

## SQL WHERE CLAUSE:

Select everything whose firstname is Jim:

The screenshot shows a SQL Server Enterprise Manager window with two tabs: 'SQLQuery2.sql - D...NKL2RA\USER (61))' and 'SQLQuery1.sql - D...NKL2RA\USER (51))\*'. The active tab displays the following SQL query:

```
SELECT * FROM EmployeeDemographics  
WHERE FirstName = 'Jim'
```

Below the query editor, the 'Results' tab is selected, showing a single row of data:

	EmployeeID	FirstName	LastName	Age	Gender
1	1001	Jim	Halpert	30	Male

Now, let's look at the not equal sign <>

Here we get everybody except Jim

The screenshot shows a SQL Server Enterprise Manager window with a single tab: 'SQLQuery1.sql - D...NKL2RA\USER (51))\*'. The active tab displays the following SQL query:

```
SELECT * FROM EmployeeDemographics  
WHERE FirstName <> 'Jim'
```

Below the query editor, the 'Results' tab is selected, showing a list of 8 rows of data:

	EmployeeID	FirstName	LastName	Age	Gender
1	1002	Pam	Beasley	30	Female
2	1003	Dwight	Schrute	29	Male
3	1004	Angela	Martin	31	Female
4	1005	Toby	Flenderson	32	Male
5	1006	Michael	Scott	35	Male
6	1007	Meredith	Palmer	32	Female
7	1008	Stanley	Hudson	38	Male
8	1009	Kevin	Malone	31	Male

Now less than and greater than and equal:

SQLQuery2.sql - D...NKL2RA\USER (61))    SQLQuery1.sql - D...NKL2RA\USER (5

```
SELECT * FROM EmployeeDemographics  
WHERE Age < 30
```

100 %

Results    Messages

	EmployeeID	FirstName	LastName	Age	Gender
1	1003	Dwight	Schrute	29	Male

SQLQuery2.sql - D...NKL2RA\USER (61))    SQLQuery1.sql - D...NKL2RA\USER (51))\*

```
SELECT * FROM EmployeeDemographics  
WHERE Age > 30
```

100 %

Results    Messages

	EmployeeID	FirstName	LastName	Age	Gender
1	1004	Angela	Martin	31	Female
2	1005	Toby	Flenderson	32	Male
3	1006	Michael	Scott	35	Male
4	1007	Meredith	Palmer	32	Female
5	1008	Stanley	Hudson	38	Male
6	1009	Kevin	Malone	31	Male

SQL tutorial | Execute

SQLQuery2.sql - D...NKL2RA\USER (61)) | SQLQuery1.sql - D...NKL2RA\USER (51))

```
SELECT * FROM EmployeeDemographics  
WHERE Age <= 31
```

100 %

Results Messages

	EmployeeID	FirstName	LastName	Age	Gender
1	1001	Jim	Halpert	30	Male
2	1002	Pam	Beasley	30	Female
3	1003	Dwight	Schrute	29	Male
4	1004	Angela	Martin	31	Female
5	1009	Kevin	Malone	31	Male

Now, And, Or

SQLQuery2.sql - D...NKL2RA\USER (61)) | SQLQuery1.sql - D...NKL2RA\USER (51))\*

```
SELECT * FROM EmployeeDemographics  
WHERE Age <= 31 AND Gender = 'Male'
```

100 %

Results Messages

	EmployeeID	FirstName	LastName	Age	Gender
1	1001	Jim	Halpert	30	Male
2	1003	Dwight	Schrute	29	Male
3	1009	Kevin	Malone	31	Male

SQLQuery2.sql - D...NKL2RA\USER (61))

SQLQuery1.sql - D...NKL2RA\

```
SELECT * FROM EmployeeDemographics
WHERE Age <= 31 OR Gender = 'Male'
```

100 %

Results Messages

	EmployeeID	FirstName	LastName	Age	Gender
1	1001	Jim	Halpert	30	Male
2	1002	Pam	Beasley	30	Female
3	1003	Dwight	Schrute	29	Male
4	1004	Angela	Martin	31	Female
5	1005	Toby	Flenderson	32	Male
6	1006	Michael	Scott	35	Male
7	1008	Stanley	Hudson	38	Male
8	1009	Kevin	Malone	31	Male

Or will show between two criteria, one can be true.

