SQL Fundamentals – TWO TABLES OR MORE

Create SQL commands to provide information for the following problems.

1. S-T: Use either join method. Use either join method. Use column aliases for every column displayed. List students and their evaluations when they were the evaluator. Show the student ID as "ID", full name (concatenated into one column) as "Evaluator", team name as "Team", evaluation ID as "Eval ID", semester as "Semester" and year as "Year". Sort by student ID and evaluation ID.
2. AW: Join tables in the FROM clause with the JOIN operator. List female marketing specialists. Show the business entity ID, the first and last name of the employee, the job title and gender. Sort by business entity ID. HINT: Be sure to put a table name in front of the business entity ID column in the SELECT clause. Also, remember the PERSON data is in a different schema than the EMPLOYEE data.
3. AW: Join tables in the WHERE clause. List female marketing specialists. Show the business entity ID, the first and last name of the employee, and the job title. Sort by business entity ID. HINT: Be sure to put a table name in front of the business entity ID column in the SELECT clause.
4. AW: Join tables in the FROM clause with the JOIN operator. List employees who work in the night shift. Show the shift, the employee’s first and last name and job title. Sort by employee's last name then first name. Hint: Join tables in the sequence of their relationships, such as shift joins employee department history which then joins employee which then joins, etc.
5. AW: Join tables in the WHERE clause. List employees who work in the night shift. Show the shift, the employee’s first and last name and job title. Sort by employee's last name then first name. Hint: Join tables in the sequence of their relationships, such as shift joins employee department history which then joins employee which then joins, etc.

What to do:

1. In one file write all the SQL commands.
2. Before each command add the problem statement as a comment line.
3. The file must be a simple text file with a TXT or SQL file extension.
4. Test your commands and make sure they are error-free before submitting the solution file.

Solutions:

/\* S-T: S-T: Use either join method. Use either join method. Use column aliases for every column displayed. List students and their evaluations when they were the evaluator. Show the student ID as "ID", full name (concatenated into one column) as "Evaluator", team name as "Team", evaluation ID as "Eval ID", semester as "Semester" and year as "Year". Sort by student ID and evaluation ID. \*/

 /\* JOIN operator \*/

select stdid as "ID", stdfname + ' ' + stdlname as "Evaluator", team\_name as "Team",

 eval\_id as "Eval ID", evalsemester as "Semester", evalyear as "Year"

from teams join students

 on teams.teamid = students.std\_teamID

 join evaluations

 on students.stdid = evaluations.evaluatorID

order by stdid, eval\_id;

 /\* WHERE Clause \*/

select stdid as "ID", stdfname + ' ' + stdlname as "Evaluator", team\_name as "Team",

 eval\_id as "Eval ID", evalsemester as "Semester", evalyear as "Year"

from teams, students, evaluations

WHERE teams.teamid = students.std\_teamID

and students.stdid = evaluations.evaluatorID

order by stdid, eval\_id;

/\* AW: Join tables in the FROM clause with the JOIN operator. List female marketing specialists. Show the business entity ID, the first and last name of the employee, the job title and gender. Sort by business entity ID. HINT: Be sure to put a table name in front of the business entity ID column in the SELECT clause. Also, remember the PERSON data is in a different schema than the EMPLOYEE data. \*/

/\* FROM clause must use the JOIN operator \*/

select employee.BusinessEntityID, FirstName, LastName, JobTitle, Gender

from AdventureWorks2008.HumanResources.employee

 join AdventureWorks2008.Person.person

 on employee.BusinessEntityID = person.BusinessEntityID

where JobTitle = 'Marketing Specialist'

and Gender = 'F';

/\* AW: Join tables in the WHERE clause. List female marketing specialists. Show the business entity ID, the first and last name of the employee, the job title and gender. Sort by business entity ID. HINT: Be sure to put a table name in front of the business entity ID column in the SELECT clause. \*/

/\* Tables joined in the WHERE clause. \*/

select employee.BusinessEntityID, FirstName, LastName, JobTitle, Gender

from AdventureWorks2008.HumanResources.employee, AdventureWorks2008.Person.person

where employee.BusinessEntityID = person.BusinessEntityID

and JobTitle = 'Marketing Specialist'

and Gender = 'F';

/\* AW: Join tables in the FROM clause with the JOIN operator.

List employees who work in the night shift and whose job title has the word ‘Production’ in it. Show the shift, the employee’s first and last name and job title.

Sort by employee's last name then first name. Hint: Join tables in the sequence of their relationships, such as shift joins employee department history which then joins employee which then joins, etc.

\*/

select Person.businessEntityID, FirstName, LastName, JobTitle, Gender

from AdventureWorks2008.HumanResources.Shift join

 AdventureWorks2008.HumanResources.EmployeeDepartmentHistory

 on Shift.ShiftID = EmployeeDepartmentHistory.ShiftID

 join AdventureWorks2008.HumanResources.Employee

 on EmployeeDepartmentHistory.BusinessEntityID = Employee.BusinessEntityID

 join AdventureWorks2008.Person.Person

 on Employee.BusinessEntityID = Person.BusinessEntityID

where JobTitle like '%Production%'

and Shift.Name = 'Night'

order by LastName, FirstName;

/\* AW: Join tables in the WHERE clause.

List employees who work in the night shift and whose job title has the word ‘Production’ in it. Show the shift, the employee’s first and last name and job title.

Sort by employee's last name then first name. Hint: Join tables in the sequence of their relationships, such as shift joins employee department history which then joins employee which then joins, etc.

\*/

select Person.businessEntityID, FirstName, LastName, JobTitle, Gender

from AdventureWorks2008.HumanResources.Shift,

 AdventureWorks2008.HumanResources.EmployeeDepartmentHistory,

 AdventureWorks2008.HumanResources.Employee,

 AdventureWorks2008.Person.Person

where shift.ShiftID = EmployeeDepartmentHistory.ShiftID

and EmployeeDepartmentHistory.BusinessEntityID = Employee.BusinessEntityID

and Employee.BusinessEntityID = Person.BusinessEntityID

and JobTitle like '%Production%'

and Shift.Name = 'Night'

order by LastName, FirstName;