

0. What to learn

1. Northwind Database

- 1.1. 1st way : Restore database from bak file
- 1.2. 2nd way : Restore database from sql file
- 1.3. Northwind Database Diagram

2. Select

3. Group By

4. Where V.S. Having

0. What to learn

- 1.
Select (DISTINCT/TOP n)..
WHERE...(AND/OR)...
ORDER BY ...(ASC/DESC)
- 2.
SQL Operator
Reference:
https://www.w3schools.com/sql/sql_operators.asp
<https://www.tutorialspoint.com/sql/sql-wildcards.htm>
<https://docs.microsoft.com/en-us/sql/t-sql/language-elements/wildcard-character-s-to-match-transact-sql>
<https://docs.microsoft.com/en-us/sql/t-sql/language-elements/wildcard-character-s-not-to-match-transact-sql>
- 2.1.
SQL Comparison Operators
 - 2.1.1. = : Equal to
 - 2.1.2. > : Greater than
 - 2.1.3. < : Less than
 - 2.1.4. >= : Greater than or equal to
 - 2.1.5. <= : Less than or equal to
 - 2.1.6. != or <> : Not equal to
- 2.2.
SQL Logical Operators
 - 2.2.1. ALL : TRUE if all of the subquery values meet the condition
 - 2.2.2. SOME : TRUE if any of the subquery values meet the condition
 - 2.2.3. ANY : TRUE if any of the subquery values meet the condition
 - 2.2.4. AND : TRUE if all the conditions separated by AND is TRUE
 - 2.2.5. OR : TRUE if any of the conditions separated by OR is TRUE
 - 2.2.6. BETWEEN : TRUE if the operand is within the range of comparisons.
 - 2.2.7. IN : TRUE if the operand is equal to one of a list of expressions
 - 2.2.8. EXISTS : TRUE if the subquery returns one or more records
 - 2.2.9. LIKE : TRUE if the operand matches a pattern
 - 2.2.10. NOT : Displays a record if the condition(s) is NOT TRUE
- 2.3.
SQL Wildcard Operators
 - 2.3.1. % : The percent sign means matches zero or more characters.
 - 2.3.2. _ : The underscore (_) means matches one character
 - 2.3.3. [] : Matches any single character within the specified range between brackets []
 - 2.3.4. [^] : Matches any single character that is NOT within the range between brackets []

3.

- In order to use aggregate, we need Group By
- aggregate include Count(), Sum(), avg(), Min(), Max().

4.

WHERE V.S. Having

4.1.

WHERE and HAVING can be used together.

Aggregates includes Count, Sum, Avg, Min and Max.

Aggregates must be happened after Group By, because Aggregates need Group By to perform.

4.2.

WHERE operator filters the rows before Group by.

Thus, WHERE operator CAN NOT be used to filter Aggregates.

HAVING operator filters the Group after Group by.

Thus, HAVING operator CAN filter Aggregates.

4.3.

HAVING is slower than WHERE.

Try to use WHERE to replace HAVING if possible.

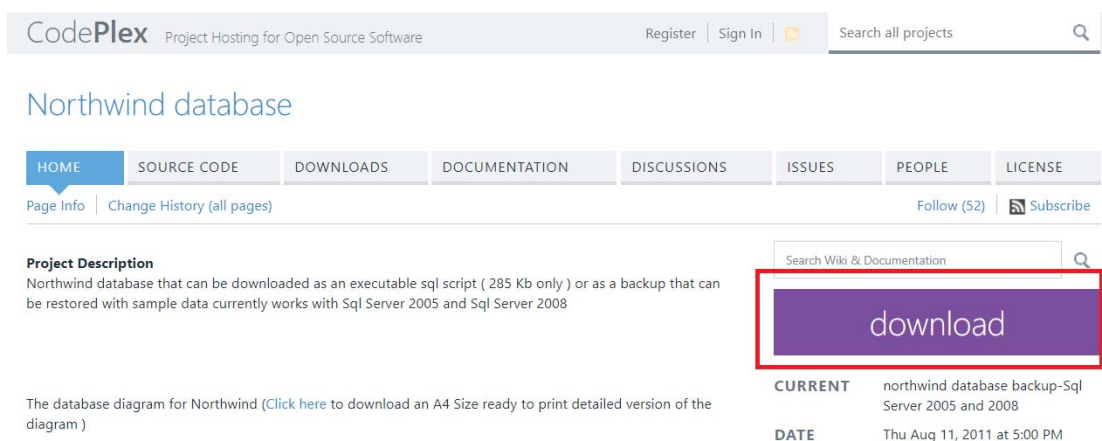
1. Northwind Database

There are 2 ways to re-store database.

1.1. 1st way : Restore database from bak file

Go the the following link and download the Northwind Database

<https://northwinddatabase.codeplex.com/>



CodePlex Project Hosting for Open Source Software Register Sign In Search all projects

Northwind database

HOME SOURCE CODE DOWNLOADS DOCUMENTATION DISCUSSIONS ISSUES PEOPLE LICENSE

Page Info Change History (all pages) Follow (52) Subscribe

Project Description
Northwind database that can be downloaded as an executable sql script (285 Kb only) or as a backup that can be restored with sample data currently works with Sql Server 2005 and Sql Server 2008

The database diagram for Northwind (Click here to download an A4 Size ready to print detailed version of the diagram)

Search Wiki & Documentation

download

CURRENT northwind database backup-Sql Server 2005 and 2008

DATE Thu Aug 11, 2011 at 5:00 PM

Press the download button and get the file "Northwind.bak.zip"

Unzip it and get the file "Northwind.bak"

If you don't have any software to unzip the file, you may use 7-zip (<http://www.7-zip.org/download.html>)

It is a very nice and free software.

In SQL server

--> Database --> Restore Database...

--> Device --> ... -->

-->Add --> paste the bak file path.

E.g.

D:\0_MyDocument\Desktop\Northwind.bak\Northwind.bak

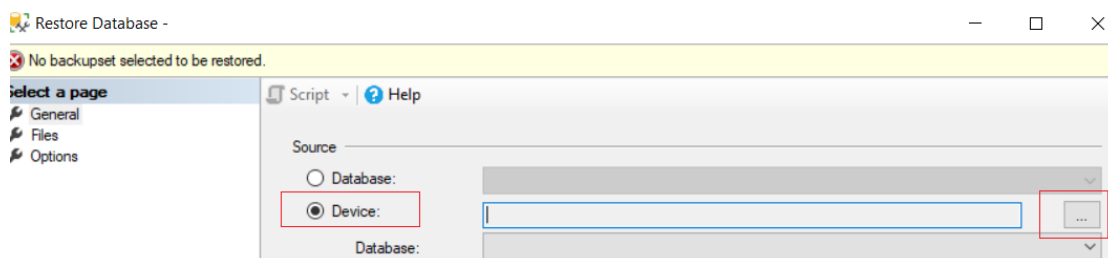
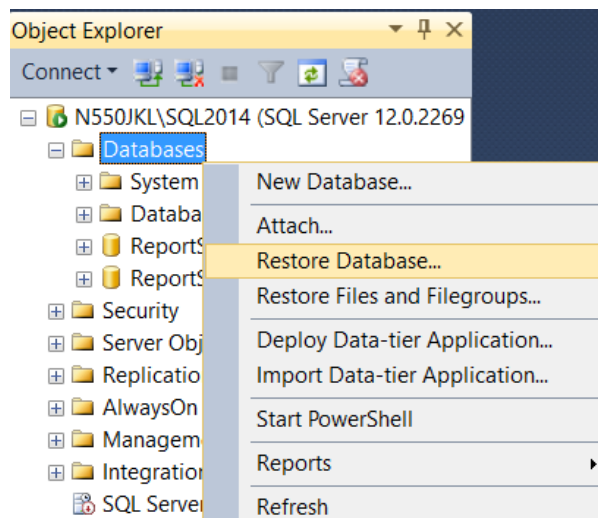
--> OK