Now, LIKE

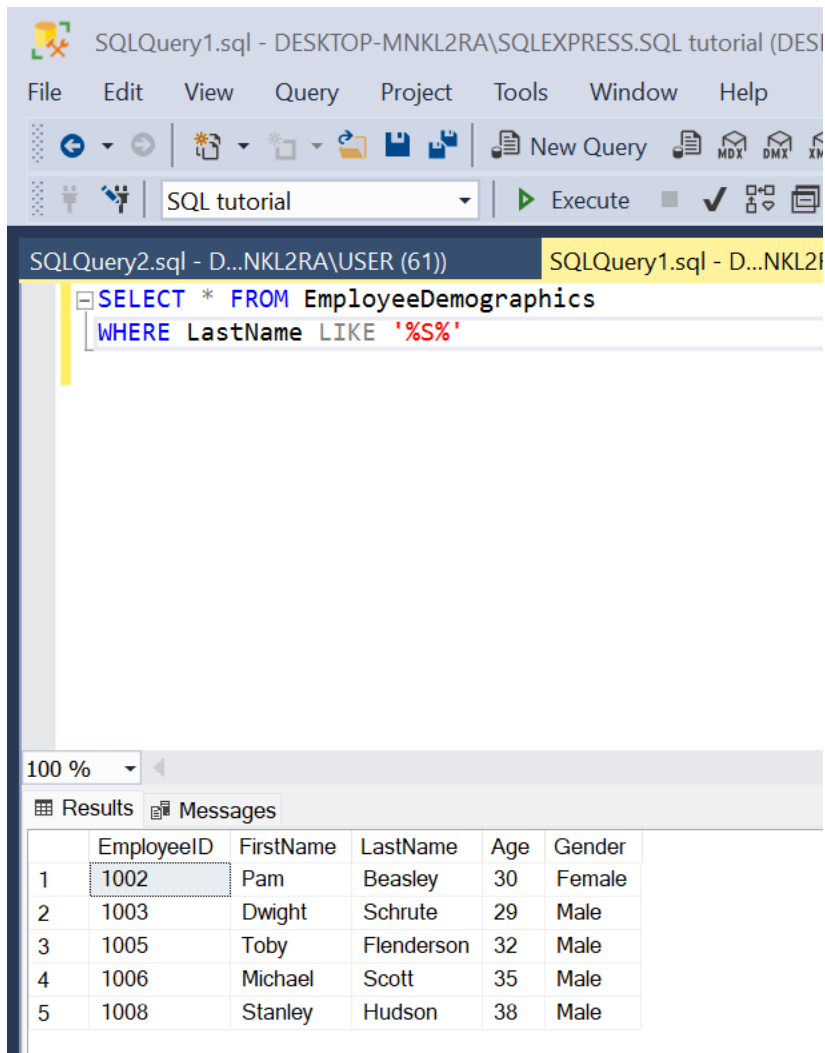
I want every name start with S:

SQLQuery2.sql - D...\NKL2RA\USER (61))SQLQuery1.sql - D...\NKL2RA\USER (51))\*

<

% is known as wildcard.

S in last name in anywhere:

