CSE 4508 – RDBMS Programming Lab

Lab 3

Task A:

A telecom company, OraclePHONE, maintains a database for its customers, to serve various purposes. The most essential information the company is interested to maintain are (but not limited to):

A unique Customer ID, Customer Name, Date of Birth, Permanent Address and Subscription Info.

The address should be handled separately. Create its own table and store information such as division (such as Khulna or Dhaka) and district (such as Dhaka or Gazipur). Ensure that this table is connected to the main table via keys.

Subscription Info should be handled separately as well. It should have a "Subscriber_Level" field with values such as "Bronze" or "Silver" or "Platinum". Most importantly, there should be a field called "Lifetime Usage" which is a value to represent the total BDT worth of usage done by that subscriber over their lifetime. Ensure that the table is similarly connected to the main table via keys.

1. First implement the above system in SQL.
2. Insert some data in all relevant tables created. (at least 5 entries)
3. Answer the following (should involve Join operations and subqueries):

- Display the name, date of birth, district and division of all Platinum status individuals whose current lifetime usage is greater than twice the current highest Silver status.

- Display the Name, Customer ID and Lifetime Usage of the top 5 highest users in OraclePHONE history.
- Update the Subscription Info table for those 5 individuals who fulfill the criteria set by the immediately previous query, such that their "Subscriber_Level" field is now set to "Elite" (Hint: Use another level of subquery)

Task B:
IUT maintains database where there are two tables. First one is employee table and the second one in salary table. The employee table has two attributes(employee_id, department_id) and the salary table has three attributes (employee_id, amount, pay_date) as shown below.

Table 1: salary table

employee_id	amount	pay_date
1	9000	2017-03-31
2	6000	2017-03-31
3	10000	2017-03-31
1	7000	2017-02-28
2	6000	2017-02-28
3	8000	2017-02-28

Table 2: employee table

employee_id	department_id
1	1
2	2
3	2

- Show the comparison result (higher/lower/same) of the average salary(per month) of employees in a department to the company's average salary using appropriate SQL query. For the above example, the result should be:

pay_month	department_id	comparision
2017-03	1	higher
2017-03	2	lower
2017-02	1	same
2017-02	2	same

Note: you might need to use case...when statement. Follow the link to learn about it.