



Module 17: Selection Commands – Filtering

This module focuses on using **SQL filtering techniques** to retrieve specific records using **IN**, **BETWEEN**, and **LIKE** with real examples from the **Sales** table.



Reference Table Used in This Module

The screenshot shows a SQL Server Management Studio interface. In the top tab bar, there are three tabs: 'SQLQuery10.sql - U...', 'SQLQuery9.sql - US...', and 'SQLQuery7.sql'. The main area contains a single line of SQL code: '1 | SELECT * FROM Sales'. Below the code is a results grid titled 'Results' showing 11 rows of data from the Sales table. The columns are: OrderDate, Region, Manager, SalesMan, Item, Units, Unit_price, and Sale_amt. The data includes various dates from January 2018 to June 2018, regions like East, Central, and West, managers like Martha, Hermann, Alexander, etc., items like Television, Home Theater, Cell Phone, and more, and unit counts ranging from 27 to 95.

	OrderDate	Region	Manager	SalesMan	Item	Units	Unit_price	Sale_amt
1	2018-01-06	East	Martha	Alexander	Television	95	1198	113810
2	2018-01-23	Central	Hermann	Shelli	Home Theater	50	500	25000
3	2018-02-09	Central	Hermann	Luis	Television	36	1198	43128
4	2018-02-26	Central	Timothy	David	Cell Phone	27	225	6075
5	2018-03-15	West	Timothy	Stephen	Television	56	1198	67088
6	2018-04-01	East	Martha	Alexander	Home Theater	60	500	30000
7	2018-04-18	Central	Martha	Steven	Television	75	1198	89850
8	2018-05-05	Central	Hermann	Luis	Television	90	1198	107820
9	2018-05-22	West	Douglas	Michael	Television	32	1198	38336
10	2018-06-08	East	Martha	Alexander	Home Theater	60	500	30000
11	2018-06-25	Central	Hermann	Sigal	Television	90	1198	107820

◆ 1. IN Operator

✓ Definition:

The **IN** operator allows you to filter rows where a column matches **any value in a specified list**.

✓ Syntax:

```
SELECT column1, column2  
FROM table_name  
WHERE column_name IN (value1, value2, ...);
```

📌 Example:

```
SELECT DISTINCT Manager, Salesman  
FROM Sales  
WHERE Region IN ('East', 'Central');
```

The screenshot shows a SQL query being run in SSMS. The query is:

```
3 | SELECT DISTINCT Manager, SalesMan from Sales where Region IN ('East', 'Central');
```

The results pane displays a table with two columns: Manager and SalesMan. The data is as follows:

Manager	SalesMan
Douglas	John
Douglas	Karen
Hermann	Luis
Hermann	Shelli
Hermann	Sigal
Martha	Alexander
Martha	Diana
Martha	Steven
Timothy	David

- ✓ Returns unique combinations of Manager and SalesMan where the Region is either East or Central.

◆ 2. BETWEEN Operator

✓ Definition:

BETWEEN filters rows where a value is within a **range (inclusive)**. You can use it on **numbers, dates, or text**.

✓ Syntax:

```
SELECT * FROM table_name  
WHERE column_name BETWEEN value1 AND value2;
```

📌 Examples:

a) Filter by Units:

```
SELECT * FROM Sales  
WHERE Units BETWEEN 50 AND 70;
```

```

10
11 SELECT * FROM SALES WHERE UNITS BETWEEN 50 AND 70; I
12
13
14
15
16
17
18
19

```

145 %

Results Messages

	OrderDate	Region	Manager	SalesMan	Item	Units	Unit_price	Sale_amt
1	2018-01-23	Central	Hermann	Shelli	Home Theater	50	500	25000
2	2018-03-15	West	Timothy	Stephen	Television	56	1198	67088
3	2018-04-01	East	Martha	Alexander	Home Theater	60	500	30000
4	2018-06-08	East	Martha	Alexander	Home Theater	60	500	30000
5	2018-10-22	East	Martha	Alexander	Cell Phone	64	225	14400
6	2018-12-12	Central	Douglas	John	Television	67	1198	80266

✓ Gets rows with unit values between 50 and 70.

b) Not Between:

```

SELECT * FROM Sales
WHERE Units NOT BETWEEN 50 AND 100;

