Assignment Day1 –SQL: Comprehensive practice

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.

```sql
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, excludes the rows that ListPrice is 0.

```sql
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 NULL for the Color column.

```sql
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.

```sql
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.

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

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

```sql
SELECT 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 [Name And Color]
FROM Production.Product
WHERE Color IS NOT NULL
```

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 = 'Black' OR Color = 'Blue'
```

10. Write a query to get a result set 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
        .........
```

```
SELECT Name, ListPrice

FROM Production.Product

WHERE Name LIKE '[A,S]%'

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 exists. Order the result set by the *Name* column.

```
SELECT Name

FROM Production.Product

WHERE Name LIKE 'SPO%' AND NAME NOT LIKE 'SPOK%'

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 DESC
```

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 ProductSubcategoryID, Color

FROM Production.Product

WHERE ProductSubcategoryID IS NOT NULL AND

Color IS NOT NULL
```

GOOD LUCK.