

Przetestowanie ogólnodostępnej bazy danych AdventureWorks2014:

1. Dodawanie nowego użytkownika do tabeli Person.Person:

Przed dodaniem wpisu do tabeli Person.Person należy najpierw dodać wpis do tabeli Person.BusinessEntity, która zawiera PrimaryKey (PK) na kolumnie: BusinessEntityID

Tabela Person.Person ma ustawioną kolumnę BusinessEntityID jako PrimaryKey (PK) i ForeignKey(FK), dlatego powyższe jest wymagane.

Struktura tabeli Person.BusinessEntity:

```
CREATE TABLE [Person].[BusinessEntity](  
    [BusinessEntityID] [int] IDENTITY(1,1) NOT FOR REPLICATION NOT NULL,  
    [rowguid] [uniqueidentifier] ROWGUIDCOL NOT NULL CONSTRAINT [DF_BusinessEntity_rowguid] DEFAULT (newid()),  
    [ModifiedDate] [datetime] NOT NULL CONSTRAINT [DF_BusinessEntity_ModifiedDate] DEFAULT (getdate()),  
    CONSTRAINT [PK_BusinessEntity_BusinessEntityID] PRIMARY KEY CLUSTERED
```

Dodaję wpis do tabeli Person.BusinessEntity:

```
INSERT INTO Person.BusinessEntity  
DEFAULT VALUES
```

2. Następnie sprawdzam, jaki ID został nadany w powyższym kroku:

```
SELECT MAX(BUSINESSENTITYID)  
FROM Person.BusinessEntity
```

Results		Messages	
		(No column name)	
1	20783		

3. Teraz mogę dodać nowego użytkownika, podając najnowszy BusinessEntityID i pozostałe wymanagowane wartości (kolumny, które w opisie mają "not null"):

Person.Person
Columns
BusinessEntityID (PK, FK, int, not null)
PersonType (nchar(2), not null)
NameStyle (NameStyle(bit), not null)
Title (nvarchar(8), null)
FirstName (Name(nvarchar(50)), not null)
MiddleName (Name(nvarchar(50)), null)
LastName (Name(nvarchar(50)), not null)
Suffix (nvarchar(10), null)
EmailPromotion (int, not null)
AdditionalContactInfo (XML(Person.AdditionalContactInfoSchemaCollection), null)
Demographics (XML(Person.IndividualSurveySchemaCollection), null)
rowguid (uniqueidentifier, not null)
ModifiedDate (datetime, not null)

```
INSERT INTO Person.Person (BusinessEntityID ,PersonType, NameStyle, FirstName,
LastName, EmailPromotion, ModifiedDate)
VALUES ( 20783,'IN', 0, 'Dylan', 'Miller', 2, GETUTCDATE())
```

Powyższe polecenie wykonałam kilka razy, najpierw generując kolejne BusinessEntityID:

Results		Messages										
	BusinessEntityID	Person Type	NameStyle	Title	FirstName	MiddleName	LastName	Suffix	EmailPromotion	AdditionalContactInfo	Demographics	rowguid
1	20783	IN	0	NULL	Dylan	NULL	Miller	NULL	2	NULL	<IndividualSurvey xmlns="http://schemas.microso...	71E3D0BD-A3DF-4D92-9766-05/
2	20782	IN	0	NULL	Dylan	NULL	Miller	NULL	2	NULL	<IndividualSurvey xmlns="http://schemas.microso...	9117468E-238B-46F6-AAE-6B0
3	20781	IN	0	NULL	Dylan	NULL	Miller	NULL	2	NULL	<IndividualSurvey xmlns="http://schemas.microso...	198E790A-98B1-40CF-B7A4-1DF
4	20780	IN	0	NULL	Dylan	NULL	Miller	NULL	2	NULL	<IndividualSurvey xmlns="http://schemas.microso...	1487C998-7207-427D-A5AD-2F9
5	20779	IN	0	NULL	Dylan	NULL	Miller	NULL	2	NULL	<IndividualSurvey xmlns="http://schemas.microso...	A5229801-3180-40DD-9DD3-335
6	18308	IN	0	NULL	Dylan	NULL	Miller	NULL	0	NULL	<IndividualSurvey xmlns="http://schemas.microso...	E2A5ED0D-AE5E-44A7-AA0-B5
7	2287	GC	0	NULL	Dylan	NULL	Miller	NULL	1	NULL	<IndividualSurvey xmlns="http://schemas.microso...	8CE9EC23-22D5-4657-83E2-F84
8	1477	SC	0	Mr.	Dylan	NULL	Miller	NULL	1	NULL	<IndividualSurvey xmlns="http://schemas.microso...	B6A6DFE5-5AF5-4052-A88A-0CA
9	7	EM	0	NULL	Dylan	A	Miller	NULL	2	NULL	<IndividualSurvey xmlns="http://schemas.microso...	C45E8AB8-01BE-4B76-B215-820

4. Sprawdzam, czy użytkownik został dodany do wybranej tabeli (wiele wpisów dla tego samego imienia i nazwiska):

```
SELECT *  
FROM Person.Person  
WHERE LastName = 'Miller'  
AND FirstName = 'Dylan'  
ORDER BY BusinessEntityID DESC
```

	BusinessEntityID	FirstName	LastName	PhoneNumber
1	20779	Dylan	Miller	NULL
2	20780	Dylan	Miller	NULL
3	20781	Dylan	Miller	NULL
4	20782	Dylan	Miller	NULL
5	20783	Dylan	Miller	NULL

5. Wiedząc, że istnieje tabela Person.PersonPhone, sprawdzam, czy dodani przeze mnie użytkownicy (lub istniejący wcześniej) nie mają przypisanego nr telefonu (nie zdefiniowałam dla nich nr telefonu przy dodawaniu):

```
SELECT Person.Person.BusinessEntityID, FirstName, LastName, PhoneNumber  
FROM Person.Person  
LEFT JOIN Person.PersonPhone  
ON Person.BusinessEntityID = PersonPhone.BusinessEntityID  
WHERE Person.PersonPhone.PhoneNumber IS NULL
```

	BusinessEntityID	FirstName	LastName	PhoneNumber
1	20779	Dylan	Miller	NULL
2	20780	Dylan	Miller	NULL
3	20781	Dylan	Miller	NULL
4	20782	Dylan	Miller	NULL
5	20783	Dylan	Miller	NULL