--procedurat

--CRUD ose RAPORTE ose Importe OSE data cleaning

--Na ka kerkuar programuesi qe te krijojme Stored Proc per

--Create (insert), Retrieve one, Retrieve list, Update, Delete

--per Objektin ProductSubCategory

--Create Category

alter procedure Production.usp\_rProductCategory\_Insert

(@name nvarchar(50), @newCategoryId int output)

as

BEGIN

INSERT INTO [Production].[ProductCategory]

([Name]

,[rowguid]

,[ModifiedDate])

VALUES

(@name

,newid()

,sysdatetime())

set @newCategoryId = @@IDENTITY

END

GO

--Thonjeza teke brenda stringut

select 'A''aron'

--Create ProductSubCategory

create proc Production.usp\_rProductSubcategory\_InsertWithNewCat(@ProductCategoryId int,

@name nvarchar(50)

,@NewProductSubCategoryId int output)

as

BEGIN

declare @newCategoryId int = @ProductCategoryId

if not exists (select \* from production.ProductCategory

where ProductCategoryID = @ProductCategoryId)

BEGIN

INSERT INTO [Production].[ProductCategory]

([Name]

,[rowguid]

,[ModifiedDate])

VALUES

(@name

,newid()

,sysdatetime())

set @newCategoryId = @@identity

end

INSERT INTO [Production].[ProductSubcategory]

([ProductCategoryID]

,[Name]

,[rowguid]

,[ModifiedDate])

VALUES

(@newCategoryId

,@name

,newid()

,getdate())

set @NewProductSubCategoryId = @@IDENTITY

END

GO

alter proc Production.usp\_rProductSubcategory\_InsertWithNewCatFromStoredProc(@ProductCategoryId int,

@name nvarchar(50)

,@NewProductSubCategoryId int output)

as

BEGIN

declare @newCategoryId int = @ProductCategoryId

if not exists (select \* from production.ProductCategory

where ProductCategoryID = @ProductCategoryId)

BEGIN

exec Production.usp\_rProductCategory\_Insert @name = @name, @newCategoryId = @newCategoryId output

--print @newCategoryId

end

INSERT INTO [Production].[ProductSubcategory]

([ProductCategoryID]

,[Name]

,[rowguid]

,[ModifiedDate])

VALUES

(@newCategoryId

,@name

,newid()

,getdate())

set @NewProductSubCategoryId = @@IDENTITY

END

GO

--Retrieve one

create proc Production.usp\_rProductSubcategory\_SelectOne(@ProductSubcategoryID int)

as

begin

select psc.ProductSubcategoryID,psc.ProductCategoryID

,psc.Name,psc.ModifiedDate,psc.rowguid

from Production.ProductSubcategory psc

where psc.ProductSubcategoryID = @ProductSubcategoryID

END

--Retrieve list,

create proc Production.usp\_rProductSubcategory\_SelectList

as

begin

select psc.ProductSubcategoryID,psc.ProductCategoryID

,psc.Name,psc.ModifiedDate,psc.rowguid

from Production.ProductSubcategory psc

END

--deklarim dhe inicializim

declare @i int = 0

--2

alter proc Production.usp\_rProductSubcategory\_SelectOneOrAll(@ProductSubcategoryID int = 0)

as

begin

select psc.ProductSubcategoryID,psc.ProductCategoryID

,psc.Name,psc.ModifiedDate,psc.rowguid

from Production.ProductSubcategory psc

where psc.ProductSubcategoryID = @ProductSubcategoryID or @ProductSubcategoryID = 0

END

--seanca #18 vazhdon ketu.....

create proc [Production].[usp\_rProductSubcategory\_InsertTryCatch](@ProductCategoryId int,

@name nvarchar(50)

,@NewProductSubCategoryId int output)

as

BEGIN

-- configurimi xact\_abort perdoret PA TRY CATCH BLOCK.

set xact\_abort on;

begin try

begin transaction

INSERT INTO [Production].[ProductSubcategory]

([ProductCategoryID]

,[Name]

,[rowguid]

,[ModifiedDate])

VALUES

(@ProductCategoryId

,@name

,newid()

,getdate())

set @NewProductSubCategoryId = @@IDENTITY

--statement 2

--statement 3

commit transaction

end try

begin catch

--KETU: te keni nje kode per te loguar problemin

rollback transaction;

throw 50008,'Error ne Insert te Sub Kategorise',0

end catch

END

--update procedure

create procedure Production.usp\_rProductSubcategory\_Update

(@ProductSubcategoryID int, @Name nvarchar(50), @ProductCategoryID int )