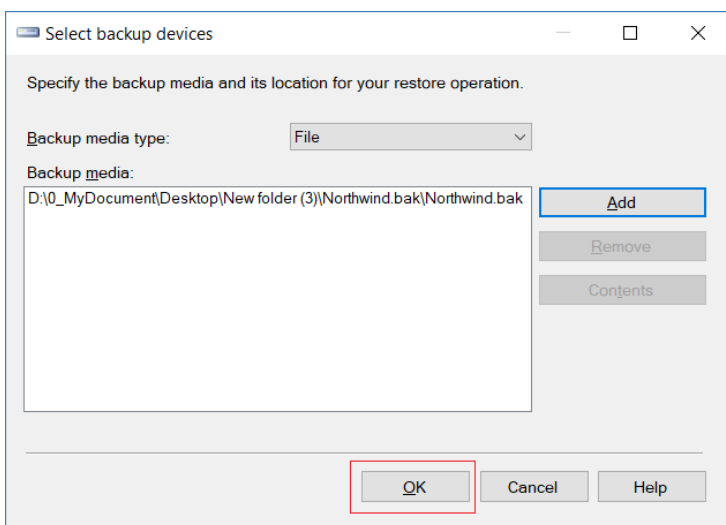
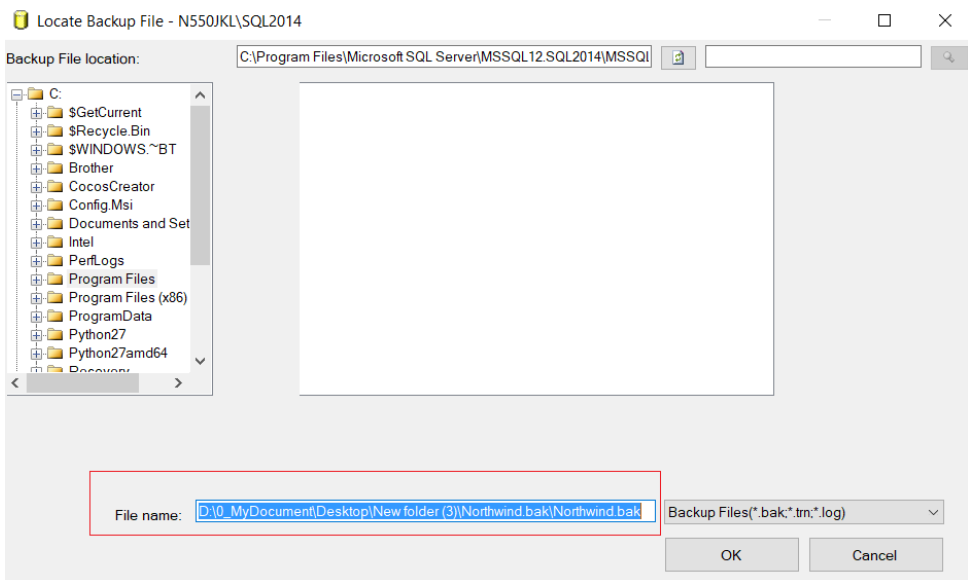
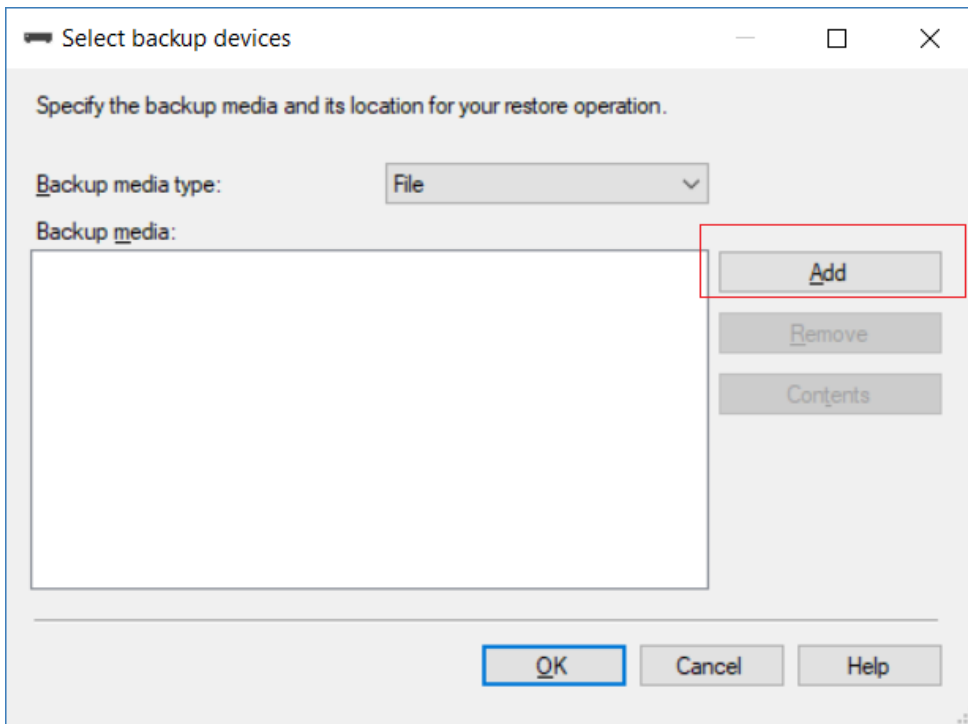
--> Verify Backup Media

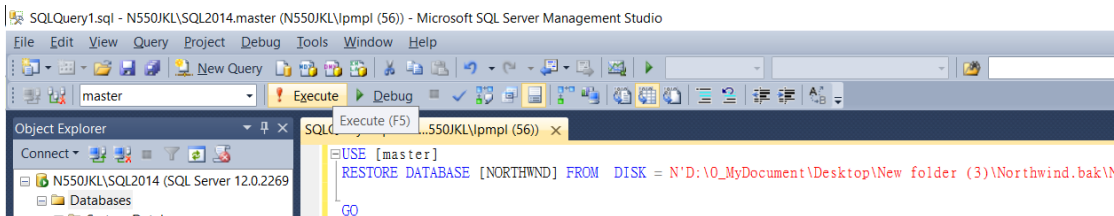
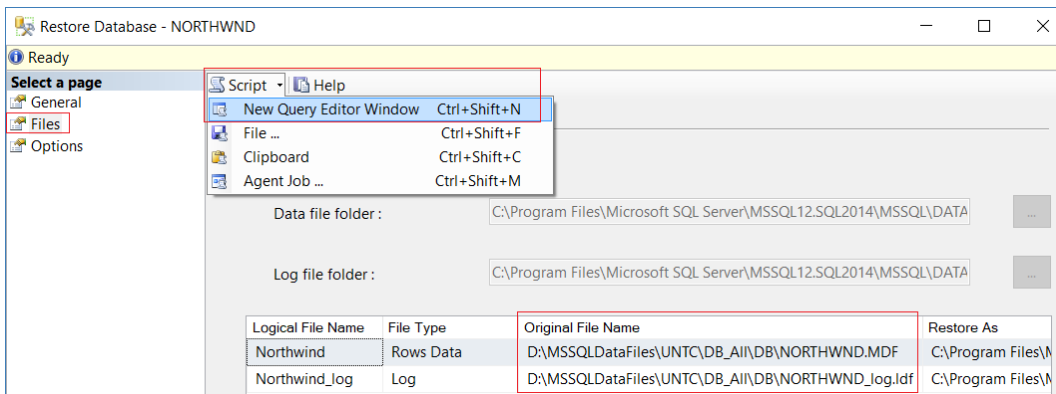
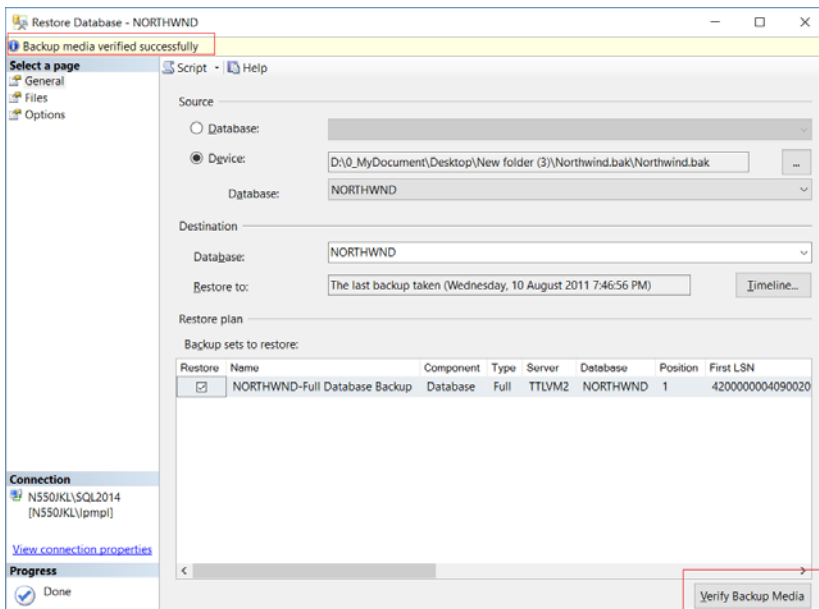
--> Files Tab --> Make sure the path is what you want.

--> Script --> New Query Editor Window

--> Execute the query





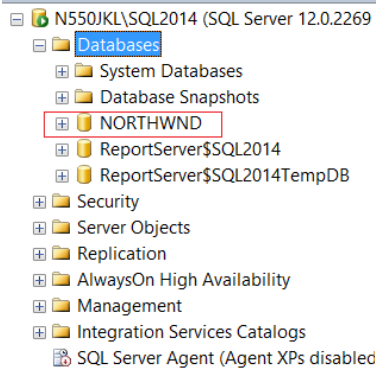


USE [master]

RESTORE DATABASE [NORTHWND] FROM DISK = N'D:\0_MyDocument\Desktop\New folder

(3)\Northwind.bak\Northwind.bak' WITH FILE = 1, MOVE N'Northwind' TO N'C:\Program Files\Microsoft SQL Server\MSSQL12.SQL2014\MSSQL\DATA\NORTHWND.MDF', MOVE N'Northwind_log' TO N'C:\Program Files\Microsoft SQL Server\MSSQL12.SQL2014\MSSQL\DATA\NORTHWND_log.ldf', NOUNLOAD, STATS = 5

GO



1.2. 2nd way : Restore database from sql file

Go to the following link to download sql file

<https://northwinddatabase.codeplex.com/releases/view/71634>

northwind database backup-Sql Server 2005 and 2008

Rating: ★★★★★ Based on 12 ratings	Released: Aug 11, 2011
Reviewed: 11 reviews	Updated: Aug 12, 2011 by binøj7
Downloads: 510462	Dev status: Alpha ?