```

```

13 SELECT * FROM SALES WHERE UNITS NOT BETWEEN 50 AND 100;
14
15
16
17
18
19

```

145 %

Results Messages

	OrderDate	Region	Manager	SalesMan	Item	Units	Unit_price	Sale_amt
1	2018-02-09	Central	Hermann	Luis	Television	36	1198	43128
2	2018-02-26	Central	Timothy	David	Cell Phone	27	225	6075
3	2018-05-22	West	Douglas	Michael	Television	32	1198	38336
4	2018-07-12	East	Martha	Diana	Home Theater	29	500	14500
5	2018-08-15	East	Martha	Alexander	Television	35	1198	41930
6	2018-09-01	Central	Douglas	John	Desk	2	125	250

✓ Gets rows where units are either < 50 or > 100.

c) Filter by Order Date:

```

SELECT * FROM Sales
WHERE OrderDate BETWEEN '2018-03-01' AND '2018-04-30';

```

The screenshot shows a SQL query window with the following code:

```

14
15 SELECT * FROM SALES WHERE ORDERDATE BETWEEN '2018-03-01' AND '2018-04-30';
16
17
18
19

```

The results pane displays a table with the following data:

	OrderDate	Region	Manager	SalesMan	Item	Units	Unit_price	Sale_amt
1	2018-03-15	West	Timothy	Stephen	Television	56	1198	67088
2	2018-04-01	East	Martha	Alexander	Home Theater	60	500	30000
3	2018-04-18	Central	Martha	Steven	Television	75	1198	89850

✓ Returns sales made between March and April 2018.

◆ 3. LIKE Operator (Pattern Matching)

✓ Definition:

`LIKE` is used to search for a **pattern** in a column (usually `VARCHAR` or `TEXT`).

✓ Syntax:

```
SELECT * FROM table_name
WHERE column_name LIKE 'pattern';
```

📌 Examples:

a) Starts with 'M':

```
SELECT DISTINCT * FROM Sales
WHERE SalesMan LIKE 'M%';
```

The screenshot shows a SQL query window with the following code:

```

20
21 SELECT * FROM SALES WHERE MANAGER LIKE 'M%';
22
23
24
25
26
27
28

```

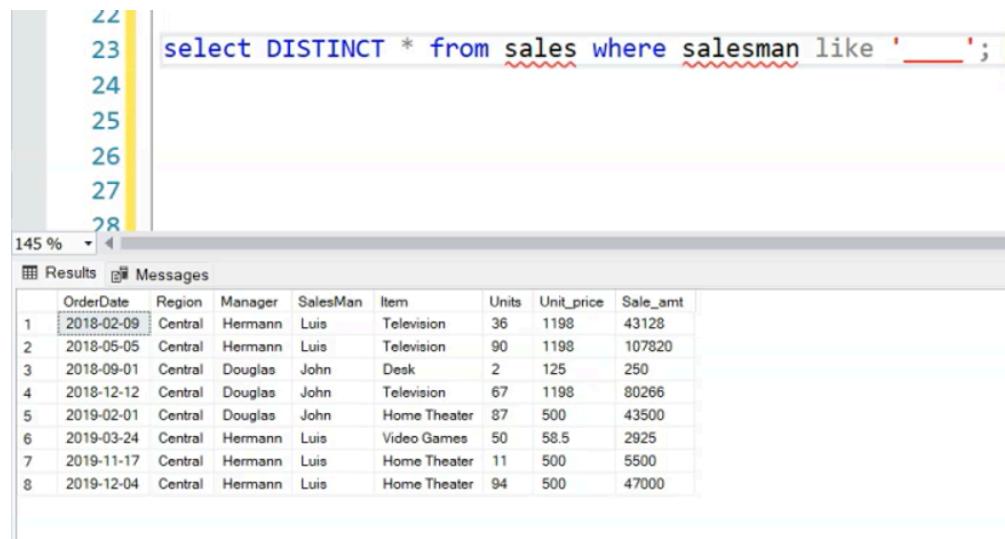
The results pane displays a table with the following data:

	OrderDate	Region	Manager	SalesMan	Item	Units	Unit_price	Sale_amt
1	2018-01-06	East	Martha	Alexander	Television	95	1198	113810
2	2018-04-01	East	Martha	Alexander	Home Theater	60	500	30000
3	2018-04-18	Central	Martha	Steven	Television	75	1198	89850
4	2018-06-08	East	Martha	Alexander	Home Theater	60	500	30000