AS

BEGIN

set xact\_abort on;

update Production.ProductSubcategory

set [ProductCategoryID] = @ProductCategoryID,[Name] = @Name

,[ModifiedDate] = getdate()

where [ProductSubcategoryID] = @ProductSubcategoryID

END

--delete Procedure

alter procedure Production.usp\_rProductSubcategory\_delete(

@ProductSubcategoryID int

) as

begin

set xact\_abort on;

--nese ka produkte ne kete SubCategory nuk duhet te fshihet SubCategoria

if not exists( select \* from production.product

where ProductSubcategoryID = @ProductSubcategoryID)

begin

delete from production.ProductSubcategory

where ProductSubcategoryID = @ProductSubcategoryID

end

end

--krijoni nje delete procedure nga tabela e produkteve ne form update isDeleted column

--isDeleted bit, deleteDate smalldatetime, deletedBy nvarchar(50)

--deletedBy = SYSTEM\_USER

create procedure Production.usp\_rProduct\_delete(

@productId int)

as

begin

set xact\_abort on;

update production.Product set

isDeleted = 1,

deleteDate = getdate(),

deleteBy = SYSTEM\_USER

where ProductID = @productId

end

--Ndryshoni INSERT PROCEDURE TE SUBCATEGORI-VE QE:

--nese nuk ekziton kategoria ta bej insert automatikisht,

--kategoria te marre te njejtin emer si subcategoria e re

--dhe pastaj insert subCategorine si normalisht nen kete kategori te re

USE [AdventureWorks2019]

GO

/\*\*\*\*\*\* Object: StoredProcedure [Production].[usp\_rProductSubcategory\_Insert] Script Date: 12/26/2022 8:23:08 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

create proc [Production].[usp\_rProductSubcategory\_Insert2](@ProductCategoryId int,

@name nvarchar(50)

,@NewProductSubCategoryId int output)

as

BEGIN

if exists (select \* from production.ProductCategory

where ProductCategoryID = @ProductCategoryId)

BEGIN

INSERT INTO [Production].[ProductSubcategory]

([ProductCategoryID]

,[Name]

,[rowguid]

,[ModifiedDate])

VALUES

(@ProductCategoryId

,@name

,newid()

,getdate())

set @NewProductSubCategoryId = @@IDENTITY

END

END

GO

---TRIGGERS--- dy tipe te ndryshme

--AFTER : runs after user's sql statement

--INSTEAD OF: runs instead of user's sql statement

CREATE TRIGGER Production.tr\_Product on Production.Product

instead of delete, insert, update

as

begin

update production.Product set

isDeleted = 1,

deleteDate = getdate(),

deleteBy = SYSTEM\_USER

where ProductID in (select ProductID from deleted)

end

--test trigger

select \* from production.Product

delete from production.Product where productid = 1009

---Execute procedurat e krijuara

exec Production.usp\_rProductCategory\_Insert @name = 'Parafango'

select \* from Production.ProductCategory

go

--test

exec Production.usp\_rProductSubcategory\_Insert @ProductCategoryId = 5, @name = 'drita te tejdukshme fosforeshente'

select \* from Production.ProductSubcategory

declare @NewProdSubCatId int

exec Production.usp\_rProductSubcategory\_Insert @ProductCategoryId = 500, @name = 'kapertona te holle fare2', @NewProductSubCategoryId = @NewProdSubCatId OUTPUT

select @NewProdSubCatId

--Retrieve on

exec Production.usp\_rProductSubcategory\_SelectOne @ProductSubCategoryId = 20

--retrieve list

Production.usp\_rProductSubcategory\_SelectList

--retrieve edhe one edhe list

exec Production.usp\_rProductSubcategory\_SelectOneOrAll

--update table subcategories

exec Production.usp\_rProductSubcategory\_Update @ProductSubcategoryId = 125,@name = 'e-devices',@ProductCategoryId = 4

--delete subcategory

exec Production.usp\_rProductSubcategory\_delete @ProductSubcategoryId = 1

--delete product

exec Production.usp\_rProduct\_delete @productId = 400

--Detyre per ju

--Krijoni Update dhe Delete procedure

select \* from Production.Product

where isdeleted is null

--new sub ne new cat

declare @newSubCatId int

exec Production.usp\_rProductSubcategory\_InsertWithNewCatFromStoredProc @ProductCategoryId = 9876, @name = 'Fire&Ice',@NewProductSubCategoryId = @newSubCatId

print @newSubCatId

select \* from Production.ProductCategory

select \* from Production.ProductSubcategory