(T38)討論SequenceObject和IdentityProperty  
CourseGUID: e48417fc-9db5-4e99-822c-706c5ccef6cc  
=======================================================================  
(T38)討論SequenceObject和IdentityProperty  
=======================================================================  
0. Summary

-----------

1. SequenceObject

1.0. Create Sequence Object by SSMS

1.1. Sequence Object Basic concept.

1.2. decrementing Sequence object

1.3. MINVALUE and MAXVALUE of Sequence object

1.4. MINVALUE, MAXVALUE and CYCLE of Sequence object

1.5. MINVALUE, MAXVALUE and CYCLE of Sequence object

1.6. Clean up

-----------

2. SequenceAndIdentity

2.1. Sequence object V.S. Identity property

2.2. Sequence object V.S. Identity property

2.3. Clean up  
=======================================================================

0. Summary

1.

Sequence Object

Reference:

<https://docs.microsoft.com/en-us/sql/t-sql/statements/create-sequence-transact-sql>

CREATE SEQUENCE [schema\_name . ] sequence\_name

    [ AS [ built\_in\_integer\_type | user-defined\_integer\_type ] ]

    [ START WITH <constant> ]

    [ INCREMENT BY <constant> ]

    [ { MINVALUE [ <constant> ] } | { NO MINVALUE } ]

    [ { MAXVALUE [ <constant> ] } | { NO MAXVALUE } ]

    [ CYCLE | { NO CYCLE } ]

    [ { CACHE [ <constant> ] } | { NO CACHE } ]

    [ ; ]

E.g.

--CREATE SEQUENCE SequenceObj4

--AS INT

--START WITH 14

--INCREMENT BY 1

--MINVALUE 10

--MAXVALUE 15

--CYCLE

--CACHE 10

1.1.

--AS DataType

E.g.

--AS INT

Default is bigint.

DataType can be Built-in integer type

(tinyint , smallint, int, bigint, decimal etc...) or

user-defined integer type.

1.2.

--START WITH N

E.g.

--START WITH 14

The sequence object starting value is N.

1.3.

--INCREMENT BY N

E.g.

--INCREMENT BY 1

The value to increment if N is positive.

or the value to decrement if N is negative.

1.4.

--MINVALUE N

E.g.

--NO MINVALUE

--MINVALUE 10

Minimum value of the sequence object

1.5.

--MAXVALUE N

E.g.

--NO MAXVALUE

--MAXVALUE 15

Maximum value of the sequence object

1.6.

E.g.

--NO CYCLE

--CYCLE

CYCLE means the sequence object will restart to min value,

when the max value (for incrementing sequence object) or

min value (for decrementing sequence object) is reached.

Default is NO CYCLE, which throws an error

when minimum or maximum value is met.

1.7.

CACHE Property

--CACHE

E.g.

--NO CACHE

--CACHE 10

Cache means the value is temporarily saved in the memory instead of disk.

Thus, CACHE improves performance.

By default, it is CACHE.

Microsoft change the default CACHE size without notice.

But we can still specify the CACHE size.

--CACHE 10

means to create the sequence object with 10 values cached.

When the 11th value is requested,

the next 10 values will be cached again.

---------------------------------------------------

2.

Sequence object V.S. Identity property

2.1.

Different 1.

2.1.1.

Identity property

is a table column property and

it can only be used in the Table column.

2.1.2.

Sequence object

is a user-defined database object.

It can be shared by multiple tables.

--------------------------

2.2.

Different 2.

2.2.1.

Identity property

will generate the next identity value

Only when using INSERT cluase to insert a row.

2.2.2.

--SELECT NEXT VALUE FOR SequenceObjName

Sequence object

can use NEXT VALUE FOR SequenceObjName to

generate the next sequence value.

It is not necessary to use INSERT cluase to insert a row.

--------------------------

2.3.

Different 3.

2.3.1.

Identity property

can not set Max and Min value.

The Max and Min value depend on the column data type.

2.3.2.

Sequence object

can set Max and Min value.

By default, the Max and Min value depend on

the Sequence object data type.

--------------------------

2.4.

Different 4.

2.4.1.

Identity property

has no CYCLE option

to automatically restart the identity values.

2.4.2.

Sequence object

can use CYCLE option

to automatically restart

when the max value (for incrementing sequence object) or

min value (for decrementing sequence object) is reached.

==========================================

1. SequenceObject

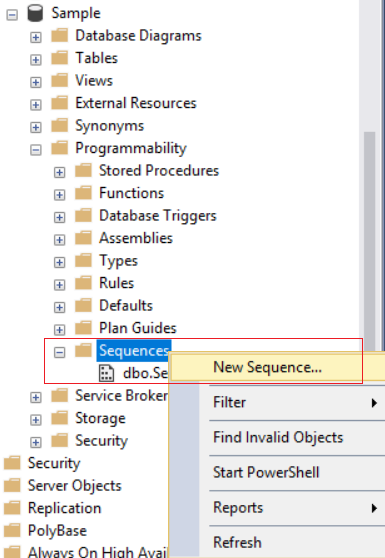
--=============================================================================

--T038\_01\_SequenceObject

--=============================================================================

1.0. Create Sequence Object by SSMS

Database --> Sequences --> New Sequence...



Graphical user interface, text, application, email

Description automatically generated

1.1. Sequence Object Basic concept.

--=============================================================================

--T038\_01\_01

--Sequence Object Basic concept.

---------------------------------------------------------------------------------------

--T038\_01\_01\_01

--Delete SEQUENCE object if exist, otherwise create it.

IF ( EXISTS ( SELECT    \*

              FROM      sys.sequences

              WHERE     name = 'SequenceObj' ) )

    BEGIN

        DROP SEQUENCE SequenceObj;

    END;

GO -- Run the previous command and begins new batch

CREATE SEQUENCE SequenceObj

AS INT

START WITH 1

INCREMENT BY 1;

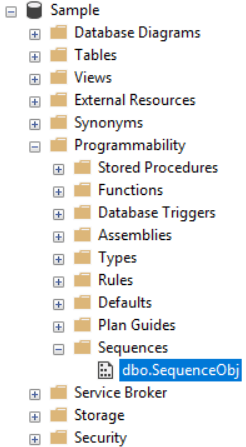
GO -- Run the previous command and begins new batch

/\*

Creating an Incrementing Sequence object

that starts with 1 and increments by 1.

\*/



---------------------------------------------------------------------------------------

--T038\_01\_01\_02

--Generating the Next Sequence Value.

--It will display 1,2,3

SELECT NEXT VALUE FOR

        SequenceObj;

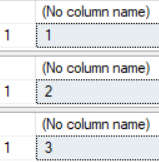
SELECT NEXT VALUE FOR

        SequenceObj;

SELECT NEXT VALUE FOR

        SequenceObj;

GO -- Run the previous command and begins new batch



---------------------------------------------------------------------------------------

--T038\_01\_01\_03

--Retrieving the current sequence value

--the current\_value will be 3

SELECT  \*

FROM    sys.sequences

WHERE   name = 'SequenceObj';

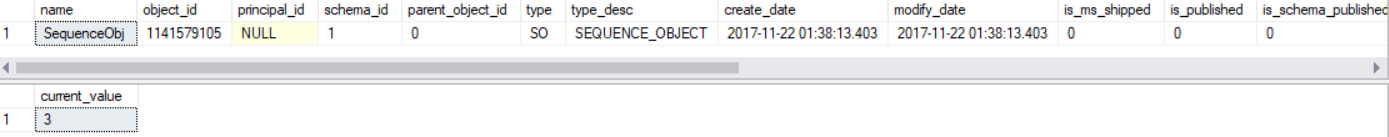
GO -- Run the previous command and begins new batch

SELECT  current\_value

FROM    sys.sequences

WHERE   name = 'SequenceObj';

GO -- Run the previous command and begins new batch



---------------------------------------------------------------------------------------

--T038\_01\_01\_04

--Manually reset the current\_value of the Sequence object to 1.

ALTER SEQUENCE SequenceObj RESTART WITH 1;

SELECT  current\_value

FROM    sys.sequences

WHERE   name = 'SequenceObj';

GO -- Run the previous command and begins new batch

A picture containing text

Description automatically generated

1.2. decrementing Sequence object

--======================================================================================

--T038\_01\_02

--decrementing Sequence object

---------------------------------------------------------------------------------------

--T038\_01\_02\_01

--Delete decrementing SEQUENCE object if exist, otherwise create it.

--it starts with 10 and decrements by 1

IF ( EXISTS ( SELECT    \*

              FROM      sys.sequences

              WHERE     name = 'SequenceObj2' ) )

    BEGIN

        DROP SEQUENCE SequenceObj2;

    END;

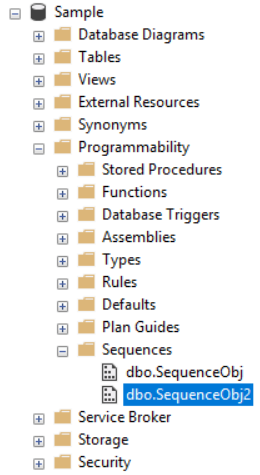
GO -- Run the previous command and begins new batch

CREATE SEQUENCE SequenceObj2

AS INT

START WITH 10

INCREMENT BY -1;



---------------------------------------------------------------------------------------

--T038\_01\_02\_02

--Generating the Next Sequence Value.

--It will display 10,9,8

SELECT NEXT VALUE FOR

        SequenceObj2;

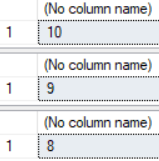
SELECT NEXT VALUE FOR

        SequenceObj2;

SELECT NEXT VALUE FOR

        SequenceObj2;

GO -- Run the previous command and begins new batch



1.3. MINVALUE and MAXVALUE of Sequence object

--======================================================================================

--T038\_01\_03

--MINVALUE and MAXVALUE of Sequence object

---------------------------------------------------------------------------------------

--T038\_01\_03\_01

--Delete decrementing SEQUENCE object if exist, otherwise create it.

--it starts with 14 and increments by 1

--MINVALUE is 10, MAXVALUE is 15, No CYCLE by default.

IF ( EXISTS ( SELECT    \*

              FROM      sys.sequences

              WHERE     name = 'SequenceObj3' ) )

    BEGIN

        DROP SEQUENCE SequenceObj3;

    END;

GO -- Run the previous command and begins new batch

CREATE SEQUENCE SequenceObj3

AS INT

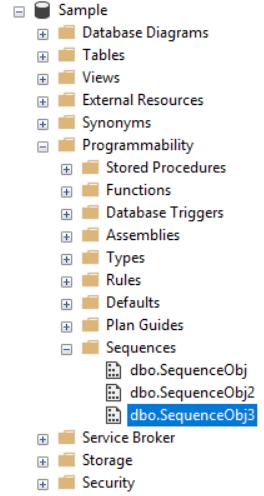
START WITH 14

INCREMENT BY 1

MINVALUE 10

MAXVALUE 15;

GO -- Run the previous command and begins new batch



---------------------------------------------------------------------------------------

--T038\_01\_03\_02

--Generating the Next Sequence Value.

SELECT NEXT VALUE FOR

        SequenceObj3;

SELECT NEXT VALUE FOR

        SequenceObj3;

SELECT NEXT VALUE FOR

        SequenceObj3;

GO -- Run the previous command and begins new batch

/\*

Generating the Next Sequence Value

It will display 14, 15, NULL.

The last statement will show the error message

because it reach its max value limit.

--Msg 11728, Level 16, State 1, Line 167

--The sequence object 'SequenceObj3' has reached

--its minimum or maximum value.

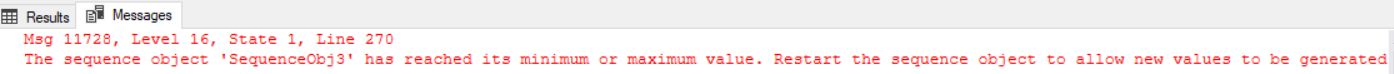
--Restart the sequence object to

--allow new values to be generated.

\*/

A picture containing timeline

Description automatically generated



1.4. MINVALUE, MAXVALUE and CYCLE of Sequence object

--======================================================================================

--T038\_01\_04

--MINVALUE, MAXVALUE and CYCLE of Sequence object

---------------------------------------------------------------------------------------

--T038\_01\_04\_01

--Delete decrementing SEQUENCE object if exist, otherwise create it.

--it starts with 14 and increments by 1

--MINVALUE is 10, MAXVALUE is 15, Set CYCLE property.

IF ( EXISTS ( SELECT    \*

              FROM      sys.sequences

              WHERE     name = 'SequenceObj4' ) )

    BEGIN

        DROP SEQUENCE SequenceObj4;

    END;

GO -- Run the previous command and begins new batch

CREATE SEQUENCE SequenceObj4

AS INT

START WITH 14

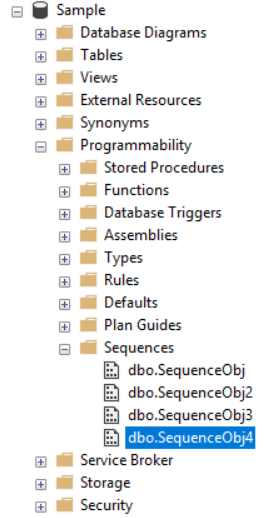
INCREMENT BY 1

MINVALUE 10

MAXVALUE 15

CYCLE;

GO -- Run the previous command and begins new batch



---------------------------------------------------------------------------------------

--T038\_01\_04\_02

--Generating the Next Sequence Value.

SELECT NEXT VALUE FOR

        SequenceObj4;

SELECT NEXT VALUE FOR

        SequenceObj4;

SELECT NEXT VALUE FOR

        SequenceObj4;

SELECT NEXT VALUE FOR

        SequenceObj4;

GO -- Run the previous command and begins new batch

/\*

Generating the Next Sequence Value

It will display 14, 15, 10, 11.

The 3rd statement will reach its max value limit.

Thus, CYCLE property will

reset it to min value which is 10.

\*/

Graphical user interface, table

Description automatically generated

1.5. MINVALUE, MAXVALUE and CYCLE of Sequence object

--======================================================================================

--T038\_01\_05

--MINVALUE, MAXVALUE and CYCLE of Sequence object

---------------------------------------------------------------------------------------

--T038\_01\_05\_01

--Delete decrementing SEQUENCE object if exist, otherwise create it.

--it starts with 14 and increments by 1

--MINVALUE is 10, MAXVALUE is 15, Set CYCLE, CACHE 10 property.

IF ( EXISTS ( SELECT    \*

              FROM      sys.sequences

              WHERE     name = 'SequenceObj5' ) )

    BEGIN

        DROP SEQUENCE SequenceObj5;

    END;

GO -- Run the previous command and begins new batch

CREATE SEQUENCE SequenceObj5

AS INT

START WITH 14

INCREMENT BY 1

MINVALUE 10

MAXVALUE 15

CYCLE

CACHE 10;

/\*

--CACHE 10

Cache means the value is temporarily saved in the memory instead of disk.

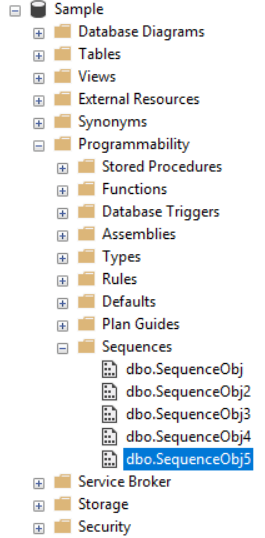
This improves performance.

CACHE 10 means to create the sequence object with 10 values cached.

When the 11th value is requested,

the next 10 values will be cached again.

\*/



1.6. Clean up

--==========================================

--Ch133\_06

--Clean up

--Delete SEQUENCE object if exist

IF ( EXISTS ( SELECT    \*

              FROM      sys.sequences

              WHERE     name = 'SequenceObj' ) )

    BEGIN

        DROP SEQUENCE SequenceObj;

    END;

GO -- Run the previous command and begins new batch

--Delete SEQUENCE object if exist

IF ( EXISTS ( SELECT    \*

              FROM      sys.sequences

              WHERE     name = 'SequenceObj2' ) )

    BEGIN

        DROP SEQUENCE SequenceObj2;

    END;

GO -- Run the previous command and begins new batch

--Delete SEQUENCE object if exist

IF ( EXISTS ( SELECT    \*

              FROM      sys.sequences

              WHERE     name = 'SequenceObj3' ) )

    BEGIN

        DROP SEQUENCE SequenceObj3;

    END;

GO -- Run the previous command and begins new batch

--Delete SEQUENCE object if exist

IF ( EXISTS ( SELECT    \*

              FROM      sys.sequences

              WHERE     name = 'SequenceObj4' ) )

    BEGIN

        DROP SEQUENCE SequenceObj4;

    END;

GO -- Run the previous command and begins new batch

IF ( EXISTS ( SELECT    \*

              FROM      sys.sequences

              WHERE     name = 'SequenceObj5' ) )

    BEGIN

        DROP SEQUENCE SequenceObj5;

    END;

GO -- Run the previous command and begins new batch

2. SequenceAndIdentity

--=============================================================================

--T038\_02\_SequenceAndIdentity

--=============================================================================

/\*

2.

Sequence object V.S. Identity property

2.1.

Different 1.

2.1.1.

Identity property

is a table column property and

it can only be used in the Table column.

2.1.2.

Sequence object

is a user-defined database object.

It can be shared by multiple tables.

--------------------------

2.2.

Different 2.

2.2.1.

Identity property

will generate the next identity value

Only when using INSERT cluase to insert a row.

2.2.2.

--SELECT NEXT VALUE FOR SequenceObjName

Sequence object

can use NEXT VALUE FOR SequenceObjName to

generate the next sequence value.

It is not necessary to use INSERT cluase to insert a row.

--------------------------

2.3.

Different 3.

2.3.1.

Identity property

can not set Max and Min value.

The Max and Min value depend on the column data type.

2.3.2.

Sequence object

can set Max and Min value.

By default, the Max and Min value depend on

the Sequence object data type.

--------------------------

2.4.

Different 4.

2.4.1.

Identity property

has no CYCLE option

to automatically restart the identity values.

2.4.2.

Sequence object

can use CYCLE option

to automatically restart

when the max value (for incrementing sequence object) or

min value (for decrementing sequence object) is reached.

\*/

2.1. Sequence object V.S. Identity property

--========================================================================

--T038\_02\_01

/\*

2.

Sequence object V.S. Identity property

2.1.

Different 1.

2.1.1.

Identity property

is a table column property and

it can only be used in the Table column.

2.1.2.

Sequence object

is a user-defined database object.

It can be shared by multiple tables.

2.

Sequence object V.S. Identity property

2.2.

Different 2.

2.2.1.

Identity property

will generate the next identity value

Only when using INSERT cluase to insert a row.

2.2.2.

--SELECT NEXT VALUE FOR SequenceObjName

Sequence object

can use NEXT VALUE FOR SequenceObjName to

generate the next sequence value.

It is not necessary to use INSERT cluase to insert a row.

\*/

----------------------------------------------------------

--T038\_02\_01\_01

/\*

1.

Different 1.

Identity property

is a table column property and

it can only be used in the Table column.

2.

Different 2.

Identity property

will generate the next identity value

Only when using INSERT cluase to insert a row.

\*/

--If Table exists then DROP it

IF ( EXISTS ( SELECT    \*

              FROM      INFORMATION\_SCHEMA.TABLES

              WHERE     TABLE\_NAME = 'Person1' ) )

    BEGIN

        TRUNCATE TABLE dbo.Person1;

        DROP TABLE Person1;

    END;

GO -- Run the previous command and begins new batch

CREATE TABLE Person1

(

  Id INT PRIMARY KEY

         IDENTITY(1, 1) ,

  [Name] NVARCHAR(50) ,

);

GO -- Run the previous command and begins new batch

INSERT  INTO Person1

VALUES  ( 'Name01' );

INSERT  INTO Person1

VALUES  ( 'Name02' );

INSERT  INTO Person1

VALUES  ( 'Name03' );

GO -- Run the previous command and begins new batch

SELECT  \*

FROM    Person1;

GO -- Run the previous command and begins new batch

Table

Description automatically generated

----------------------------------------------------------

--T038\_02\_01\_02

/\*

1.

Different 1.

Sequence object

is a user-defined database object.

It can be shared by multiple tables.

2.

Different 2.

Sequence object

can use NEXT VALUE FOR SequenceObjName to

generate the next sequence value.

It is not necessary to use INSERT cluase to insert a row.

\*/

--Delete SEQUENCE object if exist

IF ( EXISTS ( SELECT    \*

              FROM      sys.sequences

              WHERE     name = 'SequenceObjA' ) )

    BEGIN

        DROP SEQUENCE SequenceObjA;

    END;

GO -- Run the previous command and begins new batch

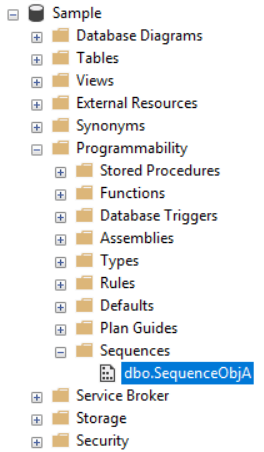
CREATE SEQUENCE SequenceObjA

AS INT

START WITH 1

INCREMENT BY 1;

GO -- Run the previous command and begins new batch



---------------------------

--If Table exists then DROP it

IF ( EXISTS ( SELECT    \*

              FROM      INFORMATION\_SCHEMA.TABLES

              WHERE     TABLE\_NAME = 'Person2' ) )

    BEGIN

        TRUNCATE TABLE dbo.Person2;

        DROP TABLE Person2;

    END;

GO -- Run the previous command and begins new batch

CREATE TABLE Person2

(

  Id INT PRIMARY KEY ,

  [Name] NVARCHAR(50) ,

);

GO -- Run the previous command and begins new batch

---------------------------

--If Table exists then DROP it

IF ( EXISTS ( SELECT    \*

              FROM      INFORMATION\_SCHEMA.TABLES

              WHERE     TABLE\_NAME = 'Person3' ) )

    BEGIN

        TRUNCATE TABLE dbo.Person3;

        DROP TABLE Person3;

    END;

GO -- Run the previous command and begins new batch

CREATE TABLE Person3

(

  Id INT PRIMARY KEY ,

  [Name] NVARCHAR(50) ,

);

GO -- Run the previous command and begins new batch

---------------------------

INSERT  INTO Person3

VALUES  ( NEXT VALUE FOR SequenceObjA, 'P3Name01' );

INSERT  INTO Person3

VALUES  ( NEXT VALUE FOR SequenceObjA, 'P3Name02' );

GO -- Run the previous command and begins new batch

--NEXT VALUE FOR SequenceObjA will be 1,2

INSERT  INTO Person2

VALUES  ( NEXT VALUE FOR SequenceObjA, 'P2Name01' );

INSERT  INTO Person2

VALUES  ( NEXT VALUE FOR SequenceObjA, 'P2Name02' );

INSERT  INTO Person2

VALUES  ( NEXT VALUE FOR SequenceObjA, 'P2Name03' );

GO -- Run the previous command and begins new batch

--NEXT VALUE FOR SequenceObjA will be 3,4,5

---------------------------

SELECT  \*

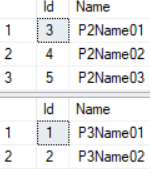
FROM    Person2;

GO -- Run the previous command and begins new batch

SELECT  \*

FROM    Person3;

GO -- Run the previous command and begins new batch



---------------------------

SELECT NEXT VALUE FOR

        SequenceObjA;

SELECT NEXT VALUE FOR

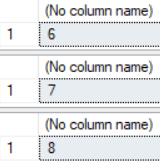
        SequenceObjA;

SELECT NEXT VALUE FOR

        SequenceObjA;

GO -- Run the previous command and begins new batch

--NEXT VALUE FOR SequenceObjA will be 6,7,8



2.2. Sequence object V.S. Identity property

--========================================================================

--T038\_02\_02

/\*

Sequence object V.S. Identity property

1.

Different 3.

1.1.