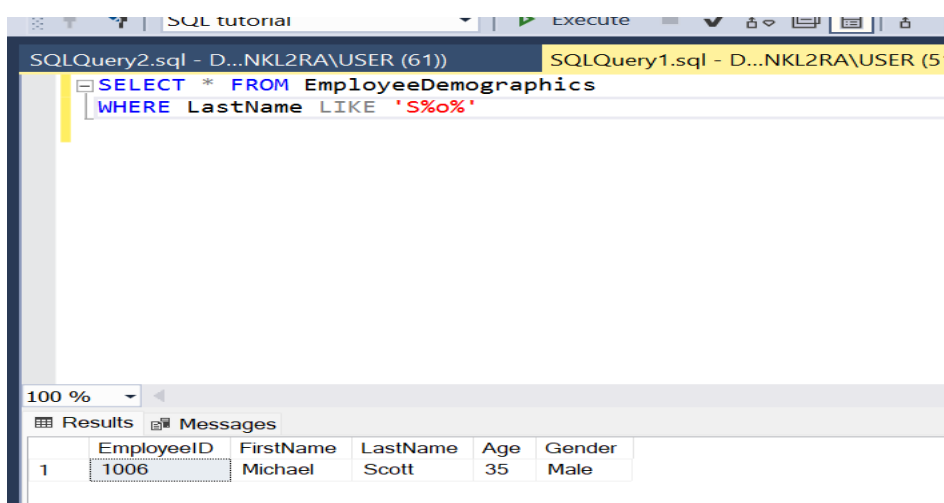


The screenshot shows the SQL Server Enterprise Manager interface. The top menu bar includes File, Edit, View, Query, Project, Tools, Window, and Help. The toolbar contains icons for New Query, MDX, DMX, and XN. The SQL query editor displays the following query:

```
SELECT * FROM EmployeeDemographics
WHERE LastName LIKE '%S%'
```

The Results tab is selected, showing a table with 5 rows and 6 columns: EmployeeID, FirstName, LastName, Age, and Gender. The data is as follows:

	EmployeeID	FirstName	LastName	Age	Gender
1	1002	Pam	Beasley	30	Female
2	1003	Dwight	Schrute	29	Male
3	1005	Toby	Flenderson	32	Male
4	1006	Michael	Scott	35	Male
5	1008	Stanley	Hudson	38	Male



The screenshot shows the SQL Server Enterprise Manager interface. The top menu bar includes File, Edit, View, Query, Project, Tools, Window, and Help. The toolbar contains icons for New Query, MDX, DMX, and XN. The SQL query editor displays the following query:

```
SELECT * FROM EmployeeDemographics
WHERE LastName LIKE 'S%o%'
```

The Results tab is selected, showing a table with 1 row and 6 columns: EmployeeID, FirstName, LastName, Age, and Gender. The data is as follows:

	EmployeeID	FirstName	LastName	Age	Gender
1	1006	Michael	Scott	35	Male

This wildcard works in order. In Scott name, o is after S that's why, it shows Scott.

NULL:

The screenshot shows a SQL Server Enterprise Manager window with two tabs: 'SQLQuery2.sql - D...NKL2RA\USER (61))' and 'SQLQuery1.sql - D...NKL2RA\USER (51))'. The active query in the 'SQLQuery2.sql' tab is:

```
SELECT * FROM EmployeeDemographics
WHERE Age IS NULL
```

Below the query editor, the 'Results' tab is selected, showing a table with the following columns: EmployeeID, FirstName, LastName, Age, and Gender. The table is currently empty, indicating no results were returned by the query.

There is not any null value so no output.