RECOMMENDED DOWNLOAD



Northwind.bak.zip

example, 525K, uploaded Aug 12, 2011 - 401000 downloads

OTHER AVAILABLE DOWNLOADS



Northwind.sql.zip

source code, 285K, uploaded Aug 12, 2011 - 109462 downloads

Download [Northwind.sql.zip](#) and unzip it.

You will get the file called "northwind.sql"

In SQL server

--> Database --> New Database

--> General Tab --> Database Name : **Northwind**

--> Option Tab --> Recovery Model : **Simple**

New Database

Select a page: General Options Filegroups

Script Help

Database name: Northwind

Owner: <default>

☒ Use full-text indexing

Database files:

Logical Name	File Type	Filegroup	Initial Size (MB)	Autogrowth / Maxsize	Path
Northwind	ROW	PRIMARY	5	By 1 MB, Unlimited	C:\Program Files\Microsoft SQL Server\90\Shared\Northwind.mdf
Northwind_log	LOG	Not Applicable	1	By 10 percent, Unlimited	C:\Program Files\Microsoft SQL Server\90\Shared\Northwind.ldf

Connection:

Server: N550JKL\SQL2014

Connection: N550JKL\pmp1

[View connection](#)

Progress: Ready

Add Remove

OK Cancel

New Database

Select a page: General Options Filegroups

Script Help

Collation: <default>

Recovery model: Simple

Compatibility level: SQL Server 2014 (120)

Containment type: None

Other options:

Auto Update Statistics Asynchronously	False
Containment	
Default Fulltext Language LCID	1033
Default Language	English
Nested Triggers Enabled	True
Transform Noise Words	False
Two Digit Year Cutoff	2049
Cursor	
Close Cursor on Commit Enabled	False
Default Cursor	GLOBAL
FILESTREAM	
FILESTREAM Directory Name	
FILESTREAM Non-Transacted Access	Off
Miscellaneous	
Allow Snapshot Isolation	False

Allow Snapshot Isolation

OK Cancel

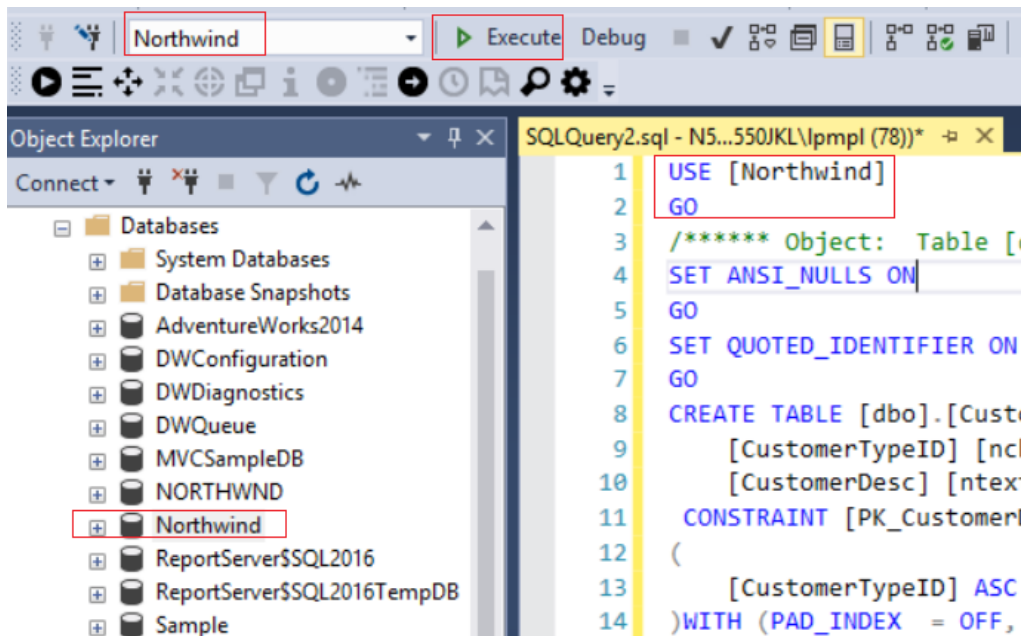
-->

Use Notepad++ to open the file "northwind.sql"

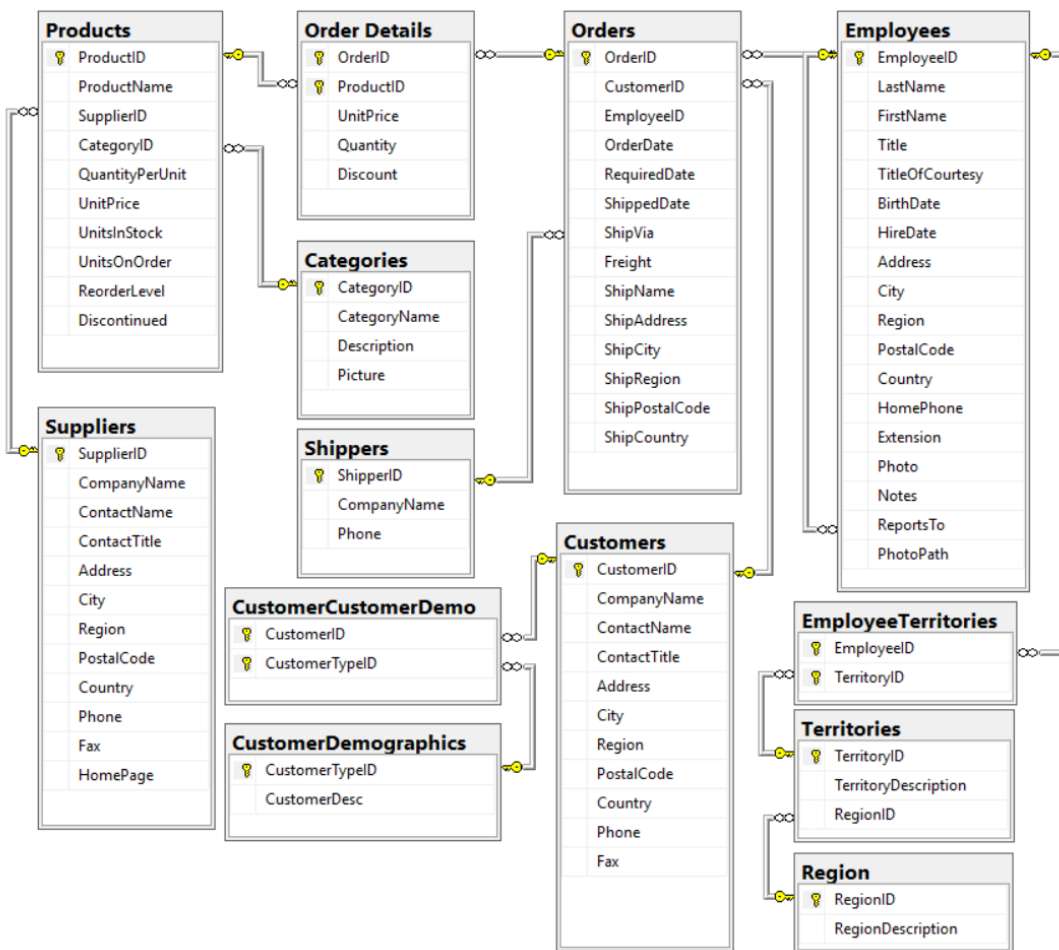
You may download Notepad++ here (<https://notepad-plus-plus.org/>)

Copy the code from "northwind.sql" into new query window.

Execute the query.



1.3. Northwind Database Diagram



2. Select

```
--=====
--T004_01_Select
--=====

/*
What to learn
1.
Select (DISTINCT/TOP n)..
WHERE...(AND/OR)...
ORDER BY ...(ASC/DESC)
2.
SQL Operator
Reference:
https://www.w3schools.com/sql/sql\_operators.asp
https://www.tutorialspoint.com/sql/sql-wildcards.htm
https://docs.microsoft.com/en-us/sql/t-sql/language-elements/wildcard-character-s-to-match-transact-sql
https://docs.microsoft.com/en-us/sql/t-sql/language-elements/wildcard-character-s-not-to-match-transact-sql
2.1.
SQL Comparison Operators
2.1.1. = : Equal to
2.1.2. > : Greater than
2.1.3. < : Less than
2.1.4. >= : Greater than or equal to
2.1.5. <= : Less than or equal to
2.1.6. != or <> : Not equal to
2.2.
SQL Logical Operators
2.2.1. ALL : TRUE if all of the subquery values meet the condition
2.2.2. SOME : TRUE if any of the subquery values meet the condition
2.2.3. ANY : TRUE if any of the subquery values meet the condition
2.2.4. AND : TRUE if all the conditions separated by AND is TRUE
2.2.5. OR : TRUE if any of the conditions separated by OR is TRUE
2.2.6. BETWEEN : TRUE if the operand is within the range of comparisons.
2.2.7. IN : TRUE if the operand is equal to one of a list of expressions
2.2.8. EXISTS : TRUE if the subquery returns one or more records
2.2.9. LIKE : TRUE if the operand matches a pattern
2.2.10. NOT : Displays a record if the condition(s) is NOT TRUE
2.3.
SQL Wildcard Operators
2.3.1. % : The percent sign means matches zero or more characters.
2.3.2. _ : The underscore (_) means matches one character
2.3.3. [] : Matches any single character within the specified range between brackets []
2.3.4. [^] : Matches any single character that is NOT within the range between brackets []
*/
```

```
--T004_01_01
```

```
--Select
```

```
--T004_01_01_01
```

```
USE Northwind;
```

```
GO -- Run the prvious command and begins new batch
```

```
SELECT *
```

```
FROM [dbo].[Customers];
```

