Session Date	:	4 Oct 2021
Semester	:	3
Subject	:	Basis Data
Topic	:	Programming in SQL Server (PL/SQL)
Activity	:	Practicing Programming (PL/SQL) in SQL Server
Duration	:	110 minutes
Rules		Individual
Deliverable	:	Softcopy
Dead line	:	End of Session
Place to deliver	:	http://ecourse.del.ac.id/
Objective	:	Students able to use PL/SQL in SQL Server
Lecturer	:	PAT/IUS/RSL

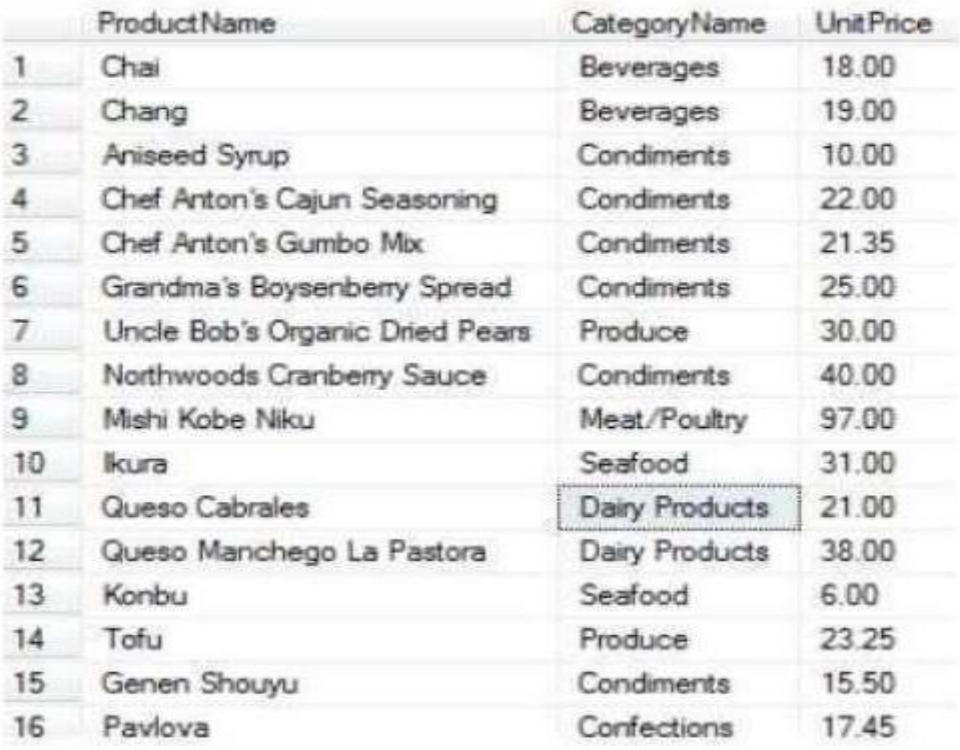
Procedure Restore database Nortwind Database to your SQL Server database. Use Nortwind database for these excercises.

Exercise 1: CASE..END statement

Execute SQL Statement below:

```
SELECT ProductName, CategoryName, UnitPrice
FROM Products INNER JOIN Categories
ON Products. CategoryID=Categories. CategoryID;
```

The result set are :

