

Database Assignment Homework #4

Homework #4 – Question 1A – Deserving of a Red Card for Loss of Contact. Deserving a Red Card for the loss of Contact for a particular race, IDRace=1 and LOCAverage >= 60 ms.

Answer) The sql statement is give below for Red Card deserving for LOCAvg >= 60 ms, with the execution result.

```
select A.Firstname,A.Lastname, VO.BibNumber
from videoobservation VO
Inner join bib B on VO.BibNumber = B.BibNumber
Inner Join Athlete A on B.IDAthlete = A.IDAthlete
where VO.Idrace=1 and VO.LOCAverage >= 60
Group by A.Lastname, A.Firstname
order by A.lastname, A.firstname;
```

The screenshot shows a database application window with the following components:

- Top Bar:** Displays the file name '1: homework4new.sql*' and the current query '7: Untitled*'. The database connection is 'DrexelRaceWalking: VideoObservation'.
- Toolbar:** Contains icons for running the query, saving, undo, redo, and other database operations.
- Database Connection:** A dropdown menu shows 'SQLite (1)'.
- SQL Editor:** The query is entered as follows:

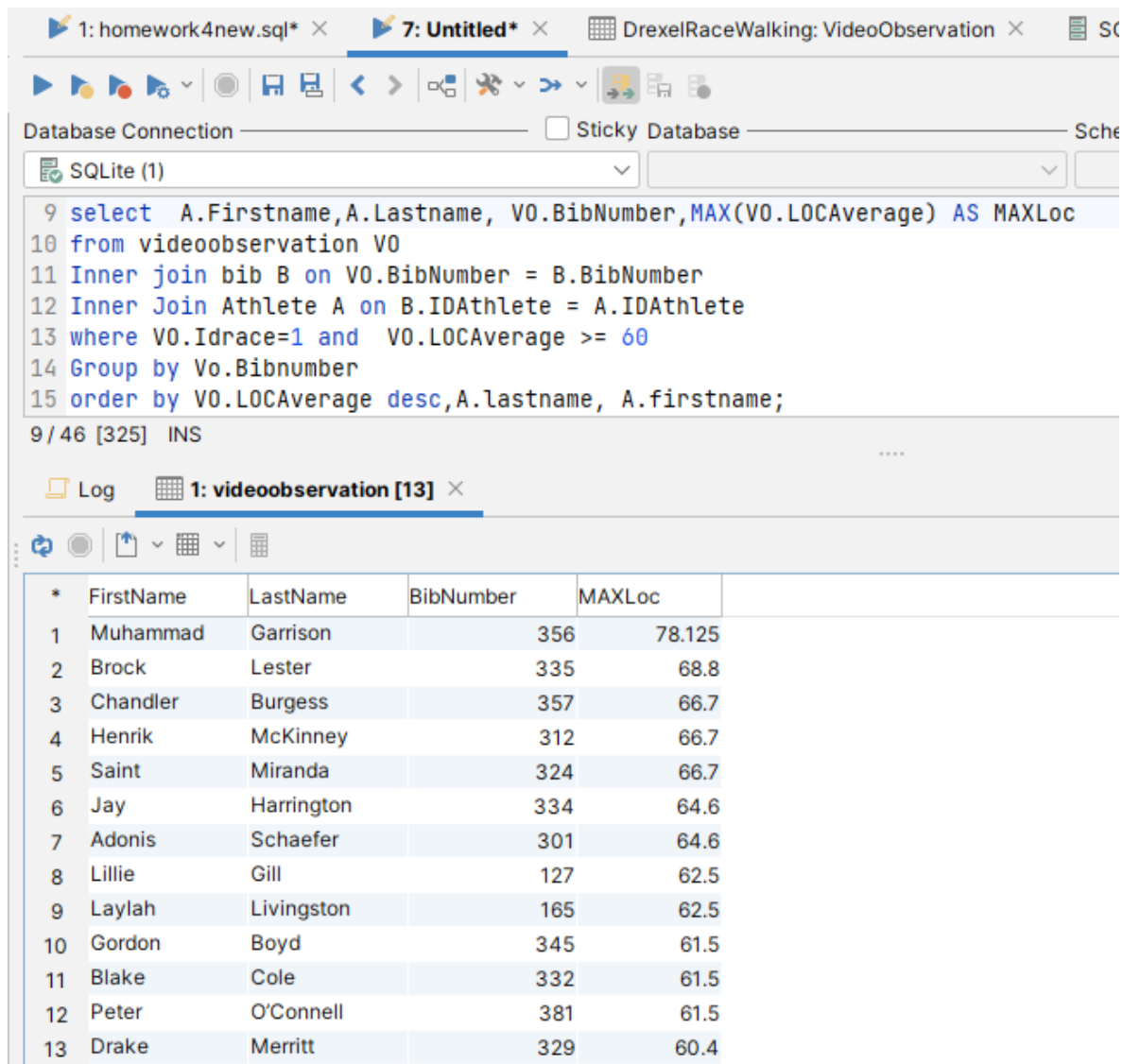
```
1 select A.Firstname,A.Lastname, VO.BibNumber
2 from videoobservation VO
3 Inner join bib B on VO.BibNumber = B.BibNumber
4 Inner Join Athlete A on B.IDAthlete = A.IDAthlete
5 where VO.Idrace=1 and VO.LOCAverage >= 60
6 Group by A.Lastname, A.Firstname
7 order by A.lastname, A.firstname;
```
- Execution Status:** Below the editor, it shows '7 / 35 [278] INS'.
- Log:** A log entry shows '1: videoobservation [13]'.
- Results Table:** The query results are displayed in a table with 4 columns: * (row number), FirstName, LastName, and BibNumber. The results are sorted by LastName and then by FirstName.