Results		Messages									
CustomerID	CompanyName	ContactName	ContactTitle	Address	City	Region	PostalCode	Country	Phone	Fax	
1	ALFKI	Alfreds Futterkiste	Maria Anders	Sales Representative	Obere Str. 57	Berlin	NULL	12209	Germany	030-0074321	030-0076545
2	ANATR	Ana Trujillo Emparedados y helados	Ana Trujillo	Owner	Avda. de la Constitución 2222	México D.F.	NULL	05021	Mexico	(5) 555-4729	(5) 555-3745
3	ANTON	Antonio Moreno Taqueria	Antonio Moreno	Owner	Mataderos 2312	México D.F.	NULL	05023	Mexico	(5) 555-3932	NULL
4	AROUT	Around the Horn	Thomas Hardy	Sales Representative	120 Hanover Sq.	London	NULL	WA1 1DP	UK	(171) 555-7788	(171) 555-6750
5	BERGS	Berglunds snabbköp	Christina Berglund	Order Administrator	Berguvsvägen 8	Luleå	NULL	S-958 22	Sweden	0921-12 34 65	0921-12 34 67
Query executed successfully.											
				N550IKL\SOL2016 (13.0 SP1)		N550IKL\lsmel (57)		Northwind		00:00:00	91

Query executed successfully.

N550IKL\SQL2016 (13.0 SP1) N550IKL\jpmpl (57) Northwind 00:00:00 91 rows

--T004_01_01_02

SELECT *

FROM [Northwind].[dbo].[Customers];

--[DatabaseName].[SchemaName].[TableName]



CustomerID	CompanyName	ContactName	ContactTitle	Address	City	Region	PostalCode	Country	Phone	Fax
1	ALFKI	Alfreds Futterkiste	Sales Representative	Obere Str. 57	Berlin	NULL	12209	Germany	030-0074321	030-0076545
2	ANATR	Ana Trujillo Emparedados y helados	Owner	Avda. de la Constitución 2222	México D.F.	NULL	05021	Mexico	(5) 555-4729	(5) 555-3745
3	ANTON	Antonio Moreno Taquería	Owner	Mataderos 2312	México D.F.	NULL	05023	Mexico	(5) 555-3932	NULL
4	AROUT	Around the Horn	Sales Representative	120 Hanover Sq	London	NULL	WA1 1DP	UK	(171) 555-7788	(171) 555-6750
5	BERGS	Berglunds snabbköp	Order Administrator	Berguvsvägen 8	Luleå	NULL	S-958 22	Sweden	0921-12 34 65	0921-12 34 67

Query executed successfully. N550IKL\SQL2016 (13.0 SP1) N550IKL\jpmpl (57) Northwind 00:00:00 91 rows

--T004_01_02

--Select (DISTINCT/TOP n)..

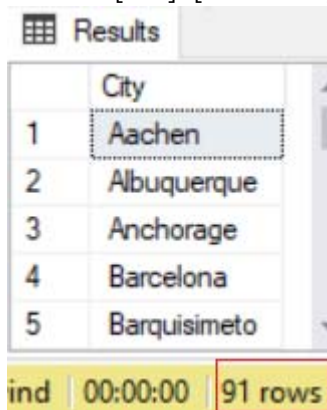
--T004_01_02_01

USE Northwind;

GO -- Run the previous command and begins new batch

SELECT [Customers].[City]

FROM [dbo].[Customers];



City
1 Aachen
2 Albuquerque
3 Anchorage
4 Barcelona
5 Barquisimeto

ind 00:00:00 91 rows

--T004_01_02_02

USE Northwind;

GO -- Run the previous command and begins new batch

SELECT DISTINCT

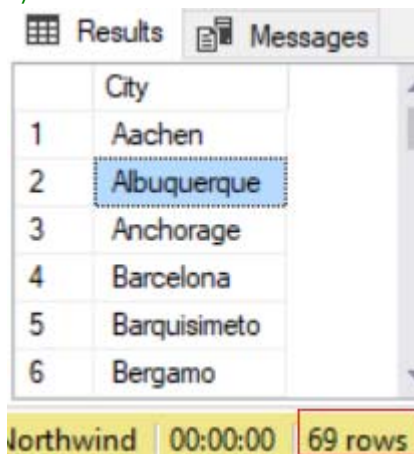
[Customers].[City]

FROM [dbo].[Customers];

/*

DISTINCT filter all the repeated [Customers].[City]

*/

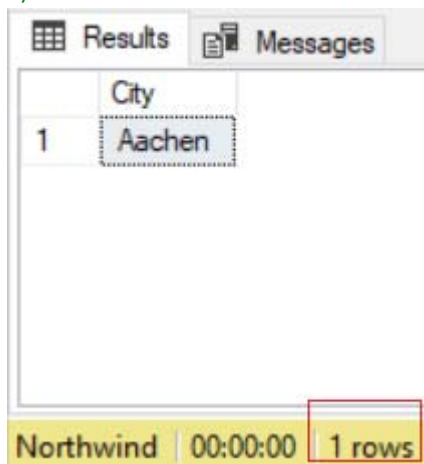


City
1 Aachen
2 Albuquerque
3 Anchorage
4 Barcelona
5 Barquisimeto
6 Bergamo

Northwind 00:00:00 69 rows

--T004_01_03

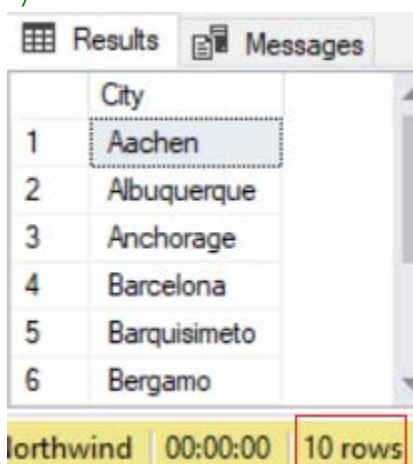
```
USE Northwind;
GO -- Run the previous command and begins new batch
SELECT DISTINCT TOP 1
    [Customers].[City]
FROM    [dbo].[Customers];
/*
1.
DISTINCT filter all the repeated [Customers].[City]
2.
TOP 1 only get the first one record.
*/
```



	City
1	Aachen

Northwind | 00:00:00 | 1 rows

```
-----
--T004_01_02_04
USE Northwind;
GO -- Run the previous command and begins new batch
SELECT DISTINCT TOP 10
    [Customers].[City]
FROM    [dbo].[Customers];
/*
1.
DISTINCT filter all the repeated [Customers].[City]
2.
TOP 10 only get the first ten records.
*/
```



	City
1	Aachen
2	Albuquerque
3	Anchorage
4	Barcelona
5	Barquisimeto
6	Bergamo

Northwind | 00:00:00 | 10 rows

```
-----
--T004_01_03
--SQL Wildcard Operators
--Select (DISTINCT/TOP n)..
--WHERE...(AND/OR)...
--Reference: https://www.tutorialspoint.com/sql/sql-wildcards.htm
-----
--T004_01_03_01
USE Northwind;
```

GO -- Run the previous command and begins new batch

```
SELECT *
FROM [dbo].[Customers]
WHERE [dbo].[Customers].[City] LIKE 'Lo%';
/*
```

Finds the customers whose City name starts with 'Lo'.

*/

Results		Messages										
	CustomerID	Company/Name	ContactName	ContactTitle	Address	City	Region	PostalCode	Country	Phone	Fax	
1	AROUT	Around the Horn	Thomas Hardy	Sales Representative	120 Hanover Sq.	London	NULL	WA1 1DP	UK	(171) 555-7788	(171) 555-6750	
2	BSBEV	B's Beverages	Victoria Ashworth	Sales Representative	Fauntleroy Circus	London	NULL	EC2 5NT	UK	(171) 555-1212	NULL	
3	CONSH	Consolidated Holdings	Elizabeth Brown	Sales Representative	Berkeley Gardens 12 Brewery	London	NULL	WX1 6LT	UK	(171) 555-2282	(171) 555-9199	
4	EASTC	Eastern Connection	Ann Devon	Sales Agent	35 King George	London	NULL	WX3 6FW	UK	(171) 555-0297	(171) 555-3373	
5	NORTS	North/South	Simon Crowther	Sales Associate	South House 300 Queensbridge	London	NULL	SW7 1RZ	UK	(171) 555-7733	(171) 555-2530	
6	SEVES	Seven Seas Imports	Hari Kumar	Sales Manager	90 Wadhurst Rd.	London	NULL	OX15 4NB	UK	(171) 555-1717	(171) 555-5646	
Query executed successfully.							N550JKL\SQL2016 (13.0 SP1)		N550JKL\jpmpl (57)		Northwind 00:00:00	6 rows

--T004_01_03_02

USE Northwind;

GO -- Run the previous command and begins new batch

```
SELECT *
FROM [dbo].[Customers]
WHERE [dbo].[Customers].[City] LIKE '%on%';
/*
```

Finds the customers whose City name has 'on' in any position.

