

Chapter 5

How to code summary queries

Exercises

1. Write a SELECT statement that returns these columns:
 - The count of the number of instructors in the Instructors table
 - The average of the AnnualSalary column in the Instructors tableInclude only those rows where the Status column is equal to "F" (Fulltime).
2. Write a SELECT statement that returns one row for each department that has instructors. The statement should return these columns:
 - The DepartmentName column from the Departments table
 - The count of the instructors in that department
 - The annual salary of the highest paid instructor in that departmentSort the result set so the department with the most instructors appears first.
3. Write a SELECT statement that returns one row for each instructor that has courses. The statement should return these columns:
 - The instructor's first and last names from the Instructors table in this format: John Doe (Note: If the instructor first name has a null value, the concatenation of the first and last name will result in a null value.)
 - A count of the number of courses for each instructor
 - The sum of the course units for each instructorSort the result set in descending sequence by the total course units for each instructor.
(Hint: You will need to concatenate the instructor first and last names again in the GROUP BY clause.)
4. Write a SELECT statement that returns one row for each course that has students enrolled. The statement should return these columns:
 - The DepartmentName column from the Departments table
 - The CourseDescription from the Courses table
 - A count of the number of students in the courseSort the result set by DepartmentName, then by the enrollment for each course.
5. Write a SELECT statement that returns one row for each student that has courses. The statement should return these columns:
 - The StudentID column from the Students table
 - The sum of the course units for each studentSort the result set in descending sequence by the total course units for each student.
6. Modify the solution to exercise 5 so it only includes students who haven't graduated and who are taking more than nine units.

7. Write a SELECT statement that answers this question: What is the total number of courses taught by **part-time** instructors only? Return these columns:

The instructor's last name and first name from the Instructors table in this format: Doe, John (Note: If the instructor first name has a null value, the concatenation of the first and last name will result in a null value.)

The total number of courses taught by each instructor

Chapter 6

How to code subqueries

Exercises

1. Write a SELECT statement that returns the same result set as this SELECT statement, but don't use a join. Instead, use a subquery in a WHERE clause that uses the IN keyword.

```
SELECT DISTINCT LastName, FirstName
FROM Instructors i
JOIN Courses c
    ON i.InstructorID = c.InstructorID
ORDER BY LastName, FirstName
```

2. Write a SELECT statement that answers this question: Which instructors have an annual salary that's greater than the average annual salary for all instructors?

Return the LastName, FirstName, and AnnualSalary columns for each Instructor.

Sort the result set by the AnnualSalary column in descending sequence.

3. Write a SELECT statement that returns the LastName and FirstName columns from the Instructors table.

Return one row for each instructor who doesn't have any courses in the Courses table. To do that, use a subquery with the NOT EXISTS operator.

Sort the result set by LastName and then by FirstName.

4. Write a SELECT statement that returns the LastName and FirstName columns from the Students table, along with a count of the number of courses each student is taking from the StudentCourses table.

Return one row for each student who is taking more than one class. To do that, use a subquery with the IN operator that groups the student course by StudentID.

Group and sort the result set by the LastName and then by the FirstName.

5. Write a SELECT statement that returns the LastName, FirstName, and AnnualSalary columns of each instructor who has a unique annual salary. In other words, don't include instructors who have the same annual salary as another instructor.

Sort the results by LastName and then by FirstName.

6. Write a SELECT statement that returns one row for each course. It should return these columns:

The CourseID column from the Courses table

The most recent enrollment date for that course

Change the SELECT statement to a CTE. Then, write a SELECT statement that returns one row per course that shows the CourseDescription for the course and the LastName, FirstName, and EnrollmentDate for the student with the most recent enrollment data.

7. Write a SELECT statement that returns one row for each student who has courses. It should return with these columns:

The StudentID column from the Students table

The sum of the course units for that student

Include only those students who are taking more than 9 units (fulltime).

Change the SELECT statement to a CTE. Then, write a SELECT statement that uses this CTE to return the student ID, sum of their course units, and their tuition. (The tuition is equal to the FullTimeCost column, plus the PerUnitCost column multiplied by the number of units.)

(Hint: You can use a cross join to add the columns from the Tuition table to the query. This works because there's only one row in the Tuition table.)