*	FirstName	LastName	BibNumber
1	Gordon	Boyd	345
2	Chandler	Burgess	357
3	Blake	Cole	332
4	Muhammad	Garrison	356
5	Lillie	Gill	127
6	Jay	Harrington	334
7	Brock	Lester	335
8	Laylah	Livingston	165
9	Henrik	McKinney	312
10	Drake	Merritt	329
11	Saint	Miranda	324
12	Peter	O'Connell	381
13	Adonis	Schaefer	301

Homework #4 – Question 1B – Deserving of a Red Card for Loss of Contact with Stats. The maximum visible loss of contact in milliseconds.

Answer) The sql statement is give below for MAX visible loss of contact LOCaverage with the execution result.

```
select A.Firstname,A.Lastname, VO.BibNumber,
MAX(VO.LOCAverage) AS MAXLoc
from videoobservation VO
Inner join bib B on VO.BibNumber = B.BibNumber
Inner Join Athlete A on B.IDAthlete = A.IDAthlete
where VO.Idrace=1 and VO.LOCAverage >= 60
Group by Vo.Bibnumber
order by VO.LOCAverage desc,A.lastname, A.firstname;
```



The screenshot shows a SQL IDE interface with a query editor and a results pane. The query editor contains the following SQL statement:

```
9 select A.Firstname,A.Lastname, VO.BibNumber,MAX(VO.LOCAverage) AS MAXLoc
10 from videoobservation VO
11 Inner join bib B on VO.BibNumber = B.BibNumber
12 Inner Join Athlete A on B.IDAthlete = A.IDAthlete
13 where VO.Idrace=1 and VO.LOCAverage >= 60
14 Group by Vo.Bibnumber
15 order by VO.LOCAverage desc,A.lastname, A.firstname;
```