*/

Results		Messages									
	CustomerID	CompanyName	ContactName	ContactTitle	Address	City	Region	PostalCode	Country	Phone	Fax
1	AROUT	Around the Horn	Thomas Hardy	Sales Representative	120 Hanover Sq.	London	NULL	WA1 1DP	UK	(171) 555-7788	(171) 555-6750
2	BSBEV	B's Beverages	Victoria Ashworth	Sales Representative	Fauntleroy Circus	London	NULL	EC2 5NT	UK	(171) 555-1212	NULL
3	CONSH	Consolidated Holdings	Elizabeth Brown	Sales Representative	Berkeley Gardens 12 Brewery	London	NULL	WX1 6LT	UK	(171) 555-2282	(171) 555-9199
4	EASTC	Eastern Connection	Ann Devon	Sales Agent	35 King George	London	NULL	WX3 6FW	UK	(171) 555-0297	(171) 555-3373
5	GALED	Galería del gastrónomo	Eduardo Saavedra	Marketing Manager	Rambla de Cataluña, 23	Barcelona	NULL	08022	Spain	(93) 203 4560	(93) 203 4561
6	MEREP	Mère Pailleurde	Jean Fresnière	Marketing Assistant	43 rue St. Laurent	Montréal	Québec	H1J 1C3	Canada	(514) 555-8054	(514) 555-8055
7	NORTS	North/South	Simon Crowther	Sales Associate	South House 300 Queensbridge	London	NULL	SW7 1RZ	UK	(171) 555-7733	(171) 555-2530
8	SEVES	Seven Seas Imports	Hari Kumar	Sales Manager	90 Wadhurst Rd.	London	NULL	OX15 4NB	UK	(171) 555-1717	(171) 555-5646
9	VICTE	Victualles en stock	Mary Saveley	Sales Agent	2, rue du Commerce	Lyon	NULL	69004	France	78.32.54.86	78.32.54.87

Query executed successfully.

N550JKL\SQL2016 (13.0 SP1)

N550JKL\jpmpl (57)

Northwind

00:00:00

9 rows

--T004_01_03_03

USE Northwind;

GO -- Run the previous command and begins new batch

```
SELECT *
FROM [dbo].[Customers]
WHERE [dbo].[Customers].[City] LIKE 'L_%_%';
/*
```

Finds the customers whose City name starts with 'L' and are at least 3 characters in length.

*/

Results		Messages									
	CustomerID	CompanyName	ContactName	ContactTitle	Address	City	Region	PostalCode	Country	Phone	Fax
1	AROUT	Around the Horn	Thomas Hardy	Sales Representative	120 Hanover Sq.	London	NULL	WA1 1DP	UK	(171) 555-7788	(171) 555-6750
2	BERGS	Berglunds snabbköp	Christina Berglund	Order Administrator	Berguvsvägen 8	Luleå	NULL	S-958 22	Sweden	0921-12 34 65	0921-12 34 67
3	BSBEV	B's Beverages	Victoria Ashworth	Sales Representative	Fauntleroy Circus	London	NULL	EC2 5NT	UK	(171) 555-1212	NULL
4	CONSH	Consolidated Holdings	Elizabeth Brown	Sales Representative	Berkeley Gardens 12 Brewery	London	NULL	WX1 6LT	UK	(171) 555-2282	(171) 555-9199
5	EASTC	Eastern Connection	Ann Devon	Sales Agent	35 King George	London	NULL	WX3 6FW	UK	(171) 555-0297	(171) 555-3373
6	FOLIG	Foies gourmandes	Martine Rancé	Assistant Sales Agent	184, chaussée de Tournai	Lille	NULL	59000	France	20 16 10 16	20 16 10 17
7	FURIB	Furia Bacalhau e Frutos do Mar	Lino Rodriguez	Sales Manager	Jardim das rosas n. 32	Lisboa	NULL	1675	Portugal	(1) 354-2534	(1) 354-2535
8	MORGK	Morgenstern Gesundkost	Alexander Feuer	Marketing Assistant	Heenstr. 22	Leipzig	NULL	04179	Germany	0342-023176	NULL
9	NORTS	North/South	Simon Crowther	Sales Associate	South House 300 Queensbridge	London	NULL	SW7 1RZ	UK	(171) 555-7733	(171) 555-2530
10	PRINI	Princesa Isabel Vinhos	Isabel de Castro	Sales Representative	Estrada da saúde n. 58	Lisboa	NULL	1756	Portugal	(1) 356-5634	NULL
11	SEVES	Seven Seas Imports	Hari Kumar	Sales Manager	90 Wadhurst Rd.	London	NULL	OX15 4NB	UK	(171) 555-1717	(171) 555-5646
12	SPLIR	Split Rail Beer & Ale	Art Braunschwe...	Sales Manager	P.O. Box 555	Lander	WY	82520	USA	(307) 555-4680	(307) 555-6525
13	VICTE	Victualles en stock	Mary Saveley	Sales Agent	2, rue du Commerce	Lyon	NULL	69004	France	78.32.54.86	78.32.54.87

Query executed successfully. N550JKL\SQL2016 (13.0 SP1) N550JKL\jpmpl (57) Northwind 00:00:00 13 rows

--T004_01_03_04

USE Northwind;

GO -- Run the previous command and begins new batch

```
SELECT *
```

```
FROM [dbo].[Customers]
WHERE [dbo].[Customers].[City] LIKE '_o%n';
```

```
/*
Finds the customers whoes City name have 'o' in the second position and end with a 'n'.
*/
```

Results Messages											
	CustomerID	CompanyName	ContactName	ContactTitle	Address	City	Region	PostalCode	Country	Phone	Fax
1	AROUT	Around the Horn	Thomas Hardy	Sales Representative	120 Hanover Sq.	London	NULL	WA1 1DP	UK	(171) 555-7788	(171) 555-6750
2	BSBEV	B's Beverages	Victoria Ashworth	Sales Representative	Fauntleroy Circus	London	NULL	EC2 5NT	UK	(171) 555-1212	NULL
3	CONSH	Consolidated Holdings	Elizabeth Brown	Sales Representative	Berkeley Gardens 12 Brewery	London	NULL	WX1 6LT	UK	(171) 555-2282	(171) 555-9199
4	EASTC	Eastern Connection	Ann Devon	Sales Agent	35 King George	London	NULL	WX3 6FW	UK	(171) 555-0297	(171) 555-3373
5	NORTS	North/South	Simon Crowther	Sales Associate	South House 300 Queensbridge	London	NULL	SW7 1RZ	UK	(171) 555-7733	(171) 555-2530
6	SEVES	Seven Seas Imports	Hari Kumar	Sales Manager	90 Wadhurst Rd.	London	NULL	OX15 4NB	UK	(171) 555-1717	(171) 555-5646
7	SIMOB	Simons bistro	Jytte Petersen	Owner	Vinbæltet 34	Kobenhavn	NULL	1734	Denmark	31 12 34 56	31 13 35 57
Query executed successfully.							N550JKL/SQL2016 (13.0 SP1) N550JKL/upmpl (57) Northwind 00:00:00 7 rows				

Query executed successfully. N550IKL\SQL2016 (13.0 SP1) N550IKL\pmp1 (57) Northwind 00:00:00 7 rows

```
--T004_01_03_05
```

```
USE Northwind;
```

```
GO -- Run the prvious command and begins new batch
```

```
SELECT *
FROM [dbo].[Customers]
WHERE [dbo].[Customers].[ContactName] LIKE '[ABC]%';
```

```
/*
Finds the customers whoes [ContactName] start with 'A' or 'B' or 'C'.
*/
```

Results Messages

	CustomerID	CompanyName	ContactName	ContactTitle	Address	City	Region	PostalCode	Country	Phone	Fax
1	ANATR	Ana Trujillo Emparedados y helados	Ana Trujillo	Owner	Avda. de la Constitución 2222	México D.F.	NULL	05021	Mexico	(5) 555-4729	(5) 555-3745
2	ANTON	Antonio Moreno Taqueria	Antonio Moreno	Owner	Mataderos 2312	México D.F.	NULL	05023	Mexico	(5) 555-3932	NULL
3	BERGS	Berglunds snabbköp	Christina Berglund	Order Administrator	Berguvsvägen 8	Luleå	NULL	S-958 22	Sweden	0921-12 34 65	0921-12 34 67
4	EASTC	Eastern Connection	Ann Devon	Sales Agent	35 King George	London	NULL	WX3 6FW	UK	(171) 555-0297	(171) 555-3373
5	FAMIA	Familia Arribas	Aria Cruz	Marketing Assistant	Rua Orós, 52	Sao Paulo	SP	05442-030	Brazil	(11) 555-9857	NULL
6	FRANR	France restauration	Carine Schmitt	Marketing Manager	54, rue Royale	Nantes	NULL	44000	France	40 32 21 21	40 32 21 20
7	GOURL	Gourmet Lanchonetes	André Fonseca	Sales Associate	Av. Brasil, 442	Campos	SP	04876-706	Brazil	(11) 555-9482	NULL

Query executed successfully.