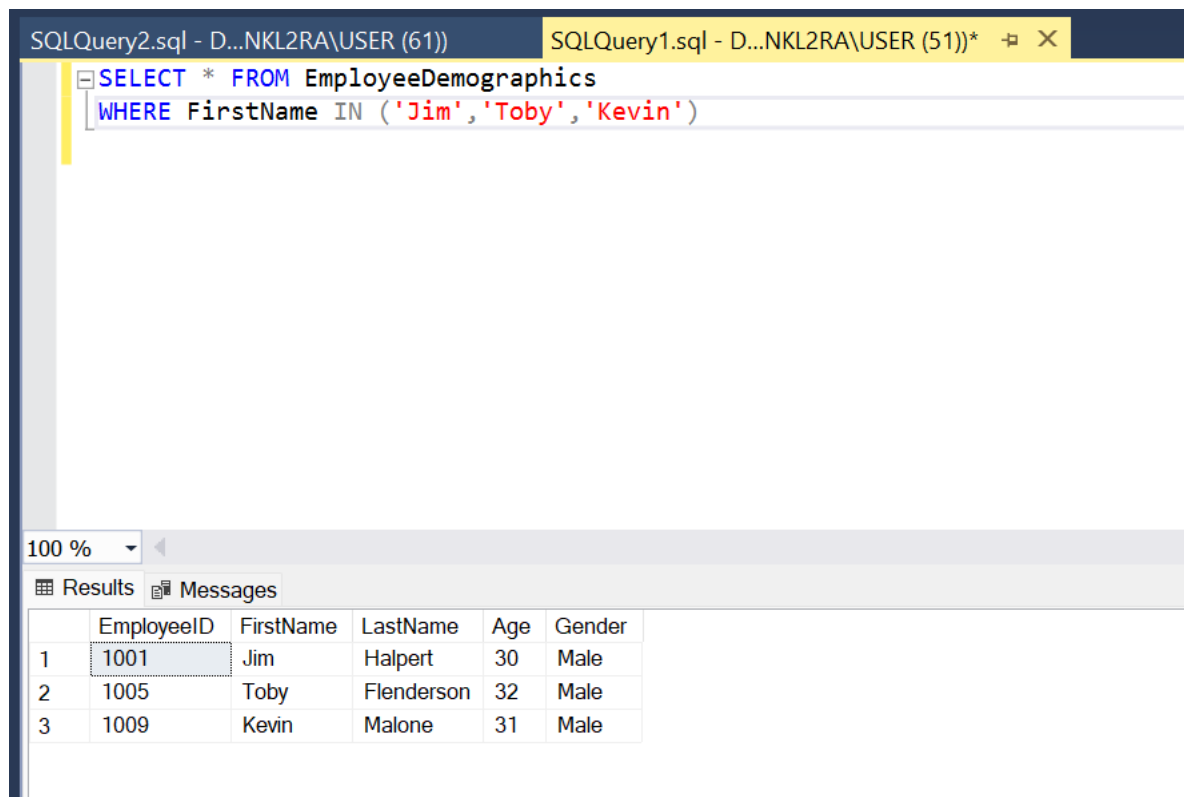
The screenshot shows a SQL Server Enterprise Manager window with two tabs: 'SQLQuery2.sql - D...NKL2RA\USER (61))' and 'SQLQuery1.sql - D...NKL2RA\USER (51))'. The active query in the 'SQLQuery2.sql' tab is:

```
SELECT * FROM EmployeeDemographics
WHERE Age IS NOT NULL
```

Below the query editor, the 'Results' tab is selected, showing a table with the following columns: EmployeeID, FirstName, LastName, Age, and Gender. The table contains 9 rows of data:

	EmployeeID	FirstName	LastName	Age	Gender
1	1001	Jim	Halpert	30	Male
2	1002	Pam	Beasley	30	Female
3	1003	Dwight	Schrute	29	Male
4	1004	Angela	Martin	31	Female
5	1005	Toby	Flenderson	32	Male
6	1006	Michael	Scott	35	Male
7	1007	Meredith	Palmer	32	Female
8	1008	Stanley	Hudson	38	Male
9	1009	Kevin	Malone	31	Male

Now, IN, in is like equal to:



The screenshot shows a SQL Server Enterprise Manager window with two tabs: 'SQLQuery2.sql - D...NKL2RA\USER (61))' and 'SQLQuery1.sql - D...NKL2RA\USER (51))\*'. The active tab displays the following SQL query:

```
SELECT * FROM EmployeeDemographics
WHERE FirstName IN ('Jim', 'Toby', 'Kevin')
```

Below the query editor, the 'Results' tab is selected, showing a table with 3 rows and 6 columns: EmployeeID, FirstName, LastName, Age, and Gender. The first row is highlighted.

	EmployeeID	FirstName	LastName	Age	Gender
1	1001	Jim	Halpert	30	Male
2	1005	Toby	Flenderson	32	Male
3	1009	Kevin	Malone	31	Male