

INSERT INTO برای درج سطر یا رکورد جدید در جدول استفاده می شود و به دو روش اینکار انجام می شود:

1- نام ستون ها و مقادیری که باید درج شوند را مشخص کنیم.

فیلدی با مقدار NULL فیلدی بدون مقدار است. اگر یک فیلد در جدول اختیاری باشد، می توان یک رکورد جدید درج کرد یا یک رکورد را بدون افزودن مقداری به این فیلد به روز کرد. سپس، فیلد با مقدار NULL ذخیره می شود.

SELECT column_names

FROM table_name

WHERE column_name IS NULL;

توجه: مقدار NULL با مقدار صفر یا فیلدی که حاوی فاصله است متفاوت است.

فیلدی با مقدار NULL فیلدی است که در حین ایجاد رکورد خالی مانده است!

The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL query:

```
SELECT DOCUMENT
FROM Production.Document
WHERE Document IS NULL;
```

The query has been executed successfully, and the results are displayed in the Results pane. The results show four rows, all with NULL values in the DOCUMENT column.

	DOCUMENT
1	NULL
2	NULL
3	NULL
4	NULL

The status bar at the bottom indicates that the query was executed successfully and returned 4 rows.

SELECT column_names

FROM table_name

WHERE column_name IS NOT NULL;

عملگر IS NOT NULL برای آزمایش مقادیر غیر خالی (مقادیر NOT NULL) استفاده می شود.

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left displays the database structure, including tables like System Tables, FileTables, External Tables, Graph Tables, and various tables in the AdventureWorks2019 database. The main query window shows the following SQL query:

```
SELECT DOCUMENT
FROM Production.Document
WHERE Document IS NOT NULL;
```

The Results pane shows the output of the query, displaying a list of document IDs (DOCUMENT) in hexadecimal format. The status bar at the bottom indicates that the query was executed successfully, returning 9 rows.

UPDATE از این دستور برای بروزرسانی رکوردهای موجود یک جدول استفاده می شود. به عبارت WHERE در عبارت UPDATE توجه کنید. بند WHERE مشخص می کند که کدام رکوردها باید به روز شود. اگر بند WHERE را حذف کنید، تمام رکوردهای جدول به روز می شوند!

```
UPDATE Person.Address
SET City= 'Bothell', PostalCode= '98011'
WHERE AddressID = 1 ;
```

The screenshot shows the Microsoft SQL Server Management Studio interface with the following SQL query in the main query window:

```
UPDATE Person.Address
SET City= 'Bothell', PostalCode= '98011'
WHERE AddressID = 1 ;
```

