1. Problem Statement: Design and Develop SQL DDL statements on Schema given. Schema: employee_master(emp_id,firstname, middle, lastname, department, managerid.) branch_master(branched , branchname)

1. Insert records in branch_master

2. Insert records in employee_master.

3. Create index on emp_name column of employee_master.

4. Create a view containing employee details

2. Problem Statement: Design following SQL DML statements: Create a university/college database containing following tables- Student (stud_id, deptnm, sem, name, yr, credits) Teaches (teacher_id,teacher_name,salary, deptnm)

1. Insert records in to all tables.

2. Update record on student tables as department name comp to IT.

3. Find the department that has highest or average salary

4. Delete the records of all teachers with salary below2000.

5. Find the sum of salary of each department.

3. Emp(empIdint,empNamevarchar(10),empSalint,empDeptId int) Dept (deptIdint,deptNamevarchar(10)) Set appropriate primary key and Foreignkey.

1. Create table Emp(empID) and Dept(deptID).

2. Add Not Null constraint to empName.

3. Insert few Records.

4. Add column deptloc varchar(10) to dept table

4. Emp(empId int,empName varchar (10), empSal int,empDeptId int) Dept(deptId int,deptName varchar(10))

1. Insert few Record.

2. List employees belonging to department 30,40,or10

3. List the employee details whose salary is between10000 to 30000.

4. List total no of employee.

5. List average sal of each deptID.

6. List employee details in ascending order of salary.

5. Demonstrate all types of JOIN on following schema customer (customer_id, first_name) orders(order_id,amount,customer_id)

6. Create a collection Employee in MongoDB(emp_id,emp_name,dept_name,sal)

1. Insert few documents in collection.

2. Find employees having salary greater than 50000

3. Find employees having salary between 50000 and80000

4. Find employees having salary more than 60000 from ‘HR’ department

5. Delete employees from ‘Finance’ department having salary less than 10000.

7. Create a collection Student in Mongo DB(stud_id,stud_name,dept_name,marks)

1. Insert few documents in collection.

2. Find students having marks greater than 50.

3. Find students having marks between50and 80.

4. Find students having marks more than 60from ‘Computer’ department.

5. Update marks of all students from ‘Civil’ department. Set marks to 30.(Use update())

6. Delete students from ‘Chemical’ department having marks less than 30

8. Create a collection Book in MongoDB(Title,Description,Author,Publisher, URL,no_of_likes)

1. Add documents in collection.

2. Display all documents in collection.

3. Display a list stating how many books are written by each author.

4. Calculate the sum of no_of_likes from all documents in the collection for each Author.

5. Calculates the average of no_of_likes from all documents in the collection for each Author.

9. Create a collection Book in MongoDB(Title,Description,Author,Publisher, URL,no_of_likes)

1. Add documents in collection.

2. Display all documents in collection.

3. Display a list stating how many books are published by each Publisher.

4. Gets the minimum of no_of_likes from all documents in the collection for each Author.

5. Gets the maximum of no_of_likes from all documents in the collection for each Author.