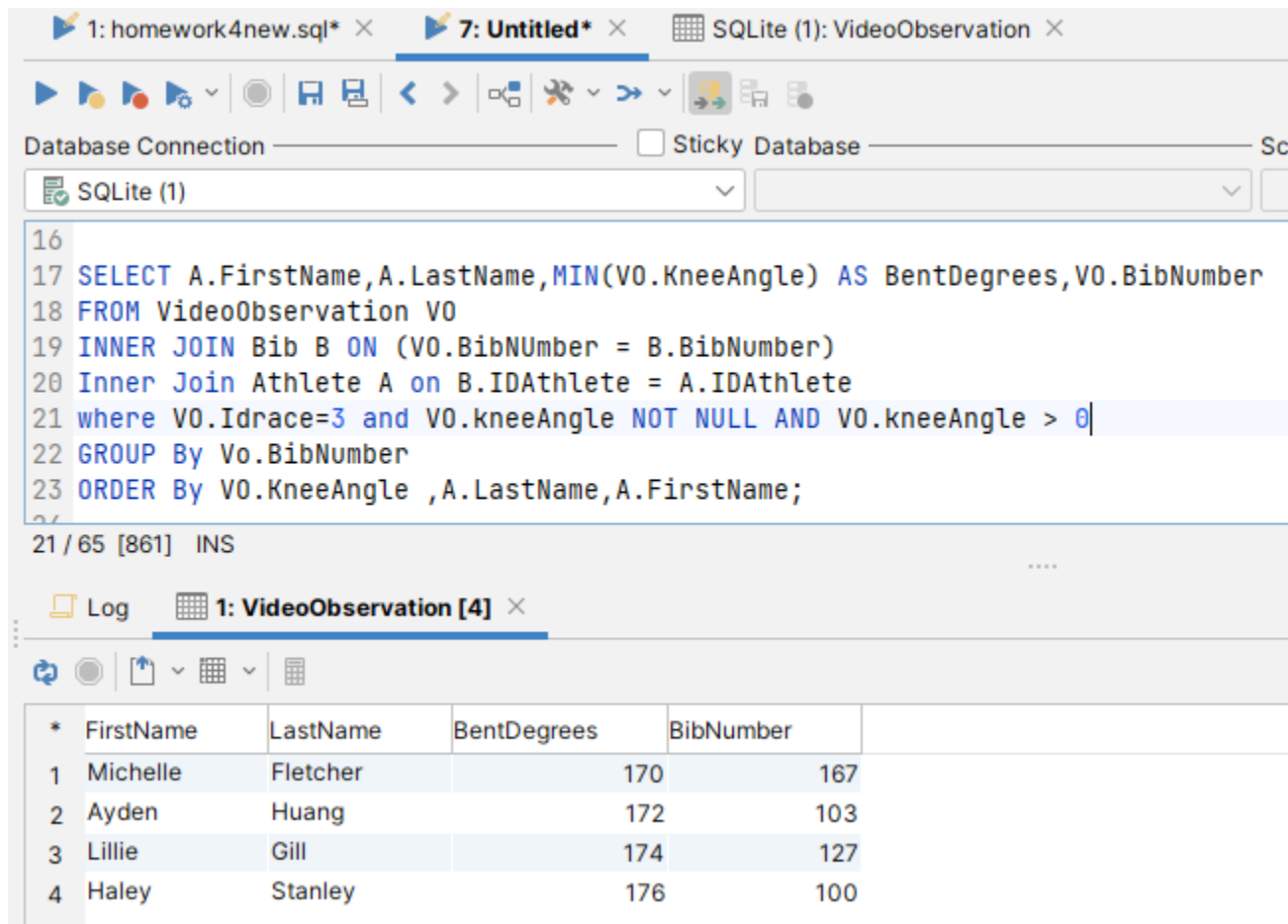
The results pane displays the execution of the query, showing 13 rows of data. The columns are: * (row number), FirstName, LastName, BibNumber, and MAXLoc. The data is sorted by MAXLoc in descending order.

