

# Lab Activity 3 - SQL

Software System Development – Monsoon 2023

**Due Date: 23 August 2023, 05:00 pm**

---

## Instructions:

- Deadline mentioned during the Lab is strictly immutable. No extensions will be given.
- Any naming convention mentioned in the lab activity must be followed strictly or marks may be deducted for the same.
- Any plagiarized content will fetch zero marks for the current lab and will be followed by strict action against the students involved. However, discussion of ideas is allowed.

## Submission Criteria:

- Create a folder with your **roll number as its name** and containing the following files corresponding to the questions:
  - **<roll\_number>\_q1\_A.sql**
  - **<README.md**
- Compress the folder as a zip file (**name should be <roll\_number>.zip**) and then upload it on the Moodle before deadline.
- **README.md** should contain steps for execution of your script and any extra information that you want the evaluator to know before running your script, such as dependencies on some external tools or libraries.
- For Example:

**2022201079.zip**

```
|__2022201079
    |__2022201079_q1_A.sql
    |__2022201079_q1_B.sql
    |__so on ...
    |__README.md
```

---

### Question 1: (40 Marks: 8 Marks each)

Welcome to Hotel Shaandaar! You're given a menu for this restaurant in the form of a table.

**menu** table – (dish\_name, price)

*{Find CREATE & INSERT scripts for **menu** table in Q1\_artifacts folder}*

You need to help the owner with the following things:

- Clean the '**dish\_name**' column by removing leading & trailing spaces. Also remove '0's in '**dish\_name**' column (if any). Update the same table. Print the table.
- All dishes are either 'Chicken' or 'Paneer'. Create a new column '**food\_type**' which can have values 'veg' or 'non-veg'. Update the same table. Print the table.
- All prices are in multiples of 100. Subtract *one* from the price for Veg dishes & *five* from the price of non-Veg dishes. Update the same table. Print the table.  
**Complete this task in just one query.**
- Give an average price for 'veg' and 'non-veg' dishes whose price is greater than or equal to 400. Display in descending order of average price. Print table with column names (**food\_type, avg\_food\_type**).
- Print all 'non-veg' dishes whose price is greater than the maximum 'veg' dish. Print table with column names (**dish\_name, price**).

### Question 2: (60 Marks: 12 Marks each)

Let's analyze a Library Management System. Following tables are being used in this system:

- **books table** (book\_id, title, author\_id, genre)
- **authors table** (author\_id, author\_name)
- **issued\_users table** (user\_id, user\_name, book\_id, date\_issued)

*{Find CREATE & INSERT scripts for all the tables in Q2\_artifacts folder}*

Please note: One user can issue multiple books. *\* If present*

Complete the following tasks to successfully understand this system and the data:

- A. Retrieve a list of all users along with the titles and authors of the books they have borrowed. Filter out records where title and/or author name is NULL. Print table with column names **(user\_name, title, author\_name)**.
- B. Count total issued users in each genre along with the genre name. Print table with column names **(genre, total\_user\_count)**.
- C. Get the list of all book names, the author names and the latest (most recent) date\_issued for that book. Print table with column names **(book\_name, author\_name, recent\_date\_issued)**.
- D. Retrieve the top three authors who have written the most books, along with the total count of books each of these authors has written. Only display the records if the book count is greater than or equal to 2. Display this in decreasing order of book count. In cases where authors have the same book count, their names should also be arranged in descending alphabetical order. Print table with column names **(author\_name, book\_count)**.
- E. Identify genres that users have not yet explored (issued any books from) and recommend those genres to them. Print the table with column names **(user\_name, genre)**.