

Question 1

1. Create the following EMPLOYEE table, and fill the following data in it.

EMPNO	EMPNAME	DEPT	SALARY	DOJ	BRANCH
E101	Amit	Production	45000	12/03/2000	Banglore
E102	Ashish	HR	70000	03/07/2002	Banglore
E103	Sunita	Manager	120000	11/01/2001	Mysore
E104	Raj	IT	90000	02/10/2000	Mysore
E105	Ashish	IT	67000	01/08/2001	Mysore
E106	Amit	Civil	145000	20/09/2003	Mumbai

DeptName	DeptBuilding	DeptBudget
Production	Kalam Complex	1000000
HR	Bhargava Building	200000
Manager	Kailash Complex	300000
IT	Information Complex	250000
Civil	Sarovar Building	400000

Perform the following operations in SQL:

- Retrieve average salary of all employee
- Retrieve number of employee
- Retrieve distinct number of employee
- Retrieve the branch name which is paying maximum total salary
- Find the department name in which Employee named as "Ashish" is not working.
- Find the employee name who is getting second maximum salary.
- Display name of employee in descending order
- Find the name of department building of the department which is located in Mysore and having maximum budget.
- Up to what normal form the above mentioned tables are normalized? Explain with reasoning.

SQL Queries and Output

- Creating Database
 - create database labAssign;
- Switching to 'labAssign' Database
 - use labAssign;
- Creating the tables 'DEPARTMENT' and 'EMPLOYEE'
 - create table DEPARTMENT (DeptName varchar(20) unique not null, DeptBuilding varchar(50), DeptBudget int, primary key (DeptName));
 - create table EMPLOYEE (EMPNO varchar(10) unique not null, EMPNAME varchar(25), DEPT varchar(20), SALARY int, DOJ date, BRANCH varchar(30), primary key (EMPNO), foreign key (DEPT) references DEPARTMENT(DeptName) on update cascade on delete cascade);
- Inserting the data in the tables 'DEPARTMENT' and 'EMPLOYEE'
 - insert into DEPARTMENT values ('Production', 'Kalam Complex', 1000000), ('HR', 'Bhargava Building', 200000), ('Manager', 'Kailash Complex', 300000), ('IT', 'Information Complex', 250000), ('Civil', 'Sarovar Building', 400000);
 - insert into EMPLOYEE values ('E101', 'Amit', 'Production', 45000, '2000-03-12', 'Bangalore'), ('E102', 'Ashish', 'HR', 70000, '2002-07-03', 'Bangalore'), ('E103', 'Sunita', 'Manager', 120000, '2001-01-11', 'Mysore'), ('E104', 'Raj', 'IT', 90000, '2000-10-02', 'Mysore'), ('E105', 'Ashish', 'IT', 67000, '2001-08-01', 'Mysore'), ('E106', 'Amit', 'Civil', 145000, '2003-09-20', 'Mumbai');

```
mysql> select * from DEPARTMENT;
+-----+-----+-----+
| DeptName | DeptBuilding | DeptBudget |
+-----+-----+-----+
| Civil    | Sarovar Building | 400000 |
| HR       | Bhargava Building | 200000 |
| IT       | Information Complex | 250000 |
| Manager  | Kailash Complex | 300000 |
| Production | Kalam Complex | 1000000 |
+-----+-----+-----+
5 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM EMPLOYEE;
```

EMPNO	EMPNAME	DEPT	SALARY	DOJ	BRANCH
E101	Amit	Production	45000	2000-03-12	Banglore
E102	Ashish	HR	70000	2002-07-03	Banglore
E103	Sunita	Manager	120000	2001-01-11	Mysore
E104	Raj	IT	90000	2000-10-02	Mysore
E105	Ashish	IT	67000	2001-08-01	Mysore
E106	Amit	Civil	145000	2003-09-20	Mumbai

```
6 rows in set (0.00 sec)
```

- Perform the following operations in SQL

- a.) Retrieve average salary of all employee

SQL Query: select avg(SALARY) as AverageSalary from EMPLOYEE;

OUTPUT:

```
mysql> select avg(SALARY) as AverageSalary from EMPLOYEE;
```

AverageSalary
89500.0000

```
1 row in set (0.00 sec)
```

- b.) Retrieve number of employee

SQL Query: select count(EMPNO) as NumberOfEmployees from EMPLOYEE;

OUTPUT:

```
mysql> select count(EMPNO) as NumberOfEmployees from EMPLOYEE;
```

NumberOfEmployees
6

```
1 row in set (0.01 sec)
```

- c.) Retrieve distinct number of employee

SQL Query: select count(distinct(EMPNO)) as DistinctNumOfEmployees from EMPLOYEE;

Output:

```
mysql> select count(distinct(EMPNO)) as DistinctNumOfEmployees from EMPLOYEE;
```

DistinctNumOfEmployees
6

```
1 row in set (0.01 sec)
```

- d.) Retrieve the branch name which is paying maximum total salary

SQL Query: select BRANCH as maxTotalSalaryBranch from EMPLOYEE where SALARY in (select max(SALARY) from EMPLOYEE);

Output:

```
mysql> select BRANCH as maxTotalSalaryBranch from EMPLOYEE where SALARY in (select max(SALARY) from EMPLOYEE);
+-----+
| maxTotalSalaryBranch |
+-----+
| Mumbai              |
+-----+
1 row in set (0.00 sec)
```

- e.) Find the department name in which Employee named as “Ashish” is not working.