Identity property

can not set Max and Min value.

The Max and Min value depend on the column data type.

1.2.

Sequence object

can set Max and Min value.

By default, the Max and Min value depend on

the Sequence object data type.

--------------------------

2.

Different 4.

2.1.

Identity property

has no CYCLE option

to automatically restart the identity values.

2.2.

Sequence object

can use CYCLE option

to automatically restart

when the max value (for incrementing sequence object) or

min value (for decrementing sequence object) is reached.

\*/

---------------------------------------------------------------------------------------

--T038\_02\_02\_01

--MINVALUE, MAXVALUE and CYCLE of Sequence object

--Delete decrementing SEQUENCE object if exist, otherwise create it.

--it starts with 14 and increments by 1

--MINVALUE is 10, MAXVALUE is 15, Set CYCLE property.

IF ( EXISTS ( SELECT    \*

              FROM      sys.sequences

              WHERE     name = 'SequenceObj4' ) )

    BEGIN

        DROP SEQUENCE SequenceObj4;

    END;

GO -- Run the previous command and begins new batch

CREATE SEQUENCE SequenceObj4

AS INT

START WITH 14

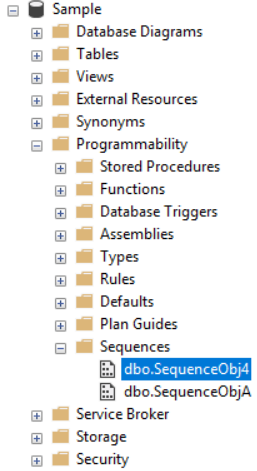
INCREMENT BY 1

MINVALUE 10

MAXVALUE 15

CYCLE;

GO -- Run the previous command and begins new batch



---------------------------------------------------------------------------------------

--T038\_02\_02\_02

--Generating the Next Sequence Value.

SELECT NEXT VALUE FOR

        SequenceObj4;

SELECT NEXT VALUE FOR

        SequenceObj4;

SELECT NEXT VALUE FOR

        SequenceObj4;

SELECT NEXT VALUE FOR

        SequenceObj4;

GO -- Run the previous command and begins new batch

/\*

Generating the Next Sequence Value

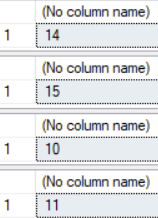
It will display 14, 15, 10, 11.

The 3rd statement will reach its max value limit.

Thus, CYCLE property will

reset it to min value which is 10.

\*/



2.3. Clean up

--========================================================================

--T038\_02\_03

--Clean up

--If Table exists then DROP it

IF ( EXISTS ( SELECT    \*

              FROM      INFORMATION\_SCHEMA.TABLES

              WHERE     TABLE\_NAME = 'Person1' ) )

    BEGIN

        TRUNCATE TABLE dbo.Person1;

        DROP TABLE Person1;

    END;

GO -- Run the previous command and begins new batch

--Delete SEQUENCE object if exist

IF ( EXISTS ( SELECT    \*

              FROM      sys.sequences

              WHERE     name = 'SequenceObjA' ) )

    BEGIN

        DROP SEQUENCE SequenceObjA;

    END;

GO -- Run the previous command and begins new batch

--If Table exists then DROP it

IF ( EXISTS ( SELECT    \*

              FROM      INFORMATION\_SCHEMA.TABLES

              WHERE     TABLE\_NAME = 'Person2' ) )

    BEGIN

        TRUNCATE TABLE dbo.Person2;

        DROP TABLE Person2;

    END;

GO -- Run the previous command and begins new batch

--If Table exists then DROP it

IF ( EXISTS ( SELECT    \*

              FROM      INFORMATION\_SCHEMA.TABLES

              WHERE     TABLE\_NAME = 'Person3' ) )

    BEGIN

        TRUNCATE TABLE dbo.Person3;

        DROP TABLE Person3;

    END;

GO -- Run the previous command and begins new batch

IF ( EXISTS ( SELECT    \*

              FROM      sys.sequences

              WHERE     name = 'SequenceObj4' ) )

    BEGIN

        DROP SEQUENCE SequenceObj4;

    END;

GO -- Run the previous command and begins new batch