Quiz 4 SQL Queries

Due Sep 27 at 10:25am **Points** 100 **Questions** 5
Available after Sep 27 at 10am **Time Limit** None

Instructions

This quiz is 25 minutes long and contains multiple choice questions.

Having an issue with the quiz? Please send an email to the course staff (rkheni@wpi.edu) (mailto:cvieira@wpi.edu) with "CS542 Quiz" included in the subject line any time during the quiz. If you require help through zoom then please join the zoom link https://wpi.zoom.us/j/2094237642 (https://wpi.zoom.us/j/2094237642) .

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	20 minutes	53.33 out of 100

Score for this quiz: **53.33** out of 100
Submitted Sep 27 at 10:22am
This attempt took 20 minutes.

Question 1 — 0 / 20 pts

Given the following two tables:

Student(ID, name, address);

Grade(Sequence, StudentID, exam1, exam2, exam3, total);

Note: Each student has at most one record in Grade table. Grade.StudentID references Student.ID

Write a SELECT statement that computes the average of each exam across all students. The output should be:

Test Average

Exam1 ---

'ou Answered

Select Avg(exam1) As 'Exam 1', Avg(exam2) as 'Exam 2', Avg(exam3) as 'Exam 3' From Grade;

This will select the averages, but in the wrong output format:

Exam 1	Exam 2	Exam 3
Average for exam 1 here	Average for exam 2 here	Average for exam 3 here

orrect Answer

Select 'Exam 1' As Test, Avg(exam1) As Average From Grade

Union

Select 'Exam 2' As Test, Avg(exam2) As Average From Grade

Union

Select 'Exam 3' As Test, Avg(exam3) As Average From Grade;

Select 'Exam 1' As Test, Avg(exam1) As Average From Student

Union

Select 'Exam 2' As Test, Avg(exam2) As Average From Student

Union

Select 'Exam 3' As Test, Avg(exam3) As Average From Student;

Select 'Exam 1' As Test, Avg(exam1) As Average From Grade;

Union

Select 'Exam 2' As Test, Avg(exam2) As Average From Grade;

Union

Select 'Exam 3' As Test, Avg(exam3) As Average From Grade;

Question 2

20 / 20 pts

Given the following two tables:

Student(ID, name, address);

Grade(Sequence, StudentID, exam1, exam2, exam3, total);

Note: Each student has at most one record in Grade table. Grade.StudentID references Student.ID

Write an Update statement to set exam3 grade for student(s) of name 'John Smith' to 55 (More than one student can have the same name).

Update Grade
Set exam3 = 55
Where StudentID in (Select name
From Student
Where name = 'John Smith');

Update Grade
Set exam3 = 55
Where name = 'John Smith';

Correct!

Update Grade
Set exam3 = 55
Where StudentID in (Select ID
From Student
Where name = 'John Smith');

Update Grade
Set total = 55
Where StudentID in (Select ID
From Student
Where name = 'John Smith');

Question 3

0 / 20 pts

Given the SQL that creates the table "Rentals", a table designed to capture rentals of books from a library, select the query that correctly shows the name ("NAME") and number of times ("TIMES") that a book has been rented, if that book has been rented more than 1 time after September 10, 2020.

Here is the table schema:

```
CREATE TABLE Rentals (
    bookName varchar(40),
    renter varchar(40),
    rentedAt date,
    constraint rentals_pk primary key (bookName, rentedAt)
);
```

Here is some example data:

bookName	renter	rentedAt
Ender's Game	John Doe	01-SEP-2019
Ender's Game	Jane Doe	05-SEP-2019
Alice in Wonderland	John Doe	11-SEP-2020
Alice in Wonderland	Jane Doe	11-SEP-2020
Ender's Game	Brian Smith	11-SEP-2020

Assuming you ran your query on this data, you should get:

NAME	TIMES
Alice in Wonderland	2

```
SELECT bookName AS NAME, COUNT(*) as TIMES FROM Rentals WHERE COUNT(*)
> 1 GROUP BY bookName HAVING rentedAt > '10-SEP-2020'
```

```
SELECT bookName AS NAME, COUNT(*) as TIMES FROM Rentals GROUP BY bookN
ame HAVING COUNT(*) > 1 and rentedAt > '10-SEP-2020'
```

```
SELECT bookName AS NAME, SUM(*) as TIMES FROM Rentals WHERE COUNT(*) >
1 and rentedAt > '10-SEP-2020' GROUP BY bookName
```

orrect Answer

```
SELECT bookName AS NAME, COUNT(*) as TIMES FROM Rentals WHERE rentedAt
> '10-SEP-2020' GROUP BY bookName HAVING COUNT(*) > 1
```

ou Answered

```
SELECT bookName AS NAME, COUNT(*) as TIMES FROM Rentals WHERE COUNT
(*) > 1 and rentedAt > '10-SEP-2020' GROUP BY bookName
```

You can't use the aggregate 'COUNT(*)' in a WHERE clause

Question 4

20 / 20 pts

Given the following two tables in the database:

Table 1: Employees	**Table 2:** Department
Columns:	**Columns:**
ssn (exactly a 9-digit integer, primary key)	id (up to a 4-digit integer, primary key)
first_name (up to 40 characters, not null)	department_name (up to 40 characters, not null)
	department_manager (foreign key for employee ssn)

last_name (up to 40 characters, not null)	

Create a query to show all of the names of the departments and the manager for each department.

Select the correct theta-join to fill in the blank of this query

```
SELECT E.first_name || ' ' || E.last_name, D.department_name FROM _____
_____;
```

Correct!

- (●) Department D, Employees E WHERE E.ssn=D.department_manager
- ( ) Department D AND Employees E WHERE E.ssn=D.department_manager
- ( ) Department D, Employees E WHERE E.ssn=D.id
- ( ) Department D AND Employees E WHERE E.ssn=D.id

Question 5

13.33 / 20 pts

Given the following two tables in a rental car company's database:

Table 1: Car	**Table 2:** RentalContract
Columns:	**Columns:**
VIN (17-character string, primary key)	id (integer, primary key)
make (up to 40 characters, not null)	carVin (17-character string, foreign key for Car VIN, not null)
model (up to 40 characters, not null)	renterFirstName (up to 40 characters, not null)
color (up to 40 characters. not null)	renterLastName (up to 40 characters, not null)
	rentedDate (date, not null)

manufactureYear (4-digit number, not null)	price (*positive* floating point number less than 1,000,000,000.00 with two decimal places, not null)

Fill in the blanks in the following query that outputs the full name (in a single column) of all people who entered into a rental agreement for over $1000 with cars manufactured in 2014.

```
SELECT [ Select ]
FROM [ Select ]
WHERE manufactureYear = 2014 AND price > 1000
;
```

Answer 1:

Correct!

renterFirstName || ' ' || renterLastName

Answer 2:

orrect Answer

Car join RentalContract on VIN=carVin

ou Answered

Car natural join RentalContract on VIN=carVin

Answer 3:

Correct!

manufactureYear = 2014 AND price > 1000

Quiz Score: **53.33** out of 100