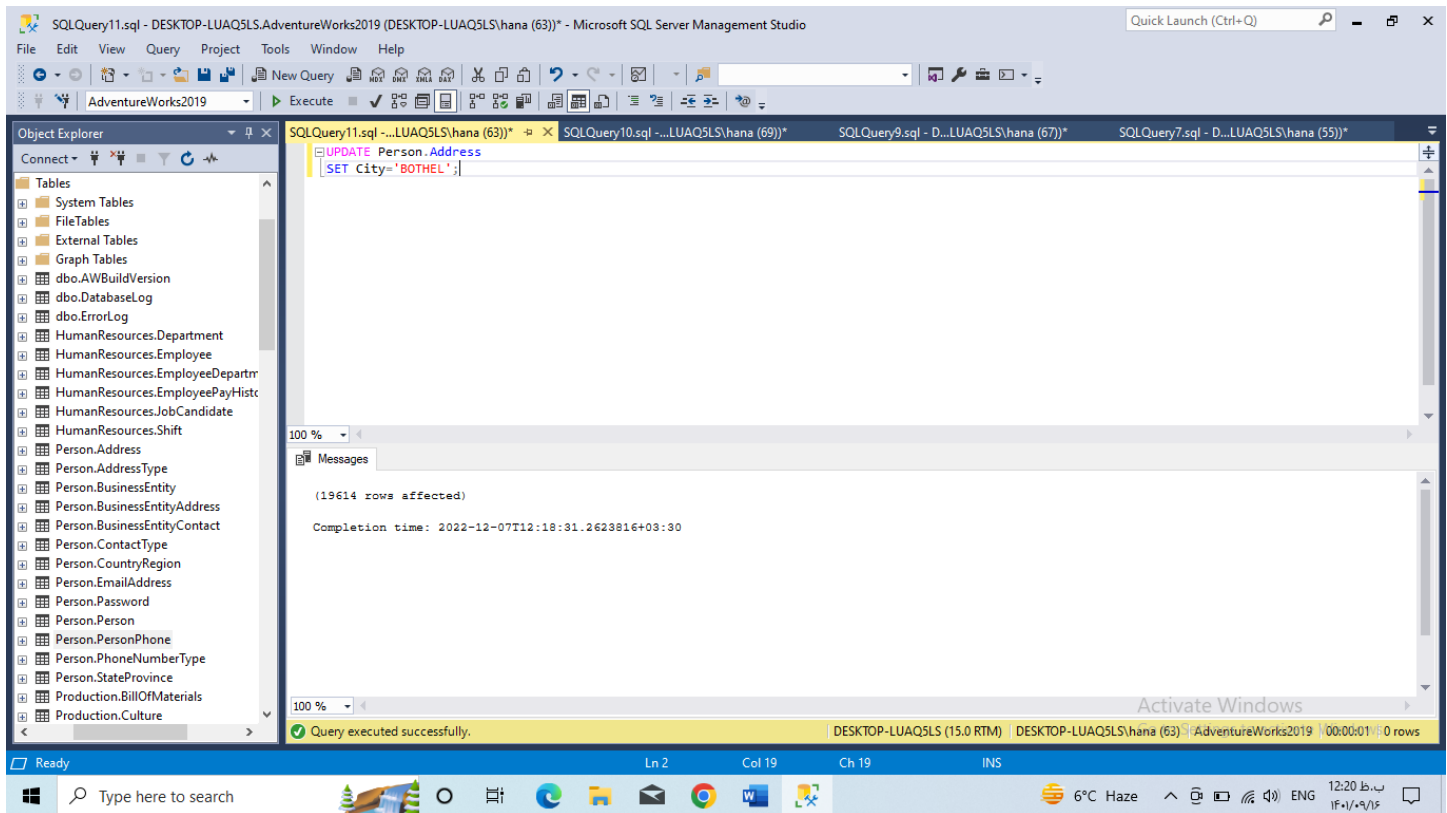
The Messages pane shows the execution results:

```
(1 row affected)
Completion time: 2022-12-07T12:13:59.5583960+03:30
```

The status bar at the bottom indicates that the query was executed successfully, affecting 1 row.

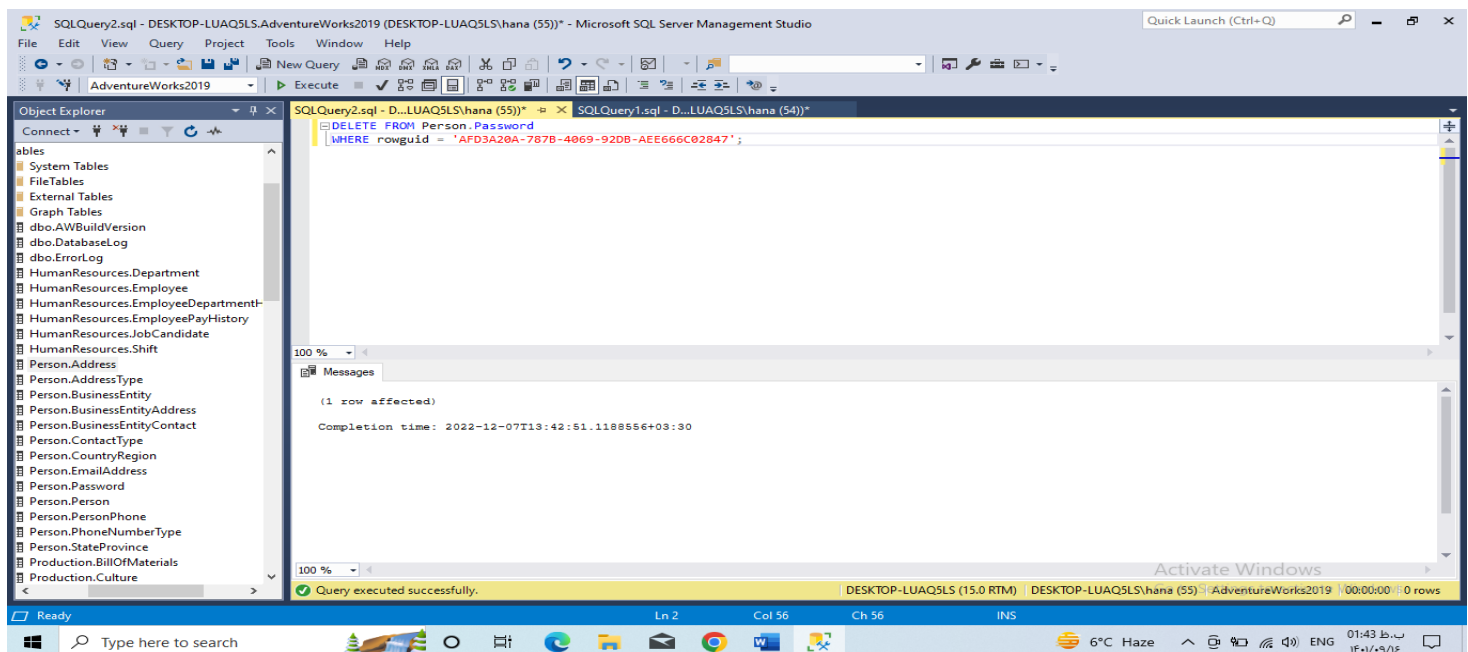
هنگام به روز رسانی سوابق مراقب باشید. اگر بند WHERE را حذف کنید، تمام رکوردها به روز می شوند!

```
UPDATE Person.Address  
SET City='BOTHEL';
```



