1\apache-jena-fuseki-4.6.1\run
[02:23:58] INFO Config :: Shiro file: file:/C:/Users\cusat\Downloads\apache-jena-fuseki-4.6.1\apache-jena-fuseki-4.6.1\run\shiro.ini
[02:40:00] INFO Config :: Template file: templates/config-mem
[02:40:05] INFO Server :: Database: in-memory
[02:40:05] INFO Server :: Path = /ds
[02:40:05] INFO Server :: System
[02:40:05] INFO Server :: Memory: 1.2 GB
[02:40:05] INFO Server :: Port: 19001
[02:40:05] INFO Server :: OS: Windows 10 10.0 amd64
[02:40:05] INFO Server :: PID: 11892
[02:40:06] INFO Server :: Started 2022/12/15 10:24:06 IST on port 3030
[02:44:40] INFO Admin :: [3] Create database : name = /geography
[02:44:59] INFO Fuseki :: [7] POST http://localhost:3030/geography/data
[02:44:59] ERROR Fuseki :: [line: 1, col: 1] Content is not allowed in prolog.
[02:45:00] INFO Fuseki :: [7] Upload error: [line: 1, col: 1] Content is not allowed in prolog.
[02:45:00] INFO Fuseki :: [7] 500 Server Error (632 ms)
[02:46:12] INFO Fuseki :: [8] POST http://localhost:3030/geography/data
[02:46:12] ERROR Fuseki :: [line: 1, col: 1] Content is not allowed in prolog.
[02:46:12] INFO Fuseki :: [8] Upload error: [line: 1, col: 1] Content is not allowed in prolog.
[02:46:12] INFO Fuseki :: [8] 500 Server Error (417 ms)
[02:46:56] INFO Fuseki :: [9] POST http://localhost:3030/geography/data
[02:46:56] ERROR Fuseki :: [line: 1, col: 1] Content is not allowed in prolog.
[02:46:56] INFO Fuseki :: [9] Upload error: [line: 1, col: 1] Content is not allowed in prolog.
[02:46:56] INFO Fuseki :: [9] 500 Server Error (433 ms)
[02:48:13] INFO Fuseki :: [25] POST http://localhost:3030/geography/data
[02:48:13] ERROR Fuseki :: [line: 1, col: 1] Content is not allowed in prolog.
[02:48:14] INFO Fuseki :: [25] Upload error: [line: 1, col: 1] Content is not allowed in prolog.
[02:48:14] INFO Fuseki :: [25] 500 Server Error (477 ms)
[02:48:48] INFO Fuseki :: [29] POST http://localhost:3030/geography/data
[02:48:49] ERROR Fuseki :: [line: 1, col: 1] Content is not allowed in prolog.
[02:48:49] INFO Fuseki :: [30] Upload error: [line: 1, col: 1] Content is not allowed in prolog.
[02:48:49] INFO Fuseki :: [30] 500 Server Error (488 ms)
[02:50:02] INFO Fuseki :: [30] POST http://localhost:3030/geography/data
[02:50:02] INFO Fuseki :: [30] Filename: geography.owl, Content-Type=application/octet-stream, Charset=null => RDF/XML : Count=3589 Triples=3589 Quads=0
[02:50:03] INFO Fuseki :: [30] 200 OK (1.272 s)
[02:50:36] INFO Fuseki :: [33] POST http://localhost:3030/geography/
[02:50:36] INFO Fuseki :: [33] Query - SELECT ?subject ?predicate WHERE { ?subject ?predicate ?object } LIMIT 25
[02:50:36] INFO Fuseki :: [33] 200 OK (65 ms)
