МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ

«ГОМЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ П. О. СУХОГО»

Факультет автоматизированных и информационных систем

Кафедра «Информационные технологии»

дисциплина «Разработка приложений баз данных для информационных систем»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ

«Разработка серверной части информационной системы   
в СУБД MS SQL Server»

Вариант № 15

Выполнил:

студент группы ИТП-31, Солодков М.А

Принял:

доцент Асенчик О.Д.

Гомель 2019

**Цель работы:**  Разработать серверную часть клиент-серверной информационной системы основанной на базе данных в заданной предметной области средствами СУБД MS SQL Server.

**Ход работы и результаты.**

Код на языке T-SQL для создания базы данных и ее таблиц:

Создание таблицы Units:

use master

go

if DB\_ID (N'Company') is null

create database Company

go

use [Company]

if OBJECT\_ID ('Units','U') is not null

drop table Units

go

create table Units

(UnitId int IDENTITY(1,1) not null primary key, FullName nvarchar(20),CountEmployees int)

go

Создание таблицы UnitValuationFact:

use master

go

if DB\_ID (N'Company') is null

create database Company

go

use [Company]

if OBJECT\_ID ('UnitsValuationFact','U') is not null

drop table UnitsValuationFact

go

create table UnitsValuationFact

(UnitValuationFactId int IDENTITY(1,1) not null primary key, FullName nvarchar(20), Income money, Cost money

,UnitId int,CONSTRAINT FK\_UnitsValuationFact\_To\_Units FOREIGN KEY (UnitId) REFERENCES Units (UnitId) ON DELETE CASCADE

ON UPDATE CASCADE);

Создание таблицы UnitValuationPlans:

use master

go

if DB\_ID (N'Company') is null

create database Company

go

use [Company]

if OBJECT\_ID ('UnitsValuationPlans','U') is not null

drop table UnitsValuationPlans

go

create table UnitsValuationPlans

(UnitValuationPlanId int IDENTITY(1,1) not null primary key, FullName nvarchar(20),Income money, Cost money,

UnitValuationFactId int, CONSTRAINT FK\_UnitsValuationPlans\_To\_UnitsValuationFact FOREIGN KEY (UnitValuationFactId) REFERENCES UnitsValuationFact (UnitValuationFactId) ON DELETE CASCADE

ON UPDATE CASCADE);

go

Создание таблицы Employees:

use master

go

if DB\_ID (N'Company') is null

create database Company

go

use [Company]

if OBJECT\_ID ('Employees','U') is not null

drop table Employees

go

create table Employees

(EmployeeId int IDENTITY(1,1) not null primary key, FullName nvarchar(20), Solution money,Profit money, Age int, UnitId int,

CONSTRAINT FK\_Employees\_To\_Units FOREIGN KEY (UnitId) REFERENCES Units (UnitId));

Go

Создание таблицы EmployeesFact:

use master

go

if DB\_ID (N'Company') is null

create database Company

go

use [Company]

if OBJECT\_ID ('EmployeeFact','U') is not null

drop table EmployeeFact

go

create table EmployeeFact

(EmployeeFactId int IDENTITY(1,1) not null primary key, FullName nvarchar(20), Quarter int, Year int, EmployeeId int, ProfitYear money, ProfitQuarter money,

CONSTRAINT FK\_EmployeeFact\_To\_Employees FOREIGN KEY (EmployeeId) REFERENCES Employees (EmployeeId));

Go

Создание таблицы EmployeesPlans:

use master

go

if DB\_ID (N'Company') is null

create database Company

go

use [Company]

if OBJECT\_ID ('EmployeePlans','U') is not null

drop table EmployeePlans

go

create table EmployeePlans

(EmployeePlanId int IDENTITY(1,1) not null primary key, FullName nvarchar(20), Quarter int, Year int, EmployeeFactId int, ProfitQuarter money, ProfitYear money,

CONSTRAINT FK\_EmployeePlans\_To\_EmployeeFact FOREIGN KEY (EmployeeFactId) REFERENCES EmployeeFact (EmployeeFactId));