DELETE برای حذف سرهای جدول استفاده می شود. به عبارت WHERE در عبارت DELETE توجه کنید. بند WHERE مشخص می کند که کدام رکوردها باید حذف شوند. اگر بند WHERE را حذف کنید، تمام رکوردهای جدول حذف خواهند شد!

```
DELETE FROM Person.Password  
WHERE rowguid = 'AFD3A20A-787B-4069-92DB-AEE666C02847';
```



```
SELECT MIN(ModifiedDate) AS MODIFIEDDATE
FROM Person.Password;
```

تابع (MIN()) کوچکترین مقدار ستون انتخاب شده را برمی گرداند.

The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL code:

```
SELECT MIN(ModifiedDate) AS MODIFIEDDATE
FROM Person.Password;
```

The query has been executed successfully, and the results are displayed in the Results pane. The results show a single row with the value '2006-06-23 00:00:00.000' under the column header 'MODIFIEDDATE'.

	MODIFIEDDATE
1	2006-06-23 00:00:00.000

The status bar at the bottom indicates 'Query executed successfully.' and '1 rows'.

```
SELECT MAX(ModifiedDate) AS MODIFIEDDATE
FROM Person.Password;
```

تابع (MAX()) بزرگترین مقدار ستون انتخاب شده را برمی گرداند.

The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL code:

```
SELECT MAX(ModifiedDate) AS MODIFIEDDATE
FROM Person.Password;
```

The query has been executed successfully, and the results are displayed in the Results pane. The results show a single row with the value '2015-04-15 16:33:33.123' under the column header 'MODIFIEDDATE'.

	MODIFIEDDATE
1	2015-04-15 16:33:33.123

The status bar at the bottom indicates 'Query executed successfully.' and '1 rows'.

تابع COUNT() تعداد ردیف هایی را برمی گرداند که با یک معیار مشخص مطابقت دارند.

```
SELECT COUNT(BusinessEntityID)
FROM Person.Password;
```

The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL code:

```
SELECT COUNT(BusinessEntityID)
FROM Person.Password;
```

The query has been executed successfully, and the results are displayed in the Results pane. The results show a single row with the value 19970.

(No column name)
19970

The status bar at the bottom indicates that the query was executed successfully and returned 1 row.

تابع AVG() مقدار متوسط یک ستون عددی را برمی گرداند.

```
SELECT AVG(ReorderPoint)
FROM Production.Product;
```

The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL code:

```
SELECT AVG(ReorderPoint)
FROM Production.Product;
```

The query has been executed successfully, and the results are displayed in the Results pane. The results show a single row with the value 401.

(No column name)
401

The status bar at the bottom indicates that the query was executed successfully and returned 1 row.

SELECT SUM(SafetyStockLevel)

FROM Production.Product;

تابع SUM() مجموع کل یک ستون عددی را برمی گرداند.