N550JKL(SQL)2016 (13.0 SP) N550JKL(vpmpl) (57) Northwind 00:00:00 16 rows

Query executed successfully. N550IKL\SQL2016 (13.0 SP1) N550IKL\pmp1 (57) Northwind 00:00:00 16 rows

```
--T004_01_03_06
```

```
USE Northwind;
```

```
GO -- Run the prvious command and begins new batch
```

```
SELECT *
FROM [dbo].[Customers]
WHERE [dbo].[Customers].[ContactName] LIKE '[^ABC]%';
```

```
/*
Finds the customers whoes [ContactName] does NOT start with 'A' or 'B' or 'C'.
*/
```

Results Messages											
	CustomerID	CompanyName	ContactName	ContactTitle	Address	City	Region	PostalCode	Country	Phone	Fax
1	ALFKI	Alfreds Futterkiste	Maria Anders	Sales Representative	Obere Str. 57	Berlin	NULL	12209	Germany	030-0074321	030-0076545
2	AROUT	Around the Horn	Thomas Hardy	Sales Representative	120 Hanover Sq.	London	NULL	WA1 1DP	UK	(171) 555-7788	(171) 555-6750
3	BLAUS	Blauer See Delikatessen	Hanna Moos	Sales Representative	Forsterstr. 57	Mannheim	NULL	68306	Germany	0621-08460	0621-08924
4	BLOPP	Blondiesdel père et fils	Frédérique Citeaux	Marketing Manager	24, place Kléber	Strasbourg	NULL	67000	France	88.60.15.31	88.60.15.32
5	BOLID	Bólido Comidas preparadas	Martín Sommer	Owner	C/ Araquil, 67	Madrid	NULL	28023	Spain	(91) 555 22 82	(91) 555 91 99
6	BONAP	Bon app'	Laurence Leblanc	Owner	12, rue des Bouchers	Marseille	NULL	13008	France	91.24.45.40	91.24.45.41
7	BOTTM	Bottom-Dollar Markets	Elizabeth Lincoln	Accounting Manager	23 Tsawassen Blvd.	Tsawassan	BC	T2F 8M4	Canada	(604) 555-4729	(604) 555-3745
✔ Query executed successfully.											
N550JKL.SQ1.16.20 (13.0 SP1) N550JKL.vpml (57) Northwind 00:00:00 75 rows											

Query executed successfully. N550IKL\SQL2016 (13.0 SP1) N550IKL\pmp1 (57) Northwind 00:00:00 75 rows

```
--T004_01_03_07
```

```
USE Northwind;
```

```
GO -- Run the prvious command and begins new batch
```

```
SELECT *
FROM [dbo].[Customers]
WHERE [dbo].[Customers].[ContactName] NOT LIKE '[ABC]%';
```

```
/*
Finds the customers whoes [ContactName] does NOT start with 'A' or 'B' or 'C'.
*/
```


Results Messages											
	CustomerID	Company Name	Contact Name	Contact Title	Address	City	Region	Postal Code	Country	Phone	Fax
1	ALFKI	Alfreds Futterkiste	Maria Anders	Sales Representative	Obere Str. 57	Berlin	NULL	12209	Germany	030-0074321	030-0076545
2	AROUT	Around the Horn	Thomas Hardy	Sales Representative	120 Hanover Sq.	London	NULL	WA1 1DP	UK	(171) 555-7788	(171) 555-6750
3	BLAUS	Blauer See Delikatessen	Hanna Moos	Sales Representative	Forststr. 57	Mannheim	NULL	68306	Germany	0621-08460	0621-08924
4	BLOMP	Blondel père et fils	Frédérique Citeaux	Marketing Manager	24, place Kléber	Strasbourg	NULL	67000	France	88.60.15.31	88.60.15.32
5	BOLID	Bólido Comidas preparadas	Martín Sommer	Owner	C/ Araquil, 67	Madrid	NULL	28023	Spain	(91) 555 22 82	(91) 555 91 99
6	BONAP	Bon app'	Laurence Leblanc	Owner	12, rue des Bouchers	Marseille	NULL	13008	France	91.24.45.40	91.24.45.41
7	BOTTM	Bottom-Dollar Markets	Elizabeth Lincoln	Accounting Manager	23 Tsawassen Blvd.	Tsawassen	BC	T2F 8M4	Canada	(604) 555-4729	(604) 555-3745
✔ Query executed successfully.											
N550KJLSQL2616 (13.0 SP1) N550KJL5pml4 (57) Northwind 00:00:07 75 rows											

Query executed successfully.

N550IKL\SQL2016 (13.0 SP1) N550IKL\lpmpl (57) Northwind 00:00:00 75 rows

--T004_01_03_08

USE Northwind;

GO -- Run the previous command and begins new batch

SELECT *

FROM [dbo].[Customers]

WHERE [dbo].[Customers].[PostalCode] LIKE '[0-9][0-9][0-9][0-9]';

/*

Finds the customers whose [PostalCode] have only 4 digits which are between 0 and 9.

*/

CustomerID	CompanyName	ContactName	ContactTitle	Address	City	Region	PostalCode	Country	Phone	Fax
1	CACTU	Cactus Comidas para llevar	Patricio Simpson	Sales Agent	Centro 333	Buenos Aires	1010	Argentina	(1) 135-5555	(1) 135-
2	CHOPS	Chop-suey Chinese	Yang Wang	Owner	Hauptstr. 29	Bern	3012	Switzerland	0452-076545	NULL
3	ERNSH	Ernst Handel	Roland Mendel	Sales Manager	Kirchgasse 6	Graz	8010	Austria	7675-3425	7675-3-
4	FURIB	Furia Bacalhau e Frutos do Mar	Lino Rodriguez	Sales Manager	Jardim das rosas n. 32	Lisboa	1675	Portugal	(1) 354-2534	(1) 354-
5	GROSR	GROSELLA-Restaurante	Manuel Pereira	Owner	5ª Ave. Los Palos Grandes	Caracas	1081	Venezuela	(2) 283-2951	(2) 283-
6	HILAA	HILARION-Abastos	Carlos Hernández	Sales Representative	Carrera 22 con Ave. Carlos Soublette #8-35	San Cristóbal	5022	Venezuela	(5) 555-1340	(5) 555-

Query executed successfully.

N550IKL\SQL2016 (13.0 SP1) N550IKL\lpmpl (57) Northwind 00:00:00 16 rows

--T004_01_04

--Select (DISTINCT/TOP n)..

--WHERE...(AND/OR)...

--ORDER BY ...(ASC/DESC)

--T004_01_04_01

USE Northwind;

GO -- Run the previous command and begins new batch

SELECT *

FROM [dbo].[Customers]

WHERE [dbo].[Customers].[ContactName] LIKE '[ABC]%'

ORDER BY [dbo].[Customers].[ContactName], [dbo].[Customers].[Country];

/*

Finds the customers whose [ContactName] start with 'A' or 'B' or 'C'.

Order by the [ContactName] firstly, then order by the [Country] secondly.

*/

CustomerID	CompanyName	ContactName	ContactTitle	Address	City	Region	PostalCode	Country	Phone	Fax
1	ROMEY	Romero y tomillo	Alejandra Camino	Accounting Manager	Gran Vía, 1	Madrid	28001	Spain	(91) 745 6200	(91) 745 6210
2	MORGK	Morgenstern Gesundkost	Alexander Feuer	Marketing Assistant	Heenstr. 22	Leipzig	04179	Germany	0342-023176	NULL
3	ANATR	Ana Trujillo Emparedados y helados	Ana Trujillo	Owner	Avda. de la Constitución 2222	México D.F.	05021	Mexico	(5) 555-4729	(5) 555-3745
4	TRADH	Tradição Hojmemercados	Anabela Domingues	Sales Representative	Av. Inês de Castro, 414	Sao Paulo	05634-030	Brazil	(11) 555-2167	(11) 555-2168
5	GOURL	Gourmet Lanchonetes	André Fonseca	Sales Associate	Av. Brasil, 442	Campinas	04876-786	Brazil	(11) 555-9482	NULL
6	EASTC	Eastern Connection	Ann Devon	Sales Agent	35 King George	London	WX3 6FW	UK	(171) 555-0297	(171) 555-3373

Query executed successfully.

N550IKL\SQL2016 (13.0 SP1) N550IKL\lpmpl (57) Northwind 00:00:00 16 rows

--T004_01_04_02

USE Northwind;

GO -- Run the previous command and begins new batch

SELECT *

FROM [dbo].[Customers]

WHERE [dbo].[Customers].[ContactName] LIKE '[ABC]%'

ORDER BY [dbo].[Customers].[ContactName] ASC, [dbo].[Customers].[Country] DESC;

/*

Finds the customers whose [ContactName] start with 'A' or 'B' or 'C'.

Order by the [ContactName] ascending firstly, then order by the [Country] descending secondly.