* (row number)	FirstName	LastName	BibNumber	MAXLoc
1	Muhammad	Garrison	356	78.125
2	Brock	Lester	335	68.8
3	Chandler	Burgess	357	66.7
4	Henrik	McKinney	312	66.7
5	Saint	Miranda	324	66.7
6	Jay	Harrington	334	64.6
7	Adonis	Schaefer	301	64.6
8	Lillie	Gill	127	62.5
9	Laylah	Livingston	165	62.5
10	Gordon	Boyd	345	61.5
11	Blake	Cole	332	61.5
12	Peter	O'Connell	381	61.5
13	Drake	Merritt	329	60.4

Homework #4 – Question 2 – Deserved a Red Card for Bent Knee. For a given race IDRace=3, give all the athletes who deserve Red card for Bent Knee. You deserve a DQ for Bent knee angle other than Null. Include the Bent Knee angle where the player Bent Knee the most, i.e. it is the smallest value greater than Zero.

Answer) The sql statement for Bent Knee Red Card Deserving Athletes fro IDRace=3.

```
SELECT A.FirstName,A.LastName,MIN(VO.KneeAngle) AS BentDegrees,VO.BibNumber
FROM VideoObservation VO
INNER JOIN Bib B ON (VO.BibNumber = B.BibNumber)
Inner Join Athlete A on B.IDAthlete = A.IDAthlete
where VO.IDrace=3 and VO.kneeAngle NOT NULL AND VO.kneeAngle > 0
GROUP By Vo.BibNumber
ORDER By VO.KneeAngle ,A.LastName,A.FirstName;
```



The screenshot shows a SQLite IDE interface with the following components:

- Database Connection:** SQLite (1)
- SQL Query Editor:** Contains the SQL query from the previous block. The line numbers 16 through 23 are visible.
- Log:** Shows the execution status: 21 / 65 [861] INS.
- Results Panel:** Displays the query results in a table format.

*	FirstName	LastName	BentDegrees	BibNumber
1	Michelle	Fletcher	170	167
2	Ayden	Huang	172	103
3	Lillie	Gill	174	127
4	Haley	Stanley	176	100

Homework #4 – Question 3 –Athlete List participating in more than one race

Answer) The sql statement for Athletes who participated in more than one race, also showing the start race date and the last race date. Various functions used here are , MAX,MIN, COUNT, DISTINCT, GROUP , HAVING. SQL is to achieve this is given below.

```
SELECT a.FirstName, a.LastName,  
       COUNT(b.IDRace) AS NbrRaces,  
       MAX(r.RaceDate) AS lastRace,  
       MIN(r.RaceDate) AS firstRace  
FROM Athlete a  
JOIN Bib b ON a.IDAthlete = b.IDAthlete  
JOIN Race r ON b.IDRace = r.IDRace  
GROUP BY a.IDAthlete, a.FirstName, a.LastName  
HAVING COUNT(b.IDRace) > 1  
ORDER BY NbrRaces, a.Lastname, a.FirstName;
```

Database Connection ☐ Sticky Database

SQLite (1)

```
25 SELECT a.FirstName, a.LastName,  
26     COUNT(b.IDRace) AS NbrRaces,  
27     MAX(r.RaceDate) AS lastRace,  
28     MIN(r.RaceDate) AS firstRace  
29 FROM Athlete a  
30 JOIN Bib b ON a.IDAthlete = b.IDAthlete  
31 JOIN Race r ON b.IDRace = r.IDRace  
32 GROUP BY a.IDAthlete, a.FirstName, a.LastName  
33 HAVING COUNT(b.IDRace) > 1  
34 ORDER BY NbrRaces, a.Lastname, a.FirstName;
```

