SQL – Assignment 5

Submission: See folder "Hướng dẫn nộp bài qua Git"

Barem:

Exercise 1 Question 1	Exercise 1 Question 2	Exercise 1 Question 3	Exercise 2 Question 1	Exercise 2 Question 2	Exercise 2 Question 3	Exercise 2 Question 4
20%	10%	10%	15%	15%	15%	15%

Exercise 1: Subquery

Question 1: Write a query that filters data and return the column "Name" from table Production.Product. The filtering of rows is achieved by a WHERE clause that compares a single value from a subquery.

The inner subquery shall return a specific ProductSubcategoryID that the outer query uses as a filter of products to include in the report. The inner query will use its own WHERE clause to deliver its value, the ProductSubcategoryID, by retrieving it where the column "Name" in table Production.ProductSubcategory have the value of 'Saddles'.

The result set should look like the following.

```
Name
----------------------------
LL Mountain Seat/Saddle
ML Mountain Seat/Saddle
HL Mountain Seat/Saddle
LL Road Seat/Saddle
ML Road Seat/Saddle
HL Road Seat/Saddle
LL Touring Seat/Saddle
ML Touring Seat/Saddle
HL Touring Seat/Saddle

(9 row(s) affected)
```

Question 2: In this exercise you can change the previous query to deliver the following result set. The WHERE clause in the subquery will now use the wildcard string 'Bo%' for a comparison.

The result set should look like the following.

```
Name
----------------------------
```

Water Bottle - 30 oz.
Mountain Bottle Cage
Road Bottle Cage
LL Bottom Bracket
ML Bottom Bracket
HL Bottom Bracket
(6 row(s) affected)

Question 3:

Write a query that return all products that has the same price as the cheapest (lowest ListPrice) Touring Bike (ProductSubcategoryID = 3). Use the MIN() aggregate function in the subquery to return the lowest ListPrice to the outer query.

The result set should look like the following.

Name

Touring-3000 Blue, 54
Touring-3000 Blue, 58
Touring-3000 Blue, 62
……
Touring-3000 Yellow, 62
Touring-3000 Blue, 44
Touring-3000 Blue, 50

(10 row(s) affected)

Exercise 2: JOIN nhiều bảng

Question 1: Write a query that lists the country and province names stored in AdventureWorks2008sample database. In the Person schema you will find the CountryRegion and StateProvince tables. Join them and produce a result set similar to the following. Notice that there is no particular sort order in the result set.

Country Province
--- ---
CanadaAlberta
United States Alaska
United States Alabama
United States Arkansas
American Samoa American Samoa
……
France Belford (Territoire de)
France Essonne
France Hauts de Seine
FranceSeine Saint Denis
France Val de Marne
France Val d'Oise

(181 row(s) affected)

Question 2: Continue to work with the previous query and add a filter to only list the countries Germany and Canada. Also notice the sort order and column headings of the result set. Your result set should look similar to the following.

```
Country                    Province
-------------------------- ------------------------
Canada                     Alberta
Canada                     British Columbia
Canada                     Brunswick
CanadaLabrador
CanadaManitoba
CanadaNewfoundland
                    ……
GermanyBrandenburg
GermanyHamburg
Germany                    Hessen
Germany                    Nordrhein-Westfalen
GermanySaarland
GermanySaxony

(20 row(s) affected
```

Question 3:

We want information about orders. From the Sales.SalesOrderHeader table we want the SalesOrderID, OrderDate and SalesPersonIDcolums. From the Sales.SalesPerson table we want the BusinessEntityID (which identifies the sales person), Bonus and the SalesYTD (how much this person sold for yet this year) columns.

(As an aside, note that joining SalesOrderHeader to SalesPerson will restrict the result to non-Internet orders (order processed on the Internet has 1 in the OnlineOrderFlag, and has NULL for the SalesPersonID column.)

Note that the time portion below has been removed from the OrderDate column for presentation purposes.

```
SalesOrderID OrderDate   SalesPersonID  BusinessEntityID Bonus
SalesYTD
------------ ----------- ------------- ---------------- ---------- ---------------------
43659        2001-07-01  279           279              6700,00    2811012,7151
43660        2001-07-01  279           279              6700,00    2811012,7151
43661        2001-07-01  282           282              5000,00    3189356,2465
43662        2001-07-01  282           282              5000,00    3189356,2465
43663        2001-07-01  276           276              2000,00    5200475,2313
43664        2001-07-01  280           280              5000,00    0,00
.....
```

71949	2004-06-01	277	277	2500,00	3857163,6332
71950	2004-06-01	279	279	6700,00	2811012,7151
71951	2004-06-01	279	279	6700,00	2811012,7151
71952	2004-06-01	275	275	4100,00	4557045,0459

(3806 row(s) affected)

Question 4:
Use above query, add JobTitle and remove the SalesPersonID and the BusinessEntityID columns. You need to join to the HumanResources.Employee table.

SalesOrderID	OrderDate	Jobtitle	Bonus	SalesYTD
43659	2001-07-01	Sales Representative	6700.00	2811012,7151
43660	2001-07-01	Sales Representative	6700.00	2811012,7151
43661	2001-07-01	Sales Representative	5000.00	3189356,2465
43662	2001-07-01	Sales Representative	5000.00	3189356,2465
.....				
71947	2004-06-01	Sales Representative	2500.00	3857163,6332
71948	2004-06-01	Sales Representative	6700.00	2811012,7151
71949	2004-06-01	Sales Representative	2500.00	3857163,6332
71950	2004-06-01	Sales Representative	6700.00	2811012,7151
71951	2004-06-01	Sales Representative	6700.00	2811012,7151
71952	2004-06-01	Sales Representative	4100.00	4557045,0459

(3806 row(s) affected)

Note:

- Write the coding convention correctly
- following the best practices
- Don’t copy other people (there will be penalties if found)