SQL Query: select DEPT from EMPLOYEE where not DEPT in (select DEPT from EMPLOYEE where EMPNAME = 'Ashish');

Output:

```
mysql> select DEPT from EMPLOYEE where not DEPT in (select DEPT from EMPLOYEE where EMPNAME = 'Ashish');
+-----+
| DEPT |
+-----+
| Civil |
| Manager |
| Production |
+-----+
3 rows in set (0.00 sec)
```

- f.) Find the employee name who is getting second maximum salary.

SQL Query: select max(SALARY) from EMPLOYEE where SALARY not in (select max(SALARY) from EMPLOYEE);

Output:

```
mysql> select max(SALARY) from EMPLOYEE where SALARY not in (select max(SALARY) from EMPLOYEE);
+-----+
| max(SALARY) |
+-----+
| 120000 |
+-----+
1 row in set (0.00 sec)
```

- g.) Display name of employee in descending order

SQL Query: select EMPNAME from EMPLOYEE order by EMPNAME DESC;

Output:

```
mysql> select EMPNAME from EMPLOYEE order by EMPNAME DESC;
+-----+
| EMPNAME |
+-----+
| Sunita |
| Raj |
| Ashish |
| Ashish |
| Amit |
| Amit |
+-----+
6 rows in set (0.00 sec)
```

h.) Find the name of department building of the department which is located in Mysore and having maximum budget.

SQL Query: select DeptBuilding from DEPARTMENT where DeptBudget in (select max(DeptBudget) from DEPARTMENT where DeptName in (select DEPT from EMPLOYEE where BRANCH = 'Mysore'));

Output:

```
mysql> select DeptBuilding from DEPARTMENT where DeptBudget in (select max(DeptBudget) from DEPARTMENT where DeptName in (select DEPT from EMPLOYEE where BRANCH = 'Mysore'));
+-----+
| DeptBuilding |
+-----+
| Kallash Complex |
+-----+
```

i.) Up to what normal form the above mentioned tables are normalized? Explain with reasoning.

The tables are normalized till BCNF. The explanation for the same is given below.

In the table EMPLOYEE, the following set of functional dependencies hold.

$EMPNO \rightarrow EMPNAME$

$EMPNO \rightarrow DEPT$

$EMPNO \rightarrow SALARY$

$EMPNO \rightarrow DOJ$

$EMPNO \rightarrow BRANCH$

Since, none of the attributes are multivalued, hence table is in 1NF.

The only candidate of the EMPLOYEE table is {EMPNO}, hence all other attributes are non-prime and they only depend on the {EMPNO}. Hence, there are no partial dependencies. Hence, the table is in 2NF.

As there are no transitive dependencies, the table is in 3NF.

As there is a candidate key on the left side of each dependency, the table is in BCNF.

In the table DEPARTMENT, the following set of functional dependencies hold.

$DeptName \rightarrow DeptBuilding$

$DeptName \rightarrow DeptBudget$

Since, none of the attributes are multivalued, hence table is in 1NF.

The only candidate of the DEPARTMENT table is {DeptName}, hence all other attributes are non-prime and they only depend on the {DeptName}. Hence, there are no partial dependencies. Hence, the table is in 2NF.

As there are no transitive dependencies, the table is in 3NF.

As there is a candidate key on the left side of each dependency, the table is in BCNF.

Question 2

2. In the EMPLOYEE table in Q1, add a column named **RAISE** having values as 10% raise of their current SALARY.

SQL Queries and Output

- Adding column 'RAISE' in 'EMPLOYEE' table

- alter table EMPLOYEE add RAISE int;
- Adding values to 'RAISE' column
 - update EMPLOYEE set RAISE = 1.1 * SALARY;

```
mysql> select * from EMPLOYEE;
```

EMPNO	EMPNAME	DEPT	SALARY	DOJ	BRANCH	RAISE
E101	Amit	Production	45000	2000-03-12	Banglore	49500
E102	Ashish	HR	70000	2002-07-03	Banglore	77000
E103	Sunita	Manager	120000	2001-01-11	Mysore	132000
E104	Raj	IT	90000	2000-10-02	Mysore	99000
E105	Ashish	IT	67000	2001-08-01	Mysore	73700
E106	Amit	Civil	145000	2003-09-20	Mumbai	159500

```
6 rows in set (0.00 sec)
```

Question 3

3. In the table given in the Q1, remove all the rows having same EMPNAME and then delete all the rows having same BRANCH names. Display the resultant table as the output.

SQL Queries and Output

```
mysql> select * from EMPLOYEE;
```

EMPNO	EMPNAME	DEPT	SALARY	DOJ	BRANCH	RAISE
E101	Amit	Production	45000	2000-03-12	Banglore	49500
E102	Ashish	HR	70000	2002-07-03	Banglore	77000
E103	Sunita	Manager	120000	2001-01-11	Mysore	132000
E104	Raj	IT	90000	2000-10-02	Mysore	99000
E105	Ashish	IT	67000	2001-08-01	Mysore	73700
E106	Amit	Civil	145000	2003-09-20	Mumbai	159500

```
6 rows in set (0.00 sec)
```

- Removing all the rows having same EMPNAME

SQL Query: