

MLDS-413 Introduction to Databases and Information Retrieval

Homework 4: INNER JOINS

Name 1: _____ Sydney (Lowan) Li

NetID 1:

_____ chr0390 _____

Name 2: _____ Ayush Agarwal

NetID 2:

_____ scg1143 _____

Instructions

You should submit this homework assignment via Canvas. Acceptable formats are word files, text files, and pdf files. Paper submissions are not allowed and they will receive an automatic zero.

As explained during lecture and in the syllabus, assignments are done in groups. The groups have been created and assigned. Each group needs to submit only one assignment (i.e., there is no need for both partners to submit individually the same homework assignment).

Each group can submit solutions multiple times (for example, you may discover an error in your earlier submission and choose to submit a new solution set). We will grade only the last submission and ignore earlier ones.

Make sure you submit your solutions before the deadline. The policies governing academic integrity, tardiness and penalties are detailed in the syllabus.

Homework Instructions

For this assignment, you will use the program "DB Browser for SQLite" (available at <http://sqlitebrowser.org/>). This is the same software we have worked with in class. I posted several sample database files on Canvas in the Lecture Materials page. These database files can be opened with the DB Browser for SQLite. The database files we will use for this homework are:

- [EntertainmentAgency.sqlite](#)
- [BowlingLeague.sqlite](#)
- [SchoolScheduling.sqlite](#)

For every question, we expect to see both your SQL code and the resulting data. Copy and paste both the SQL code and the results into a document and submit it following the submission instructions.

Here is an example question that applies to the SalesOrders.sqlite database:

Question: What bikes cost more than \$1000?

Answer:

```
SELECT ProductName, RetailPrice
FROM Products JOIN Categories
  ON Products.CategoryID = Categories.CategoryID
WHERE CategoryDescription = "Bikes"
  AND RetailPrice > 1000;
```

Output:

"Trek 9000 Mountain Bike"	"1200"
"Eagle FS-3 Mountain Bike"	"1800"
"GT RTS-2 Mountain Bike"	"1650"

You must answer each question with a single query.

You may find it helpful to use the "Basic SQL Cheat Sheet" posted on Canvas.

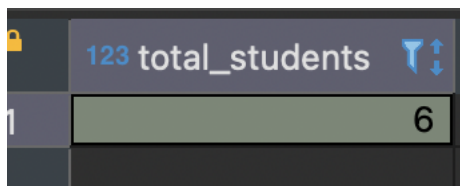
Each one of the questions below is worth **10 points**.

SchoolScheduling.sqlite

- 1) How many students are majoring in English or Mathematics? To receive credit you **must not** use subqueries anywhere (i.e., no nested SELECT clauses at all).

```
SELECT COUNT(*) as total_students
FROM Students s
inner join
Majors m
on
s.StudMajor = m.MajorID
where
m.Major in ('English', 'Mathematics')
```

There are 6 students majoring in English or Mathematics



total_students
6

- 2) What is the full name of the instructor of the class that has the highest average students' grade? Your output should list the full name of the instructor, the Class ID, and the average students' grade of that class. The full name of the instructor can be formed by concatenating the last name, a comma, a space, and the first name. For example, the full name of the professor in this MSiA-413 class is the string "Hardavellas, Nikos". To receive credit you **must not** use subqueries anywhere (i.e., no nested SELECT clauses at all).

```
SELECT s2.StfFirstName || ',' || s2.StfLastname as instructor_name , c.ClassID as
class_id, AVG(s.Grade) as max_avg_grade
FROM
Staff s2
inner join
Faculty_Classes fc
on
s2.StaffID = fc.StaffID
inner join
```

```

Classes c
on fc.ClassID = c.ClassID
inner join
Student_Schedules s
on
c.ClassID = s.ClassID
group by
c.ClassID
order by
AVG(s.Grade) DESC
limit 1

```

Ans: Deb Waldal is the instructor name of the class_id 2410 with maximum average grade of 93.64.

ABC instructor_name	123 class_id	123 max_avg_grade
Deb,Waldal	2,410	93.6433333333

- 3) What is the percentage of students with majors in English or Mathematics? To receive full credit for this question you **must not** use subqueries anywhere (i.e., no nested SELECT clauses at all). To receive partial credit you **must** use the JOIN operator and you **must not** use subqueries in the WHERE clause (subqueries elsewhere are fine).

```

SELECT
    ROUND(100.0 * SUM(CASE WHEN m.Major = 'English' or m.Major = 'Mathematics' then
1 else 0 END) / COUNT(DISTINCT s.StudentID),2) as eng_math_perc
FROM Students s
inner join
Majors m
on
s.StudMajor = m.MajorID

```

Ans : 33.33 % students major in English or Mathematics

123 eng_math_perc
33.33

EntertainmentAgency.sqlite

- 4) What percentage of all entertainer members are male entertainer members whose musical style is Jazz, and what percentage of all entertainer members are female entertainer members whose musical style is Jazz? You should provide a single query that outputs the percentages of each gender separately and indicates which is which. To receive full credit you **must not** use subqueries anywhere (i.e., no nested SELECT clauses at all). To receive partial credit you **must** use the JOIN operator and you **must not** use subqueries in the FROM and WHERE clauses (subqueries elsewhere are fine).

```
SELECT
    ROUND(100.0 * SUM(CASE WHEN m.Gender='M' and ms.StyleName='Jazz' then 1 else 0
END) / COUNT(DISTINCT em.MemberID),2) as MaleJazzPerc,
    ROUND(100.0 * SUM(CASE WHEN m.Gender='F' and ms.StyleName='Jazz' then 1 else 0
END) / COUNT(DISTINCT em.MemberID),2) as FemaleJazzPerc
FROM
Members m
inner join
Entertainer_Members em
on m.MemberID = em.MemberID
inner join
Entertainer_Styles es
on
em.EntertainerID = es.EntertainerID
inner join
Musical_Styles ms
on es.StyleID = ms.StyleID
```

123 MaleJazzPerc	123 FemaleJazzPerc
16	12

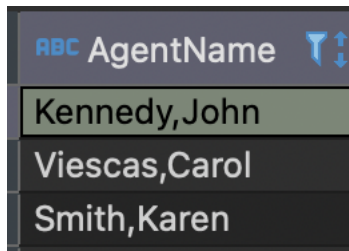
- 5) What is the full name (in the form “LastName, FirstName”) of the top 3 agents who have the highest average commission per engagement? The commission can be calculated by multiplying the contract price with the agent’ s commission rate. To receive credit you **must not** use subqueries anywhere (i.e., no nested SELECT clauses at all).

```
SELECT a.AgtLastName || ', ' || a.AgtFirstName as AgentName
FROM Agents a
inner join
```

```

Engagements e
on a.AgentID = e.AgentID
group by
a.AgentID
order by
AVG(a.CommissionRate * e.ContractPrice)
DESC
limit 3;

```



ABC AgentName 
Kennedy,John
Viescas,Carol
Smith,Karen

- 6) What is the total income of the Jazz entertainers (i.e., the sum of all Jazz entertainers' income across all of their engagements) and the total income of the Salsa entertainers? The income of each entertainer for each engagement is the ContractPrice of the engagement minus the agent's commission. To receive credit you **must not** use subqueries anywhere (i.e., no nested SELECT clauses at all).

```

SELECT ms.StyleName , SUM(e2.ContractPrice - e2.ContractPrice * a.CommissionRate)
as TotalIncome
FROM
Musical_Styles ms
inner join
Entertainer_Styles es
on ms.StyleID = es.StyleID
inner join
Entertainers e
on es.EntertainerID = e.EntertainerID
inner join
Engagements e2
on e.EntertainerID = e2.EntertainerID
inner join
Agents a
on a.AgentID = e2.AgentID
where
ms.StyleName in ('Jazz', 'Salsa')
group by

```

ms.StyleName

	ABC StyleName 🔼🔼	123 TotalIncome 🔼🔼
1	Jazz	19,623.3
2	Salsa	19,115.7

- 7) What are the top 5 musical styles that have the highest number of unique customers, and how many customers each of these styles has? To receive credit you **must not** use subqueries anywhere (i.e., no nested SELECT clauses at all).

```
SELECT ms.StyleName , COUNT(DISTINCT c.CustomerID) as customerCount
FROM
Musical_Styles ms
inner join
Musical_Preferences mp
on ms.StyleID = mp.StyleID
inner join
Customers c
on mp.CustomerID = c.CustomerID
group by
ms.StyleID
order by
COUNT(DISTINCT c.CustomerID)
DESC
limit 5;
```

ABC StyleName 🔼🔼	123 customerCount 🔼🔼
Standards	4
Rhythm and Blues	3
Jazz	3
Contemporary	3
Salsa	2

BowlingLeague.sqlite

- 8) Which teams have captains with the same last name? Each such team must be listed exactly once, along with the team captain's full name (in the form "LastName,

FirstName”). To receive full credit you **must not** use subqueries anywhere (i.e., no nested SELECT clauses at all).

```
SELECT t1.TeamID , t1.TeamName , b1.BowlerLastName || ',' || b1.BowlerFirstName as
Captain
FROM
Teams t1
inner join
Bowlers b1
on t1.CaptainID = b1.BowlerID
inner join
Teams t2
on t1.TeamID != t2.TeamID
inner join
Bowlers b2
on t2.CaptainID = b2.BowlerID and b1.BowlerLastName = b2.BowlerLastName
```

Ans:

TeamID	TeamName	Captain
5	Dolphins	Viescas,Suzanne
7	Manatees	Viescas,Michael

- 9) In question 8 you identified the bowling teams that have captains with the same last name. List all the matches in which any of these teams participates. The output should provide the TourneyDate, TourneyLocation, odd and even Team Names, and Lanes. You can use the team names identified in question #8 here to make the query easier. To receive credit you **must not** use subqueries anywhere (i.e., no nested SELECT clauses at all).

```
SELECT
    ts.TourneyDate,
    ts.TourneyLocation,
    tm1.TeamName AS OddTeamName,
    tm2.TeamName AS EvenTeamName,
    tm.Lanes
FROM
Tourney_Matches AS tm
inner join
Teams AS tm1 ON tm.OddLaneTeamID = tm1.TeamID
```

inner join

Teams AS tm2 ON tm.EvenLaneTeamID = tm2.TeamID

inner join

Tournaments AS ts ON tm.TourneyID = ts.TourneyID

WHERE

tm1.TeamName IN ('Dolphins','Manatees')

OR

tm2.TeamName IN ('Dolphins','Manatees');

Ans:

	ABC TourneyDate	ABC TourneyLocation	ABC OddTeamName	ABC EvenTeamName	ABC Lanes
1	1999-06-05	Red Rooster Lanes	Dolphins	Orcas	05-06
2	1999-06-05	Red Rooster Lanes	Manatees	Swordfish	07-08
3	1999-06-12	Thunderbird Lanes	Dolphins	Manatees	25-26
4	1999-06-19	Bolero Lanes	Manatees	Orcas	19-20
5	1999-06-19	Bolero Lanes	Dolphins	Swordfish	21-22
6	1999-06-26	Imperial Lanes	Marlins	Dolphins	09-10
7	1999-06-26	Imperial Lanes	Terrapins	Manatees	13-14
8	1999-07-03	Sports World Lanes	Dolphins	Sharks	13-14
9	1999-07-03	Sports World Lanes	Manatees	Barracudas	17-18
10	1999-07-10	Totem Lanes	Marlins	Manatees	05-06
11	1999-07-10	Totem Lanes	Terrapins	Dolphins	07-08
12	1999-07-17	Acapulco Lanes	Manatees	Sharks	15-16
13	1999-07-17	Acapulco Lanes	Dolphins	Barracudas	19-20
14	1999-07-24	Red Rooster Lanes	Manatees	Dolphins	25-26
15	1999-07-31	Thunderbird Lanes	Orcas	Manatees	19-20
16	1999-07-31	Thunderbird Lanes	Swordfish	Dolphins	21-22
17	1999-08-07	Bolero Lanes	Dolphins	Marlins	09-10
18	1999-08-07	Bolero Lanes	Manatees	Terrapins	13-14
19	1999-08-14	Imperial Lanes	Sharks	Dolphins	13-14
20	1999-08-14	Imperial Lanes	Barracudas	Manatees	17-18
21	1999-08-21	Sports World Lanes	Manatees	Marlins	05-06
22	1999-08-21	Sports World Lanes	Dolphins	Terrapins	07-08
23	1999-08-28	Totem Lanes	Sharks	Manatees	15-16
24	1999-08-28	Totem Lanes	Barracudas	Dolphins	19-20
25	1999-09-04	Acapulco Lanes	Orcas	Dolphins	05-06
26	1999-09-04	Acapulco Lanes	Swordfish	Manatees	07-08

- 10) How many teams have different players in the same team with the same last name?
To receive credit you **must not** use subqueries anywhere (i.e., no nested SELECT clauses at all).

SELECT COUNT(DISTINCT b1.TeamID) as teamCount

FROM

Bowlers b1

```
inner join
```

```
Bowlers as b2
```

```
on b1.TeamID = b2.TeamID and b1.BowlerID != b2.BowlerID and b1.BowlerLastName =  
b2.BowlerLastName
```

Ans: 5 Teams are there which have different players within same team with same last name