Go

Создание таблицы ProgressEmployees:

use master

go

if DB\_ID (N'Company') is null

create database Company

go

use [Company]

if OBJECT\_ID ('ProgressEmployees','U') is not null

drop table ProgressEmployees

go

create table ProgressEmployees

(ProgressEmployeeId int IDENTITY(1,1) not null primary key, FullName nvarchar(20), Progress nvarchar(50), EmployeeId int,

CONSTRAINT FK\_ProgressEmployees\_To\_Employees FOREIGN KEY (EmployeeId) REFERENCES Employees (EmployeeId));

Go

Код на языке T-SQl для генерации тестового набора данных:

Код для генерации тестового набора для таблицы Units:

--таблица Units находится на стороне отношений "ОДИН"

use Company

go

DELETE FROM Units

DBCC CHECKIDENT ('Units', RESEED, 0)

go

DECLARE @Symbol CHAR(52)= 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz',

@Position int,

@i int,

@NameLimit int,

@FullName nvarchar(20),

@RowCount INT,

@MinNumberSymbols int,

@MaxNumberSymbols int,

@CountEmployees int,

@AmountOfData int

SET @AmountOfData = 500

BEGIN TRAN

SELECT @i=1 FROM dbo.Units WITH (TABLOCKX) WHERE 1=0

-- Заполнение данными таблицы Units

SET @RowCount=1

SET @MinNumberSymbols=5

SET @MaxNumberSymbols=50

WHILE @RowCount<=@AmountOfData

BEGIN

SET @NameLimit=@MinNumberSymbols+RAND()\*(@MaxNumberSymbols-@MinNumberSymbols) -- имя от 5 до 50 символов

SET @i=1

SET @FullName=''

WHILE @i<=@NameLimit

BEGIN

SET @Position=RAND()\*52

SET @FullName = @FullName + SUBSTRING(@Symbol, @Position, 1)

SET @i=@i+1

END

SET @CountEmployees = CAST((100 + RAND()\*1000) as int)

INSERT INTO dbo.Units(FullName, CountEmployees)

SELECT @FullName, @CountEmployees

SET @RowCount +=1

END

SELECT \* FROM dbo.Units

Код для генерации тестового набора для таблицы UnitsValuationFact:

use Company

go

DELETE FROM dbo.UnitsValuationFact

DBCC CHECKIDENT ('UnitsValuationFact', RESEED, 0)

go

DECLARE @Symbol CHAR(52)= 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz',

@Position int,

@i int,

@NameLimit int,

@FullName nvarchar(20),

@Income int,

@Cost int,

@RowCount INT,

@MinNumberSymbols int,

@MaxNumberSymbols int,

@AmountOfData int,

@NumberUnitId int

SET @AmountOfData =20000

BEGIN TRAN

SELECT @i=1 FROM dbo.UnitsValuationFact WITH (TABLOCKX) WHERE 1=0

-- Заполнение данными таблицы UnitsValuationFact

SET @RowCount=1

SET @MinNumberSymbols=5

SET @MaxNumberSymbols=50

WHILE @RowCount<=@AmountOfData

BEGIN

SET @NameLimit=@MinNumberSymbols+RAND()\*(@MaxNumberSymbols-@MinNumberSymbols) -- имя от 5 до 50 символов

SET @i=1

SET @FullName=''

WHILE @i<=@NameLimit

BEGIN

SET @Position=RAND()\*52

SET @FullName = @FullName + SUBSTRING(@Symbol, @Position, 1)

SET @i=@i+1

END

SET @Income = CAST((1000 + RAND()\*1000) as float)

SET @Cost = CAST((100 + RAND()\*100) as float)

SET @NumberUnitId = 1 + RAND()\*(500-1)

INSERT INTO dbo.UnitsValuationFact(FullName, Income, Cost, UnitId)

SELECT @FullName, @Income, @Cost, @NumberUnitId

SET @RowCount +=1

END

SELECT \* FROM dbo.UnitsValuationFact

Код для генерации тестового набора для таблицы UnitsValuationPlans:

use Company

go

DELETE FROM dbo.UnitsValuationPlans

DBCC CHECKIDENT ('UnitsValuationPlans', RESEED, 0)

go

DECLARE @Symbol CHAR(52)= 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz',

@Position int,

@i int,

@NameLimit int,

@FullName nvarchar(20),

@Income int,

@Cost int,

@RowCount INT,

@MinNumberSymbols int,

@MaxNumberSymbols int,

@AmountOfData int,

@UnitValuationFactId int

SET @AmountOfData = 20000

BEGIN TRAN

SELECT @i=1 FROM dbo.UnitsValuationPlans WITH (TABLOCKX) WHERE 1=0

-- Заполнение данными таблицы UnitsValuationPlans

SET @RowCount=1

SET @MinNumberSymbols=5

SET @MaxNumberSymbols=50

WHILE @RowCount<=@AmountOfData

BEGIN

SET @NameLimit=@MinNumberSymbols+RAND()\*(@MaxNumberSymbols-@MinNumberSymbols) -- имя от 5 до 50 символов

SET @i=1

SET @FullName=''

WHILE @i<=@NameLimit

BEGIN

SET @Position=RAND()\*52

SET @FullName = @FullName + SUBSTRING(@Symbol, @Position, 1)

SET @i=@i+1

END

SET @Income = CAST((10000 + RAND()\*10000) as float)

SET @Cost = CAST((1000 + RAND()\*1000) as float)

SET @UnitValuationFactId = 1 + RAND()\*(20000 - 1)

INSERT INTO dbo.UnitsValuationPlans(FullName, Income, Cost, UnitValuationFactId)

SELECT @FullName, @Income, @Cost, @UnitValuationFactId

SET @RowCount +=1

END

SELECT \* FROM dbo.UnitsValuationPlans

Код для генерации тестового набора для таблицы Employees:

use Company

go

DELETE FROM dbo.Employees

DBCC CHECKIDENT ('Employees', RESEED, 0)

go

DECLARE @Symbol CHAR(52)= 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz',

@Position int,

@i int,

@NameLimit int,

@FullName nvarchar(20),

@RowCount INT,

@MinNumberSymbols int,

@MaxNumberSymbols int,

@AmountOfData int,

@Age int,

@Solution int,

@Profit int,

@UnitId int

SET @AmountOfData =20000

BEGIN TRAN

SELECT @i=1 FROM dbo.UnitsValuationFact WITH (TABLOCKX) WHERE 1=0

-- Заполнение данными таблицы Employees

SET @RowCount=1

SET @MinNumberSymbols=5

SET @MaxNumberSymbols=50

WHILE @RowCount<=@AmountOfData

BEGIN

SET @NameLimit=@MinNumberSymbols+RAND()\*(@MaxNumberSymbols-@MinNumberSymbols) -- имя от 5 до 50 символов

SET @i=1

SET @FullName=''

WHILE @i<=@NameLimit

BEGIN

SET @Position=RAND()\*52

SET @FullName = @FullName + SUBSTRING(@Symbol, @Position, 1)

SET @i=@i+1

END

SET @Solution = CAST((1 + RAND()\*10000) as int)

SET @Profit = CAST((1 + RAND()\*100000) as int)

SET @UnitId = CAST((1+RAND()\*(500-1)) as int)

SET @Age = 20 + RAND()\*30;

INSERT INTO dbo.Employees(FullName, Solution, Profit, Age, UnitId)

SELECT @FullName, @Solution, @Profit, @Age, @UnitId

SET @RowCount +=1

END

SELECT \* FROM dbo.Employees

Код для генерации тестового набора для таблицы EmployeeFact:

use Company

go

DELETE FROM dbo.EmployeeFact

DBCC CHECKIDENT ('EmployeeFact', RESEED, 0)

go

DECLARE @Symbol CHAR(52)= 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz',

@Position int,

@i int,

@NameLimit int,

@FullName nvarchar(20),

@RowCount INT,

@MinNumberSymbols int,

@MaxNumberSymbols int,

@AmountOfData int,

@Quarter int,

@Year int,

@EmployeeId int

SET @AmountOfData = 20000

BEGIN TRAN

SELECT @i=1 FROM dbo.UnitsValuationFact WITH (TABLOCKX) WHERE 1=0

-- Заполнение данными таблицы Units

SET @RowCount=1

SET @MinNumberSymbols=5

SET @MaxNumberSymbols=50

WHILE @RowCount<=@AmountOfData

BEGIN

SET @NameLimit=@MinNumberSymbols+RAND()\*(@MaxNumberSymbols-@MinNumberSymbols) -- имя от 5 до 50 символов