SQLQuery8.sql - DESKTOP-LUAQ5LS.AdventureWorks2019 (DESKTOP-LUAQ5LS\hana (76)) - Microsoft SQL Server Management Studio

Object Explorer: Tables, System Tables, FileTables, External Tables, Graph Tables, dbo.AWBuildVersion, dbo.DatabaseLog, dbo.ErrorLog, HumanResources.Department, HumanResources.Employee, HumanResources.EmployeeDepartmentH, HumanResources.EmployeePayHistory, HumanResources.JobCandidate, HumanResources.Shift, Person.Address, Person.AddressType, Person.BusinessEntity, Person.BusinessEntityAddress, Person.BusinessEntityContact, Person.ContactType, Person.CountryRegion, Person.EmailAddress, Person.Password, Person.Person, Person.PersonPhone, Person.PhoneNumberType, Person.StateProvince, Production.BillOfMaterials, Production.Culture.

SQLQuery8.sql - D:\LUAQ5LS\hana (76)*

```
SELECT SUM(SafetyStockLevel)
FROM Production.Product;
```

Results: (No column name)

1	269716
---	--------

Query executed successfully.

SELECT *

FROM Person.Person

WHERE FirstName IN ('micheal', 'john');

عملگر IN به شما امکان می دهد چندین مقدار را در یک عبارت WHERE مشخص کنید.

عملگر IN خلاصه ای برای شرایط OR چنگانه است.

SQLQuery2.sql - DESKTOP-LUAQ5LS.AdventureWorks2019 (DESKTOP-LUAQ5LS\hana (66)) - Microsoft SQL Server Management Studio

Object Explorer: Tables, System Tables, FileTables, External Tables, Graph Tables, dbo.AWBuildVersion, dbo.DatabaseLog, dbo.ErrorLog, HumanResources.Department, HumanResources.Employee, HumanResources.EmployeeDepartmentH, HumanResources.EmployeePayHistory, HumanResources.JobCandidate, HumanResources.Shift, Person.Address, Person.AddressType, Person.BusinessEntity, Person.BusinessEntityAddress, Person.BusinessEntityContact, Person.ContactType, Person.CountryRegion, Person.EmailAddress, Person.Password, Person.Person, Person.PersonPhone, Person.PhoneNumberType, Person.StateProvince, Production.BillOfMaterials, Production.Culture.

SQLQuery2.sql - D:\LUAQ5LS\hana (66)*

```
SELECT *
FROM Person.Person
WHERE FirstName IN ('micheal', 'john');
```

Results:

	BusinessEntityID	Person Type	Name Style	Title	FirstName	MiddleName	LastName	Suffix	EmailPromotion	AdditionalContactInfo	Demographics	rowguid
9	18352	IN	0	NULL	John	D	White	NULL	0	NULL	<IndividualSurvey xmlns="http://schemas.microso...	ADE02
10	18371	IN	0	NULL	John	M	Walker	NULL	0	NULL	<IndividualSurvey xmlns="http://schemas.microso...	CDE01
11	14591	IN	0	NULL	Micheal	NULL	Vazquez	NULL	0	NULL	<IndividualSurvey xmlns="http://schemas.microso...	3DBD3
12	14583	IN	0	NULL	Micheal	NULL	Travers	NULL	1	NULL	<IndividualSurvey xmlns="http://schemas.microso...	DCD26
13	18359	IN	0	NULL	John	I	Thomps...	NULL	1	NULL	<IndividualSurvey xmlns="http://schemas.microso...	1570AE
14	18344	IN	0	NULL	John	W	Thomas	NULL	2	NULL	<IndividualSurvey xmlns="http://schemas.microso...	9A912E
15	18329	IN	0	NULL	John	NULL	Taylor	NULL	0	NULL	<IndividualSurvey xmlns="http://schemas.microso...	A23E22
16	14600	IN	0	NULL	Micheal	NULL	Suarez	NULL	1	NULL	<IndividualSurvey xmlns="http://schemas.microso...	98CB22
17	18222	IN	0	NULL	John	NULL	Smith	NULL	0	NULL	<IndividualSurvey xmlns="http://schemas.microso...	170138
18	14594	IN	0	NULL	Micheal	T	Serrano	NULL	0	NULL	<IndividualSurvey xmlns="http://schemas.microso...	CBA668

Query executed successfully.

عملگر BETWEEN مقادیری را در یک محدوده مشخص انتخاب می کند. مقادیر می توانند اعداد، متن یا تاریخ باشند.

SELECT *

FROM Sales.SalesTaxRate

where StateProvinceID BETWEEN 49 and 63;

عملگر BETWEEN شامل مقادیر شروع و پایان است.

