

Unit 09 Instructions - Sub Queries, Unions and Views

Introduction to Enterprise Relational Databases

Instructions

Read these instructions all the way through before you start.

Download the answer document. Be sure to put your name at the top of the answer document where that's indicated. Answer the questions below in that document and when you are done, upload it into Blackboard.

Overview

This unit focuses on different ways of combining queries and with turning them into views.

The first approach – subquery is where one SELECT statement uses another SELECT statement in its WHERE clause. The SELECT statement embedded in the WHERE clause is called the subquery. It's used by the main query to filter records.

The second approach – unions – combines the results of two or more different queries by stacking (concatenating) the rows of one query on top of another query.

Preparing your SQL

Be sure to follow these points in writing your queries.

- Make sure you include the comment and USE statement at the top of your SQL

```
-- Unit 9.1 - Your Name  
  
USE college ;  
  
-- Students whose scholarship  
-- is greater than the average  
-- scholarship in the school
```

- Beautify your SQL.
- In your screen shots of the Result Grid you need to snip only first 6-10 rows. If you're not sure about this, see Unit 02 instruction for details on how to snip.

Note: Your data may be different than the data in the rows of the examples show below. The data itself may be different and you may get more or fewer rows that shown in the examples.

Unit 09 Instructions - Sub Queries, Unions and Views

Introduction to Enterprise Relational Databases

Steps

In each of these queries you're shown the columns to display in your result. Make your column headers look exactly like the examples shown, including capitalization and spacing.

1. [10] Write a query that uses a subquery to list students whose scholarship is greater than the average scholarship for the school. In calculating the average, be sure to exclude students without a scholarship since including those would incorrectly reduce the average. Tip: Write the base query first so you get a list of all students like below. Then write the subquery to calculate the average separately and get it working. While you're working on the subquery, use it to see what the average is so you check your results. Then add the subquery to the main query. Sort on last name.

Last	First	Major	Scholarship
Berry	Jose	Information Technology	3,500.00
Crawford	Joshua	Criminal Justice	4,000.00

This page may help: <http://dev.mysql.com/doc/refman/5.7/en/subqueries.html>

2. [10] Write a query that uses a UNION statement to combine both students and faculty members into a single result set. You should show only faculty from the Nursing department and students who are majoring in Nursing. Sort on Last Name.

Last	First	Type
Bennett	Betty	Student
Berry	Jose	Student
Black	Justin	Faculty
Cruz	Donna	Student

This is a bit tricky so here's some tips:

- You'll combine two separate SELECT statements with a UNION statement – one for faculty and the other for students.
- Get the two separate SELECT statements working by themselves before you combine them with the UNION.
- Make sure you have the sort working properly.
- This page may help
<http://dev.mysql.com/doc/refman/5.7/en/union.html>

Unit 09 Instructions - Sub Queries, Unions and Views

Introduction to Enterprise Relational Databases

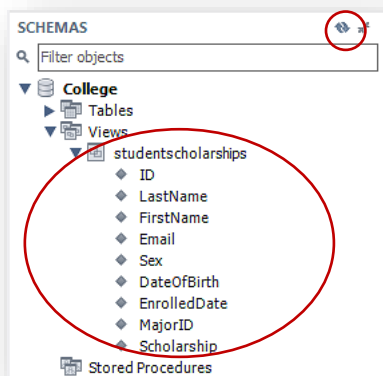
Tip

Since views are just stored SELECT statements, start out by creating your SELECT statement and getting it to work just the way you want. Then add the CREATE VIEW statement to store it in the database.

- [5] Create a view named **StudentScholarships** that shows all students receiving a scholarship. Return the columns shown below from the Student table, but only for students with a non-zero scholarship.

LastName	FirstName	Email	Scholarship
----------	-----------	-------	-------------

Snip the Schemas window showing that your view is in the College database. Expand the view so I can see its columns. You'll need to click on the Refresh button to see your new view.



Then write a SQL statement to run the view (SELECT * FROM **StudentScholarships**) and paste it's results into the homework.

TIP: If you create your views with create or replace instead of just create, you can run your script repeatedly which helps in tweaking scripts. Like this:

```
CREATE OR REPLACE VIEW Scholarships
```

- [10] Create a view named **StudentFlatten** produces the result below. Note that this requires an OUTER JOIN, is sorted on Name, and note the column headings. Show the view in the Schema window with its columns expanded. Then write a SQL statement to run the view and paste it's results into the homework. Then write a sentence or two discussing how this kind of view would be

Unit 09 Instructions - Sub Queries, Unions and Views

Introduction to Enterprise Relational Databases

useful.

	ID	Name	Major	Email	Sex	Date of Birth	Enrolled	Scholarship	MajorID
▶	20	Alvarez, Kenneth	Nursing	KennethAlvarez@college.edu	Male	1994-09-03	2013-03-01	0.00	5
	22	Armstrong, Jose	Criminal Justice	JoseArmstrong@college.edu	Male	1987-06-06	2014-04-22	2000.00	6
	2	Bailey, Christine	NULL	ChristineBailey@college.edu	Female	1990-07-24	2015-12-17	1000.00	NULL
	65	Barnes, Joyce	NULL	JoyceBarnes@college.edu	Female	1994-11-30	2012-10-29	4000.00	NULL
	36	Bechtel, Gerald	Biology	GeralBechtel@college.edu	Male	1993-04-28	2013-08-27	3500.00	4
	97	Bell, Mark	NULL	MarkBell@college.edu	Male	1993-08-16	2013-06-01	2000.00	NULL
	66	Bennett, Betty	Computer Science	BettyBennett@college.edu	Female	1993-10-11	2012-05-20	0.00	1
	34	Bennett, Julia	Information Systems	JuliaBennett@college.edu	Female	1987-04-19	2014-11-23	0.00	2

- [5] Create a view named **NursingStudents** that lists all the students who are majoring in Nursing. Base this view on the StudentFlatten view. Do not select on the field named Major, instead, select on the MajorID. Return all the columns from the StudentFlatten view. Show the view in the Schema window with its columns expanded. Then write a SQL statement to run the view and select only the nursing student with scholarship. Paste it's results into the homework. Then write one or two sentences on why you shouldn't select on name of the major.
- [10] Roll-your-own view. In real-world systems, reports are based views. Create a view of your own design that could be used to produce a report. Your view must have a useful purpose and must join at least three tables. At least one of the joins must be an outer join. Don't just recreate the StudentFlatten view. Make sure the columns are all ready to display as report column headings. For example, don't have a column DateOfBirth, make it Date of Birth. Explain the purpose of your view, and show the view in the Schema window with its columns expanded. Then write a SQL statement to run the view and paste it's results into the homework.