DATABASE MANAGEMENT SYSTEMS

Course Code: CSE2007	Slot: L13+L14
Class Number: AP2023246000686	Venue: CB-102
Assignment No.: 6	Date:07-03-2024

Reg. No:22BCE8609

Name:Gourab Choudhury

School & Programme: B.Tech CSE Core

Instructions

1. Submit Your Assignment at the following link. Click Here or https://forms.gle/ChkVB8gP5uckTUYr8
2. Submit the Assignment in PDF format.
3. Submit the assignment on or before 04/March/2024.
4. Copy the code, take a picture of the output, and paste it into a Word document. Convert the word doc into PDF and submit.
5. The file name will be your registration number and name.
6. **Assignments Programs:**

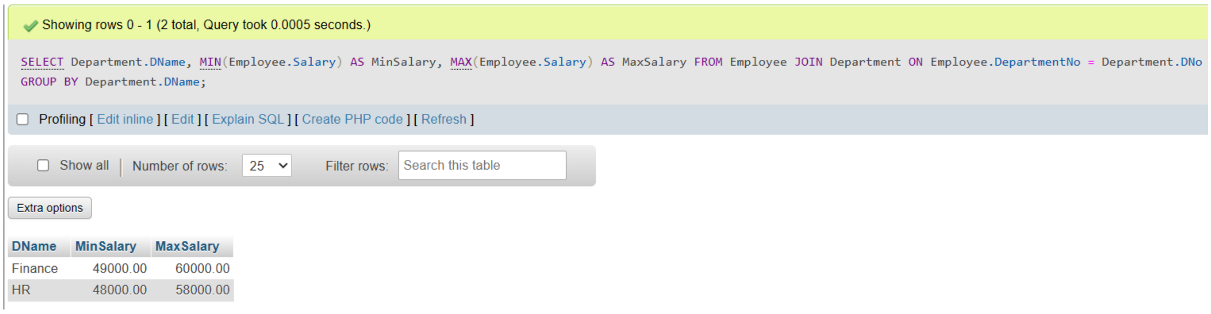
Solve the following using machine learning techniques.	
A.	For each department display the minimum and maximum employee salaries
B.	Print the average annual salary.
C.	Count the number of employees over 30 age.
D.	Print the Department name and average salary of each department.
E.	Create a view to display the employee details who is working in IT department.
F.	Create a logical table to store employee details who is getting salary more than 10000.
G.	Create a table to store the employees details based on the department no
H.	List the names of all managers who have no dependents.
I.	List the employee's names and the department names if they happen to manage a department.

Program No:A

Title: For each department display the minimum and maximum employee salaries

```
SELECT Department.DName,
       MIN(Employee.Salary) AS MinSalary,
       MAX(Employee.Salary) AS MaxSalary
FROM Employee
JOIN Department ON Employee.DepartmentNo = Department.DNo
GROUP BY Department.DName;
```

Output:



✓ Showing rows 0 - 1 (2 total, Query took 0.0005 seconds.)

```
SELECT Department.DName, MIN(Employee.Salary) AS MinSalary, MAX(Employee.Salary) AS MaxSalary FROM Employee JOIN Department ON Employee.DepartmentNo = Department.DNo
GROUP BY Department.DName;
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
□ Show all | Number of rows: 25
Filter rows: Search this table
Extra options

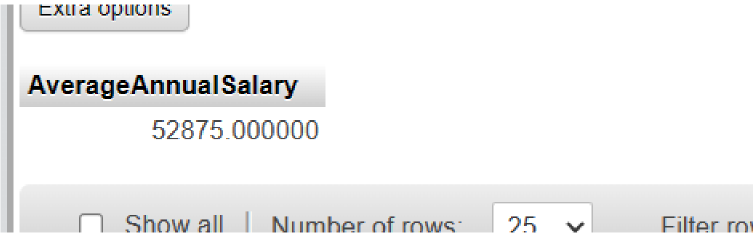
DName	MinSalary	MaxSalary
Finance	49000.00	60000.00
HR	48000.00	58000.00

Program No:B

Title: Print the average annual salary.

```
SELECT AVG(Salary) AS AverageAnnualSalary
FROM Employee;
```

Output:



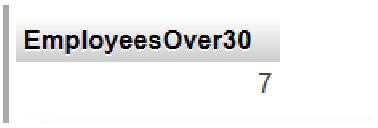
AverageAnnualSalary
52875.000000

Program No:C

Title: Count the number of employees over 30 age.

```
SELECT COUNT(*) AS EmployeesOver30

FROM Employee

WHERE YEAR(NOW()) - YEAR(Birthday) > 30;
```

Output:



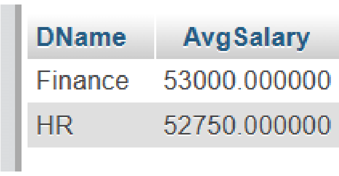
EmployeesOver30
7

Program No:D

Title: Print the Department name and average salary of each department.

```
SELECT Department.DName,

    AVG(Employee.Salary) AS AvgSalary

FROM Employee

JOIN Department ON Employee.DepartmentNo = Department.DNo

GROUP BY Department.DName;
```

Output:



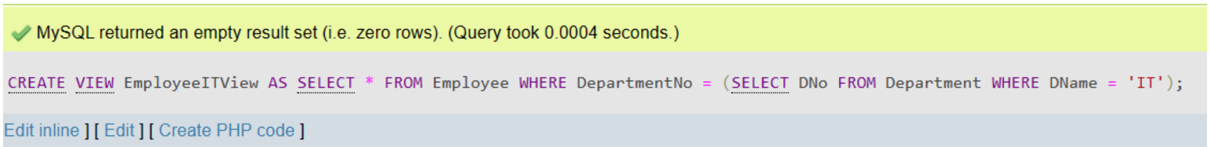
DName	AvgSalary
Finance	53000.000000
HR	52750.000000

Program No:E

Title: Create a view to display the employee details who is working in IT department.

```
CREATE VIEW EmployeeITView AS

SELECT *

FROM Employee

WHERE DepartmentNo = (SELECT DNo FROM Department WHERE DName = 'IT');
```

Output:



✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0004 seconds.)

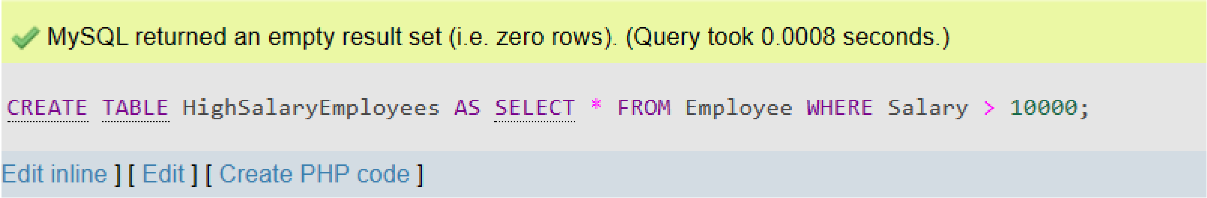
```
CREATE VIEW EmployeeITView AS SELECT * FROM Employee WHERE DepartmentNo = (SELECT DNo FROM Department WHERE DName = 'IT');
```

Program No:F

Title: Create a logical table to store employee details who is getting salary more than 10000.

```
CREATE TABLE HighSalaryEmployees AS

SELECT *

FROM Employee

WHERE Salary > 10000;
```

Output:



✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0008 seconds.)

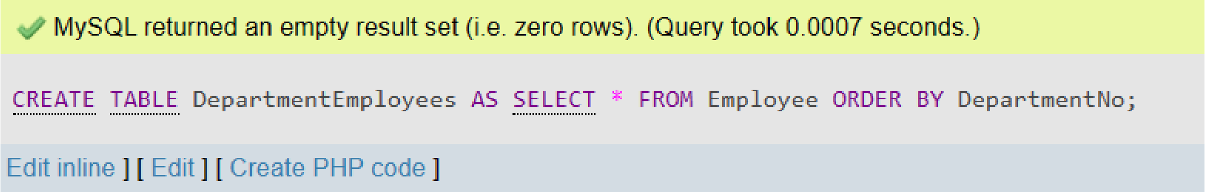
```
CREATE TABLE HighSalaryEmployees AS SELECT * FROM Employee WHERE Salary > 10000;
```

Program No:G

Title: Create a table to store the employees details based on the department no

```
CREATE TABLE DepartmentEmployees AS

SELECT *

FROM Employee

ORDER BY DepartmentNo;
```

Output:



✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0007 seconds.)

```
CREATE TABLE DepartmentEmployees AS SELECT * FROM Employee ORDER BY DepartmentNo;
```

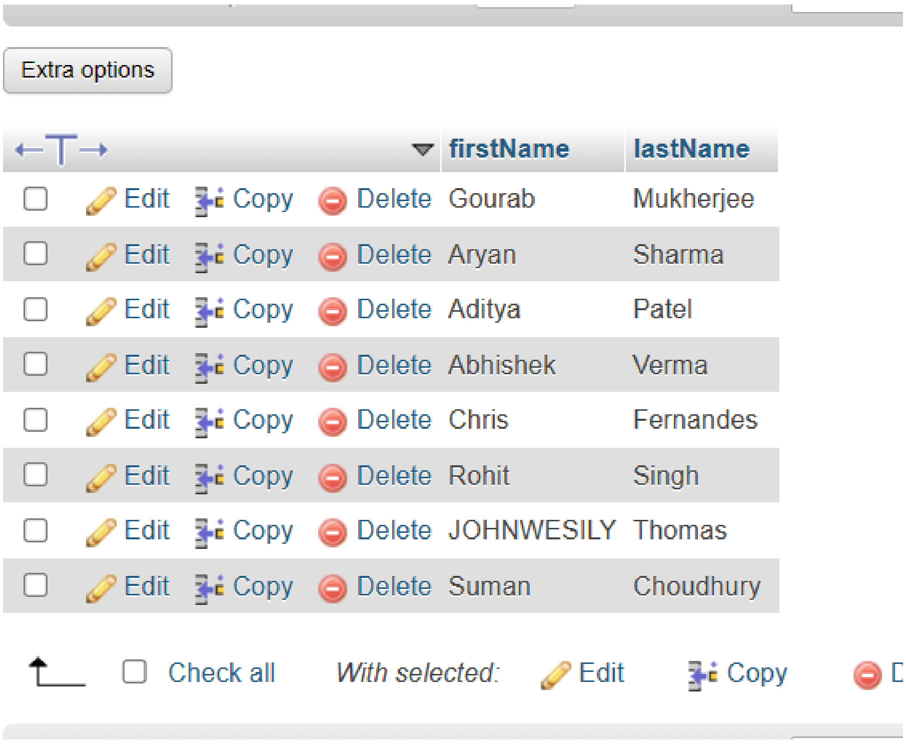
Program No:H

Title: List the names of all managers who have no dependents.

```
SELECT Manager.firstName, Manager.lastName

FROM Employee AS Manager
```

```
WHERE Manager.SSN NOT IN (SELECT DISTINCT SupervisorSSN FROM Employee WHERE
SupervisorSSN IS NOT NULL);
```

Output:



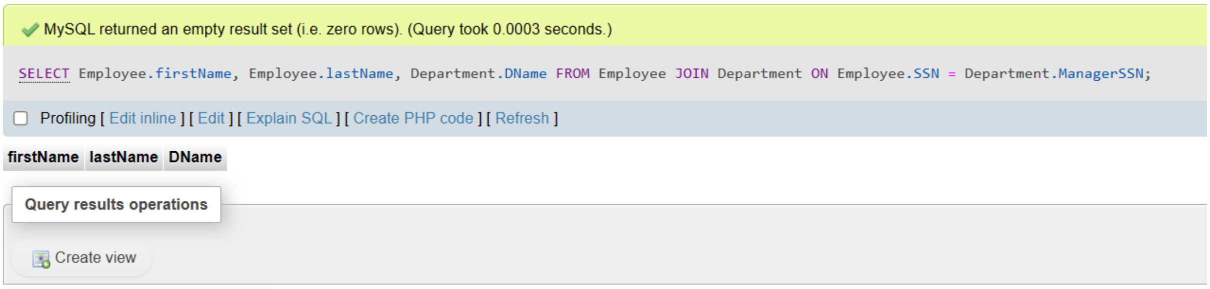
firstName	lastName
Gourab	Mukherjee
Aryan	Sharma
Aditya	Patel
Abhishek	Verma
Chris	Fernandes
Rohit	Singh
JOHNWESILY	Thomas
Suman	Choudhury

Program No:I

Title: List the employee’s names and the department names if they happen to manage a department.

```
SELECT Employee.firstName, Employee.lastName, Department.DName

FROM Employee

JOIN Department ON Employee.SSN = Department.ManagerSSN;
```

Output:



✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0003 seconds.)

```
SELECT Employee.firstName, Employee.lastName, Department.DName FROM Employee JOIN Department ON Employee.SSN = Department.ManagerSSN;
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

firstName	lastName	DName

Query results operations
Create view