

1.

Mike is a software developer in a Marketing company. The business analysts have provided Mike a requirement to develop an application which stores the department and employee information. The application should also provide a feature for users to query the records based on employee id.

The following are the information which needs to be stored

Employees information such as employee id, employee name, salary, Employee address, contact number, and the department where he works.

Department information such as of the firm such as department id, department name, department head and number of employees in the department.

Additional Requirement:

1. Duplicate Employee data should not be stored in the system.
2. Duplicate department data should not be stored in the system.
3. Salary should be between 1000 and 30001.

Problem # 1 Creating Tables : Create following tables using My SQL command line Client and DDL's.

i. Create Department table

- a. Department\_ID – Primary Key - Number
- b. Department\_Name - Varchar
- c. Department\_Head – Varchar
- d. Department\_Description – Varchar

## OUTPUT:

```
mysql> CREATE TABLE Department (  
->     Department_ID INT NOT NULL AUTO_INCREMENT,  
->     Department_Name VARCHAR(255) NOT NULL,  
->     Department_Head VARCHAR(255),  
->     Department_Description VARCHAR(255),  
->     PRIMARY KEY (Department_ID)  
-> );  
Query OK, 0 rows affected (0.00 sec)
```

ii. Create Employee table

- a. Employee\_Id- Primarykey- Number
- b. Employee\_Name- Varchar
- c. Employee\_Address- Varchar
- d. Employee\_Salary- Decimal Number
- e. Employee\_Contact\_No- Number
- f. Department\_Id- Number (Foreign Key)

## OUTPUT:

```
mysql> CREATE TABLE Employee (  
->     Employee_Id INT NOT NULL AUTO_INCREMENT,  
->     Employee_Name VARCHAR(255) NOT NULL,  
->     Employee_Address VARCHAR(255),  
->     Employee_Salary DECIMAL(10, 2),  
->     Employee_Contact_No BIGINT,  
->     Department_Id INT,  
->     PRIMARY KEY (Employee_Id),  
->     FOREIGN KEY (Department_Id) REFERENCES Department(Department_ID)  
-> );  
Query OK, 0 rows affected (0.00 sec)
```

iii. Create a constraint on salary to ensure salary between 1000 and 1000000.

## OUTPUT:

```
mysql> ALTER TABLE Employee  
-> ADD CONSTRAINT chk_salary  
-> CHECK (Employee_Salary BETWEEN 1000 AND 1000000);  
Query OK, 0 rows affected (0.01 sec)  
Records: 0 Duplicates: 0 Warnings: 0
```