

How to create, validate, and modify the Identity column in Microsoft SQL Server

Microsoft SQL Server has a built-in Identity column, used to generate important values. It can reference an AutoNumber field in Microsoft Access or a range in Oracle. This article from TipsMake will guide you on how to create, check, and modify the Identity column in Microsoft SQL Server.

An Identity attribute in a column will generate a new value based on the current value and increments when a new record is inserted into the table. Each new value generated for a specific transaction will differ from other concurrent transactions in the table. This Identity column can be used as a single column when it is designated as the primary key.

Readers can find detailed instructions on how to create, check, and modify the Identity column in Microsoft SQL Server in the article below by TipsMake.

How to create, validate, and modify the Identity column in Microsoft SQL Server

Step 1: Create a table using the Identity property.

In this tutorial, TipsMake will create a new table called TechJourney, which includes three columns: TechJourneyID, TechJourneyCode, and Description.

- SQL syntax:

```
Create Table TableName  
( ID int IDENTITY(1,1) NOT NULL,  
Column1 datatype,  
Column2 datatype,  
Column3 datatype,  
PRIMARY KEY (ID))
```

```
Create Table TableName  
( ID int IDENTITY(1,1) NOT NULL,  
Column1 datatype,  
Column2 datatype,  
Column3 datatype,  
PRIMARY KEY (ID))
```

Run the following SQL command in SQL Management Studio:

```
CREATE TABLE TechJourney (  
TechJourneyID int IDENTITY(1,1) NOT NULL,  
TechJourneyCode [nvarchar](10) NULL,
```

**Description [nvarchar](100) NULL,
PRIMARY KEY(TechJourneyID))**

Column Name	Data Type	Allow Nulls
TechJourneyID	int	<input type="checkbox"/>
TechJourneyCode	nvarchar(10)	<input checked="" type="checkbox"/>
Description	nvarchar(100)	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

Column Properties	
Has Non-SQL Server Subscriber	No
Identity Specification	Yes
(Is Identity)	Choose Yes Yes
Identity Increment	1
Identity Seed	1
Indexable	Yes
Is Columnset	No

Next, insert the two records below into the TechJourney table to check the Identity value:

Insert into TechJourney(TechJourneyCode, Description)

Values('TJ1', 'Tech Journey 1')

And:

Insert into TechJourney(TechJourneyCode, Description)

Values('TJ2', 'Tech Journey 2')

```
Insert into TechJourney
values('TJ', 'Tech Journey')

Insert into TechJourney
values('TN', 'Tech News')

select * from TechJourney
```

	TechJourneyID	TechJourneyCode	Description
1	1	TJ	Tech Journey
2	2	TN	Tech News

Step 2: Check the current Identity value

- SQL syntax:

DBCC CHECKIDENT(table_name, NORESEED)

Run the following SQL command in SQL Management Studio:

Note : NORESEED is the current Identity value and should not be changed.

The next step is to check the current value after inserting the two records.

Run the following SQL command in SQL Management Studio:

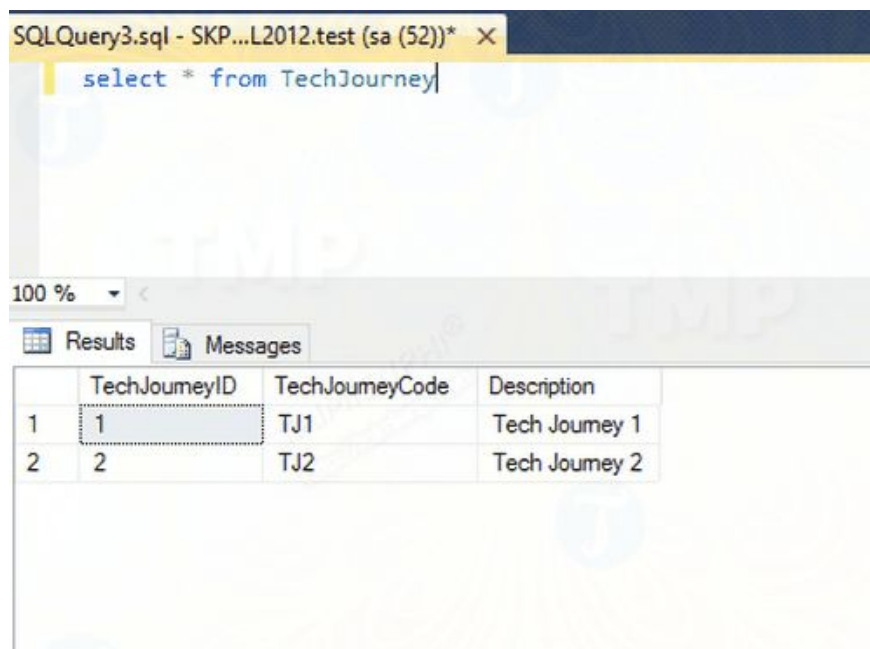
DBCC CHECKIDENT(TechJourney, NORESEED)

It will return the following message:

"Checking identity information: current identity value '2'.

DBCC execution completed. If DBCC printed error messages, contact your system administrator".

Next, select the records in the TechJourney table to view the TechJourneyID value as shown below:



Step 3: Force the current Identity value to the new value.

- SQL syntax:

DBCC CHECKIDENT(table_name, RESEED, new_reseed_value)

Suppose you want to change the current value from 2 to 20. Run the following SQL command in SQL Management Studio:

DBCC CHECKIDENT(TechJourney, RESEED, 20)

It will return the following message:

"Checking identity information: current identity value '2'.

DBCC execution completed. If DBCC printed error messages, contact your system administrator".

Verify the current Identity rental price again by running the following SQL command in SQL Management Studio:

DBCC CHECKIDENT(TechJourney, NORESEED)

It will return the following message:

"Checking identity information: current identity value '20', current column value '2'.

DBCC execution completed. If DBCC printed error messages, contact your system administrator".

In the previous step, we inserted two records into the table, and the Identity value is now 2.

Insert the two records into the TechJourney table again by running the SQL command below:

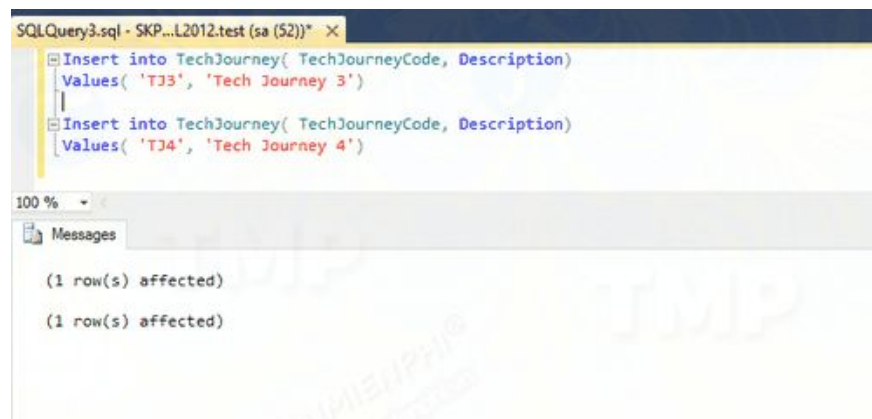
Insert into TechJourney(TechJourneyCode, Description)

Values('TJ3', 'Tech Journey 3')

And

Insert into TechJourney(TechJourneyCode, Description)

Values('TJ4', 'Tech Journey 4')



```
SQLQuery3.sql - SKP...L2012.test (sa (52)) X
Insert into TechJourney( TechJourneyCode, Description)
Values( 'TJ3', 'Tech Journey 3')
Insert into TechJourney( TechJourneyCode, Description)
Values( 'TJ4', 'Tech Journey 4')
```

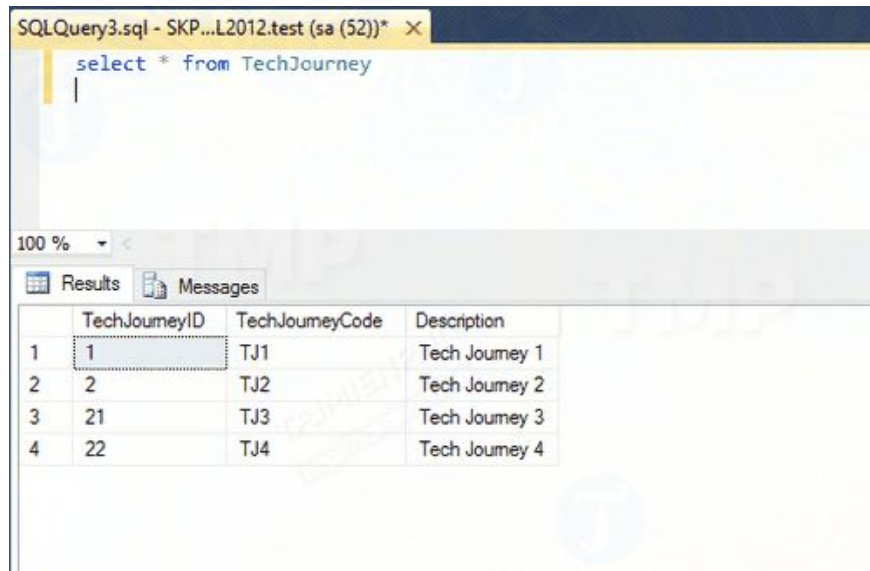
100 %

Messages

(1 row(s) affected)

(1 row(s) affected)

Select the records in the TechJourney table to view the TechJourneyID value as shown below:



TechjourneyID for T3 is 21 = 20 (new identity value) + 1 (increase value).

The article on TipsMake just guided you on how to create, check, and modify the Identity column in Microsoft SQL Server. Other SQL Server tutorials, such as how to **reset a lost SA password on SQL Server**, will help you gain a better understanding of this system. If you have any questions, please leave your comments in the section below the article.

You finished reading the article "**How to create, validate, and modify the Identity column in Microsoft SQL Server**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.