[02:56:18] INFO Fuseki :: [34] POST http://localhost:3030/geography/
[02:56:18] INFO Fuseki :: [34] Query - prefix table:<https://www.mooney.net/geo> select ?name where { ?geo table:isCityOf ?city }
[02:56:18] INFO Fuseki :: [34] 200 OK (6 ms)
[02:56:59] INFO Fuseki :: [35] POST http://localhost:3030/geography/
[02:56:59] INFO Fuseki :: [35] Query - prefix table:<https://www.mooney.net/geo#> select ?name where { ?geo table:isCityOf ?city }
[02:56:59] INFO Fuseki :: [35] 200 OK (5 ms)
[02:58:29] INFO Fuseki :: [36] POST http://localhost:3030/geography/
[02:58:29] INFO Fuseki :: [36] Query - prefix table:<https://www.mooney.net/geo#> select ?name where { ?geo table:isCityOf ?city }
[02:58:29] INFO Fuseki :: [36] 200 OK (6 ms)
[02:59:45] INFO Fuseki :: [37] POST http://localhost:3030/geography/
[02:59:45] INFO Fuseki :: [37] Query - prefix table:<https://www.mooney.net/geo#> select ?name where { ?geo table:isCityOf ?city }
[02:59:45] INFO Fuseki :: [37] 200 OK (4 ms)
[02:59:48] INFO Fuseki :: [38] POST http://localhost:3030/geography/
[02:59:48] INFO Fuseki :: [38] Query - prefix table:<https://www.mooney.net/geo#> select ?name where { ?geo table:isCityOf ?city }
[02:59:48] INFO Fuseki :: [38] 200 OK (5 ms)
[02:59:50] INFO Fuseki :: [39] POST http://localhost:3030/geography/
[02:59:50] INFO Fuseki :: [39] Query - prefix table:<https://www.mooney.net/geo#> select ?name where { ?geo table:isCityOf ?city }

```



JAVA BASE CONNECTIVITY

AIM

Develop program to implement Java Database Connectivity

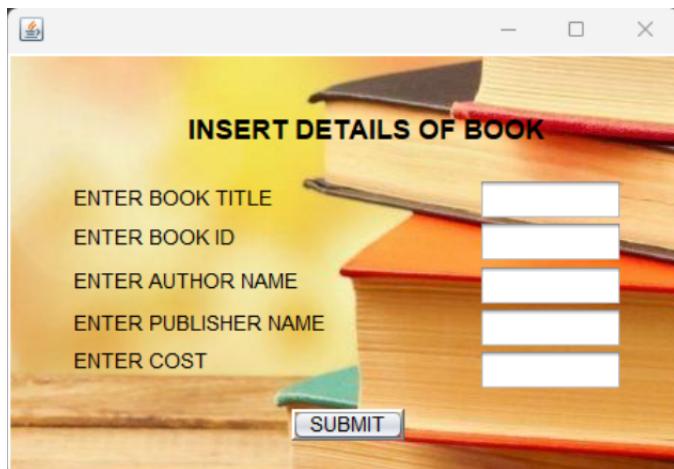
Question : 1

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

PROGRAM

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {  
    try {  
        Class.forName("com.mysql.cj.jdbc.Driver");  
    }  
    catch (ClassNotFoundException ex) {  
        Logger.getLogger(insert.class.getName()).log(Level.SEVERE, null, ex);  
    }  
    try (Connection con = DriverManager.getConnection  
        ("jdbc:mysql://localhost:3306/book","root","root")) {  
        String sql = "insert into books values(?,?,?,?,?)";  
        PreparedStatement ps = con.prepareStatement(sql);  
        ps.setString(1,jTextField1.getText());  
        ps.setString(2,jTextField2.getText());  
        ps.setString(3, jTextField3.getText());  
        ps.setString(4, jTextField4.getText());  
        ps.setInt(5,Integer.parseInt(jTextField5.getText()));  
        ps.execute();  
        JOptionPane.showMessageDialog(this,"data saved successfully");  
    }  
    catch(HeadlessException | NumberFormatException | SQLException e){  
        JOptionPane.showMessageDialog(this,e);  
    }  
}
```