The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL code:

```
SELECT *
FROM Sales.SalesTaxRate
where StateProvinceID BETWEEN 49 and 63;
```

The query has been executed successfully, and the results are displayed in the Results pane. The results table has 8 columns: SalesTaxRateID, StateProvinceID, TaxType, TaxRate, Name, rowguid, and ModifiedDate. The data is as follows:

SalesTaxRateID	StateProvinceID	TaxType	TaxRate	Name	rowguid	ModifiedDate
1	57	1	14/25	Canadian GST + Ontario Provincial Tax	05C4FFDB-4F84-4CDF-ABE5-FDF3216EA74E	2008-04-30 00:00:00.000
2	63	1	14/25	Canadian GST + Quebec Provincial Tax	D4EDB557-56D7-403C-B538-4DF5E7302588	2008-04-30 00:00:00.000
3	57	2	7/00	Canadian GST	7E0E97A2-878B-476F-A648-05A3DD4450ED	2008-04-30 00:00:00.000
4	63	2	7/00	Canadian GST	1E285D2C-8AF7-47AA-B06A-762CF4D93ACD	2008-04-30 00:00:00.000
5	49	3	7/00	Canadian GST	FB7607EE-8F12-41EA-B461-10885C6F1533	2008-04-30 00:00:00.000
6	51	3	7/00	Canadian GST	06DF529D-EB11-446F-9570-9EE97B8EA1BF	2008-04-30 00:00:00.000

عملگر NOT BETWEEN مقادیری را در خارج محدوده مشخص انتخاب می کند. مقادیر می توانند اعداد، متن یا تاریخ باشند.

SELECT *

FROM Sales.SalesTaxRate

where StateProvinceID not BETWEEN 49 and 63;

The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL code:

```
SELECT *
FROM Sales.SalesTaxRate
where StateProvinceID not BETWEEN 49 and 63;
```

The query has been executed successfully, and the results are displayed in the Results pane. The results table has 8 columns: SalesTaxRateID, StateProvinceID, TaxType, TaxRate, Name, rowguid, and ModifiedDate. The data is as follows:

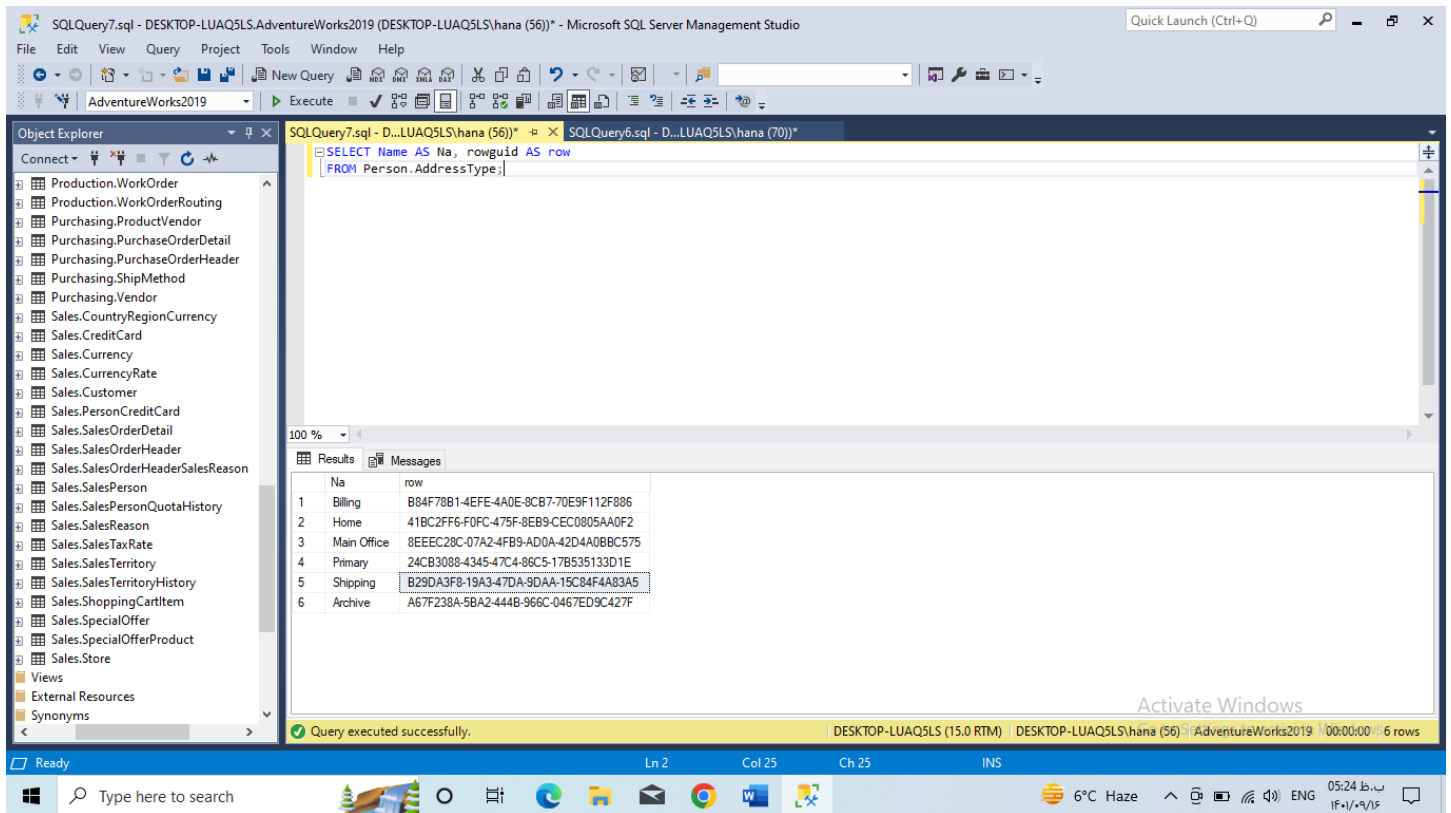
SalesTaxRateID	StateProvinceID	TaxType	TaxRate	Name	rowguid	ModifiedDate
13	30	1	7/00	Massachusetts State Sales Tax	207C7024-85FE-42CB-8F74-9FE88E2F4E1B	2008-04-30 00:00:00.000
14	22	1	7/25	Michigan State Sales Tax	F9D1A850-6474-43B5-A7EA-BED59E00439A	2008-04-30 00:00:00.000
15	23	1	6/75	Minnesota State Sales Tax	1753D75F-8357-4497-8C92-543F17B8F514	2008-04-30 00:00:00.000
16	24	1	7/25	Tennessee State Sales Tax	ECBA1BAB-A7A6-4CC2-8D0F-C094A38E5C0E	2008-04-30 00:00:00.000
17	25	1	7/50	Texas State Sales Tax	D50102E1-0150-43F3-93AD-12AB8C1DEB06	2008-04-30 00:00:00.000
18	26	1	5/00	Utah State Sales Tax	5E6D8621-36AF-40CE-972B-61AAEF0556D8	2008-04-30 00:00:00.000
19	27	1	8/80	Washington State Sales Tax	9C84C0AF-41CD-4670-98BC-5026E188541F	2008-04-30 00:00:00.000
20	28	3	10/00	Taxable Supply	F53B6A7E-7DA4-49E8-AC09-D57FE7FAC6D3	2008-04-30 00:00:00.000
21	29	3	16/00	Germany Output Tax	78C8A559-4D0F-45B6-B7A3-E11FF9A702F4	2008-04-30 00:00:00.000
22	30	3	19/60	France Output Tax	5446EA04-FA7C-4B02-A157-A05F44BF47B6	2008-04-30 00:00:00.000
23	31	3	17/50	United Kingdom Output Tax	ABD185A9-5367-44E1-8A6E-71D083943FC3	2008-04-30 00:00:00.000

```
SELECT Name AS Na, rowguid AS row
FROM Person.AddressType;
```

ALIASES SQL برای دادن نام موقت به جدول یا ستونی در جدول استفاده می شود.