*/

Results		Messages										
	CustomerID	CompanyName	ContactName	ContactTitle	Address	City	Region	PostalCode	Country	Phone	Fax	
1		ROMEY	Romero y tomillo	Alejandra Camino	Accounting Manager	Gran Vía, 1	Madrid	NULL	28001	Spain	(91) 745 6200	(91) 745 6210
2		MORGK	Morgenstern Gesundkost	Alexander Feuer	Marketing Assistant	Heenstr. 22	Leipzig	NULL	04179	Germany	0342-023176	NULL
3		ANATR	Ana Trujillo Emparedados y helados	Ana Trujillo	Owner	Avda. de la Constitución 2222	Mexico D.F.	NULL	05021	Mexico	(5) 555-4729	(5) 555-3745
4		TRADH	Tradição Hipermarcados	Anabela Domingues	Sales Representative	Av. Inês de Castro, 414	Sao Paulo	SP	05634-030	Brazil	(11) 555-2167	(11) 555-2168
5		GOURL	Gourmet Lanchonetes	André Fonseca	Sales Associate	Av. Brasil, 442	Campinas	SP	04876-796	Brazil	(11) 555-9482	NULL
6		EASTC	Eastern Connection	Ann Devon	Sales Agent	35 King George	London	NULL	WX3 6FW	UK	(171) 555-0297	(171) 555-3373

Query executed successfully.

N550JKL\SQL2016 (13.0 SP1)N550JKL\lpmp1 (57)Northwind00:00:0016 rows

3. Group By

```

--=====
--T004_02_GroupBy
--=====
/*
What to learn
- In order to use aggregate, we need Group By
- aggregate include Count(), Sum(), avg(), Min(), Max().
*/
--=====
--T004_02_01
--Aggregate : Min(), Max(), avg()
USE Northwind;
GO -- Run the prvious command and begins new batch
SELECT ProductID ,
        ( ( [UnitPrice] - [Discount] ) * [Quantity] ) AS TotalSales
FROM    [Northwind].[dbo].[Order Details];
SELECT [ProductID] ,
        [ProductName] ,
        [SupplierID] ,
        [CategoryID] ,
        [QuantityPerUnit] ,
        [UnitPrice] ,
        [UnitsInStock] ,
        [UnitsOnOrder] ,
        [ReorderLevel] ,
        [Discontinued]
FROM    [Northwind].[dbo].[Products];
SELECT MIN([UnitPrice])
FROM    [Northwind].[dbo].[Products];
SELECT MAX([UnitPrice])
FROM    [Northwind].[dbo].[Products];
SELECT AVG([UnitPrice])
FROM    [Northwind].[dbo].[Products];
SELECT MIN(( [UnitPrice] - [Discount] ) * [Quantity]) AS TotalSalesOfTheProductOfTheOrder
FROM    [Northwind].[dbo].[Order Details];
SELECT MAX(( [UnitPrice] - [Discount] ) * [Quantity]) AS TotalSalesOfTheProductOfTheOrder
FROM    [Northwind].[dbo].[Order Details];
GO -- Run the prvious command and begins new batch

```

```

/*
1.
1.1.
-- SELECT MIN([UnitPrice])
The cheapest product unit price.
1.2.
-- SELECT MAX([UnitPrice])
The most expensive product unit price.
1.3.
-- SELECT AVG([UnitPrice])
The average product unit price.
2.
--SELECT MIN(( [UnitPrice] - [Discount] ) * [Quantity]) AS TotalSalesOfTheProductOfTheOrder
--SELECT MAX(( [UnitPrice] - [Discount] ) * [Quantity]) AS TotalSalesOfTheProductOfTheOrder
2.1.
The total sales of each Product of each OrderID is
( [UnitPrice] - [Discount] ) * [Quantity]
2.2.
--SELECT MIN(( [UnitPrice] - [Discount] ) * [Quantity]) AS TotalSalesOfTheProductOfTheOrder
The cheapest total sales of each Product of each OrderID
2.3.
--SELECT MAX(( [UnitPrice] - [Discount] ) * [Quantity]) AS TotalSalesOfTheProductOfTheOrder
The most expensive total sales of each Product of each OrderID
*/

```

	ProductID	TotalSales
1	11	168
2	42	98
3	72	174
4	14	167.4

(67) Northwind 00:00:00 2155 rows

Results

Messages

	ProductID	ProductName	SupplierID	CategoryID	QuantityPerUnit	UnitPrice	UnitsInStock	UnitsOnOrder	ReorderLevel	Discontinued
1	1	Chai	1	1	10 boxes x 20 bags	18.00	39	0	10	0
2	2	Chang	1	1	24 - 12 oz bottles	19.00	17	40	25	0
3	3	Aniseed Syrup	1	2	12 - 550 ml bottles	10.00	13	70	25	0
4	4	Chef Anton's Cajun Seasoning	2	2	48 - 6 oz jars	22.00	53	0	0	0
(No column name)										
1	2.50									
(No column name)										
1	263.50									
(No column name)										
1	28.8663									
TotalSalesOfTheProductOfTheOrder										
1	4.8									
TotalSalesOfTheProductOfTheOrder										
1	15810									

Query executed successfully. N550JKL\SQL2016 (13.0 SP1) N550JKL\lpmp1 (55) Northwind 00:00:00 82 rows

```

-----
--T004_02_02
--Aggregate : Min(SUM()) cause ERROR
USE Northwind;
GO -- Run the prvious command and begins new batch
SELECT MIN(SUM(( [UnitPrice] - [Discount] ) * [Quantity])) AS TotalSalesOfTheOrder
FROM [Northwind].[dbo].[Order Details];
GO -- Run the prvious command and begins new batch
/*
1.
-- SELECT MIN(SUM(( [UnitPrice] - [Discount] ) * [Quantity])) AS TotalSalesOfTheOrder
1.1.
The total sales of each Product of each OrderID is
( [UnitPrice] - [Discount] ) * [Quantity]
1.2.
The total sales of each OrderID is
SUM(( [UnitPrice] - [Discount] ) * [Quantity])
1.3.

```



```
--SELECT MIN(SUM(( [UnitPrice] - [Discount] ) * [Quantity])) AS TotalSalesOfTheOrder
The cheapest sales of the OrderID.
```

2.

Output ERROR message

```
--Msg 130, Level 15, State 1, Line 60
```

```
--Cannot perform an aggregate function on an expression containing an aggregate or a subquery.
```

```
*/
```

```
=====
```

```
--T004_02_03
```

```
--Aggregate : SUM(), COUNT()
```

```
USE Northwind;
```

```
GO -- Run the previous command and begins new batch
```

```
SELECT [OrderID] ,
       [ProductID] ,
       [UnitPrice] ,
       [Quantity] ,
       [Discount]
FROM [Northwind].[dbo].[Order Details];
SELECT [OrderID] ,
       SUM(( [UnitPrice] - [Discount] ) * [Quantity]) AS TotalSalesOfTheOrder ,
       COUNT([ProductID]) AS NumberOfProductIDInTheSale
FROM [Northwind].[dbo].[Order Details]
GROUP BY [OrderID]
ORDER BY [OrderID];
```

```
GO -- Run the previous command and begins new batch
```

```
/*
```

1.

```
-- COUNT([ProductID]) AS NumberOfProductIDInTheSale
```

Each OrderID has several ProductID.

Thus, number of productID in the sales is COUNT([ProductID])

2.

```
-- SUM(( [UnitPrice] - [Discount] ) * [Quantity]) AS TotalSalesOfTheOrder
```

2.1.

The total sales of each Product of each OrderID is

```
( [UnitPrice] - [Discount] ) * [Quantity]
```

2.2.

The total sales of each OrderID is

```
SUM(( [UnitPrice] - [Discount] ) * [Quantity])
```

3.

In order to use aggregate, we need Group By

aggregate include Count(), Sum(), avg(), Min(), Max().

3.1.

[OrderID] is the only thing without aggregate in the SELECT clause.

Thus, we need to GROUP BY [OrderID]

```
*/
```

	OrderID	ProductID	UnitPrice	Quantity	Discount
1	10248	11	14.00	12	0
2	10248	42	9.80	10	0
3	10248	72	34.80	5	0
4	10249	14	18.60	9	0
5	10249	51	42.40	40	0

	OrderID	TotalSalesOfTheOrder	NumberOfProductIDInTheSale
1	10248	440	3
2	10249	1863.40000915527	2
3	10250	1805.5	3
4	10251	669.75	3
5	10252	3726.75	3
6	10253	1444.79998779297	3
7	10254	619.80001449585	3
8	10255	2490.5	4
9	10256	517.799995422363	2
10	10257	1119.89993286133	3

```

=====
--T004_02_04
--Aggregate : SUM()
--Different between WHERE and HAVING
--Ch11_04_01
USE Northwind;
GO -- Run the previous command and begins new batch
SELECT [OrderID] ,
       [ProductID] ,
       SUM(( [UnitPrice] - [Discount] ) * [Quantity]) AS TotalSalesByProduct
FROM   [Northwind].[dbo].[Order Details]
WHERE  OrderID BETWEEN 10249 AND 10328
GROUP BY [OrderID] ,
         [ProductID]
--HAVING OrderID BETWEEN 10249 AND 10328
ORDER BY [OrderID];
--Ch11_04_02
SELECT [OrderID] ,
       [ProductID] ,
       SUM(( [UnitPrice] - [Discount] ) * [Quantity]) AS TotalSalesByProduct
FROM   [Northwind].[dbo].[Order Details]
--WHERE OrderID BETWEEN 10249 AND 10328
GROUP BY [OrderID] ,
         [ProductID]
HAVING OrderID BETWEEN 10249 AND 10328
ORDER BY [OrderID];
GO -- Run the previous command and begins new batch
/*
1.
1.1.
WHERE filters rows before aggregation (GROUP BY)
and can be used with - Select, Insert, Update statements.
1.2.
HAVING filters groups after the aggregations (GROUP BY)

```

and can only be used with the Select statement

2.

```
--WHERE OrderID BETWEEN 10249 AND 10328
```

and

```
--HAVING OrderID BETWEEN 10249 AND 10328
```

both will get the same result.