✓ Matches names like **Michael, Martha, etc.**

b) Exactly 7-character names:

```
SELECT DISTINCT * FROM Sales  
WHERE SalesMan LIKE '_____';
```



The screenshot shows a SQL query window with the following content:

```
22  
23 select DISTINCT * from sales where salesman like '____';  
24  
25  
26  
27  
28
```

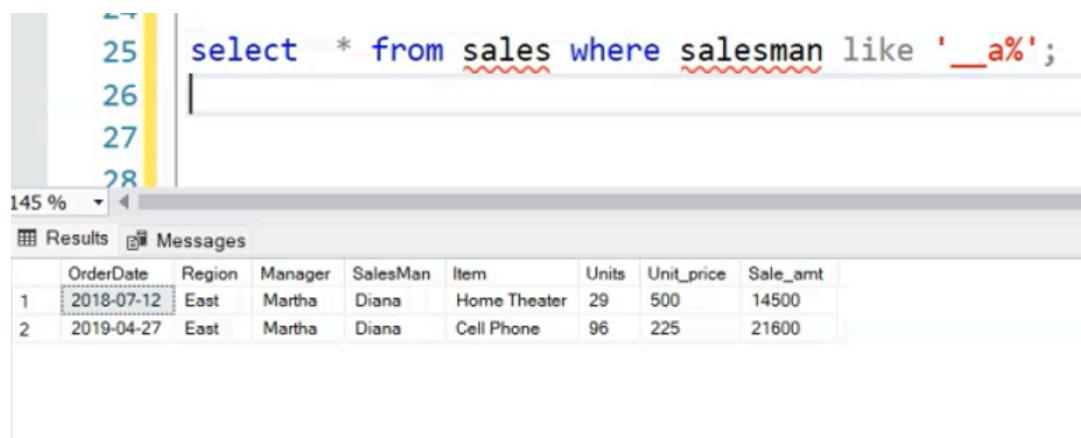
The results pane displays the following data:

	OrderDate	Region	Manager	SalesMan	Item	Units	Unit_price	Sale_amt
1	2018-02-09	Central	Hermann	Luis	Television	36	1198	43128
2	2018-05-05	Central	Hermann	Luis	Television	90	1198	107820
3	2018-09-01	Central	Douglas	John	Desk	2	125	250
4	2018-12-12	Central	Douglas	John	Television	67	1198	80266
5	2019-02-01	Central	Douglas	John	Home Theater	87	500	43500
6	2019-03-24	Central	Hermann	Luis	Video Games	50	58.5	2925
7	2019-11-17	Central	Hermann	Luis	Home Theater	11	500	5500
8	2019-12-04	Central	Hermann	Luis	Home Theater	94	500	47000

✓ Matches names that are **exactly 7 characters** long.

c) Third character is 'a':

```
SELECT DISTINCT * FROM Sales  
WHERE SalesMan LIKE '_a%';
```



The screenshot shows a SQL query window with the following content:

```
25 select * from sales where salesman like '_a%';  
26  
27  
28
```

The results pane displays the following data:

	OrderDate	Region	Manager	SalesMan	Item	Units	Unit_price	Sale_amt
1	2018-07-12	East	Martha	Diana	Home Theater	29	500	14500
2	2019-04-27	East	Martha	Diana	Cell Phone	96	225	21600

✓ Matches names where **3rd character is 'a'**, like **Diana**.

⌚ Wildcards in LIKE

Wildcard	Meaning	Example Pattern	Matches Examples
%	Any number of characters	'M%'	Martha, Michael
_	Exactly one character	'_a%'	Diana, Alexander
[a-c]	Any single char in the range	'[A-C]%'	Alice, Bob, Carl
[^a-c]	Any char not in range	'[^A-C]%'	Names not starting with A, B, or C

🔔 Note: The last two patterns ([]) work in SQL Server but not in all database engines.

✓ Summary – Key Learnings

Operator	Use	Example
IN	Match values from a list	WHERE Region IN ('East', 'West')
BETWEEN	Match values within a range	WHERE Units BETWEEN 50 AND 100
LIKE	Match text patterns with wildcards	WHERE SalesMan LIKE 'M%'