**Домашняя работа №8**

**Архитектура MS SQL**

**Задания**

1. Создайте вашу базу данных: таблицы, ограничения, ключи.

CREATE DATABASE Ecommerce;

GO

USE Ecommerce;

GO

DROP TABLE IF EXISTS Product;

CREATE TABLE Product (

ProductID INT IDENTITY(1,1) PRIMARY KEY,

ManufacturerName NVARCHAR(30),

ProductName NVARCHAR(50) NOT NULL,

Descriptions NVARCHAR(150),

CategoryName nvarchar(30),

Price MONEY NOT NULL,

CostPrice MONEY NOT NULL,

);

GO

DROP TABLE IF EXISTS Manager;

CREATE TABLE Manager (

ManagerID INT IDENTITY(1,1) PRIMARY KEY,

FirstName NVARCHAR(50) NOT NULL,

LastName NVARCHAR(50) NOT NULL,

BirthDate DATE,

PhoneNumber NVARCHAR(15),

PassportNumber NVARCHAR(25) NOT NULL UNIQUE,

);

GO

DROP TABLE IF EXISTS Shop;

CREATE TABLE Shop (

ShopID INT IDENTITY(1,1) PRIMARY KEY,

ShopName NVARCHAR(30) NOT NULL,

City NVARCHAR(30),

Address NVARCHAR(30),

);

GO

DROP TABLE IF EXISTS Delivery;

CREATE TABLE Delivery (

DeliveryID INT IDENTITY(1,1) PRIMARY KEY,

DeliveryType NVARCHAR(20) NOT NULL,

);

GO

DROP TABLE IF EXISTS Buyer;

CREATE TABLE Buyer (

BuyerID INT IDENTITY(1,1) PRIMARY KEY,

FirstName NVARCHAR(50) NOT NULL,

LastName NVARCHAR(50),

email NVARCHAR(50) NOT NULL UNIQUE,

);

GO

DROP TABLE IF EXISTS Sale;

CREATE TABLE Sale (

SaleID INT IDENTITY(1,1) PRIMARY KEY,

ProductID INT,

ManagerID INT,

BuyerID INT,

DeliveryID INT,

SellingAmount INT NOT NULL,

Date DATE NOT NULL,

ShopID INT,

CONSTRAINT FK\_ProductID FOREIGN KEY (ProductID) REFERENCES Product(ProductID),

CONSTRAINT FK\_ManagerID FOREIGN KEY (ManagerID) REFERENCES Manager(ManagerID),

CONSTRAINT FK\_BuyerID FOREIGN KEY (BuyerID) REFERENCES Buyer(BuyerID),

CONSTRAINT FK\_DeliveryID FOREIGN KEY (DeliveryID) REFERENCES Delivery(DeliveryID),

CONSTRAINT FK\_ShopID FOREIGN KEY (ShopID) REFERENCES Shop(ShopID),

);

Physical model

Изображение выглядит как текст, карта, внутренний, несколько

Автоматически созданное описание

1. Напишите скрипт для получения 1 млн человек с различными именами и фамилиями. Полезная [ссылка](https://mockaroo.com/).

Не понимаю почему не импортирует данные?!

CREATE DATABASE Test;

GO

USE Test;

GO

CREATE TABLE dbo.DifferentNames (

FirstName NVARCHAR(50),

LastName NVARCHAR(50),

);

BULK INSERT DifferentNames FROM 'C:\DB\DifferentNames.csv'

WITH (

firstrow = 2,

FIELDTERMINATOR = ',',

ROWTERMINATOR = '\n'

);

GO

В итоге импортировать файлы с .csv не получилось :( не мой день

Пришлось сгенерировать два файла .sql: first\_name и last\_name по 1000 значений в каждом и создать по ним соответствующие таблицы.

SELECT

first\_name,

last\_name

FROM dbo.first\_name FN

CROSS JOIN dbo.last\_name LN;

На выходе 1 000 000 строк

Изображение выглядит как текст

Автоматически созданное описание

1. Познакомьтесь с [обобщенным табличным выражением WITH](https://docs.microsoft.com/ru-ru/sql/t-sql/queries/with-common-table-expression-transact-sql?view=sql-server-ver15). Когда можем использовать данную структуру? Чем отличается от подзапроса?

Обобщенное табличное выражение WITH можно использовать:

для улучшения читаемости запроса в случае сложных запросов (уменьшают размер кода);

в случаях, когда нужно много раз обращаться к одним и тем же таблицам/выборкам из таблиц;

для замены представлений в тех случаях, когда использование представления не оправдано, то есть тогда, когда нет необходимости сохранять в метаданных базы его определение;

для написания рекурсивных запросов.

Обобщенное табличное выражение WITH отличается от подзапроса тем, что:

вложенный запрос повторяется для каждой строки из нашей выборки, что повышает стоимость выполнения запроса;

может включать ссылки на само себя.

P.S. Падение «вылечилось» полной переустановкой SQL Server и SSMS