However, try to eliminate rows that you don't want as early as possible.

Thus, in this case, using WHERE is better than using HAVING.

```
*/
```

Results		Messages	
	OrderID	ProductID	TotalSalesByProduct
1	10249	14	167.400009155273
2	10249	51	1696
3	10250	41	77
4	10250	51	1478.75
5	10250	65	249.75
6	10251	22	100.5
7	10251	57	233.25

	OrderID	ProductID	TotalSalesByProduct
1	10249	14	167.400009155273
2	10249	51	1696
3	10250	41	77
4	10250	51	1478.75
5	10250	65	249.75
6	10251	22	100.5
7	10251	57	233.25

Query executed successfully.

```
-----
```

```
--T004_02_05
```

```
--Aggregate : SUM()
```

```
--Aggregation (GROUP BY) can not be used in the WHERE clause.
```

```
--but it is ok in HAVING clause.
```

```
USE Northwind;
```

```
GO -- Run the previous command and begins new batch
```

```
SELECT [OrderID] ,  
       [ProductID] ,  
       SUM(( [UnitPrice] - [Discount] ) * [Quantity]) AS TotalSalesByProduct  
FROM   [Northwind].[dbo].[Order Details]  
--WHERE SUM(( [UnitPrice] - [Discount] ) * [Quantity]) > 500 --Error  
GROUP BY [OrderID] ,  
         [ProductID]  
HAVING SUM(( [UnitPrice] - [Discount] ) * [Quantity]) > 500  
ORDER BY [OrderID];
```

```
GO -- Run the previous command and begins new batch
```

```
/*
```

```
1.
```

```
1.1.
```

WHERE filters rows before aggregation (GROUP BY)

and can be used with - Select, Insert, Update statements.

```
1.2.
```

HAVING filters groups after the aggregations (GROUP BY)

and can only be used with the Select statement

2.

aggregation (GROUP BY) can be used in Having clause.

However, aggregation (GROUP BY) can not be used in the WHERE clause, unless aggregation (GROUP BY) is in a sub query with a HAVING clause.

*/

Results		Messages	
	OrderID	ProductID	TotalSalesByProduct
1	10249	51	1696
2	10250	51	1478.75
3	10252	20	2590
4	10252	60	1088
5	10253	39	604.799987792969
6	10253	49	640
7	10255	59	1320
8	10257	27	877.499938964844
9	10258	2	750
10	10258	5	1092
11	10260	57	780
12	10260	62	587.25
13	10263	16	819
14	10263	30	1227
15	10264	2	532
16	10265	17	936

1550JKL\lpmpl (55) | Northwind | 00:00:00 | 799 rows

```
--=====
--T004_02_06
--Aggregate : SUM()
--WHERE with HAVING
USE Northwind;
GO -- Run the prvious command and begins new batch
SELECT [OrderID] ,
       [ProductID] ,
       SUM(( [UnitPrice] - [Discount] ) * [Quantity]) AS TotalSalesByProduct
FROM   [Northwind].[dbo].[Order Details]
WHERE  OrderID BETWEEN 10249 AND 10328
GROUP BY [OrderID] ,
         [ProductID]
HAVING SUM(( [UnitPrice] - [Discount] ) * [Quantity]) > 500
ORDER BY [OrderID];
GO -- Run the prvious command and begins new batch
/*
1.
1.1.
WHERE filters rows before aggregation (GROUP BY)
and can be used with - Select, Insert, Update statements.
1.2.
HAVING filters groups after the aggregations (GROUP BY)
and can only be used with the Select statement
2.
aggregation (GROUP BY) can be used in Having clause.
However, aggregation (GROUP BY) can not be used in the WHERE clause,
```

unless aggregation (GROUP BY) is in a sub query with a HAVING clause.
 3.
 In this case,
 it will execute WHERE clause,
 then execute aggregation (GROUP BY, SUM()),
 then execute HAVING clause.
 */

Results		Messages	
	OrderID	ProductID	TotalSalesByProduct
1	10249	51	1696
2	10250	51	1478.75
3	10252	20	2590
4	10252	60	1088
5	10253	39	604.799987792969
6	10253	49	640
7	10255	59	1320
8	10257	27	877.499938964844
9	10258	2	750
10	10258	5	1092
11	10260	57	780
12	10260	62	587.25
13	10263	16	819
14	10263	30	1227
15	10264	2	532
16	10265	17	936

N550JKL\lpmpl (55) | Northwind | 00:00:00 | 69 rows

=====

4. Where V.S. Having

```

-----
--T004_03_Where_Having
-----
/*
1.
WHERE V.S. Having
1.1.
WHERE and HAVING can be used together.
Aggregates includes Count, Sum, Avg, Min and Max.
Aggregates must be happened after Group By,
because Aggregates need Group By to perform.
1.2.
WHERE operator filters the rows before Group by.
Thus, WHERE operator CAN NOT be used to filter Aggregates.
HAVING operator filters the Group after Group by.

```

Thus, HAVING operator CAN filter Aggregates.

1.3.

HAVING is slower than WHERE.

Try to use WHERE to replace HAVING if possible.

*/

=====

--T004_03_01

--Create Sample Data

--If Table exists then DROP it

```
IF ( EXISTS ( SELECT      *
                FROM        INFORMATION_SCHEMA.TABLES
                WHERE        TABLE_NAME = 'StoreSales' ) )
    BEGIN
        TRUNCATE TABLE dbo.StoreSales;
        DROP TABLE StoreSales;
    END;
```

GO -- Run the previous command and begins new batch

CREATE TABLE StoreSales

```
(
    Id INT IDENTITY(1, 1)
        PRIMARY KEY ,
    StoreName NVARCHAR(100) ,
    Sales MONEY
);
```

GO -- Run the previous command and begins new batch

```
INSERT INTO StoreSales
VALUES ( 'Store01', 350 );
INSERT INTO StoreSales
VALUES ( 'Store02', 690 );
INSERT INTO StoreSales
VALUES ( 'Store01', 700 );
INSERT INTO StoreSales
VALUES ( 'Store02', 150 );
INSERT INTO StoreSales
VALUES ( 'Store05', 880 );
INSERT INTO StoreSales
VALUES ( 'Store02', 860 );
INSERT INTO StoreSales
VALUES ( 'Store03', 210 );
INSERT INTO StoreSales
VALUES ( 'Store04', 750 );
```

GO -- Run the previous command and begins new batch

```
SELECT *
FROM    StoreSales;
```

GO -- Run the previous command and begins new batch

	Id	StoreName	Sales
1	1	Store01	350.00
2	2	Store02	690.00
3	3	Store01	700.00
4	4	Store02	150.00
5	5	Store05	880.00
6	6	Store02	860.00
7	7	Store03	210.00
8	8	Store04	750.00

```

=====
--T004_03_02
SELECT  StoreName ,
        SUM(Sales) AS Total
FROM    StoreSales
GROUP BY StoreName;
GO -- Run the previous command and begins new batch
/*
GROUP BY StoreName will get the total sales of each store,
*/

```

	StoreName	Total
1	Store01	1050.00
2	Store02	1700.00
3	Store03	210.00
4	Store04	750.00
5	Store05	880.00

```

=====
--T004_03_03
SELECT  StoreName ,
        SUM(Sales) AS Total
FROM    StoreSales
GROUP BY StoreName
HAVING  SUM(Sales) > 1100;
GO -- Run the previous command and begins new batch
/*
GROUP BY StoreName will get the total sales of each store,
HAVING operator filter the Group
which otal sales of each store is greater than 1100.
*/

```

	StoreName	Total
1	Store02	1700.00

```

=====
--T004_03_04
SELECT  StoreName ,
        SUM(Sales) AS Total
FROM    StoreSales
WHERE   StoreName IN ( 'Store01', 'Store02' )
GROUP BY StoreName;
GO -- Run the previous command and begins new batch
/*
WHERE operator filter the rows before Group by.
*/

```

```
--WHERE StoreName in ('Store01', 'Store02')
Only get the Store01 and Store02 rows.
Then GROUP BY StoreName will
get the total sales of Store01 and Store02
*/
```

	StoreName	Total
1	Store01	1050.00
2	Store02	1700.00

```
=====
--T004_03_05
```

```
SELECT StoreName ,
        SUM(Sales) AS Total
FROM StoreSales
GROUP BY StoreName
HAVING StoreName IN ( 'Store01', 'Store02' );
GO -- Run the previous command and begins new batch
/*
```

```
HAVING operator filter the Group after Group by.
--HAVING StoreName in ('Store01', 'Store02')
Only get the Store01 and Store02 Groups.
Then GROUP BY StoreName will
get the total sales of Store01 and Store02.
*/
```

	StoreName	Total
1	Store01	1050.00
2	Store02	1700.00

```
=====
--T004_03_06
```

```
--Clean up
--If Table exists then DROP it
IF ( EXISTS ( SELECT *
               FROM INFORMATION_SCHEMA.TABLES
               WHERE TABLE_NAME = 'StoreSales' ) )
BEGIN
    TRUNCATE TABLE dbo.StoreSales;
    DROP TABLE StoreSales;
END;
GO -- Run the previous command and begins new batch
```