

# INFSCI 2710 Database Management, Fall 2022

## Homework 1: Relational Algebra, SQL

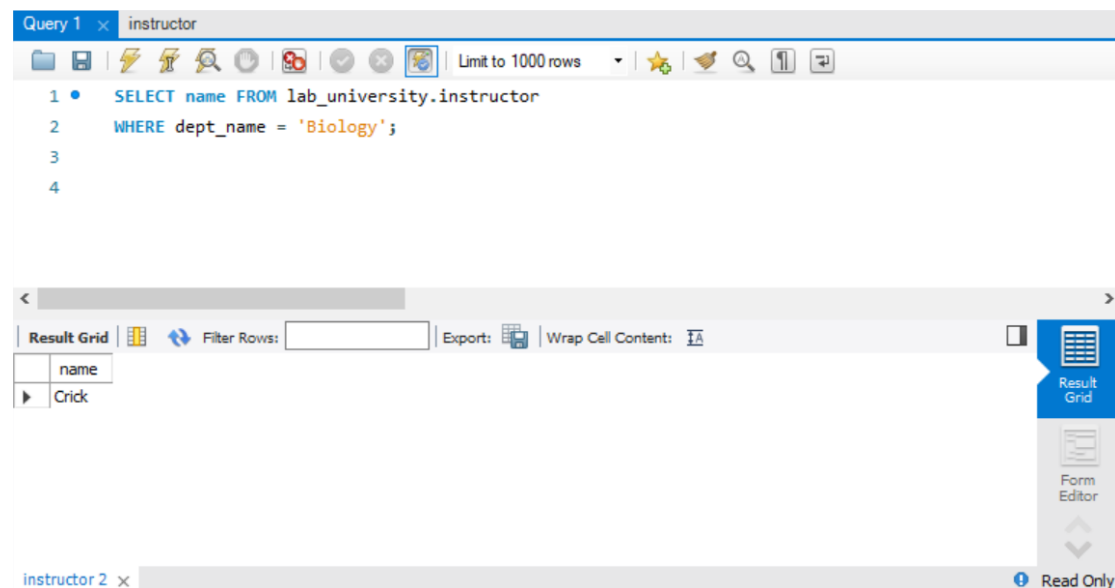
**Q1 [10 pt]** Specify a relational algebra expression and an SQL query to find the names of all the instructors from Biology department.

A:

$$\pi_{name} \left( \sigma_{dept\_name='Biology'}(instructor) \right)$$

SELECT name FROM instructor

WHERE dept\_name = 'Biology';



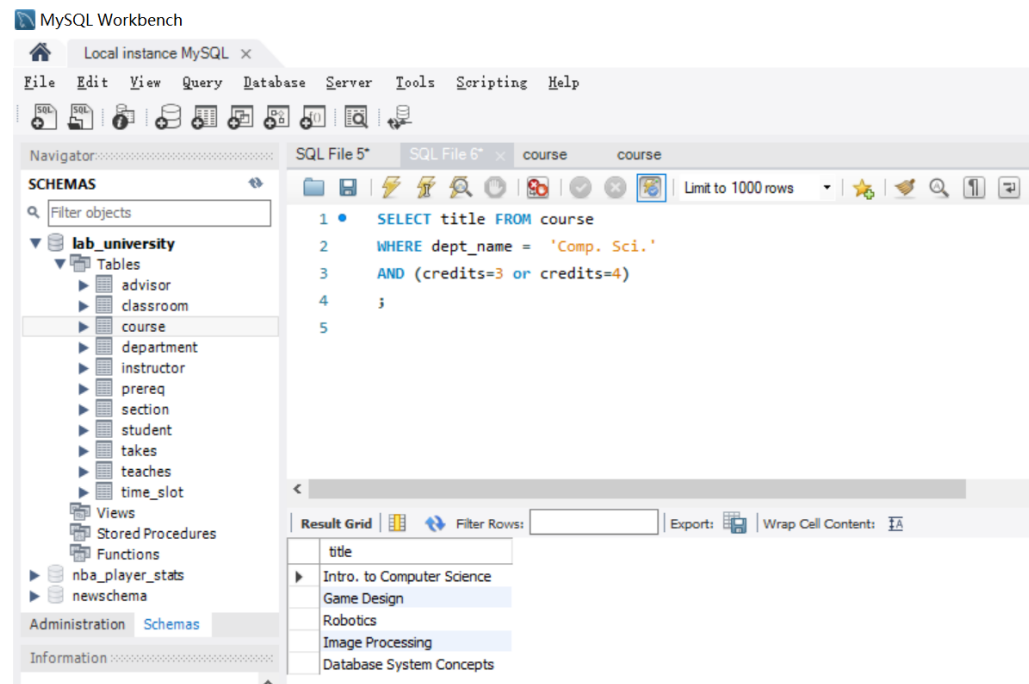
**Q2 [10 pt]** Specify a relational algebra expression and an SQL query to find the names of courses in Computer science department which have 3 credits.

A:

$$\pi_{title} \left( \sigma_{dept\_name='Comp. Sci.' \wedge credits=3} (course) \right)$$

SELECT title FROM course

WHERE dept\_name = 'Comp. Sci.' AND credits=3;



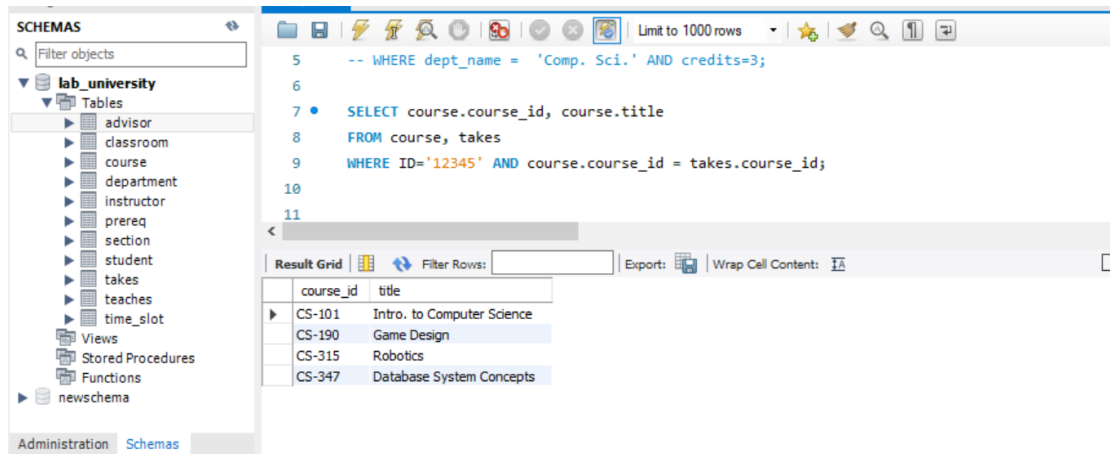
**Q3 [5 pt]** Specify a SQL query to find all course\_id and title of all courses taken by the student with ID 12345.

A:

SELECT course.course\_id, course.title

FROM course, takes

WHERE ID='12345' AND course.course\_id = takes.course\_id;



**Q4 [5 pt]** Specify a SQL query to find names of all the courses taught by computer science department in 2009 Fall.

A:

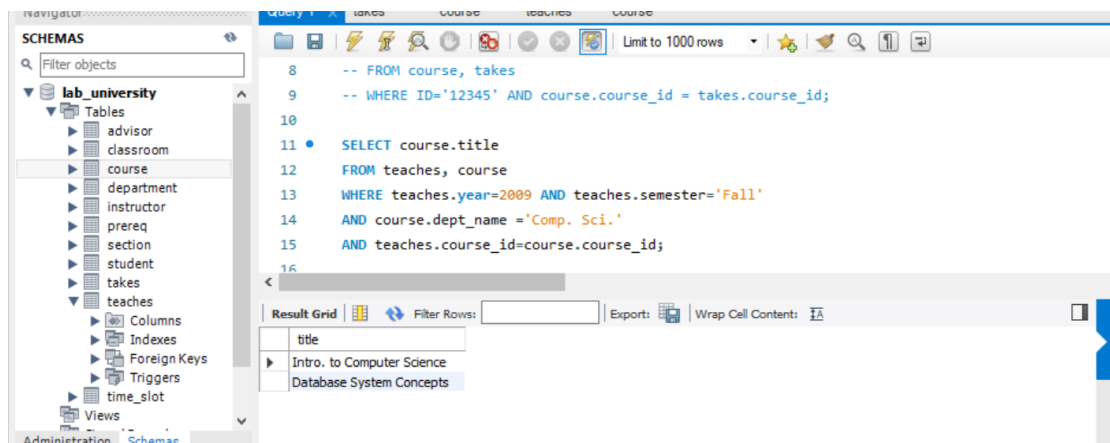
SELECT course.title

FROM teaches, course

WHERE teaches.year=2009 AND teaches.semester='Fall'

AND course.dept\_name ='Comp. Sci.'

AND teaches.course\_id=course.course\_id;



Consider another relational database that describe NBA player stats

(Underlines attributes in bold are the primary keys.)

Player (**ID**, name, position, height, weight, teamID)

Team (**ID**, name, city)

Game(**gameID**, homeTeamID, awayTeamID, homeScore, awayScore)

GameStats (**playerID**, **gameID**, points, assists, rebounds)

Hints:

(1) Two teams may play each other multiple times each season

(2) GameStats records the performance statistics of a player within a game.

A player may not play in every game, in which case it will not have its statistics recorded for that game.

**Q5[20 pt]** Write SQL DDL statements to create the above tables.

Make sure that you capture the primary and foreign key constraints (if applicable), choose appropriate domain (data) type and constraints for each attribute.

A:

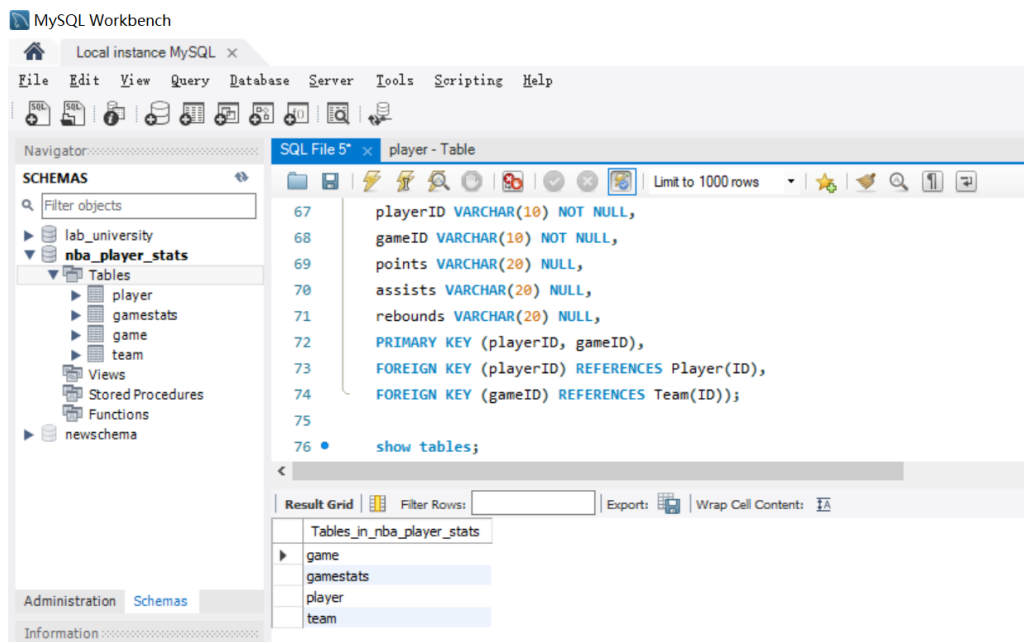
```
CREATE TABLE team (  
    ID VARCHAR(10) NOT NULL,  
    name VARCHAR(50) NULL,  
    city VARCHAR(20) NULL,  
    PRIMARY KEY (ID));
```

```
CREATE TABLE player(  
    ID VARCHAR(10) NOT NULL,  
    name VARCHAR(50) NULL,  
    position VARCHAR(20) NULL,  
    height CHAR(10) NULL,  
    weight CHAR(10) NULL,  
    teamID VARCHAR(10) NULL,  
    PRIMARY KEY (ID),  
    FOREIGN KEY (teamID) REFERENCES Team(ID) );
```

```
CREATE TABLE Game (  
    gameID VARCHAR(10) NOT NULL,  
    homeTeamID VARCHAR(10) NULL,  
    awayTeamID VARCHAR(10) NULL,  
    homeScore VARCHAR(20) NULL,  
    awayScore VARCHAR(20) NULL,  
    PRIMARY KEY (gameID),  
    FOREIGN KEY (homeTeamID) REFERENCES Team(ID),  
    FOREIGN KEY (awayTeamID) REFERENCES Team(ID));
```

```
CREATE TABLE GameStats (  
    playerID VARCHAR(10) NOT NULL,
```