SET @i=1

SET @FullName=''

WHILE @i<=@NameLimit

BEGIN

SET @Position=RAND()\*52

SET @FullName = @FullName + SUBSTRING(@Symbol, @Position, 1)

SET @i=@i+1

END

SET @Year = CAST((DATEPART(YEAR, GETDATE() - RAND()\*3000)) as int)

SET @Quarter = CAST((DATEPART(QUARTER, GETDATE() - RAND()\*3000)) as int)

SET @EmployeeId = 1 + RAND()\*(2000-1)

INSERT INTO dbo.EmployeeFact(FullName, Quarter, Year, EmployeeId)

SELECT @FullName, @Quarter, @Year, @EmployeeId

SET @RowCount +=1

END

SELECT \* FROM dbo.EmployeeFact

Код для генерации тестового набора для таблицы EmployeePlans:

use Company

go

DELETE FROM dbo.EmployeePlans

DBCC CHECKIDENT ('EmployeePlans', RESEED, 0)

go

DECLARE @Symbol CHAR(52)= 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz',

@Position int,

@i int,

@NameLimit int,

@FullName nvarchar(20),

@RowCount INT,

@MinNumberSymbols int,

@MaxNumberSymbols int,

@AmountOfData int,

@Quarter int,

@Year int,

@EmployeeFactId int

SET @AmountOfData =20000

BEGIN TRAN

SELECT @i=1 FROM dbo.UnitsValuationPlans WITH (TABLOCKX) WHERE 1=0

-- Заполнение данными таблицы Units

SET @RowCount=1

SET @MinNumberSymbols=5

SET @MaxNumberSymbols=50

WHILE @RowCount<=@AmountOfData

BEGIN

SET @NameLimit=@MinNumberSymbols+RAND()\*(@MaxNumberSymbols-@MinNumberSymbols) -- имя от 5 до 50 символов

SET @i=1

SET @FullName=''

WHILE @i<=@NameLimit

BEGIN

SET @Position=RAND()\*52

SET @FullName = @FullName + SUBSTRING(@Symbol, @Position, 1)

SET @i=@i+1

END

SET @Year = CAST((DATEPART(YEAR, GETDATE() - RAND()\*2000)) as int)

SET @Quarter = CAST((DATEPART(QUARTER, GETDATE() - RAND()\*3000)) as int)

SET @EmployeeFactId = 1 + RAND()\*(20000-1)

INSERT INTO dbo.EmployeePlans(FullName, Quarter, Year, EmployeeFactId)

SELECT @FullName,@Quarter, @Year, @EmployeeFactId

SET @RowCount +=1

END

SELECT \* FROM dbo.EmployeePlans

Код для генерации тестового набора для таблицы ProgressEmployees:

use Company

go

DELETE FROM dbo.ProgressEmployees

DBCC CHECKIDENT ('ProgressEmployees', RESEED, 0)

go

DECLARE @Symbol CHAR(52)= 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz',

@Position int,

@i int,

@NameLimit int,

@FullName nvarchar(20),

@RowCount INT,

@MinNumberSymbols int,

@MaxNumberSymbols int,

@AmountOfData int,

@Progress nvarchar(50),

@EmployeeId int

SET @AmountOfData =20000

BEGIN TRAN

SELECT @i=1 FROM dbo.ProgressEmployees WITH (TABLOCKX) WHERE 1=0

-- Заполнение данными таблицы Units

SET @RowCount=1

SET @MinNumberSymbols=5

SET @MaxNumberSymbols=50

WHILE @RowCount<=@AmountOfData

BEGIN

SET @NameLimit=@MinNumberSymbols+RAND()\*(@MaxNumberSymbols-@MinNumberSymbols) -- имя от 5 до 50 символов

SET @i=1

SET @FullName=''

SET @Progress=''

WHILE @i<=@NameLimit

BEGIN

SET @Position=RAND()\*52

SET @FullName = @FullName + SUBSTRING(@Symbol, @Position, 1)

