

Assignment Day1 –SQL: Comprehensive practice

Amber Han

Write queries for following scenarios

Using AdventureWorks Database

1. Write a query that retrieves the columns ProductID, Name, Color and ListPrice from the Production.Product table, with no filter.

```
SELECT ProductID, Name, Color, ListPrice FROM Production.Product
```

2. Write a query that retrieves the columns ProductID, Name, Color and ListPrice from the Production.Product table, the rows that are 0 for the column ListPrice

```
SELECT ProductID, Name, Color, ListPrice FROM Production.Product WHERE  
ListPrice = 0
```

3. Write a query that retrieves the columns ProductID, Name, Color and ListPrice from the Production.Product table, the rows that are rows that are NULL for the Color column.

```
SELECT ProductID, Name, Color, ListPrice FROM Production.Product WHERE Color IS  
NULL
```

4. Write a query that retrieves the columns ProductID, Name, Color and ListPrice from the Production.Product table, the rows that are not NULL for the Color column.

```
SELECT ProductID, Name, Color, ListPrice FROM Production.Product WHERE Color  
IS NOT NULL
```

5. Write a query that retrieves the columns ProductID, Name, Color and ListPrice from the Production.Product table, the rows that *are not* NULL for the column Color, and the column ListPrice has a value greater than zero.

```
SELECT ProductID, Name, Color, ListPrice FROM Production.Product WHERE Color IS  
NOT NULL AND ListPrice > 0
```

6. Generate a report that concatenates the columns Name and Color from the Production.Product table by excluding the rows that are null for color.

Name

```
SELECT Name + ' ' + Color as Name_Color FROM Production.Product WHERE Color IS NOT  
NULL
```

7. Write a query that generates the following result set from Production.Product:
Name And Color

```
-----  
NAME: LL Crankarm -- COLOR: Black  
NAME: ML Crankarm -- COLOR: Black  
NAME: HL Crankarm -- COLOR: Black  
NAME: Chainring Bolts -- COLOR: Silver
```

NAME: Chainring Nut -- COLOR: Silver

NAME: Chainring -- COLOR: Black

.....

```
SELECT 'Name:' + Name + '--COLOR:' + Color AS 'Name And Color' FROM
Production.Product WHERE Name LIKE '%Crankarm%' OR Name LIKE '%Chainring%'
ORDER BY ProductID
```

8. Write a query to retrieve the to the columns ProductID and Name from the Production.Product table filtered by ProductID from 400 to 500

```
SELECT ProductID, Name FROM Production.Product WHERE ProductID BETWEEN '400'
AND '500'
```

9. Write a query to retrieve the to the columns ProductID, Name and color from the Production.Product table restricted to the colors black and blue

```
SELECT ProductID, Name, Color FROM Production.Product WHERE Color = 'blue' OR
Color = 'black'
```

10. Write a query to generate a report on products that begins with the letter S.

```
SELECT ProductID, Name, Color FROM Production.Product WHERE Name LIKE 's%'
```

11. Write a query that retrieves the columns Name and ListPrice from the Production.Product table. Your result set should look something like the following. Order the result set by the Name column.

Name	ListPrice
Seat Lug	0,00
Seat Post	0,00
Seat Stays	0,00
Seat Tube	0,00
Short-Sleeve Classic Jersey, L	53,99
Short-Sleeve Classic Jersey, M	53,99

```
SELECT Name, ListPrice FROM Production.Product WHERE Name LIKE 's%' ORDER BY
Name
```

12. Write a query that retrieves the columns Name and ListPrice from the Production.Product table. Your result set should look something like the following. Order the result set by the Name column. The products name should start with either 'A' or 'S'

Name	ListPrice
Adjustable Race	0,00
All-Purpose Bike Stand	159,00
AWC Logo Cap	8,99
Seat Lug	0,00
Seat Post	0,00

.....

```

LIKE 'a%'
SELECT Name, ListPrice FROM Production.Product WHERE Name LIKE 's%' OR Name
ORDER BY Name

```

13. Write a query so you retrieve rows that have a Name that begins with the letters SPO, but is then *not* followed by the letter K. After this zero or more letters can exist. Order the result set by the *Name* column.

```

SELECT Name, ListPrice FROM Production.Product WHERE Name LIKE 'spo[^k]%' ORDER
BY Name

```

14. Write a query that retrieves *unique* colors from the table Production.Product. Order the results in descending manner

```

SELECT DISTINCT Color FROM Production.Product ORDER BY Color

```

15. Write a query that retrieves the unique combination of columns ProductSubcategoryID and Color from the Production.Product table. Format and sort so the result set accordingly to the following. We do not want any rows that are NULL in any of the two columns in the result.

```

SELECT DISTINCT Color, ProductSubcategoryID FROM Production.Product
WHERE Color IS NOT NULL AND ProductSubcategoryID IS NOT NULL

```

16. Something is “wrong” with the WHERE clause in the following query.
 We do not want any Red or Black products from any SubCategory than those with the value of 1 in column ProductSubCategoryID, unless they cost between 1000 and 2000.

Note:

The LEFT() function will be covered in a forthcoming module.

```

SELECT ProductSubCategoryID
      , LEFT([Name],35) AS [Name]
      , Color, ListPrice
FROM Production.Product
WHERE Color IN ('Red','Black')
      OR ListPrice BETWEEN 1000 AND 2000
      AND ProductSubCategoryID = 1
ORDER BY ProductID
SELECT ProductSubCategoryID
      , LEFT([Name],35) AS [Name]
      , Color, ListPrice
FROM Production.Product
WHERE Color NOT IN ('Red','Black')
      AND ProductSubCategoryID = 1
      OR (ListPrice BETWEEN 1000 AND 2000)
ORDER BY ProductID

```

17. Write the query in the editor and execute it. Take a look at the result set and then adjust the query so it delivers the following result set.

ProductSubcategoryID	Name	Color	ListPrice
14	HL Road Frame - Black, 58	Black	1431,50
14	HL Road Frame - Red, 58	Red	1431,50
14	HL Road Frame - Red, 62	Red	1431,50
14	HL Road Frame - Red, 44	Red	1431,50
14	HL Road Frame - Red, 48	Red	1431,50
14	HL Road Frame - Red, 52	Red	1431,50
14	HL Road Frame - Red, 56	Red	1431,50
12	HL Mountain Frame - Silver, 42	Silver	1364,50
12	HL Mountain Frame - Silver, 44	Silver	1364,50
12	HL Mountain Frame - Silver, 48	Silver	1364,50
.....			
2	Road-350-W Yellow, 44	Yellow	1700,99
2	Road-350-W Yellow, 48	Yellow	1700,99
1	Mountain-500 Black, 40	Black	539,99
1	Mountain-500 Black, 42	Black	539,99
1	Mountain-500 Black, 44	Black	539,99
1	Mountain-500 Black, 48	Black	539,99
1	Mountain-500 Black, 52	Black	539,99

```
SELECT ProductSubcategoryID, LEFT([Name], 35) AS [Name], Color, ListPrice
FROM Production.Product WHERE Color IN ('Red', 'Black') AND ProductSubcategoryID = 1 OR
(ListPrice BETWEEN 1000 AND 2000)
ORDER BY ProductID
```

GOOD LUCK.