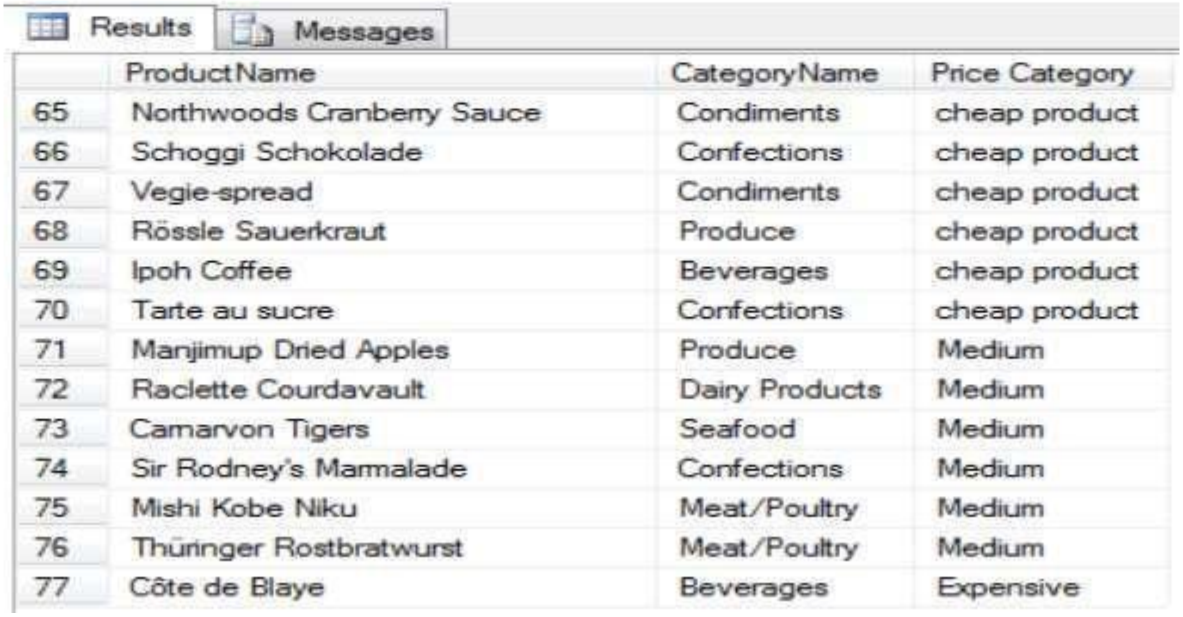


	ProductName	CategoryName	UnitPrice
1	Chai	Beverages	18.00
2	Chang	Beverages	19.00
3	Aniseed Syrup	Condiments	10.00
4	Chef Anton's Cajun Seasoning	Condiments	22.00
5	Chef Anton's Gumbo Mix	Condiments	21.35
6	Grandma's Boysenberry Spread	Condiments	25.00
7	Uncle Bob's Organic Dried Pears	Produce	30.00
8	Northwoods Cranberry Sauce	Condiments	40.00
9	Mishi Kobe Niku	Meat/Poultry	97.00
10	Ikura	Seafood	31.00
11	Queso Cabrales	Dairy Products	21.00
12	Queso Manchego La Pastora	Dairy Products	38.00
13	Konbu	Seafood	6.00
14	Tofu	Produce	23.25
15	Genen Shouyu	Condiments	15.50
16	Pavlova	Confections	17.45

Use **Case…End** statement to create t-sql query to display the products' price and category as **Not Yet Priced, Cheap Products**, **Medium**, and **Expensive**. The condition for categorization is as follows:

- If certain products not priced (the value in column price is empty/null), then 'Not Yet Priced'.
- If less than or equal 20, then 'Widihh Murah Banget'.
- If 20<price< =50, then 'Produk Murah Aja.
- If 50<price< =100, then 'Agak Mahal nih.
- If 100<price< =150, then 'Fix Mahal!!'.
- If >150 then 'Khusus Sultan!'.

The figure below depicts the result if your query executed successfully (result set: 77 rows). **Use join between products and categories table.**



	ProductName	CategoryName	Price Category
65	Northwoods Cranberry Sauce	Condiments	cheap product
66	Schoggi Schokolade	Confections	cheap product
67	Vegie-spread	Condiments	cheap product
68	Rössle Sauerkraut	Produce	cheap product
69	Ipoh Coffee	Beverages	cheap product
70	Tarte au sucre	Confections	cheap product
71	Manjimup Dried Apples	Produce	Medium
72	Raclette Courdavault	Dairy Products	Medium
73	Carnarvon Tigers	Seafood	Medium
74	Sir Rodney's Marmalade	Confections	Medium
75	Mishi Kobe Niku	Meat/Poultry	Medium
76	Thüringer Rostbratwurst	Meat/Poultry	Medium
77	Côte de Blaye	Beverages	Expensive

Exercise 2: IF..THEN…ELSE..Statement

Task -1 Execute SQL Statement below:

```
use  TennisDB
DECLARE @CharTown Char(1),
@Town Char(11);
SET @CharTown = 'E' ;
SET @Town =
CASE  @CharTown
WHEN 'S' THEN 'Stratford'
WHEN 'I' THEN 'Inglewood'
WHEN 'E' THEN 'Eltham'
WHEN 'M' THEN 'Midhurst'
WHEN 'D' THEN 'Douglas'
```

```
E ND;
```

```
SELECT * FROM PLAYERS
WHERE TOWN=@Town;
```

Modify the SQL statement above using IF…Then…Else….Statement.

Task -2 Use TennisDB Database. Display in text field (**not in grid**), the information (*playerno*,*name*, *nr_won*) about the player who won more than 1 matches.

