Министерство образования Республики Беларусь

Учреждение образования

«Брестский Государственный технический университет»

Кафедра ИИТ

**Лабораторная работа №4**

По дисциплине «Проектирование баз знаний»

Тема: «Хранимые процедуры, функции и триггеры в MS SQL Server 2005/2008.»

**Выполнил:**

Студент 3 курса

Группы ИИ-21

Карагодин Д.Л.

**Проверил:**

Савонюк В.А.

Брест 2023

**Цель работы:** Изучение и решение заданий по реализации хранимых процедур, функций и триггеровразных видов и фильтров в базах данных.

**Ход работы:**

**Вариант 5**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Вариант 5 | - | Услуги |  |  |
| 1 | **Код услуги** | название | стоимость | продолжительность |
| 2 | **фирма** | адрес | телефон |  |
| 3 | **Код услуги** | **фирма** | **дата заказа** | кол.заказов |

Задание.

1. Изучить материал, приведенный в “К лабораторной работе 4.doc”.

2. На основании логической модели (в соответствии с вариантом Лабораторной работы №1) создать и проверить на работоспособность хранимые процедуры, функции и триггеры для каждой из таблиц БД: реализовать хранимые процедуры, функции и триггеры разных видов для каждой таблицы (не менее 3 на каждую таблицу).

**Фирмы:**

|  |  |  |
| --- | --- | --- |
| Фирма | Адрес | Телефон |
| Новая Фирма | Новый Адрес | Новый Телефон |
| Фирма 1 | Адрес 1 | 111-1111 |
| Фирма 10 | Адрес 10 | 555-35-35 |
| Фирма 11 | Адрес 11 | 1111-1111 |
| Фирма 12 | Адрес 12 | 1212-1212 |
| Фирма 13 | Адрес 13 | 1313-1313 |
| Фирма 14 | Адрес 14 | 1414-1414 |
| Фирма 15 | Адрес 15 | 1515-1515 |
| Фирма 16 | Адрес 16 | 1616-1616 |
| Фирма 17 | Адрес 17 | 1717-1717 |
| Фирма 18 | Адрес 18 | 1818-1818 |
| Фирма 19 | Адрес 19 | 1919-1919 |
| Фирма 2 | Адрес 2 | 222-2222 |
| Фирма 20 | Адрес 20 | 2020-2020 |
| Фирма 4 | Адрес 4 | 444-4444 |
| Фирма 5 | Адрес 5 | 555-5555 |
| Фирма 6 | Адрес 6 | 666-6666 |
| Фирма 7 | Адрес 7 | 777-7777 |
| Фирма 8 | Адрес 8 | 888-8888 |
| Фирма 9 | Адрес 9 | 999-9999 |

**Заказы:**

|  |  |  |  |
| --- | --- | --- | --- |
| Код услуги | Фирма | Дата заказа | Кол-во заказов |
| 1 | Фирма 1 | 2023-10-01 | 5 |
| 5 | Фирма 5 | 2023-10-05 | 4 |
| 6 | Фирма 6 | 2023-10-06 | 2 |
| 7 | Фирма 7 | 2023-10-07 | 3 |
| 8 | Фирма 8 | 2023-10-08 | 1 |
| 9 | Фирма 9 | 2023-10-09 | 2 |
| 10 | Фирма 10 | 2023-10-10 | 4 |
| 11 | Фирма 11 | 2023-10-11 | 3 |
| 13 | Фирма 13 | 2023-10-13 | 2 |
| 14 | Фирма 14 | 2023-10-14 | 5 |
| 16 | Фирма 16 | 2023-10-16 | 2 |
| 17 | Фирма 17 | 2023-10-17 | 1 |
| 19 | Фирма 19 | 2023-10-19 | 4 |
| 20 | Фирма 20 | 2023-10-20 | 2 |
| 21 | Фирма 5 | 2023-10-01 | 9 |

**Услуги:**

|  |  |  |  |
| --- | --- | --- | --- |
| КодУслуги | Название | Стоимость | Продолжительность |
| 1 | Услуга 1 | 50,00 | 01:00:00.000000 |
| 2 | Услуга 2 | 90,00 | 01:30:00.000000 |
| 3 | Услуга 3 | 60,00 | 00:45:00.000000 |
| 5 | Услуга 5 | 90,00 | 01:30:00.000000 |
| 6 | Услуга 6 | 40,00 | 00:40:00.000000 |
| 7 | Услуга 7 | 55,00 | 01:15:00.000000 |
| 8 | Услуга 8 | 70,00 | 01:30:00.000000 |
| 9 | Услуга 9 | 80,00 | 01:45:00.000000 |
| 10 | Услуга 10 | 65,00 | 01:15:00.000000 |
| 11 | Услуга 11 | 95,00 | 02:30:00.000000 |
| 13 | Услуга 13 | 45,00 | 00:30:00.000000 |
| 14 | Услуга 14 | 85,00 | 01:30:00.000000 |
| 16 | Услуга 16 | 70,00 | 01:45:00.000000 |
| 17 | Услуга 17 | 55,00 | 01:15:00.000000 |
| 19 | Услуга 19 | 90,00 | 01:45:00.000000 |
| 20 | Услуга 20 | 40,00 | 00:40:00.000000 |
| 21 | Новая Услуга | 80,00 | 01:30:00.000000 |

**Задание №2**

**Фирмы:**

**Процедура:**

**Код программы:**

-- Хранимая процедура для вставки новой записи:

CREATE PROCEDURE InsertFirm

@firm NVARCHAR(255),

@addr NVARCHAR(255),

@tel NVARCHAR(15)

AS

BEGIN

INSERT INTO [Фирмы] ([Фирма], [Адрес], [Телефон])

VALUES (@firm, @addr, @tel)

END

**Запрос пользователя:**

EXEC InsertFirm "Фирма инс","ореховая 78","555-64-54"

**Результат запроса:**

(строк обработано: 1)



**Функция:**

**Код программы:**

-- Хранимая функция для получения информации о фирме по её телефону:

USE [Услуги]

GO

/\*\*\*\*\*\* Object: UserDefinedFunction [dbo].[GetFirmInfoByTel] Script Date: 17.10.2023 9:49:14 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER FUNCTION [dbo].[GetFirmInfoByTel] (@FirmTel nchar(15))

RETURNS TABLE

AS

RETURN (

SELECT [Фирма], [Адрес], [Телефон]

FROM [Фирмы]

WHERE [Телефон] = @FirmTel

)

**Запрос пользователя:**

USE [Услуги]

GO

SELECT \* FROM [dbo].[GetFirmInfoByTel] ('555-35-35')

GO

**Результат запроса:**



**Триггер:**

**Код программы:**

-- Триггер для отслеживания вставок в таблицу Фирмы:

CREATE TRIGGER FirmInsertTrigger

ON Фирмы

AFTER INSERT

AS

BEGIN

print('Был добавлен новый элемент')

END

**Запрос пользователя:**

INSERT INTO Фирмы

VALUES('1','2','3')

**Результат запроса:**

Был добавлен новый элемент (строк обработано: 1)



**Услуги:**

**Процедура:**

**Код программы:**

-- Хранимая процедура для вставки записи в таблицу "Услуги":

CREATE PROCEDURE InsertService

@КодУслуги INT,

@Название NVARCHAR(50),

@Стоимость DECIMAL(10, 2),

@Продолжительность TIME

AS

BEGIN

INSERT INTO Услуги (КодУслуги, Название, Стоимость, Продолжительность)

VALUES (@КодУслуги, @Название, @Стоимость, @Продолжительность);

END

**Запрос пользователя:**

EXEC InsertService 228, 'массаж', 50.00, '01:15:00'

**Результат запроса:**



**Функция:**

**Код программы:**

-- Функция для расчета средней стоимости услуг:

CREATE FUNCTION CalculateAverageCost()

RETURNS DECIMAL(10, 2)

AS

BEGIN

DECLARE @AvgCost DECIMAL(10, 2);

SELECT @AvgCost = AVG(Стоимость)

FROM Услуги;

RETURN @AvgCost;

END

**Запрос пользователя:**

USE [Услуги]

GO

SELECT [dbo].[CalculateAverageCost] ()

GO

**Результат запроса:**



**Триггер:**

**Код программы:**

-- Триггер для автоматического обновления средней стоимости при вставке новых записей

CREATE TRIGGER UpdateAverageCost

ON Услуги

AFTER INSERT

AS

BEGIN

DECLARE @AvgCost DECIMAL(10, 2);

SELECT @AvgCost = AVG(Стоимость)

FROM Услуги;

print('Новая средняя')

print(@AvgCost)

END

**Запрос пользователя:**

INSERT INTO Услуги

VALUES(148, 'поездка', 100.00, '02:15:00')

**Результат запроса:**



**Заказы:**

**Процедура:**

**Код программы:**

-- хранимая процедура для вставки новой записи в таблицу "Заказы"

CREATE PROCEDURE InsertOrder

@ServiceCode INT,

@Company NVARCHAR(255),

@OrderDate DATE,

@Quantity INT

AS

BEGIN

INSERT INTO Заказы ([Код услуги], Фирма, [Дата заказа], [Кол-во заказов])

VALUES (@ServiceCode, @Company, @OrderDate, @Quantity)

END;

**Запрос пользователя:**

EXEC InsertOrder 7, 'Фирма 5', '2023-10-01', 6

**Результат запроса:**



**Функция:**

**Код программы:**

-- Функция для расчета общего количества заказов для определенной фирмы:

CREATE FUNCTION dbo.ОбщееКоличествоЗаказов(@Фирма NVARCHAR(50))

RETURNS INT

AS

BEGIN

DECLARE @ОбщееКоличествоЗаказов INT

SELECT @ОбщееКоличествоЗаказов = SUM([Кол-во заказов])

FROM Заказы

WHERE Фирма = @Фирма

RETURN @ОбщееКоличествоЗаказов

END

**Запрос пользователя:**

USE [Услуги]

GO

SELECT [dbo].[ОбщееКоличествоЗаказов] ('Фирма 8')

GO

**Результат запроса:**



**Триггер:**

**Код программы:**

-- триггер будет срабатывать после вставки новой записи в таблицу "Заказы”

CREATE TRIGGER AfterOrderInsert

ON Заказы

AFTER INSERT

AS

BEGIN

print('добавлен элемент в заказы')

END;

**Запрос пользователя:**

INSERT INTO Заказы

VALUES(8, 'Фирма 12', '2023-10-02', 10)

**Результат запроса:**



**Вывод:** Изучил и решил задания по реализации хранимых процедур, функций и триггеровразных видов и фильтров в базах данных.