OUTPUT

**Question : 2**

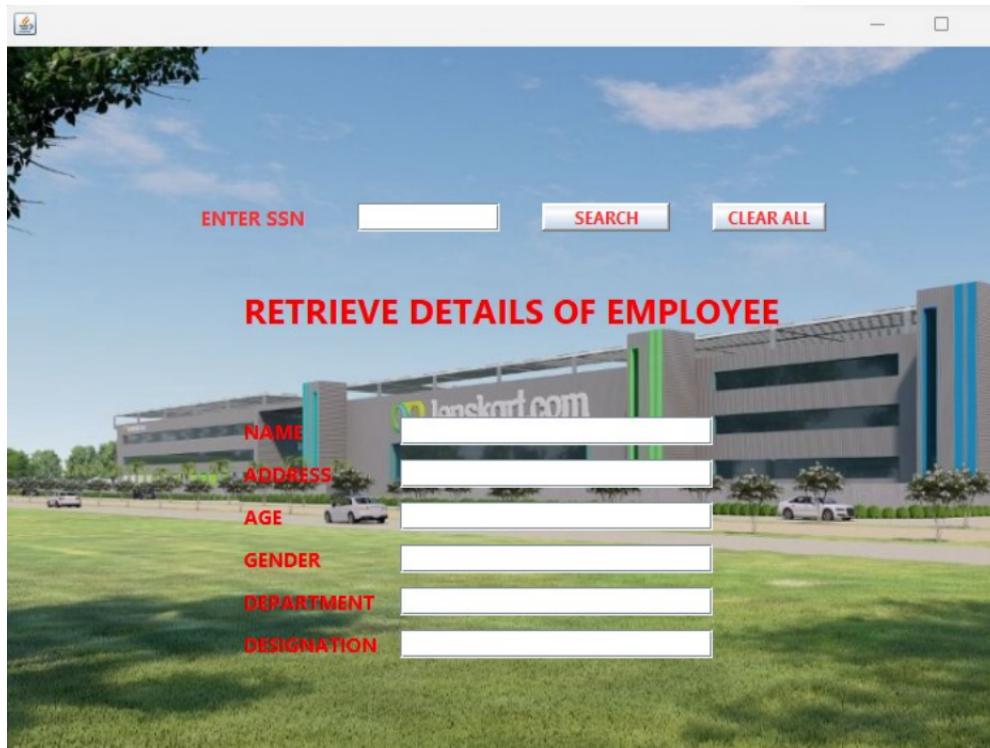
Write a program which connects to an online Employee database and retrieve the details of the employees in the database as per the schema.

PROGRAM

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {  
    try {  
        Class.forName("com.mysql.cj.jdbc.Driver");  
    }  
    catch (ClassNotFoundException ex) {  
        Logger.getLogger(Retrieve.class.getName()).log(Level.SEVERE, null, ex);  
    }  
    try (Connection con = DriverManager.getConnection  
        ("jdbc:mysql://localhost:3306/company", "root", "root")) {  
        String sql = "select * from employee where SSN = ?";  
        PreparedStatement ps = con.prepareStatement(sql);  
        ps.setString(1, ssn.getText());  
        ResultSet rs = ps.executeQuery();  
        if(rs.next()){  
            Name.setText(rs.getString("Name"));  
            address.setText(rs.getString("Address"));  
            Age.setText(rs.getString("Age"));  
            gender.setText(rs.getString("Sex"));  
        }  
        else{  
            JOptionPane.showMessageDialog(this, "data Not Found");  
        }  
    }  
    catch(HeadlessException | NumberFormatException | SQLException e){
```

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

OUTPUT



Question : 3

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

PROGRAM

```
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
try {
Class.forName("com.mysql.cj.jdbc.Driver");
}
catch (ClassNotFoundException ex) {
Logger.getLogger(Update.class.getName()).log(Level.SEVERE, null, ex);
}
try (Connection con = DriverManager.getConnection
("jdbc:mysql://localhost:3306/hospital","root","root")) {
```

