* 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;

* 1. Write a query that retrieves the columns ProductID, Name, Color and ListPrice from the Production.Product table, excludes the rows that ListPrice is 0.

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

* 1. 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;

* 1. 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;

* 1. 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;

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

SELECT Name + ' ' + Color FROM Production.Product WHERE Color IS NOT NULL;

* 1. 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 Color IS NOT NULL;

* 1. 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;

* 1. 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 IN ('Black', 'Blue');

* 1. 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%';

* 1. 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;

* 1. 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%' OR Name LIKE 'S%' ORDER BY Name;

* 1. 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, ListPrice FROM Production.Product WHERE Name LIKE 'SPO[^K]%' ORDER BY Name;

* 1. 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;

* 1. 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;