National University

of Computer & Emerging Sciences Peshawar Campus

Student Name: _____   **Roll No:** _____

Program: CS & SE   Examination: Final

Semester: Fall-2022   Total Marks: 60 Weightage: 50

Time Allowed: 02 hours 40 minutes   Instructor: Muhammad Usman

Course: Database Systems Lab   Date: 13th December,2022

NOTE:
- Don't rush, carefully understand the problem and then go for solution.
- Understanding the question is the part of Exam.
- An individual may be assigned a straight-forward 0 if the submitted assessed task is copied/cheated from another individual/internet/found using internet.
- Create a folder named as Roll number followed by your, name (20P-8721_Zain), copy all the files (Q1.txt,Q2.txt), in that folder and submit the folder.
- Your queries should be correct with respect to the design of this dataset. In other words, we should be able to use your SQL queries on another dataset with the same schema and get correct answers even if the actual data within is different.

Question # 01   [45 Marks] 3 per query

Run the script sailors.sql (provided), you will get the database of a sailors club with the following three

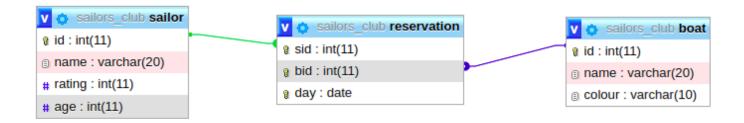tables about boats, sailors, and boat reservations as shown in below figure 1

Figure 1

Design SQL queries that answer the questions listed below (one query per question).

1) List, for every boat, the number of times it has been reserved, including those boats that have never been reserved (list the id, name and the count).

2) List those sailors who have reserved every red boat (list the id and the name).

3) List those sailors who have reserved only red boats *(hint: We are looking for sailors for whom there does not exist a reservation for a non-red boat).*

4) For which boat are there the most reservations?

5) Select all sailors who have never reserved a red boat.

6) Create a **View***(virtual table)* to List the names of all boats having an attribute of color red.

7) List the names of the red boats that have been reserved by a sailor of rating 8.

8) List the names of the red boats that have never been reserved by a sailor of rating 8.

9) How many red boats does the sailors club own?

10) Return the names of the boats that have only been reserved by sailors of rating 8?

11) List, for every sailor, the number of times it has reserved the boats. (List the id, name and the count).

12) List those boats who have reserved more than two times (list the boat id and name).

13) Create a **stored procedure** for sailor table to insert data. The stored procedures must take values as parameters.

14) Write a query which generates a list of passwords, using the first three alphabets of name in Upper Case, combined with age and length of rating attribute using sailor table. Label the column as Sailors_Password.

15) Show the data of boat table as :

The (name) whose id is (id) has (colour).

Example:

The Interlakewhose id is 101 has bluecolour

The Clipperwhose id is 103 has greencolour

The Marinewhose id is 104 has redcolour

Question # 02

- Create a database named your rollno_name (i.e. 19P0084_zain).
- Create a collection named movie.
- Import the data from movies.json file (provided) in to the movie's collection.

After successfully importing the file you would be having the documents layout
as shown in Figure 2.

```
_id: ObjectId("573a1391f29313caabcd7e30")
plot: "A dramatized account of a great Russian naval mutiny and a resulting s..."
v genres: Array
    0: "History"
    1: "War"
  runtime: 66
  rated: "UNRATED"
  num_mflix_comments: 1
  title: "Battleship Potemkin"
  fullplot: "Based on the historical events the movie tells the story of a riot at ..."
> languages: Array
> writers: Array
  lastupdated: "2015-08-18 00:22:19.750000000"
  year: 1925
v imdb: Object
    rating: 8
    votes: 36901
    id: 15648
> countries: Array
  type: "movie"
  metacritic: 1
```

Figure 2: Document Layout

Design queries that answer the questions listed below (one query per question).

1) Retrieve all movies which were made in or before 1970 but after 1965? Your query should return title, year and rated fields only.

2) List the movies which had a runtime longer than 3 hours (180 sec) and had a imdb.rating higher than 8.

3) Find movies which were tagged as one of the following genres: "Sport", "Talk-Show", "News".

4) Write a query to list movies whose imdb.rating is greater than 8.5 or metacritic rating is greater than 85.

5) Find movies which were both: History and War genres, return the documents based on the descending order of year and then uses the limit caluse to limit the number of documents being returned to just 2.