```
String sql = "select * from patients where phone = ?";  
PreparedStatement ps = con.prepareStatement(sql);  
ps.setString(1, phone.getText());  
ResultSet rs = ps.executeQuery();  
if(rs.next()) {  
    String sql2 = "update patients set Name=? ,Gender = ?,  
    bld_grp = ?,Age = ?,disease = ? where Phone = ?;";  
    PreparedStatement ps2 = con.prepareStatement(sql2);  
    ps2.setString(1, Name.getText());  
    ps2.setString(2, gender.getText());  
    ps2.setString(3, bld_grp.getText());  
    ps2.setInt(4, Integer.parseInt(Age.getText()));  
    ps2.setString(5, disease.getText());  
    ps2.setInt(6, Integer.parseInt(phone.getText()));  
    ps2.execute();  
    JOptionPane.showMessageDialog(this, "Data Updated Successfully");  
}  
else {  
    JOptionPane.showMessageDialog(this, "data Not Found");  
}  
}  
}  
}  
}  
catch (HeadlessException | NumberFormatException | SQLException e) {  
    JOptionPane.showMessageDialog(this, e);  
}  
}
```

OUTPUT



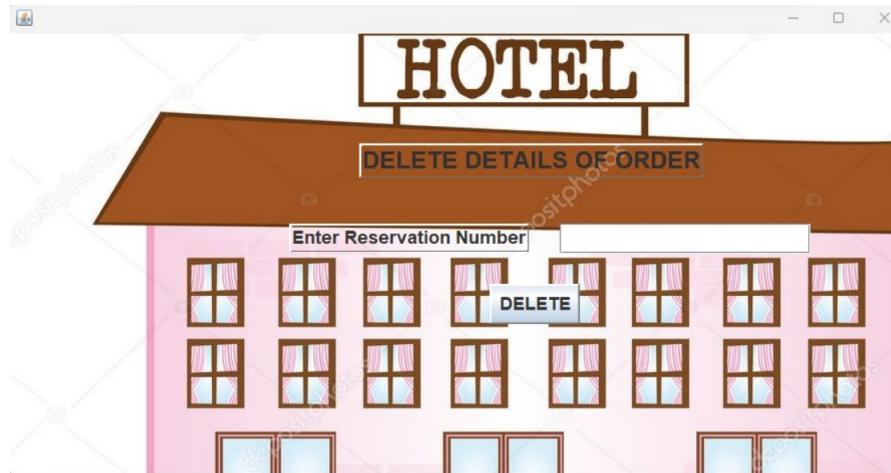
Question : 4

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

PROGRAM

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    String res = jTextField1.getText();
    try {
        Class.forName("com.mysql.cj.jdbc.Driver");
    }
    catch (ClassNotFoundException ex) {
        Logger.getLogger(delete_order.class.getName()).log(Level.SEVERE, null, ex);
    }
    try (Connection con = DriverManager.getConnection
        ("jdbc:mysql://localhost:3306/hotel","root","root")) {
        if (res.equals("")){
            JOptionPane.showMessageDialog(this,"Empty Field Not Allowed");
        }
        PreparedStatement statement = (PreparedStatement)con.prepareStatement
        ("select * from orders where reservation_no = ?;");
        statement.setString(1, res);
        ResultSet result = statement.executeQuery();
        if (!(result.next())){
            JOptionPane.showMessageDialog(this,"Data not in the DataBase");
        }
        else{
            PreparedStatement ps = (PreparedStatement)con.prepareStatement
            ("delete from orders where reservation_no = ?;");
            ps.setString(1,res);
            ps.execute();
            JOptionPane.showMessageDialog(this,"data deleted successfully");
            con.close();
            statement.close();
            result.close();
            ps.close();
        }
    }
    catch(HeadlessException | NumberFormatException | SQLException e){
        JOptionPane.showMessageDialog(this,e);
    }
}
```

OUTPUT



PROJECT

AIM

Develop an Application software using java and mySQL for an Information Management Purpose.

PROJECT DESCRIPTION

The project, Restaurant Management System is a GUI-based application. This project is the implementation of a restaurant management system that provides proper management of data in a centralized and organized manner and also provides a user friendly interface with which the customer and admin can interact easily with the just little or elementary knowledge of operating computers. Admin can alter the menu and user can order the food from the existing menu and can generate a bill. The main advantage is that , we can minimize the staff count as the customer and admin can do most of the services without depending on anyone.

