

NYU Tandon School of Engineering

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

Prof Phyllis Frankl

## HOMework #3

Instructions:

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

Hand in a **.txt** or **.sql** file with the queries for this assignment. Any text other than the queries should be commented out with “--” so the file can be imported and executed.

### Problem 1:

Revisit our database modeling student organizations, this time we want to write some advanced queries on it, recall that the schema is shown by:

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)

Assumption: We assume all students attend the events they registered for. Each year has only two semesters, Fall and Spring.

Write queries for the following tasks:

1. For each student, output the number of events the student registered for in Fall 2023. (Hint: As there may be students attending 0 events, a natural join is not enough.)
2. Output the name of the club that has the greatest increase in member fees in Spring 2024 compared with Fall 2023.
3. Output the id, name, and date of the event co-organized by the largest number of clubs.
4. Output the id and name of the events held in Fall 2023 that have a number of attendees equal to its maximum people allowance.
5. Fill in the blanks in following query to output the IDs and names of students (other than Bob) who belonged in Fall 2023 to all of the clubs that 'Bob' (sid 12345) belonged to that semester:

Method 1:

```
SELECT s.sid, s.sname FROM [      ]
WHERE s.sid <> [      ]
AND NOT EXISTS
(
  -- Bob's 2023 clubs EXCEPT this student's clubs
  SELECT * From [      ]
  WHERE sid = 12345 AND semester = 'Fall' AND year = 2023
  AND cid NOT in
  -- Bob's F'23 clubs
  (SELECT [      ] FROM [      ]
   WHERE [      ] = s.sid AND semester = 'Fall' AND year = 2023))
```

Method 2:

```
SELECT s.sid, s.sname FROM Student s
WHERE s.sid <> 12345
AND
( -- number of clubs Bob is in during Fall 2023
SELECT COUNT(*)
```

```
FROM Membership WHERE [      ])
```

```
=
```

```
(
```

```
-- number of clubs Bob and current student are both in during Fall 2023
```

```
SELECT COUNT(*)
```

```
FROM Membership AS m1 JOIN Membership AS m2 USING (cid,  
semester, year)
```

```
WHERE semester = 'Fall' AND year = 2023 AND [      ])
```

6. Output the sid, sname, cid, cname, for each student and club such that the student has been a member of the club every semester since Fall 2023.
7. Output the id and name of the club that either hasn't held an event or held an event that no one registered for.
8. Output the id and name of the club that has the highest average rating of events held in Fall 2023.