Ярославский филиал Московского финансово-юридического университета МФЮА

**ПРАКТИЧЕСКАЯ РАБОТА №2**

**ПО ДИСЦИПЛИНЕ**

**«ОСНОВЫ ПРОЕКТИРОВАНИЯ БАЗ ДАННЫХ»**

Автор:

Студенты группы 31ИПо9481

Салманов.М.Д

Сафарян.К.Ф

Руководитель:

Фурмавнин С. А

Ярославль 2024

**Тема: Простейшие SQL- запросы. DDL.**

**Цель работы:**

Научиться создавать базу данных в СУБД MS SQL Management Studio/PostreSQL/MySQL согласно схеме, созданной в проекте. Научиться строить SQL-запросы CREATE, ALTER и TABLE.

**Основная часть:**

(приводятся теоретические сведения об используемых командах SQL-запроса, его структура и синтаксис в общем виде. Описывается необходимость и обусловленность использования тех или иных ключевых слов в ваших запросах.)

Основные команды:

* CREATE – создать что-либо.
* CREATE TABLE – создать таблицу.
* IDENTITY (1, 1) – присваивает id, начиная с 1 прибавляй 1.
* PRIMARY KEY – (первичный ключ) это уникальный атрибут позволяющий точно определить его принадлежность к чему либо (как серия и номер паспорта у человека).
* FOREIGN KEY – (внешний ключ) это атрибут являющийся первичным ключом в другой таблице, служащий для их связи.
* REFERENCES – сочетание с внешним ключом, указывает таблицу и атрибут на который ссылается.
* NULL – может быть пустым.
* NOT NULL – не может быть пустым.

**Задания:**

Задание 1. Написать скрипт создания связанных таблиц с ограничениями целостности вашей базы данных с использованием CREATE.

CREATE TABLE Zavod (

inn VARCHAR(12) PRIMARY KEY NOT NULL,

naimenovanie\_procesa VARCHAR(150) NOT NULL,

adres VARCHAR(100) NOT NULL,

email VARCHAR(50) NOT NULL,

);

CREATE TABLE Zakazchik (

no\_pasp VARCHAR(10) PRIMARY KEY NOT NULL,

inn VARCHAR(12) NOT NULL,

adres VARCHAR(100) NOT NULL,

fio VARCHAR(100) NOT NULL,

no\_telefona VARCHAR(11) NOT NULL,

FOREIGN KEY (inn) REFERENCES Zavod(inn)

);

CREATE TABLE Tovar (

no\_tovara INT PRIMARY KEY IDENTITY,

naimenovanie\_tovara VARCHAR(100) NOT NULL,

);

CREATE TABLE Zakaz (

no\_zakaza INT PRIMARY KEY IDENTITY NOT NULL,

no\_dogovora VARCHAR(50) NOT NULL,

data\_zaklyucheniya DATE NOT NULL,

no\_tovara INT NOT NULL,

planirovannaya\_postavka DATE NOT NULL,

cena\_tovara DECIMAL(10, 2) NOT NULL,

no\_pozitsii INT NOT NULL,

no\_pasp VARCHAR(10) NOT NULL,

FOREIGN KEY (no\_pasp) REFERENCES Zakazchik(no\_pasp),

FOREIGN KEY (no\_tovara) REFERENCES Tovar(no\_tovara),

);

CREATE TABLE Otgruzka (

no\_otgruzki INT PRIMARY KEY IDENTITY,

no\_zakaza INT NOT NULL,

data\_otgruzki DATE NOT NULL,

otgruzheno\_tovara INT NOT NULL,

FOREIGN KEY (no\_zakaza) REFERENCES Zakaz(no\_zakaza)

);

CREATE TABLE Kurer (

no\_pasp VARCHAR(10) PRIMARY KEY NOT NULL,

fio VARCHAR(100) NOT NULL,

no\_telefona VARCHAR(11) NOT NULL,

FOREIGN KEY (no\_pasp) REFERENCES Zakazchik(no\_pasp)

);

Задание 2. Написать скрипт создания несвязанных таблиц вашей базы данных без ограничений целостности с использованием CREATE.

CREATE TABLE Zavod (

inn VARCHAR(12),

naimenovanie\_procesa VARCHAR(150),

adres VARCHAR(100),

email VARCHAR(50),

);

CREATE TABLE Zakazchik (

no\_pasp VARCHAR(10),

inn VARCHAR(12),

adres VARCHAR(100),

fio VARCHAR(100),

no\_telefona VARCHAR(11),

);

CREATE TABLE Tovar (

no\_tovara INT,

naimenovanie\_tovara VARCHAR(100),

);

CREATE TABLE Zakaz (

no\_zakaza INT,

no\_dogovora VARCHAR(50),

data\_zaklyucheniya DATE,

no\_tovara INT,

planirovannaya\_postavka DATE,

cena\_tovara DECIMAL(10, 2),

no\_pozitsii INT,

no\_pasp VARCHAR(10),

);

CREATE TABLE Otgruzka (

no\_otgruzki INT,

no\_zakaza INT,

data\_otgruzki DATE,

otgruzheno\_tovara INT,

);

CREATE TABLE Kurer (

no\_pasp VARCHAR(10),

fio VARCHAR(100),

no\_telefona VARCHAR(11),

);

Задание 3. Написать скрипт добавления связей между таблицами и ограничений целостности в них с использованием ALTER.

IF NOT EXISTS (SELECT \*FROM sys.databases

WHERE [name] = 'TOVAR')

BEGIN

CREATE DATABASE TOVAR

END

GO

USE TOVAR;

GO

ALTER TABLE Zavod ADD CONSTRAINT PK\_Zavod PRIMARY KEY (inn);

ALTER TABLE Zakazchik ADD CONSTRAINT PK\_Zakazchik PRIMARY KEY (nomer\_pasporta);

ALTER TABLE Tovar ADD CONSTRAINT PK\_Tovar PRIMARY KEY (nomer\_tovara);

ALTER TABLE Zakaz ADD CONSTRAINT PK\_Zakaz PRIMARY KEY (id\_zakaza);

ALTER TABLE Otgruzka ADD CONSTRAINT PK\_Otgruzka PRIMARY KEY (nomer\_otgruzki);

ALTER TABLE Kurer ADD CONSTRAINT PK\_Kurer PRIMARY KEY (nomer\_pasporta);

Задание 4. Напишите скрипт, который удаляет все ваши таблицы с использованием DROP.

ALTER TABLE Kurer NOCHECK CONSTRAINT ALL;

ALTER TABLE Otgruzka NOCHECK CONSTRAINT ALL;

ALTER TABLE Zakaz NOCHECK CONSTRAINT ALL;

ALTER TABLE Zakazchik NOCHECK CONSTRAINT ALL;

ALTER TABLE Tovar NOCHECK CONSTRAINT ALL;

ALTER TABLE Zavod NOCHECK CONSTRAINT ALL;

DROP TABLE IF EXISTS Kurer;

DROP TABLE IF EXISTS Otgruzka;

DROP TABLE IF EXISTS Zakaz;

DROP TABLE IF EXISTS Zakazchik;

DROP TABLE IF EXISTS Tovar;

DROP TABLE IF EXISTS Zavod;

**Вывод:**

В ходе лабораторной работы, мы научились создавать базу данных в СУБД MS SQL Management Studio/PostreSQL/MySQL согласно схеме, созданной в проекте.