gameID VARCHAR(10) NOT NULL,  
points VARCHAR(20) NULL,  
assists VARCHAR(20) NULL,  
rebounds VARCHAR(20) NULL,  
PRIMARY KEY (playerID, gameID),  
FOREIGN KEY (playerID) REFERENCES Player(ID),  
FOREIGN KEY (gameID) REFERENCES Team(ID));



**Q6 [10 pt]** Specify a SQL query to find the gameID, scores of both team of the games that are played at the Philadelphia home court.

A:

```
SELECT Game.gameID, Game.homeScore, Game.awayScore  
FROM Game,Team  
WHERE Team.city= 'Philadelphia'  
AND Game.homeTeamID = Team.ID;
```

**Q7 [10 pt]** Specify a SQL query to find points, assist and rebounds of the player whose name is “James” in all the home game he played for his team.

A:

```
SELECT GameStats.points, GameStats.assists, GameStats.rebounds
FROM GameStats, Player, Game
WHERE Player.name = 'James'
AND Player.ID = GameStats.playerID
AND GameStats.gameID=Game.gameID
AND Game.homeTeamID = Player.teamID;
```

**Q8[10 pt]** Specify a SQL query to find the gameID of games that Warriors won (including both home and away games).

A:

```
SELECT Game.gameID
FROM Game, Team
WHERE Team.name = 'Warriors'
AND ((Game.homeTeamID = Team.ID and homeScore > awayScore)
OR (Team.ID = Game. awayTeamID and awayScore > homeScore));
```

**Q9[20 pt]** Consider the following relational algebra expression.

$$\pi_{GameStats.playerID}(\sigma_{(GameStats.points > 10) \wedge (GameStats.assist > 10) \wedge (GameStats.rebound > 10)} GameStats)$$

1) How many attributes will the result have?

A:

Just one attribute and it's playerID.

2) Write in English what question the expression is trying to answer (e.g. describe what would be the result of the expression).

A:

In the GameStats table, try to find the playerID of a person whose points, assist and rebound are all greater than 10.

3) Translate the expressions in 3) into SQL.

A:

SELECT GameStats. playerID

FROM GameStats

WHERE GameStats.points > 10

AND GameStats.assists > 10

AND GameStats.rebounds > 10

;