

## Assignment

### Part A: Exploring SQL Constraints in tables

#### Objective:

The objective of this assignment is to deepen your understanding of SQL constraints, specifically focusing on NOT NULL, UNIQUE, DEFAULT, PRIMARY KEY, and FOREIGN KEY constraints. By completing this assignment, you will gain practical experience in creating tables with various constraints and manipulating data while ensuring data integrity.

#### Scenario:

You have been tasked with designing a database for a small online bookstore. The bookstore needs to manage information about books, authors, customers, and orders. To ensure data integrity and efficiency, you need to apply appropriate constraints to the tables.

#### Tasks:

##### 1. Design the Database BookStore with the following tables:

- `Books`: Contains information about books such as ISBN, title, author, genre, and price.
- `Authors`: Stores details about authors including their ID, name, and nationality.
- `Customers`: Holds customer information like customer ID, name, email, and address.
- `Orders`: Records orders made by customers, including order ID, customer ID, order date, and total amount.

##### 2. Apply Constraints:

- Define appropriate constraints for each table to ensure data integrity:
- Use NOT NULL to enforce mandatory fields.
- Apply UNIQUE constraint where necessary to ensure uniqueness of data.
- Set DEFAULT values for certain fields where applicable.
- Designate PRIMARY KEY for unique identifiers.
- Implement FOREIGN KEY constraints to establish relationships between tables.

### 3. Write SQL Statements:

- Write SQL statements to create the tables with the defined constraints.
- Insert sample data into each table to demonstrate the constraints in action.
- Enter data in the tables to check to verify the constraints.

## **Part B: Titanic MySQL Query Exercise: Click [Titanic Data Set](#)**

### Dataset:

The Titanic dataset contains information about passengers aboard the Titanic, including their names, ages, genders, ticket numbers, fares paid, decks of embarkation, and survival status.

### Objective:

The objective of this assignment is to familiarize you with fundamental SELECT statements with basic clauses. By using the Titanic dataset as an exercise, students will practice retrieving specific data subsets, filtering records based on conditions, and using various operators to refine query results.

### Instructions:

1. Connect to the MySQL server and select the Titanic database.
2. Perform the following queries using appropriate MySQL syntax.
3. Write SQL queries for each task and execute them to obtain the desired results.
4. Save your SQL queries in a text file along with the results obtained after executing each query.

Tasks:

1. Retrieve all columns for all passengers from the Titanic table.
2. Retrieve distinct values of the "Embarked" column from the Titanic table.
3. Retrieve the names and ages of passengers who survived from the Titanic table.
4. Retrieve the names and ticket numbers of passengers who paid fares greater than 60000.
5. Retrieve the names and ages of passengers whose names start with 'J'.
6. Retrieve the names and ages of passengers who boarded deck numbers either 90 or 70.
7. Retrieve the names and ages of passengers with ages between 20 and 30.
8. Retrieve the names and ages of passengers whose ages belongs to 20, 25, or 30.