USERS AND FUNCTIONALITIES

Admin functionalities

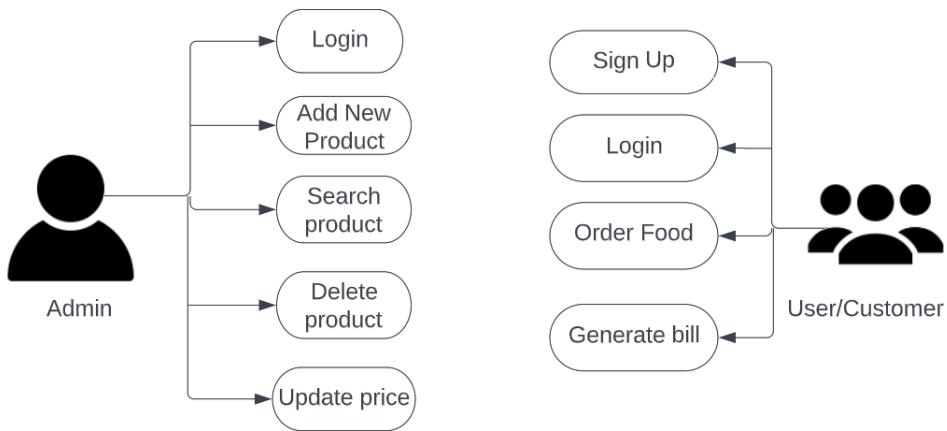
- Admin can login to the page where he can perform various operations.
- Admin can add items to the menu.
- Admin can search for an item.
- Admin can delete items from the menu.
- Admin can update the price of items from the menu.

User functionalities

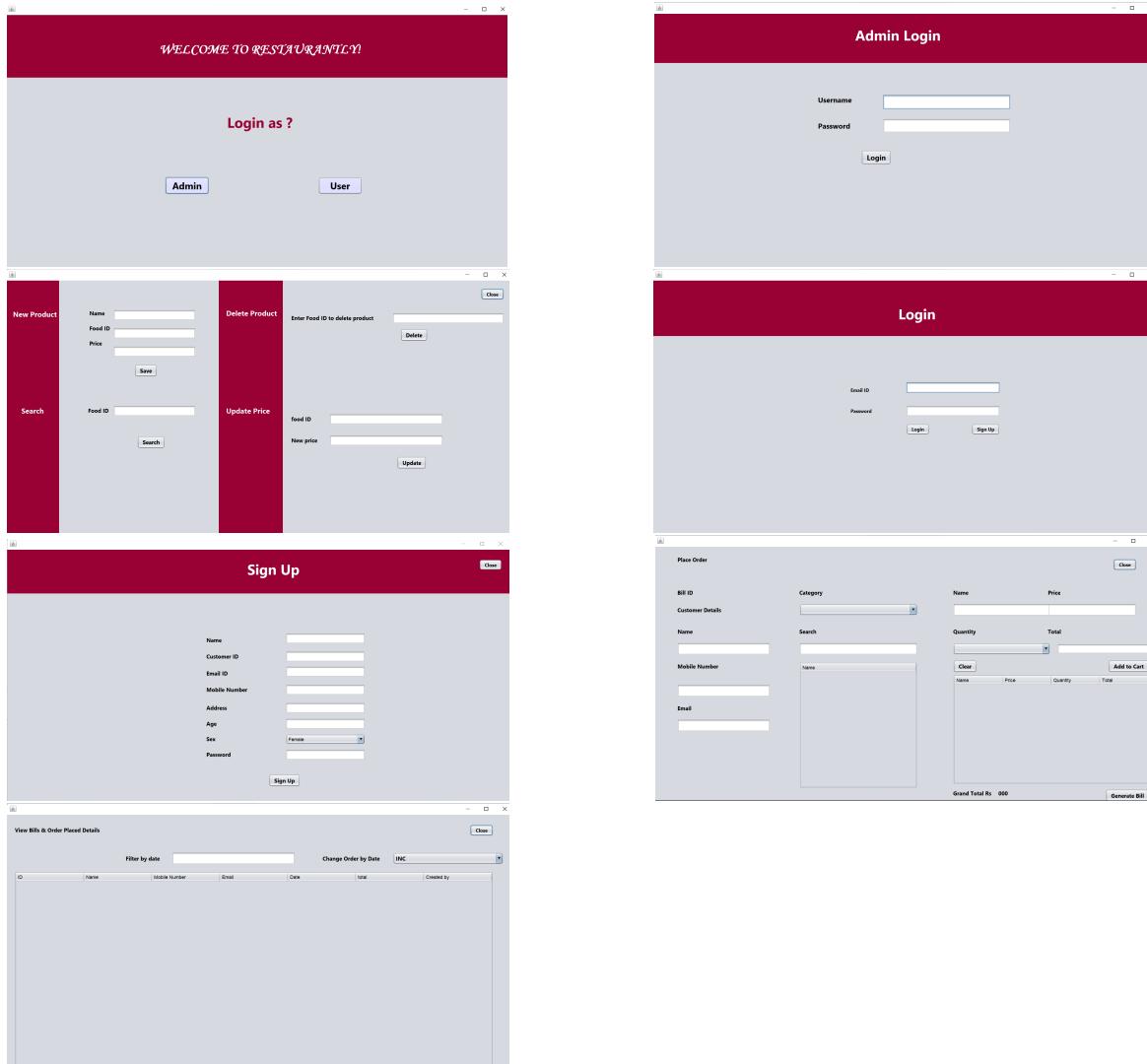
- User can login and perform operations if he/she is an existing user.
- User can sign Up and then perform if he/she is a new user.
- User can view the menu and select items to order
- User can generate the bill.