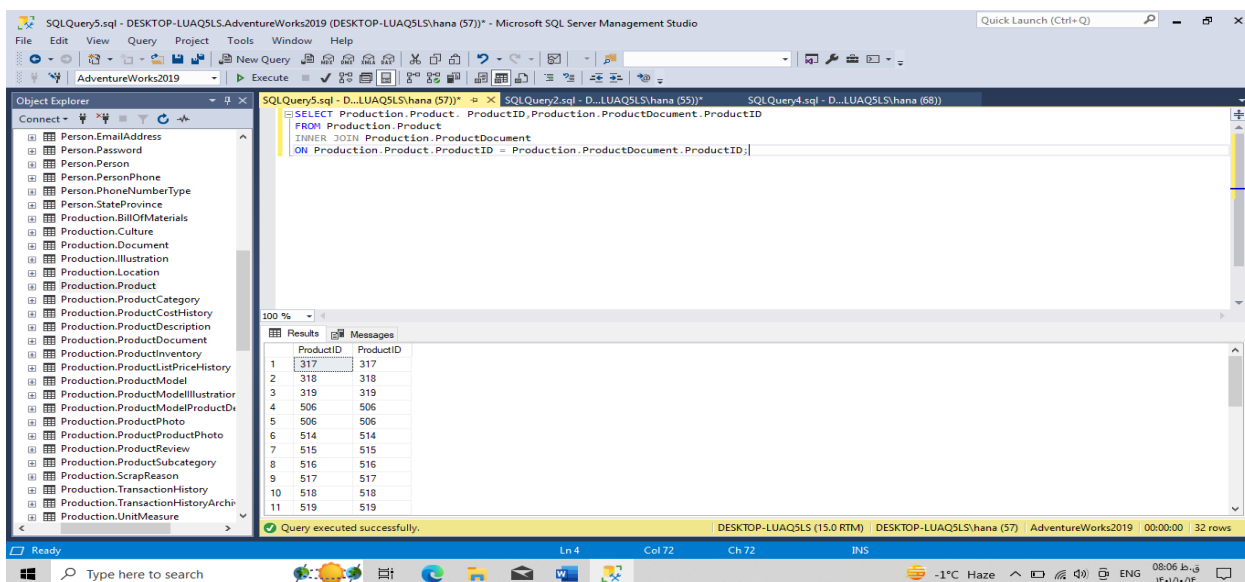
ALIASES نام مستعار اغلب برای خوانایی بیشتر نام ستون ها استفاده می شود.

ALIASES نام مستعار فقط برای مدت زمان آن پرس و جو وجود دارد. با کلمه کلیدی AS ایجاد می شود.



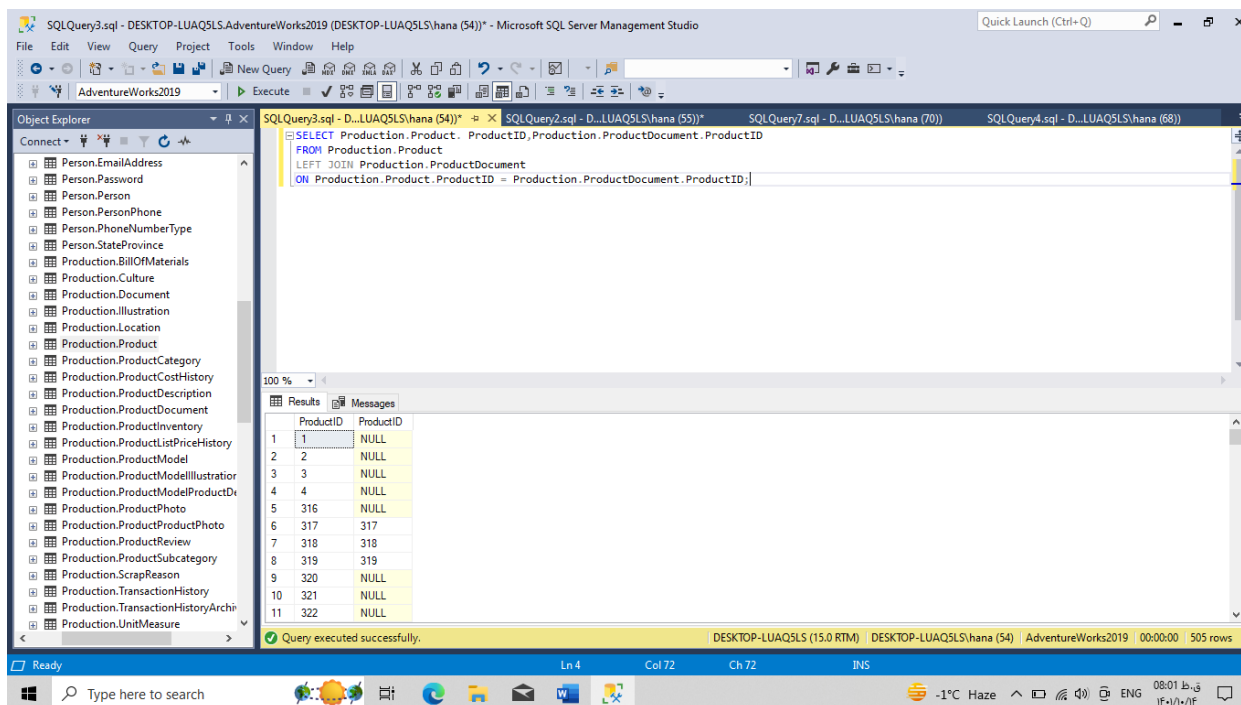
کلمه کلیدی INNER JOIN رکوردهایی را انتخاب می کند که دارای مقادیر منطبق در هر دو جدول هستند. همه سطرها را از هر دو جدول تا زمانی که بین ستون ها مطابقت داشته باشد انتخاب می کند

```
SELECT Production.Product.ProductID, Production.ProductDocument.ProductID
FROM Production.Product
INNER JOIN Production.ProductDocument
ON Production.Product.ProductID = Production.ProductDocument.ProductID;
```



