Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра программного обеспечения информационных технологий

Дисциплина: Базы Данных (БД)

ОТЧЕТ

по лабораторной работе №4

Выполнил

студент: гр. 651006 Шепелевич

Проверил: Салей

Минск 2018

**Задачи**:

**№1**

**Подзадача 1.**

Запрос:

CREATE TABLE Sales.CurrencyHst (

ID INT IDENTITY(1, 1) PRIMARY KEY,

Action NVARCHAR(20) CHECK(Action IN ('INSERT', 'UPDATE', 'DELETE')),

ModifiedDate DATETIME NOT NULL,

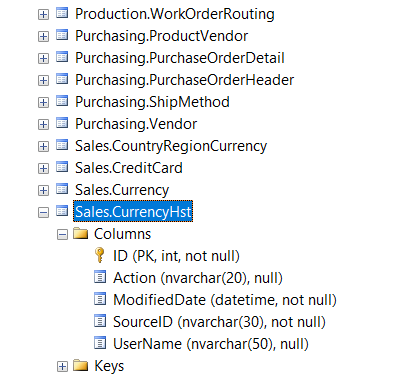
SourceID NVARCHAR(30) NOT NULL,

UserName NVARCHAR(50)

);

GO

Результат :



**Подзадача 2.**

Запрос:

CREATE TRIGGER currencyAfterInsert ON Sales.Currency

FOR INSERT

AS

DECLARE @sourceID NVARCHAR(30);

SELECT @sourceID = inserted.Name FROM inserted;

INSERT INTO Sales.CurrencyHst(Action, ModifiedDate, SourceID, UserName)

VALUES('INSERT', GETDATE(), @sourceID, CURRENT\_USER);

GO

CREATE TRIGGER currencyAfterUpdate ON Sales.Currency

FOR UPDATE

AS

DECLARE @sourceID NVARCHAR(30);

SELECT @sourceID = inserted.Name FROM inserted;

INSERT INTO Sales.CurrencyHst(Action, ModifiedDate, SourceID, UserName)

VALUES('UPDATE', GETDATE(), @sourceID, CURRENT\_USER);

GO

CREATE TRIGGER currencyAfterDelete ON Sales.Currency

FOR DELETE

AS

DECLARE @sourceID NVARCHAR(30);

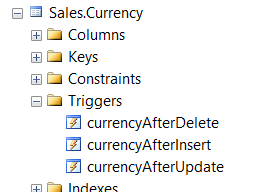
SELECT @sourceID = deleted.Name FROM deleted;

INSERT INTO Sales.CurrencyHst(Action, ModifiedDate, SourceID, UserName)

VALUES('DELETE', GETDATE(), @sourceID, CURRENT\_USER);

GO

Результат :



**Подзадача 3.**

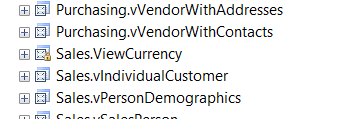
Запрос:

CREATE VIEW Sales.ViewCurrency

WITH ENCRYPTION

AS SELECT \* FROM Sales.Currency;

Результат :



**Подзадача 4.**

Запрос:

INSERT INTO Sales.ViewCurrency(CurrencyCode, ModifiedDate, Name)

VALUES('TST', GETDATE(), 'Hello!!!')

GO

UPDATE Sales.ViewCurrency

SET ModifiedDate= GETDATE(), Name= 'Hello!!!123'

WHERE CurrencyCode='TST'

GO

DELETE FROM Sales.ViewCurrency

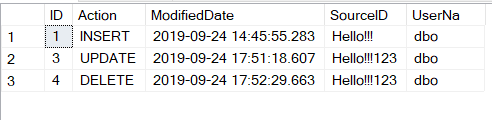
WHERE CurrencyCode='TST'

GO

SELECT \* FROM Sales.CurrencyHst;

GO

Результат :



**№2**

**1)**

CREATE VIEW Sales.ViewCurrency2

WITH SCHEMABINDING

AS

SELECT

currencyRate.CurrencyRateID,

currencyRate.CurrencyRateDate,

currencyRate.FromCurrencyCode,

currency.Name,

currency.CurrencyCode,

currencyRate.AverageRate,

currencyRate.EndOfDayRate

FROM Sales.Currency AS currency

INNER JOIN Sales.CurrencyRate AS currencyRate

ON currency.CurrencyCode = currencyRate.ToCurrencyCode

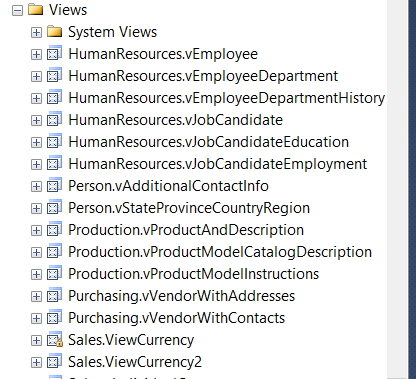
GO

CREATE UNIQUE CLUSTERED INDEX IX\_CurrencyRateID

ON Sales.ViewCurrency2 (CurrencyRateID)