REFERENCE DESIGN

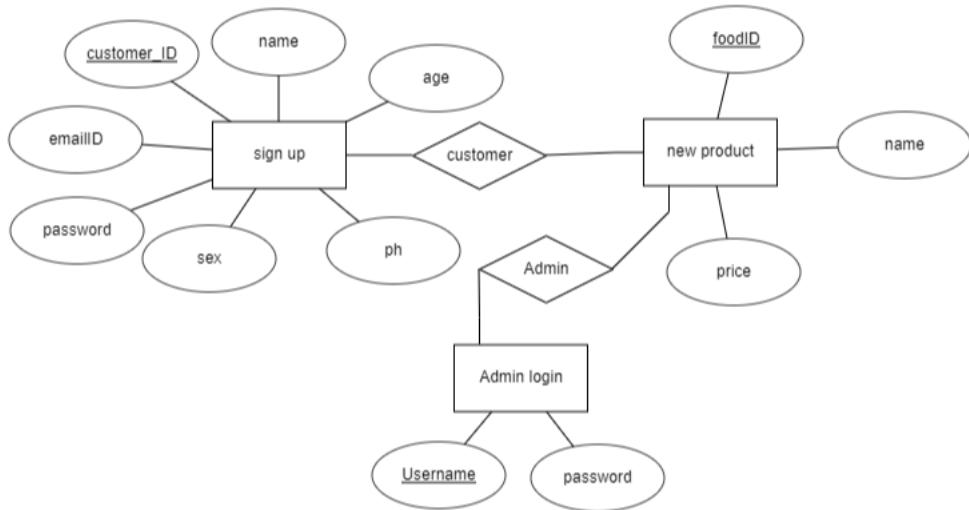
UseCase-Diagram



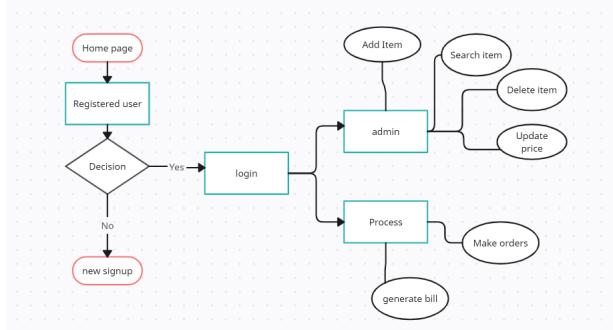
Wireframes



ER-Diagram



Activity-Diagram



SOFTWARE TOOLS

Apache Netbeans

Apache NetBeans is top level Apache Project dedicated to providing rock solid software development products (the Apache NetBeans IDE and the Apache NetBeans Platform) that address the needs of developers, users and the businesses who rely on NetBeans as a basis for their products; particularly, to enable them to develop these products quickly, efficiently and easily by leveraging the strengths of the Java platform and other relevant industry standards.

The two base products, the Apache NetBeans IDE and Apache NetBeans Platform, are free for commercial and non-commercial use, under the Apache license. The source code to both is available to anyone to reuse as they see fit, within the terms of use.

The Apache NetBeans project is also a vibrant community in which people from across the globe can ask questions, give advice, contribute and ultimately share in the success of our products. On the NetBeans mailing lists and forums, you will find posts from students, developers from top companies, and individuals looking to expand their skills.

MySQL

MySQL is an open-source relational database management system. As with other relational

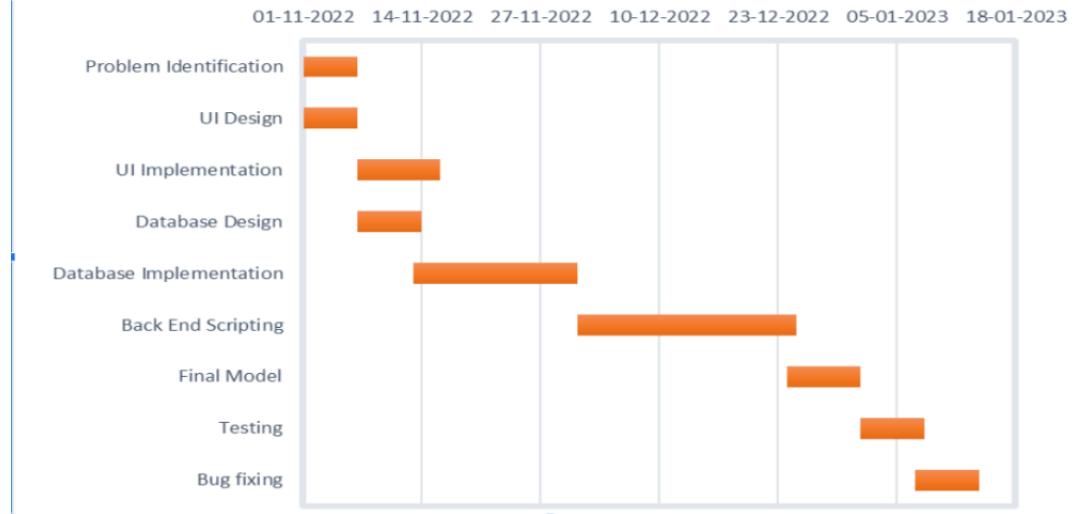
databases, MySQL stores data in tables made up of rows and columns. Users can define, manipulate, control, and query data using Structured Query Language, more commonly known as SQL.

Packages

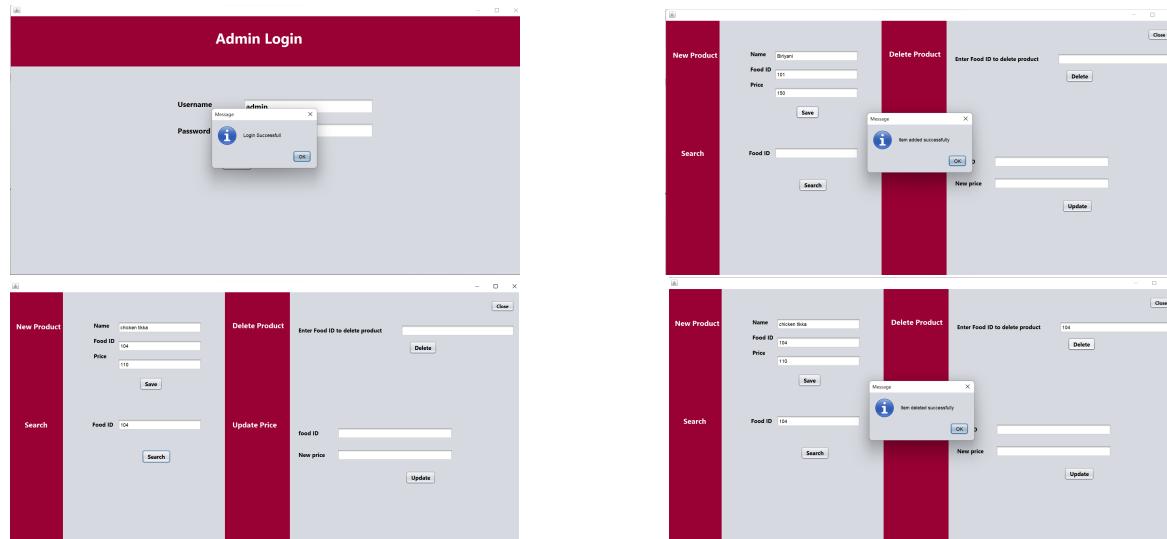
```
package com.mycompany.cafe;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;
import javax.swing.JFrame;
import javax.swing.JOptionPane;
```