کلمه کلیدی LEFT JOIN همه رکوردها را از جدول سمت چپ (جدول 1) و رکوردهای منطبق از جدول سمت راست (جدول 2) را برمی گرداند. در صورت عدم تطابق، نتیجه 0 رکورد از سمت راست است

```
SELECT Production.Product.ProductID, Production.ProductDocument.ProductID
FROM Production.Product
LEFT JOIN Production.ProductDocument
ON Production.Product.ProductID = Production.ProductDocument.ProductID;
```



The screenshot displays the Microsoft SQL Server Management Studio interface. The query editor shows the following SQL code:

```
SELECT Production.Product.ProductID, Production.ProductDocument.ProductID
FROM Production.Product
LEFT JOIN Production.ProductDocument
ON Production.Product.ProductID = Production.ProductDocument.ProductID;
```

The Results pane shows the following data:

ProductID	ProductID
1	NULL
2	NULL
3	NULL
4	NULL
5	NULL
6	317
7	318
8	319
9	320
10	321
11	322

The status bar at the bottom indicates that the query was executed successfully, returning 505 rows.

کلمه کلیدی RIGHT JOIN همه رکوردها را از جدول سمت راست (جدول 2) و رکوردهای منطبق از جدول سمت چپ (جدول 1) برمی گرداند. در صورت عدم تطابق، نتیجه 0 رکورد از سمت چپ است.

```
SELECT Production.Product.ProductID, Production.ProductDocument.ProductID
FROM Production.Product
RIGHT JOIN Production.ProductDocument
ON Production.Product.ProductID = Production.ProductDocument.ProductID;
```

SQLQuery4.sql - DESKTOP-LUAQ5LS.AdventureWorks2019 (DESKTOP-LUAQ5LS\hana (62)) - Microsoft SQL Server Management Studio

```

SELECT Production.Product.ProductID, Production.ProductDocument.ProductID
FROM Production.Product
RIGHT JOIN Production.ProductDocument
ON Production.Product.ProductID = Production.ProductDocument.ProductID;

```

ProductID	ProductID
317	317
318	318
319	319
506	506
506	506
514	514
515	515
516	516
517	517
518	518
519	519

Query executed successfully. DESKTOP-LUAQ5LS (15.0 RTM) DESKTOP-LUAQ5LS\hana (62) AdventureWorks2019 00:00:00 32 rows

کلمه کلیدی FULL OUTER JOIN همه رکوردها را زمانی که در سوابق جدول چپ (جدول 1) یا راست (جدول 2) مطابقت داشته باشد، برمی گرداند.

```

SELECT Production.Product.ProductID, Production.ProductDocument.ProductID
FROM Production.Product
FULL JOIN Production.ProductDocument
ON Production.Product.ProductID = Production.ProductDocument.ProductID;

```

SQLQuery6.sql - DESKTOP-LUAQ5LS.AdventureWorks2019 (DESKTOP-LUAQ5LS\hana (66)) - Microsoft SQL Server Management Studio

```

SELECT Production.Product.ProductID, Production.ProductDocument.ProductID
FROM Production.Product
FULL JOIN Production.ProductDocument
ON Production.Product.ProductID = Production.ProductDocument.ProductID;

```

ProductID	ProductID
1	NULL
2	NULL
3	NULL
4	NULL
316	NULL
317	317
318	318
319	319
320	NULL
321	NULL
322	NULL

Query executed successfully. DESKTOP-LUAQ5LS (15.0 RTM) DESKTOP-LUAQ5LS\hana (66) AdventureWorks2019 00:00:00 505 rows