Exercise 3:While...Statement

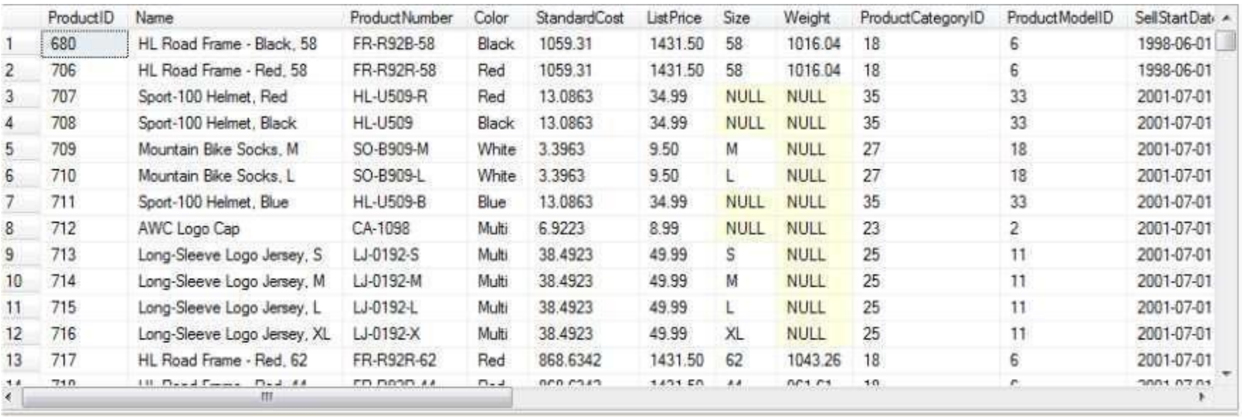
Task -1

Attach the AdventureWorks database (Databases -> Right Click -> Attach -> Click Add -> Choose location of "AdventureWorksLT2008_Data" -> Click OK -> Click OK).

Execute sql statement below:

```
Select * FROM SalesLT. Product
```

The result set are:



	ProductID	Name	ProductNumber	Color	StandardCost	ListPrice	Size	Weight	ProductCategoryID	ProductModelID	SellStartDat
1	680	HL Road Frame - Black, 58	FR-R92B-58	Black	1059.31	1431.50	58	1016.04	18	6	1998-06-01
2	706	HL Road Frame - Red, 58	FR-R92R-58	Red	1059.31	1431.50	58	1016.04	18	6	1998-06-01
3	707	Sport-100 Helmet, Red	HL-U509-R	Red	13.0863	34.99	NULL	NULL	35	33	2001-07-01
4	708	Sport-100 Helmet, Black	HL-U509	Black	13.0863	34.99	NULL	NULL	35	33	2001-07-01
5	709	Mountain Bike Socks, M	SO-B909-M	White	3.3963	9.50	M	NULL	27	18	2001-07-01
6	710	Mountain Bike Socks, L	SO-B909-L	White	3.3963	9.50	L	NULL	27	18	2001-07-01
7	711	Sport-100 Helmet, Blue	HL-U509-B	Blue	13.0863	34.99	NULL	NULL	35	33	2001-07-01
8	712	AWC Logo Cap	CA-1098	Multi	6.9223	8.99	NULL	NULL	23	2	2001-07-01
9	713	Long-Sleeve Logo Jersey, S	LJ-0192-S	Multi	38.4923	49.99	S	NULL	25	11	2001-07-01
10	714	Long-Sleeve Logo Jersey, M	LJ-0192-M	Multi	38.4923	49.99	M	NULL	25	11	2001-07-01
11	715	Long-Sleeve Logo Jersey, L	LJ-0192-L	Multi	38.4923	49.99	L	NULL	25	11	2001-07-01
12	716	Long-Sleeve Logo Jersey, XL	LJ-0192-X	Multi	38.4923	49.99	XL	NULL	25	11	2001-07-01
13	717	HL Road Frame - Red, 62	FR-R92R-62	Red	868.6342	1431.50	62	1043.26	18	6	2001-07-01

Then, find the average of ListPrice:

```
Select AVG(ListPrice) FROM SalesLT. Product,
```

Then result are 744.5952 Consider the condition below and use **while** to build PL/SQL statement for the condition. If the average ListPrice of product is less than $1000, use **WHILE** to :

- doubles the ListPrice for every products and
- If the maximum ListPrice is less than or equal to $4000, then WHILE loop restarts and doubles the prices again. This loop continues doubling the prices until the maximum price is greater than $4000, and then exits the WHILE loop and prints a message.