МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

ИНСТИТУТ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ И ПРОГРАММИРОВАНИЯ

КАФЕДРА компьютерных технологий и программной инженерии

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| старший преподаватель |  |  |  | Н.В. Путилова |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №6 |
| «Хранимые процедуры» |
| по курсу: Проектирование баз данных |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 4932 |  | 28.09.21 |  | А.И.Белов |
|  |  |  | подпись, дата |  | инициалы, фамилия |

1.  Физическая модель БД
2. Назначение, тексты ХП и их вызовов
3. Добавление нового участка, в случае если указанной линии не существует, добавить ее

Создание:

CREATE PROCEDURE ins\_area

(

@areaNum int,

@lineNum int,

@size int,

@price int

) AS

BEGIN

DECLARE @id\_line\_new int

DECLARE @id\_area\_new int

IF NOT EXISTS (SELECT \* FROM Line WHERE Line.Number=@lineNum)

INSERT INTO Line VALUES (@lineNum)

SET @id\_line\_new = (SELECT TOP 1 Line.ID FROM Line WHERE Line.ID=@lineNum)

INSERT INTO Area VALUES (@areaNum, @id\_line\_new, @size, @price, @price)

END

Вызов:

EXEC ins\_area

@areaNum = 8,

@lineNum = 3,

@size = 10,

@price = 100000

1. Удаление участка, проверка наличия строений на участке, удаление линии если на ней не осталось участков

Создание:

CREATE PROCEDURE del\_area

(

@areaNum int

) AS

BEGIN

IF NOT EXISTS (SELECT \* FROM Building WHERE Building.Number\_Area=@areaNum) BEGIN

DECLARE @lineDelId int

SET @lineDelId=(SELECT Area.ID\_Line FROM Area WHERE Area.Number=@areaNum)

DELETE FROM Area WHERE Area.Number = @areaNum

IF NOT EXISTS (SELECT \* FROM Area WHERE Area.ID\_Line=@lineDelId)

DELETE FROM Line WHERE Line.ID=@lineDelId

END

ELSE PRINT 'На участке зарегестрированы здания'

END

Вызов:

EXEC del\_area

@areaNum = 1

1. Каскадное удаление владельца

Создание:

CREATE PROCEDURE del\_cascade\_owner

(

@ownerID int

) AS

BEGIN

DELETE FROM Owner\_Area WHERE Owner\_Area.Id\_Owner=@ownerID

DELETE FROM Owner\_Payment WHERE Owner\_Payment.Id\_Owner=@ownerID

DELETE FROM Owner WHERE Owner.ID = @ownerID

END

Вызов:

EXEC del\_cascade\_owner

@ownerID = 1

1. Подсчет кол-ва участков в садоводстве

Создание:

CREATE PROCEDURE count\_areas

AS

BEGIN

RETURN (SELECT COUNT(Area.Number) FROM Area)

END

Вызов:

DECLARE @res INT;

EXEC @res = count\_areas;

SELECT 'Return Status' = @res;

GO

1. Создание временной статистической таблицы с информацией об кол-ве участков и строений у каждого владельца, а так-же вывод данных о среднем кол-ве участков и строений

Создание:

CREATE PROCEDURE infoGard

AS

BEGIN

CREATE TABLE #GardInfo1

(

[ID] [int] PRIMARY KEY IDENTITY (1,1),

[OwnerID] [int],

[OwnerName] [nvarchar](50),

[OwnerSurname] [nvarchar](50),

[CountAreas] [int],

[CountBuilds] [int],

)

INSERT INTO #GardInfo1(OwnerId,OwnerName,OwnerSurname)

SELECT Owner.ID, Owner.Name, Owner.Surname FROM Owner

UPDATE #GardInfo1 SET CountAreas =

(SELECT COUNT(an) FROM (SELECT Area.Number as an FROM Area JOIN Owner\_Area ON Area.Number=Owner\_Area.Number\_Area WHERE Owner\_Area.Id\_Owner=OwnerID) as temp)

UPDATE #GardInfo1 SET CountBuilds =

(SELECT COUNT(bi) FROM (SELECT Building.ID as bi

FROM Building JOIN Area ON Area.Number= Building.Number\_Area JOIN Owner\_Area ON Owner\_Area.Number\_Area = Area.Number WHERE Owner\_Area.Id\_Owner=OwnerID) as temp)

SELECT \* FROM #GardInfo1

SELECT AVG (CountAreas) as 'Среднее кол-во участков',AVG(CountBuilds) as 'Среднее кол-во строений'FROM #GardInfo1

DROP TABLE #GardInfo1

END

Вызов:

DECLARE @res INT;

EXEC @res = count\_areas;

SELECT 'Return Status' = @res;

GO

1. Наборы данных, возвращаемые ХП
2. Добавление нового участка, в случае если указанной линии не существует, добавить ее

Вызов:

EXEC ins\_area

@areaNum = 8,

@lineNum = 3,

@size = 10,

@price = 100000

|  |  |
| --- | --- |
| До | После |
| Line | Line |
|  |  |
| Area | Area |
|  |  |

1. Удаление участка, проверка наличия строений на участке, удаление линии если на ней не осталось участков

Вызов:

EXEC del\_area

@areaNum = 8

|  |  |
| --- | --- |
| До | После |
| Line | Line |
|  |  |
| Area | Area |
|  |  |

1. Каскадное удаление владельца

Вызов:

EXEC del\_cascade\_owner

@ownerID = 3

|  |  |
| --- | --- |
| До | После |
| Owner | Owner |
|  |  |
| Owner\_Payment | Owner\_Payment |
|  |  |

1. Подсчет кол-ва участков в садоводстве

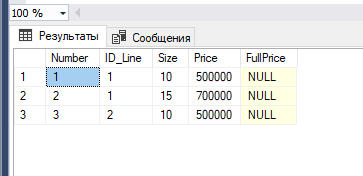
Вызов:

DECLARE @res INT;

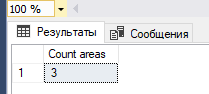
EXEC @res = count\_areas;

SELECT 'Count areas' = @res; GO

Area:



Результат:



1. Создание временной статистической таблицы с информацией об кол-ве участков и строений у каждого владельца, а так-же вывод данных о среднем кол-ве участков и строений

Вызов: EXEC infoGard

Результат:

