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 № 4

Выполнил:

студент группы ИТП-31, Бурлаков А.О.

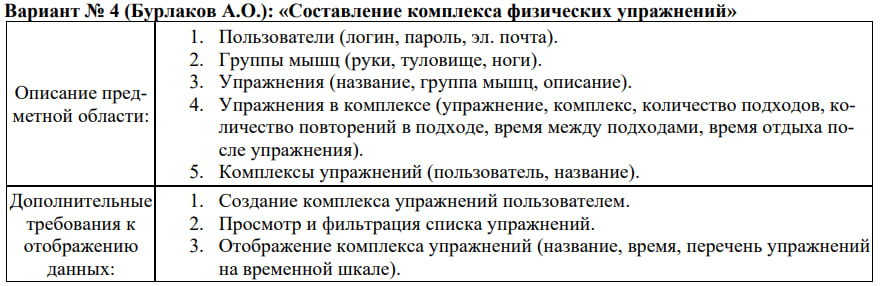
Принял:

Гуменников Е.Д.

Гомель 2024

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

На рисунке 1 представлено задание варианта 4



**Ход работы**

Connection string: Server=db8398.databaseasp.net;Database=db8398;User Id=db8398;Password=\*\*\*\*\*;Encrypt=False;MultipleActiveResultSets=True;

На рисунке 2 представлена диаграмма разработанной БД

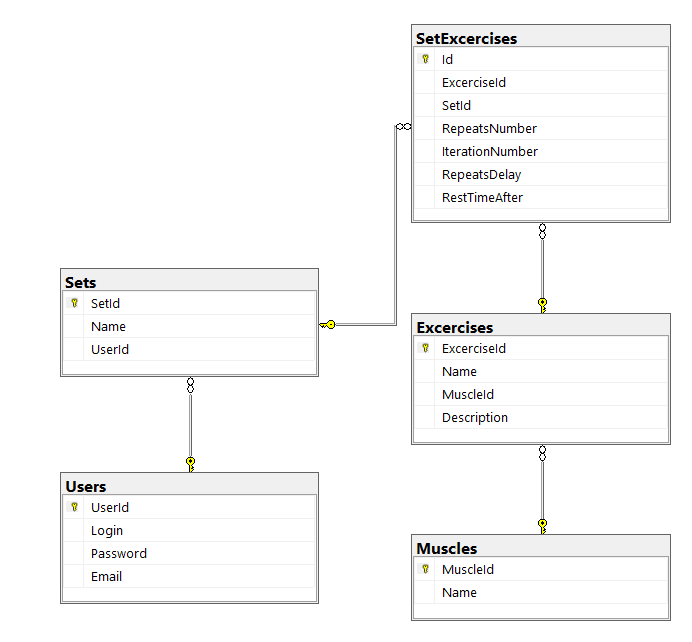


Рисунок 2 – Диаграмма БД

На рисунках 3,4,5,6,7 представлены таблицы бд. Скрипт создания таблиц представлен в приложении А.

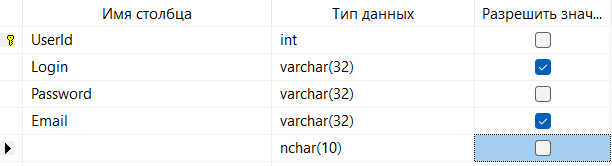


Рисунок 3 – Таблица *Users*

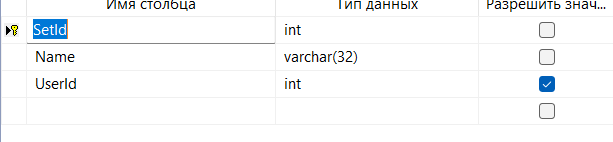


Рисунок 4 – Таблица *Sets*

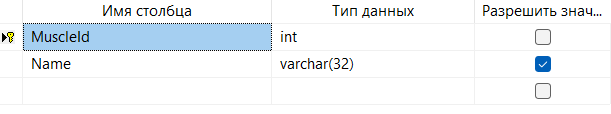


Рисунок 5 – Таблица Muscles

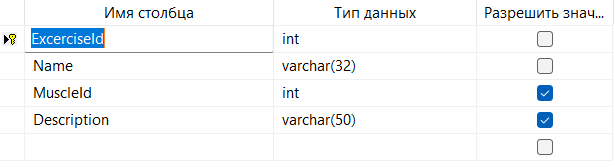


Рисунок 6 – Таблица Excercises

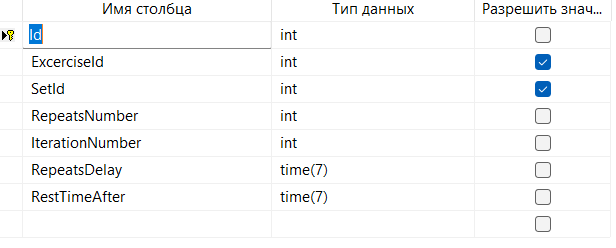


Рисунок 7 – Таблица SetExcercise

На рисунке 8 представлен результат создания Бд на удаленном сервере

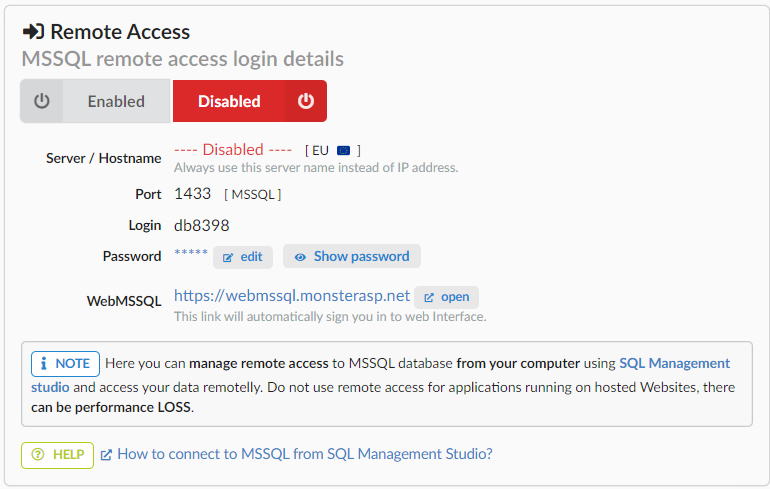


Рисунок 8 – Создание БД на удаленном сервере

Вывод: в ходе работы была Разработана серверная часть клиент-серверной информационной системы, основанной на базе данных в заданной предметной области средствами СУБД MS SQL Server.

ПРИЛОЖЕНИЕ А

Листинг

CreateTables.sql

create table Users(

UserId int identity(1,1) primary key,

[Login] varchar(32) unique not null,

[Password] varchar(32) not null,

[Email] varchar(32) unique not null

);

create table Muscles(

MuscleId int identity(1,1) primary key,

[Name] varchar(32) unique not null

);

create table [Sets](

SetId int identity(1,1) primary key,

[Name] varchar(32) not null,

[UserId] int references Users (UserId) on delete cascade

);

create table [Excercises](

ExcerciseId int identity(1,1) primary key,

[Name] varchar(32) unique not null,

[MuscleId] int references Muscles (MuscleId) on delete cascade,

[Description] varchar(50)

);

create table [SetExcercises](

Id int identity(1,1) primary key,

[ExcerciseId] int references [Excercises] (ExcerciseId) on delete cascade,

[SetId] int references [Sets] (SetId) on delete cascade,

[RepeatsNumber] int not null,

[IterationNumber] int not null,

[RepeatsDelay] time(0) not null,

[RestTimeAfter] time(0) not null

);

InsertTransactions.sql

use SportDb

set nocount on

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

@i int,

@RowCount int,

@NumberUsers int,

@NumberSets int,

@NumberSetExcercises int,

@MinNumSymbols int,

@MaxNumSymbols int,

@NameLimit int,

@Login varchar(32),

@Password varchar(20),

@Email varchar(32),

@Position int,

@PasswordLimit int,

@SetName varchar(32),

@MinUserId int,

@MaxUserId int,

@UserId int,

@ExcerciseId int,

@SetId int,

@TimeDelay time,

@RestTimeAfter time,

@Minute int,

@Second int

set @NumberUsers = 500

set @NumberSets = 20000

set @NumberSetExcercises = 20000

set @MinNumSymbols = 4

set @MaxNumSymbols = 20

begin transaction

select @i=0 from dbo.Users with (tablockx) where 1=0

set @RowCount=1

set @PasswordLimit=5

set @MinNumSymbols=3

set @MaxNumSymbols=18

while @RowCount<=@NumberUsers

begin

set @NameLimit=@MinNumSymbols+RAND()\*(@MaxNumSymbols-@MinNumSymbols) -- имя от 3 до 32 символов

set @i=1

set @Login=''

set @Password=''

set @Email=''

while @i<=@NameLimit

begin

set @Position=Rand()\*52

set @Login = @Login + SUBSTRING(@Symbol,@Position,1)

set @i=@i+1

end

set @Email=@Login + '@gmail.com'

set @i=0

while @i<=@PasswordLimit

begin

set @Position=RAND()\*52

set @Password=@Password+SUBSTRING(@Symbol,@Position,1)

set @i=@i+1

end

set @i=0

insert into dbo.Users ([Login],[Password],[Email]) select @Login,@Password,@Email

set @RowCount+=1

end

select @i=0 from [Sets] with (tablockx) where 1=0

set @RowCount=1

set @MinUserId=(select Min(Users.UserId) from Users)

set @MaxUserId=(select Max(Users.UserId) from Users)

while @RowCount<=@NumberSets

begin

set @NameLimit = @MinNumSymbols+RAND()\*(@MaxNumSymbols-@MinNumSymbols)

set @i=0

set @SetName=''

while @i<=@NameLimit

begin

set @Position=RAND()\*52

set @SetName=@SetName+SUBSTRING(@Symbol,@Position,1)

set @i+=1

end

insert into dbo.[Sets]([Name],[UserId]) select @SetName, (@MinUserId+Rand()\*(@MaxUserId-@MinUserId))

set @RowCount+=1

end

select @i=0 from SetExcercises with (tablockx) where 1=0

set @i=1

set @RowCount=1

while @RowCount<=@NumberSetExcercises

begin

set @ExcerciseId = (select top 1 ExcerciseId from Excercises order by NEWID())

set @SetId = (select top 1 SetId from [Sets] order by NEWID())

set @TimeDelay = '00:05:00'

set @RestTimeAfter = '00:08:00'

insert into SetExcercises(ExcerciseId,SetId,RepeatsNumber,IterationNumber,RepeatsDelay,RestTimeAfter) select @ExcerciseId,@SetId,Rand()\*4+1,RAND()\*15+1,@TimeDelay,@RestTimeAfter

set @RowCount+=1

end

commit transaction

CreateViews.sql

use SportDb

go

create or alter view ExcerciseView as

select ExcerciseId, ex.[Name] as Excercise,mus.[Name] as Muscle, ex.[Description]

from Excercises as ex

join Muscles as mus on ex.MuscleId=mus.MuscleId

go

go

create or alter view SetExcerciseView as

select setex.Id,s.[Name] as [Set],ex.[Name] as Excercise,ex.[Description],setex.RepeatsNumber,setex.IterationNumber,setex.RepeatsDelay,setex.RestTimeAfter

from SetExcercises as setex

join [Sets] as s on setex.SetId=s.SetId

join Excercises as ex on setex.ExcerciseId=ex.ExcerciseId

go

CreateProcedures.sql

create or alter proc InsertExcercise(@Name varchar(32),@Muscle varchar(32), @Description varchar(50) = '')

as

declare @MuscleId int

set @MuscleId = (Select MuscleId from Muscles Where @Muscle = @Name)

begin tran

insert into Excercises select @Name,@MuscleId,@Description

commit tran

go

create or alter proc InsertSet(@Name varchar(32), @User varchar(32))

as

declare @UserId int

set @UserId = (select UserId from Users Where @User = [Login])

begin tran

insert into [Sets] select @Name,@UserId

commit tran

go

create or alter proc InsertSetExcercise(

@Excercise varchar(32),

@Set varchar(32),

@RepeatsNumber int,

@IterationNumber int,

@RepeatsDelay varchar(6),

@RestTimeAfter varchar(6))

as

declare @ExcerciseId int,

@SetId int,

@RepDelayTime time(0),

@RestTime time(0)

set @ExcerciseId = (select ExcerciseId from Excercises Where @Excercise = [Name])

set @SetId = (select SetId from [Sets] Where @Set = [Name])

set @RepDelayTime = (SELECT CAST(STUFF(STUFF(@RepeatsDelay,3,0,':'),6,0,':') AS TIME(0)))

set @RestTime = (SELECT CAST(STUFF(STUFF(@RestTimeAfter,3,0,':'),6,0,':') AS TIME(0)))

begin tran

insert into SetExcercises select @ExcerciseId,@SetId,@RepeatsNumber,@IterationNumber,@RepDelayTime,@RestTime

commit tran

go