SQL Fundamentals – IN-LINE SUBQUERIES

Often times there is more than one way to write an SQL statement to answer a question. However, for these assignment be sure to use the technique taught in the lesson if you can think of more than one way to write the SQL. The point of the lessons and assignments is to learn and practice each technique.

1. AW: Combine employee department history with the sift table. Show the department name (not the ID), shift name (not just the #), business entity ID, start date, and end date.
2. AW: Combine information from the employee table and the person table. List the businessEntityID and the first and last name as one field called emp\_name.
3. AW: Use the two previous queries in the FROM clause and show people working in the evening shift if they are not in the production department and the end date is not null (they still work that shift in that department). List the department name, the shift name, the employee name, and start date. Sort by department name then employee name.

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:

/\* AW: Combine employee department history with the sift table. Show the department name (not the ID), shift name (not just the #), business entity ID, start date, and end date. \*/

select Department.Name as dept\_Name, shift.name as Shift\_Name,

 EmployeeDepartmentHistory.businessEntityID as empID,

 startDate, endDate

from AdventureWorks2008.HumanResources.Shift join

 AdventureWorks2008.HumanResources.EmployeeDepartmentHistory

 on shift.shiftID = EmployeeDepartmentHistory.ShiftID

 join AdventureWorks2008.HumanResources.Department

 on EmployeeDepartmentHistory.DepartmentID = Department.DepartmentID;

/\* AW: Combine information from the employee table and the person table. List the businessEntityID and the first and last name as one field called emp\_name. \*/

select employee.BusinessEntityID as e\_id,

 FirstName + ' ' + lastName as emp\_name

from AdventureWorks2008.HumanResources.Employee

 join AdventureWorks2008.Person.Person

 on employee.businessEntityID = person.businessEntityID;

/\* AW: Use the two previous queries in the FROM clause and show people working in the evening shift if they are not in the production department and the end date is not null (they still work that shift in that department). List the department name, the shift name, the employee name, and start date. Sort by

department name then employee name. \*/

select dept\_name, shift\_name, emp\_name

from

 (select Department.Name as dept\_Name, shift.name as Shift\_Name,

 EmployeeDepartmentHistory.businessEntityID as empID,

 startDate, endDate

 from AdventureWorks2008.HumanResources.Shift

 join AdventureWorks2008.HumanResources.EmployeeDepartmentHistory

 on shift.shiftID = EmployeeDepartmentHistory.ShiftID

 join AdventureWorks2008.HumanResources.Department

 on EmployeeDepartmentHistory.DepartmentID = Department.DepartmentID

 ) ShiftInfo,

 (select employee.BusinessEntityID as e\_id,

 FirstName + ' ' + lastName as emp\_name

 from AdventureWorks2008.HumanResources.Employee

 join AdventureWorks2008.Person.Person

 on employee.businessEntityID = person.businessEntityID

 ) Deptinfo

where shiftInfo.empID = deptInfo.e\_id

and dept\_name <> 'Production'

and shift\_name = 'Evening'

and endDate is null

order by dept\_name, emp\_name;