

NYU Tandon School of Engineering

CS-GY 6083, Principles of Database Systems, Spring 2024

Prof Phyllis Frankl

## HOMework #2

Instructions:

You may discuss these problems with other students but should write up and hand in the solutions **yourself**.

There will be two GradeScope links for this assignment:

**HW 2A:** Hand in a **.txt** or **.sql** file with the queries from Problem 1 and Problem 4. Any text other than the queries should be commented out with “--” so the file can be imported and executed.

**HW 2B:** Hand in a **.pdf** with all other problems, marking which is which, as usual. You may either typeset the RA and DRC queries or hand-write them neatly. And if you choose to type it in latex, pay attention to the issue of too long to fit in one line.

### Problem 1

In this problem, you have to write SQL queries for a database modeling student organizations (student clubs) at a university, given by the following relational schema:

Student (sid, sname, semail, sphone);

Club (cid, cname, cdescription);

Event (eid, ename, edescription, edate, memprice, nonmemprice, maxpeople)

Membership (sid, cid, semester, year, memberfee);  
    sid references Student(sid)  
    cid references Club(cid)

HoldsEvent (eid, cid);  
    eid references Event(eid)  
    cid references Club(cid)

Register(eid, sid, price, rating);  
    eid references Event(eid)  
    sid references Student(sid)

In the schema, we have students identified by a ***sid***, and clubs identified by a ***cid***. Students can join clubs on a per-semester basis by paying a membership fee. Clubs organize events, such as “Annual Dinner” or “Camping Trip”, where each event is identified by an ***eid***. An event can be jointly organized by several clubs, and the price of taking part in the event usually depends on whether the student is a member of one of the organizing clubs or not. Finally, the schema keeps track of which student registers for which event, and afterward the student can rate an event from one to five stars.

a. Write the following queries:

(i) Find all Events that are held by the “French Literature Club” and allow more than 10 people to attend.

(ii) List students’ names along with all the clubs that held events the student has taken in 2023 and spent less than 10\$ to register.

(iii) List students’ names along with all the events that they attended as non-member.

(iv) List students along with all their membership clubs in 2022 Fall whose membership cost is lower than 20\$.

(v) Output the name of any club that has never held an event with any rating below 4.

(vi) Output the name of any student who has attended an event jointly organized by the Chinese and Japanese Student Associations while not being a member of either one.

- b. Write the queries for creating all tables for this database. You may assume any reasonable domains for the attributes.

## Problem 2

Same as Problem 1, parts 1 through 4, using Relational Algebra to write the queries

## Problem 3

Same as Problem 1, parts 1 through 4, using Domain Relational Calculus to write the queries

## Problem 4

Back to our Theme Park Database in Homework 1 and assume we have the following table schema with some simplifications.

### **Tourist**

tid, first\_name, last\_name, DOB, Email

### **ThemePark**

ParkTitle, Addr, Rating

### **Ride**

RideTitle, Avg\_wt, MinHeightRestriction

### **Visit**

tid, ParkTitle, Addr, Stars

tid REFERENCES Tourist (tid),

(ParkTitle, Addr) REFERENCES ThemePark (ParkTitle, Addr)

### **Favorite**

Tid, RideTitle

tid REFERENCES Tourist (tid),

RideTitle REFERENCES Ride (RideTitle)

### **Locate**

RideTitle, ParkTitle, Addr, IP

RideTitle REFERENCES Ride (RideTitle)

(ParkTitle, Addr) REFERENCES ThemePark (ParkTitle, Addr)

Write SQL queries for the following statements:

1. List all **Rides** in the **Theme Park** with the title MagicKingdomPark located at Bay Lake.
2. List all **Tourists** who were born in 2001.
3. List the names and emails of **Tourists** who have visited Disney'sHollywoodStudios (the string here is exactly the value in the test database) located at Bay Lake.
4. List the names of **Tourists** whose favorite rides have a waiting time longer than 30 minutes.
5. List the names of **Tourists** who give a non-5 star to the **Theme Park** where their favorite rides are located. (For example, A has a favorite ride in LegoLand, but he gives a 3-star rating to LegoLand)

You may use the following statements to create tables in MYSQL.

```
CREATE TABLE Tourist (  
    tid VARCHAR(7) PRIMARY KEY,  
    first_name VARCHAR(50) NOT NULL,  
    last_name VARCHAR(50) NOT NULL,  
    DOB date,  
    email VARCHAR(100) NOT NULL  
);
```

```
CREATE TABLE ThemePark (  
    ParkTitle VARCHAR(50) NOT NULL,  
    Addr VARCHAR(50) NOT NULL,  
    Rating FLOAT NOT NULL,  
    PRIMARY KEY (ParkTitle, Addr)  
);
```

```
CREATE TABLE Ride (  
    RideTitle VARCHAR(50) PRIMARY KEY,  
    Avg_wt TIME NOT NULL,  
    MinHeightRestriction INT
```

);

```
CREATE TABLE Visit (  
    tid VARCHAR(7),  
    ParkTitle VARCHAR(50),  
    Addr VARCHAR(50),  
    Stars INT NOT NULL,  
    FOREIGN KEY (tid) REFERENCES Tourist (tid),  
    FOREIGN KEY (ParkTitle, Addr) REFERENCES ThemePark (ParkTitle, Addr)  
);
```

```
CREATE TABLE Favorite (  
    tid VARCHAR(7),  
    RideTitle VARCHAR(50),  
    FOREIGN KEY (tid) REFERENCES Tourist (tid),  
    FOREIGN KEY (RideTitle) REFERENCES Ride (RideTitle)  
);
```

```
CREATE TABLE Locate (  
    RideTitle VARCHAR(50),  
    ParkTitle VARCHAR(50),  
    Addr VARCHAR(50),  
    IP VARCHAR(50) NOT NULL,  
    FOREIGN KEY (RideTitle) REFERENCES Ride (RideTitle),  
    FOREIGN KEY (ParkTitle, Addr) REFERENCES ThemePark (ParkTitle, Addr)  
);
```