SET @i=@i+1

END

SET @i=1

WHILE @i<=@NameLimit

BEGIN

SET @Position=RAND()\*52

SET @Progress = @Progress + SUBSTRING(@Symbol, @Position, 1)

SET @i=@i+1

END

SET @EmployeeId = 1 + RAND()\*(20000-1)

INSERT INTO dbo.ProgressEmployees(FullName, Progress, EmployeeId)

SELECT @FullName,@Progress, @EmployeeId

SET @RowCount +=1

END

SELECT \* FROM dbo.ProgressEmployees

Код на языке T-SQL для создания не менее трех представлений:

Код представления (View) для вывода достижений конкретных сотрудников:

use Company

go

drop view EmployeesAndTheirProgress

go

create view EmployeesAndTheirProgress as

select dbo.ProgressEmployees.Progress as Progress,

dbo.Employees.FullName as Name,

dbo.Employees.Age as Age

from Employees inner join ProgressEmployees on dbo.Employees.EmployeeId = dbo.ProgressEmployees.EmployeeId

go

select \* from EmployeesAndTheirProgress

Код представления (View) для сравнения фактических и плановых показателей подразделений (Units):

use Company

go

drop view CompareUnitsFactAndUnitsPlans

go

create view CompareUnitsFactAndUnitsPlans as

select dbo.UnitsValuationFact.Income as IncomeUnitsFact,

dbo.UnitsValuationPlans.Income as IncomeUnitsPlans,

(dbo.UnitsValuationFact.Income-dbo.UnitsValuationPlans.Income) as Different

from dbo.UnitsValuationFact inner join dbo.UnitsValuationPlans on dbo.UnitsValuationFact.UnitValuationFactId = dbo.UnitsValuationPlans.UnitValuationFactId

go

select \* from CompareUnitsFactAndUnitsPlans

Код представления (View) для сравнения фактических и плановых показателей сотрудников (Employees) за год и квартал:

use Company

go

drop view CompareEmployeeFactAndEmployeePlans

go

create view CompareEmployeeFactAndEmployeePlans as

select dbo.EmployeeFact.Year as YearEmployeeFact,

dbo.EmployeeFact.ProfitYear as ProfitYearEmployeeFact,

dbo.EmployeePlans.Year as YearEmployeePlans,

dbo.EmployeePlans.ProfitYear as ProfitYearEmployeePlans,

(dbo.EmployeePlans.ProfitYear - dbo.EmployeeFact.ProfitYear) as Different

from dbo.EmployeeFact inner join dbo.EmployeePlans on dbo.EmployeeFact.EmployeeFactId = dbo.EmployeePlans.EmployeePlanId

go

select \* from CompareEmployeeFactAndEmployeePlans

Код на языке T-SQL для создания не менее трех процедур:

Код процедуры для добавления нового сотрудника (Employee):

use Company

go

drop procedure AddEmployee

go

create procedure AddEmployee

@FullName nvarchar(20),

@Solution real,

@Profit real,

@Age int,

@UnitId int

as

insert into dbo.Employees(FullName, Solution, Profit, Age, UnitId)

values (@FullName,@Solution,@Profit,@Age,@UnitId)

Код процедуры для добавления нового подразделения (Unit):

use Company

go

drop procedure AddUnits

go

create procedure AddUnits

@FullName nvarchar(20),

@CountEmployees int

as

insert into dbo.Units(FullName, CountEmployees)

values (@FullName,@CountEmployees)

Код процедуры для подсчета средней, минимальной и максимальной заработной платой сотрудников:

use Company

go

drop procedure AverageSolutionEmployee

go

create procedure AverageSolutionEmployee

@averageSolution varchar(20) output,

@minSolution varchar(20) output,

@maxSolution varchar(20) output

as

select @averageSolution = AVG(Solution), @minSolution = MIN(Solution), @maxSolution = MAX(Solution)

from dbo.Employees

**Вывод:** была разработана серверная часть клиент-серверной информационной системы основанной на базе данных в заданной предметной области средствами СУБД MS SQL Server, а также повторен материал по созданию хранимых процедур и представлений.