Final Project Proposal

**Employee Payroll Management System**

**Name of Project: Employee Payroll Management System**

Summary:

1. Employee Information

Employee data is very essential in order to maintain a proper record of the employees and there personal information for various purposes like contacting them for inviting for certain summit, feedback of the company from the employee data

2. Maintaining Salary

Very important to keep this data which will help not only the managers and the HR to keep a track of the employee salaries but also help the company or its board to analyze what amount they are spending on a particular employee of a particular company

3. Work Location

It is very much important for an organization small or big to have a record of all the work locations they operate from to see how they can develop in that particular region and also increase the hiring in that region.so that the organization can increase their Market Outreach that area.

4. Projects

In order to be successful company should be involved in various projects, so they also need to maintain the record of the salaries each employee is being paid for a particular type of project he/she is working on

Company List of Entities:

Employee

Employee table will include all the personal details of the employee and would be very much cover overall information of that particular employee

Salary

Salary Table will cover all the current and previous salaries an employee had or currently has. This table will help a manager/ an HR to analyze which employee has been given promotion on which date or when did his salary grade changed

Department

Department Table maintains the data of the all the possible departments an employee can belong to

Account Details

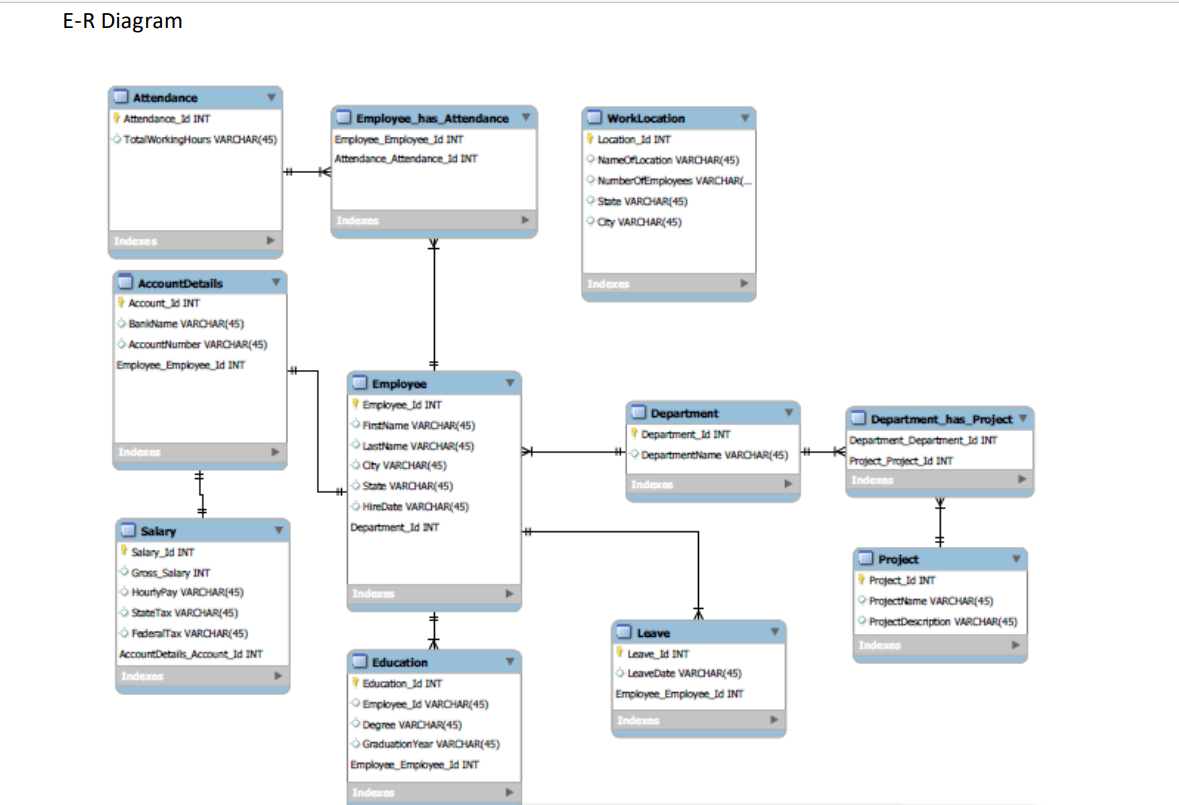
Account Details Table will maintain the data regarding the accounts which the employee has connected with the company for his/her salary to be credited Attendance. This table includes all the data of the employees attendance which includes the number of hours an employee has worked in a week Project . This table includes the data of all the projects a particular company is working on or the projects on which the company is going to work in the future Education. The Education Table keeps the track of the education of the employee including his degrees achieved until now

Work Location

The name of the table tells you most of the things. This table includes the location of the office, which city is it located, which state it is in and also tracks the number of employees in a particular location

Leave

Leave table keeps the record of the number of leaves an employee takes or has taken over the course of any month or an year



**Exercise:**

1. Create a Database and call it “payroll”
2. Understand the ER diagram and write at least 10 sentences.
3. Create the tables as per E-R diagram (you can refer to DDL.txt).

*Note: the column names without symbols (diamond/flash symbol) is for reference only. For ex: Education table has only 4 fields. The last field (Employee\_employee\_id) is just for reference to tell you that you can join Education with employee table using employee\_id)*

1. Alter the table Employee and add column department\_id INT.
2. Alter the table work location and add Column Employee\_id INT.
3. Insert the data into tables in ER diagram (Refer to DML.txt)
4. What are the different ways to insert or load data into a table? (Research on these 3 ways : INSERT ,INSERT ALL , CTAS) and explain with an example.
5. Create a view call it as “Education\_View” and the view must contain Degree and count of degrees:

Output data format: (Degree| COUNT (DEGREE))

1. Create a trigger on Leaves table whenever there is an entry (INSERT) into Leaves table. Trg\_Leaves\_Log should have an entry (INSERT).

\*Insert a record in Leaves table and test this scenario.

1. Create a trigger on Salary table whenever there is an INSERT\UPDATE\DELETE into Salary table. Trg\_Salary\_log should have an entry (INSERT).

\*Insert, Delete and update a record in salary table and test this scenario.

1. Create a trigger on Employee table whenever there is an UPDATE into Salary table. Trg\_Employee\_log should have an entry (INSERT).
2. Bring the table back to earlier state (Leaves and Salary) tables must contain same value and records after sec.9 and sec.10
3. Create a function “Employee\_salary”: The function should accept employee\_id and return his Gross salary along with Employee Name
4. Create a function “Employee\_Leaves”: The function should accept employee name and return how many days employees has taken leave**. If an employee has not taken any leaves then it must return ‘0’**

**\*If any attributes is not available then it must be replaced as “blank or NULL”**

1. Create procedure “calculate\_salaries”: The procedure should calculate the employee with least salary after tax and provide and increment of 10% salary (update the salary table).
2. A Select query to provide Salary\_range and count of employee

*(3 ranges : 0- 50K 🡪 LOW,50-100K 🡪MEDIUM, >100 🡪HIGH*

*Output Format : Salary\_range|COUNT*

1. *Write a query to give employees who has never taken a leave.*
2. *Write a query to gives employess who has taken maximum and minimum number of leaves*
3. *Write a query to find employee details (name,hire\_date etc.,) who work in ‘New york’location*
4. *Write a query to find the employee details (name,hire\_date etc.,) who work in ‘CA’ location*
5. *Write a Query to diplay employee\_name and sort the data based on desc salary (using rank,dense\_rank and rownumber)*
6. Write a query to find the employee with 4th highest salary using rank and with out rank
7. Create a procedure “updating\_employee\_dept”: The proc should accept 2 parameters:

Employee\_id and department\_id. The procedure should update the employees department.

For example : CALL updating\_employee\_dept (‘101’,8) should result in employee 101 moving from 1 to 8.

1. Check step#11, and see if the trigger worked.(Trg\_Employee\_log).
2. Create a view “Employee\_details”: It must have following columns : Employee\_id, First\_name,Degree,Salary,Location,Project,Education and leave.
3. Repeat the step#20 and create a materialized view