

## Lab 1 & 2 -- Questions & Instructions ( Aug 26, 2021 & Sep 2, 2021 )

**Total points = 90**

### Instructions:

After the demos (on Aug 26, 2021 ), you are now required to:

- Working hands-on on the demos and writing queries for the following questions; feel free to innovate and try new things :)...
- Submission = A singular pdf file (File Name: <RollNo\_FirstName\_Lab1.pdf>) with solutions and observations per question (Deadline: Sep 3, 2021 )
- It is expected that the next lab session ( Sep 2, 2021 ) will involve interactions, discussions and concerns on the following questions, and any others you may tried working on; the deadline is thus the next working day
- It would be great if you could work on the questions in the [Non-graded sheet](#), during Lab hours as well, for initiation of discussions and resolution of problems....

✓ 1. Create a table: Employee\_Info(emp\_id, emp\_name, dept, salary) **(5 points)**

✓ Write SQL queries to: **(3 \* 5 = 15 points)**

- ✓ Find the third\_highest salary from the Employee\_Info table
- ✓ Display the first and last record from the Employee\_Info table
- ✓ Copy all rows of the Employee\_Info table into another new table Emp\_Details

✓ 2. Create the following four tables: **(10 points)**

Employee(emp\_name, street, city)

Works(emp\_name, company\_name, salary)

Company(company\_name, city)

Managers(emp\_name, manager\_name)

✓ a. Identify the Primary Key-Foreign Key relationships between the tables. You may design your own Primary Keys, if required. Clearly state assumptions, if any. **(5 points)**

✓ b. Write SQL queries for the following: **(4 \* 5 = 20 points)**

- ✓ 1. Find names of all employees who work for SBI
  - ✓ 2. Find cities of residence of all employees who work for SBI
  - ✓ 3. Find names of all employees who don't work for SBI
  - ✓ 4. Find names of all employees who have worked for all branches of SBI
- c. Simulate examples of various anomalies like Insertion, Deletion, and Update on referenced as well as referencing relations in the aforementioned database. Specify what anomalies would violate the Referential integrity constraint and what could be a potential solution for the same. **(10 points)**

3. Create the following tables: **(10 points)**

Employee(emp\_name, email, contact\_no., department)

Department(emp\_name, salary, emp\_designation)

Awardee(emp\_name, email, department, experience)

- a. Identify Primary Key-Foreign Key relationships between the tables. You may design your own Primary Keys, if required. Clearly state assumptions, if any. **(5 points)**
- b. Write SQL queries for the following: **(2 \* 5 = 10 points)**
  1. If an update is made on an entry for a particular employee (X) in one table, then it will automatically update corresponding values for X in all other associated tables
  2. Find no. of awardees from each department

Reference:

- a) Bayross's Textbook on writing queries
- b) Regular Database Textbooks
- c) DBMS Handwritten Notes