34 / 44 [1,332] INS

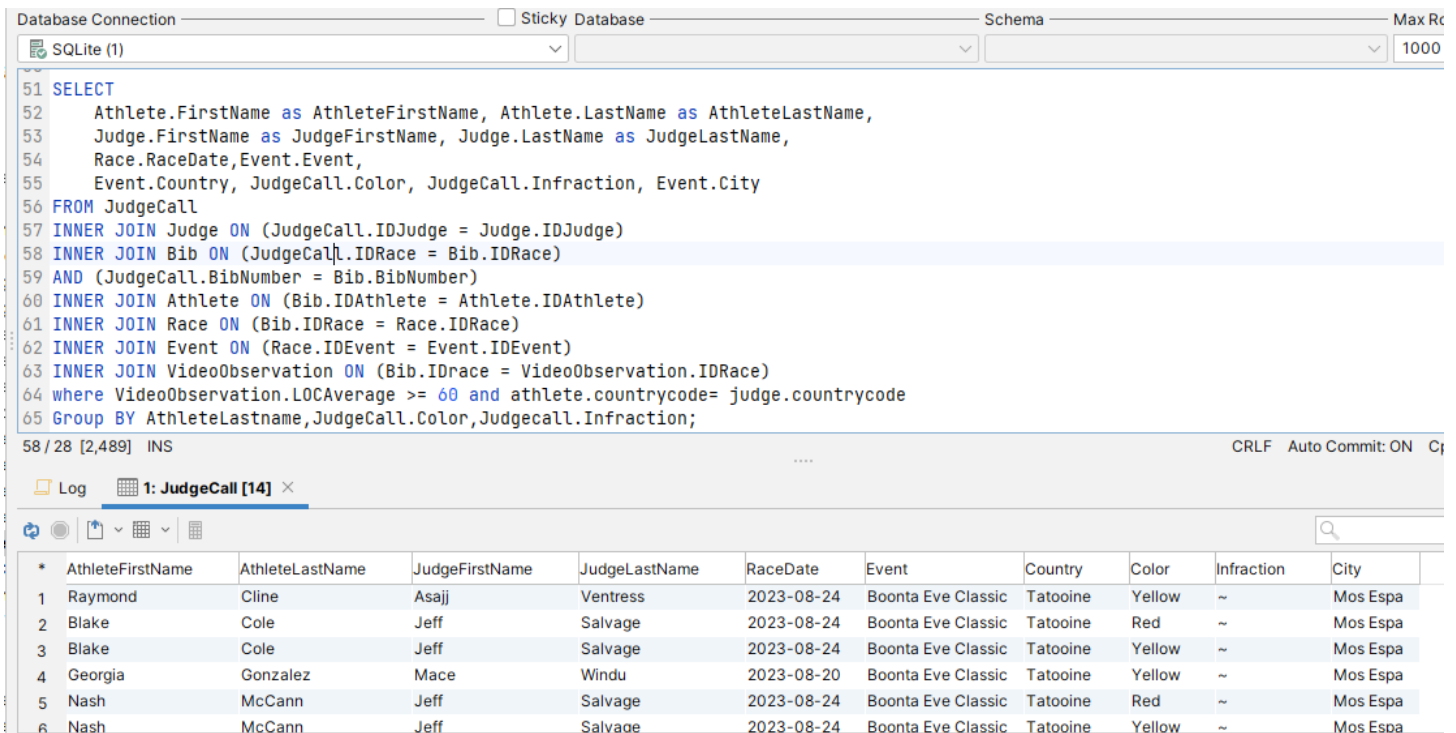
Log 1: Athlete [26] X

*	FirstName	LastName	NbrRaces	lastRace	firstRace
1	Emory	Bravo	2	2023-08-24	2023-08-19
2	Armando	Day	2	2023-08-24	2023-08-19
3	Jerry	Escobar	2	2023-08-24	2023-08-19
4	Michelle	Fletcher	2	2023-08-24	2023-08-20
5	Paige	Foster	2	2023-08-24	2023-08-20
6	Gwendolyn	French	2	2023-08-24	2023-08-20
7	Addyson	Gentry	2	2023-08-24	2023-08-20
8	Lillie	Gill	2	2023-08-24	2023-08-20
9	Fernanda	Hobbs	2	2023-08-24	2023-08-20
10	Paula	Hudson	2	2023-08-24	2023-08-20
11	Kaison	Knight	2	2023-08-24	2023-08-19

Homework #4 – Question 4 – Received a red card from a judge from their own country

Answer) Here in this question multiple tables and joins are required. This question is completed, by taking into consideration total six tables and five joins. The tables used here are Athlete, Bib, Judge, JudgeCall, Event and Race. The Bib table acts as a pivot table that integrates almost all information.

```
SELECT
    Athlete.FirstName as AthleteFirstName, Athlete.LastName as AthleteLastName,
    Judge.FirstName as JudgeFirstName, Judge.LastName as JudgeLastName,
    Race.RaceDate, Event.Event,
    Event.Country, JudgeCall.Color, JudgeCall.Infraction, Event.City
FROM JudgeCall
INNER JOIN Judge ON (JudgeCall.IDJudge = Judge.IDJudge)
INNER JOIN Bib ON (JudgeCall.IDRace = Bib.IDRace)
AND (JudgeCall.BibNumber = Bib.BibNumber)
INNER JOIN Athlete ON (Bib.IDAthlete = Athlete.IDAthlete)
INNER JOIN Race ON (Bib.IDRace = Race.IDRace)
INNER JOIN Event ON (Race.IDEvent = Event.IDEvent)
INNER JOIN VideoObservation ON (Bib.IDRace = VideoObservation.IDRace)
where VideoObservation.LOCAverage >= 60 and athlete.countrycode= judge.countrycode
Group BY AthleteLastname, JudgeCall.Color, Judgecall.Infraction;
```



* AthleteFirstName	AthleteLastName	JudgeFirstName	JudgeLastName	RaceDate	Event	Country	Color	Infraction	City
1 Raymond	Cline	Asajj	Ventress	2023-08-24	Boonta Eve Classic	Tatooine	Yellow	~	Mos Espa
2 Blake	Cole	Jeff	Salvage	2023-08-24	Boonta Eve Classic	Tatooine	Red	~	Mos Espa
3 Blake	Cole	Jeff	Salvage	2023-08-24	Boonta Eve Classic	Tatooine	Yellow	~	Mos Espa
4 Georgia	Gonzalez	Mace	Windu	2023-08-20	Boonta Eve Classic	Tatooine	Yellow	~	Mos Espa
5 Nash	McCann	Jeff	Salvage	2023-08-24	Boonta Eve Classic	Tatooine	Red	~	Mos Espa
6 Nash	McCann	Jeff	Salvage	2023-08-24	Boonta Eve Classic	Tatooine	Yellow	~	Mos Espa

Homework #4 – Question 5 – Returns average loss of contact observed by an Observer, FirstName, LastName, ID of Observer with highest Loss of contact reported by an Observer for IDRace=1. Exclude any judge whose average is above 28ms.

Answer) We are using here nested select query to complete the task. First we find max(LOCAverage) along with FirstName, LastName, LOCAverage <=28 and IDObserver. Then to match the output shown in the assignment , we exclude max column.

```
SELECT O.FirstName, O.LastName,V.LOCAverage, V.IDObserver, MAX(V.LOCAverage)
FROM VideoObservation as V
INNER JOIN Observer as O ON (V.IDObserver = O.IDObserver)
where V.LOCAverage <=28.0 AND V.IDRace=1);
```

The screenshot shows a SQLite IDE interface with the following components:

- Top Bar:** Displays the current file '1: homework4new.sql*' and a new file '7: Untitled*'. The database connection is 'SQLite (1): Athlete'.
- Toolbar:** Contains icons for running queries, saving, and other database operations.
- Database Connection:** A dropdown menu showing 'SQLite (1)' and a 'Sticky Database' checkbox.
- SQL Editor:** Contains the following SQL query:

```
87 SELECT O.FirstName, O.LastName,V.LOCAverage, V.IDObserver, MAX(V.LOCAverage)
88 FROM VideoObservation as V
89 INNER JOIN Observer as O ON (V.IDObserver = O.IDObserver)
90 where V.LOCAverage <=28.0 AND V.IDRace=1);
91
92
93
```
- Results Panel:** Shows the execution status '86 / 4 [3,712] INS'. Below it, a table titled '1: VideoObservation [1]' displays the results of the query.

* IDObserver	FirstName	LastName	Average
1	3 JeffDavid	SalvageHarriman	28.0

Table References