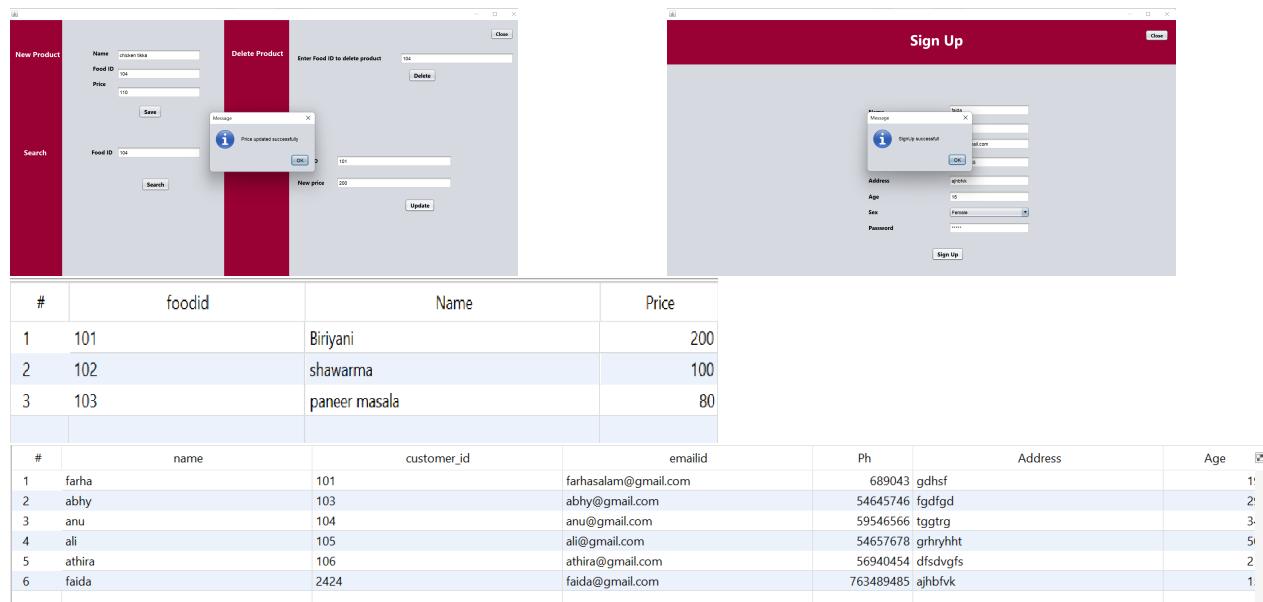
IMPLEMENTATION

Gantt-Chart



RESULT AND OUTPUT





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 Unsucces fully	Signin Unsucces 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

Restaurant Management System project provides facilities like adding new items to the menu ,searching items from the menu , deleting and updating existing items from the menu etc. The system allows to manage a restaurant in an efficient way.

REFERENCES

- MySQL
- JAVA Database Connectivity
- JAVA Swing