GO

Результат :



**2)**

-- 2

CREATE TRIGGER InsteadViewCurrency2Trigger ON Sales.ViewCurrency2

INSTEAD OF INSERT, UPDATE, DELETE

AS

BEGIN

DECLARE @currencyCode NVARCHAR(50);

--delete

IF NOT EXISTS (SELECT \* FROM inserted)

BEGIN

SELECT @currencyCode = deleted.CurrencyCode FROM deleted;

DELETE

FROM Sales.CurrencyRate

WHERE ToCurrencyCode = @currencyCode;

DELETE

FROM Sales.Currency

WHERE CurrencyCode = @currencyCode

END;

--insert

ELSE IF NOT EXISTS (SELECT \* FROM deleted)

BEGIN

IF NOT EXISTS (

SELECT \*

FROM Sales.Currency AS sc

JOIN inserted ON inserted.CurrencyCode = sc.CurrencyCode)

BEGIN

INSERT INTO Sales.Currency (

CurrencyCode,

Name,

ModifiedDate)

SELECT

CurrencyCode,

Name,

GETDATE()

FROM inserted

END

ELSE

UPDATE

Sales.Currency

SET

Name = inserted.Name,

ModifiedDate = GETDATE()

FROM

inserted

WHERE

Currency.CurrencyCode = inserted.CurrencyCode

INSERT INTO Sales.CurrencyRate(

CurrencyRateDate,

FromCurrencyCode,

ToCurrencyCode,

AverageRate,

EndOfDayRate,

ModifiedDate)

SELECT

CurrencyRateDate,

FromCurrencyCode,

CurrencyCode,

AverageRate,

EndOfDayRate,

GETDATE()

FROM inserted

END;

--update

ELSE

BEGIN

UPDATE Sales.Currency

SET

Name = inserted.Name,

ModifiedDate = GETDATE()

FROM Sales.Currency AS currencies

JOIN inserted ON inserted.CurrencyCode = currencies.CurrencyCode

UPDATE Sales.CurrencyRate

SET

CurrencyRateDate= inserted.CurrencyRateDate,

AverageRate= inserted.AverageRate,

EndOfDayRate= inserted.EndOfDayRate,

ModifiedDate= GETDATE()

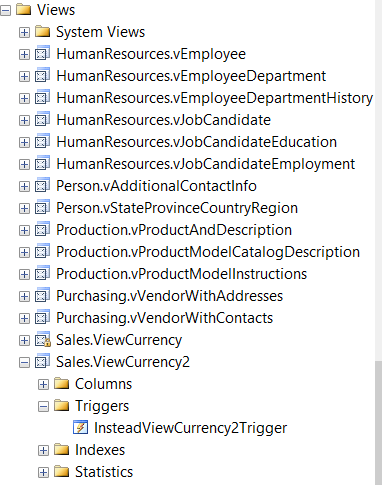
FROM Sales.CurrencyRate AS currencyRates

JOIN inserted ON inserted.CurrencyRateID = currencyRates.CurrencyRateID

END;

END;

Результат :



**3)**

--3

INSERT INTO Sales.ViewCurrency2(

CurrencyRateDate,

FromCurrencyCode,

CurrencyCode,

Name,

AverageRate,

EndOfDayRate)

VALUES(GETDATE(), 'USD','TST', 'TEST1', 2.01, 1.65)

GO

UPDATE Sales.ViewCurrency2

SET

Name='TERT',

AverageRate = 2.33,

EndOfDayRate=3.1

WHERE CurrencyCode = 'TST'

GO

DELETE

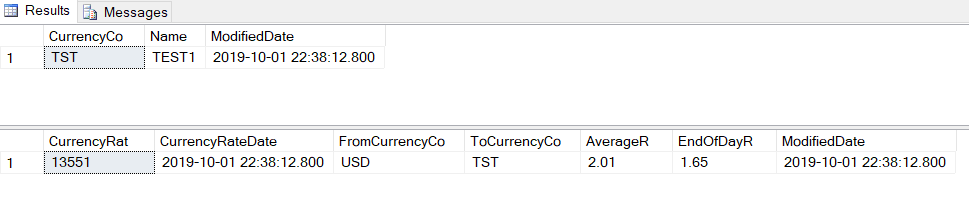
FROM Sales.ViewCurrency2

WHERE CurrencyCode = 'TST'

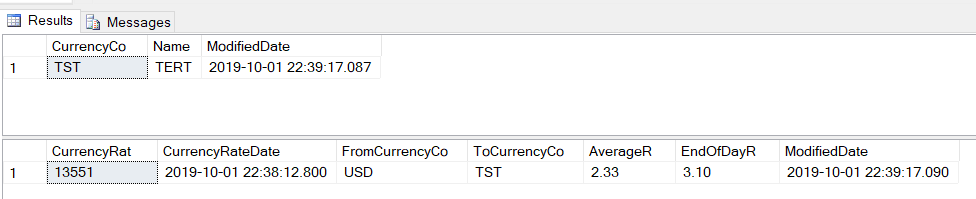
GO

Результат :

INSERT:



UPDATE:



DELETE:

