Chapter 3
How to retrieve data from a single table

Exercises

Enter and run your own SELECT statements

In these exercises, you'll enter and run your own SELECT statements.

1. Write a SELECT statement that returns all of the columns from the Courses table. Then, run this statement to make sure it works correctly.
2. Write a SELECT statement that returns three columns from the Courses table: CourseNumber, CourseDescription, and CourseUnits. Then, run this statement to make sure it works correctly.

   Add an ORDER BY clause to this statement that sorts the result set by CourseNumber in ascending sequence. Then, run this statement again to make sure it works correctly. This is a good way to build and test a statement, one clause at a time.
3. Write a SELECT statement that returns one column from the Students table named FullName that joins the LastName and FirstName columns.

   Format this column with the last name, a comma, a space, and the first name like this:

   ```
   Doe, John
   ```

   Sort the result set by last name in ascending sequence.

   Return only the students whose last name begins with a letter from A to M.
4. Write a SELECT statement that returns these column names and data from the Instructors table:

LastName	The LastName column
FirstName	The FirstName column
AnnualSalary	The AnnualSalary column

   Return only the rows with an annual salary that's greater than or equal to 60,000.

   Sort the result set in descending sequence by the AnnualSalary column.
5. Write a SELECT statement that returns these column names and data from the Instructors table:

LastName	The LastName column
FirstName	The FirstName column
HireDate	The HireDate column

   Return only the rows with a hire date that's in 2022.

   Sort the result set in ascending sequence by the HireDate column.

6. Write a SELECT statement that returns these column names and data from the Students table:

FirstName	The FirstName column
LastName	The LastName column
EnrollmentDate	The EnrollmentDate column
CurrentDate	The current date
MonthsAttended	A column that's calculated by getting the difference between the enrollment date and the current date

   To get the value of the months attended, use the DATEDIFF function with the month argument.

   Sort the result set in ascending sequence by the MonthsAttended column.

7. Write a SELECT statement that returns these column names and data from the Instructors table:

FirstName	The FirstName column
LastName	The LastName column
AnnualSalary	The AnnualSalary column

   Return only the top 20 percent of instructors based on annual salary.

8. Write a SELECT statement that returns these column names and data from the Students table:

LastName	The LastName column
FirstName	The FirstName column

   Return only the rows where the LastName column starts with the letter 'G'. To do that, use the LIKE phrase.

   Sort the result set by last name in ascending sequence.

9. Write a SELECT statement that returns these column names and data from the Students table:

LastName	The LastName column
FirstName	The FirstName column
EnrollmentDate	The EnrollmentDate column
GraduationDate	The GraduationDate column

   Return only the rows where the EnrollmentDate column is greater than 12-01-2022 and the GraduationDate column contains a null value.

10. Write a SELECT statement that returns these columns and data from the Tuition table, along with a constant value and two calculated values:

FullTimeCost	The FullTimeCost column
PerUnitCost	The PerUnitCost column
Units	12
TotalPerUnitCost	A column that's calculated by multiplying the per-unit cost by the units
TotalTuition	A column that's calculated by adding the full-time cost to the total per unit cost

Chapter 4

How to retrieve data from two or more tables

Exercises

1. Write a SELECT statement that joins the Courses table to the Departments table and returns these columns: DepartmentName, CourseNumber, CourseDescription.

   Sort the result set by DepartmentName and then by CourseNumber in ascending order.

2. Write a SELECT statement that joins the Instructors table to the Courses table and returns these columns: LastName, FirstName, CourseNumber, CourseDescription.

   Return all courses for each instructor with a status of "P" (part time).

   Sort the result set by LastName and then by FirstName in ascending order.

3. Write a SELECT statement that joins the Departments, Courses, and Instructors tables. This statement should return these columns: DepartmentName, CourseDescription, FirstName, and LastName.

   Use aliases for the tables, and return only those courses with three units.

   Sort the result set by DepartmentName and then by CourseDescription in ascending sequence.

4. Write a SELECT statement that joins the Departments, Courses, StudentCourses, and Students tables. This statement should return these columns: DepartmentName, CourseDescription, LastName, and FirstName.

   Return all courses in the English department.

   Sort the result set by CourseDescription in ascending sequence.

5. Write a SELECT statement that joins the Instructors and Courses tables and returns these columns: LastName, FirstName, and CourseDescription.

   Return at least one row for each instructor, even if that instructor isn't teaching any courses.

   Sort the result set by LastName and then by FirstName.

6. Use the UNION operator to generate a result set consisting of five columns from the Students table:

Status	A calculated column that contains a value of UNDERGRAD or GRADUATED
FirstName	The FirstName column
LastName	The LastName column
EnrollmentDate	The EnrollmentDate column
GraduationDate	The GraduationDate column

   If the student doesn't have a value in the GraduationDate column, the Status column should contain a value of UNDERGRAD. Otherwise, it should contain a value of GRADUATED.

   Sort the final result set by EnrollmentDate.

7. Write a SELECT statement that returns these two columns:

DepartmentName	The DepartmentName column from the Departments table
CourseID	The CourseID column from the Courses table

   Return all departments with no courses. (Hint: Use an outer join and only return rows where the CourseID column contains a null value.)

8. Write a SELECT statement that returns these columns:

InstructorDept	The DepartmentName column from the Departments table for an instructor
LastName	The LastName column from the Instructors table
FirstName	The FirstName column from the Instructors table
CourseDescription	The CourseDescription column from the Courses table
CourseDept	The DepartmentName column from the Departments table for a course

   Return one row for each course that's in a different department than the department of the instructor assigned to teach that course. (Hint: You will need to join the Departments table to both the Instructors table and the Courses table, which will require you to use table aliases to distinguish the two tables.)