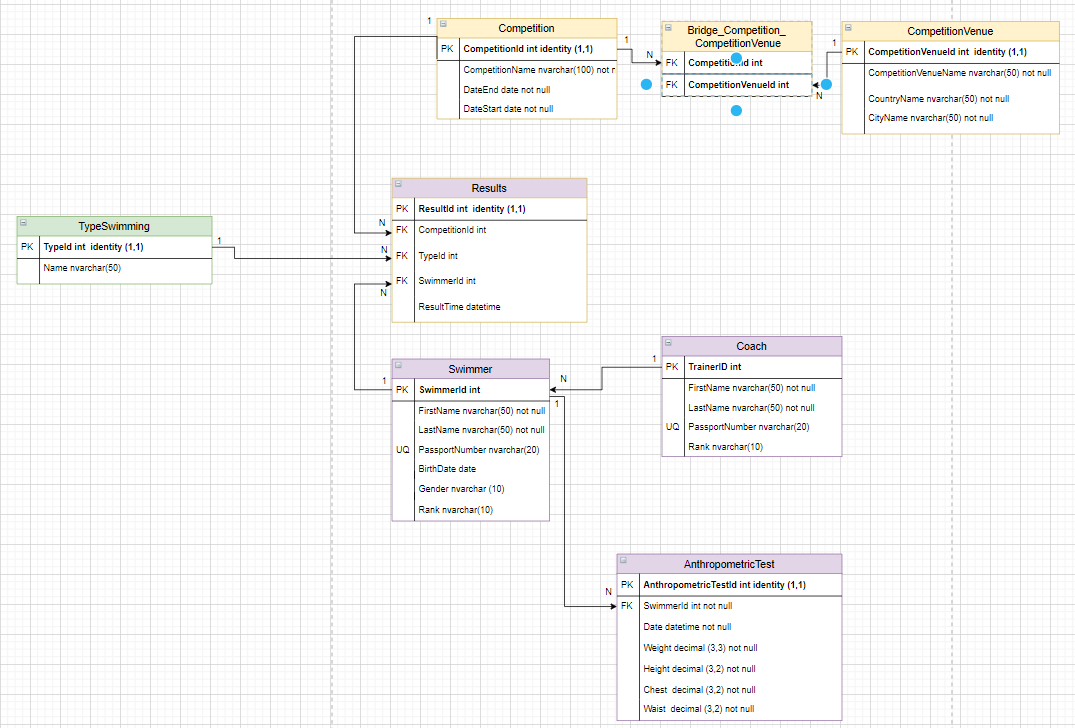
**Отчет**

**Домашняя работа 8**

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



CREATE DATABASE DiplomDB

CREATE SCHEMA Person

CREATE TABLE Person.Swimmer

(

SwimmerId INT PRIMARY KEY IDENTITY(1,1),

FirstName NVARCHAR(100) NOT NULL,

LastName NVARCHAR(100) NOT NULL,

PassportNumber NVARCHAR(20) UNIQUE,

BirthDate Date,

Gender NVARCHAR(1) NOT NULL,

Rank NVARCHAR(10)

)

CREATE TABLE Person.Trainer

(

TrainerId INT PRIMARY KEY IDENTITY(1,1),

FirstName NVARCHAR(100) NOT NULL,

LastName NVARCHAR(100) NOT NULL,

PassportNumber NVARCHAR(20) UNIQUE,

Rank NVARCHAR(10)

)

CREATE TABLE Person.AnthropometricTest

(

AnthropometricTestId INT PRIMARY KEY IDENTITY(1,1),

SwimmerId INT NOT NULL FOREIGN KEY REFERENCES Person.Swimmer(SwimmerId),

Date DATETIME NOT NULL,

Weight DECIMAL (3,3) NOT NULL,

Height DECIMAL (3,2) NOT NULL,

Сhest DECIMAL (3,2) NOT NULL,

Waist DECIMAL (3,2) NOT NULL

)

CREATE SCHEMA Swimming

CREATE TABLE Swimming.TypeSwimming

(

TypeId INT PRIMARY KEY IDENTITY(1,1),

Name NVARCHAR(50)

)

CREATE SCHEMA Place

CREATE TABLE Place.Сompetition

(

СompetitionId INT PRIMARY KEY IDENTITY(1,1),

СompetitionName NVARCHAR(100) NOT NULL,

DateEnd DATE NOT NULL,

DateStart DATE NOT NULL

)

CREATE TABLE Place.СompetitionVenue

(

СompetitionVenueId INT PRIMARY KEY IDENTITY(1,1),

СompetitionVenueName NVARCHAR(50) NOT NULL,

CountryName NVARCHAR(50) NOT NULL,

CityName NVARCHAR(50) NOT NULL

)

CREATE TABLE Place.BridgeCompetition\_Venue

(

СompetitionId INT NOT NULL FOREIGN KEY REFERENCES Place.Сompetition(СompetitionId),

СompetitionVenueId INT NOT NULL FOREIGN KEY REFERENCES Place.СompetitionVenue(СompetitionVenueId)

)

CREATE TABLE Person.Results

(

ResultId INT PRIMARY KEY IDENTITY(1,1),

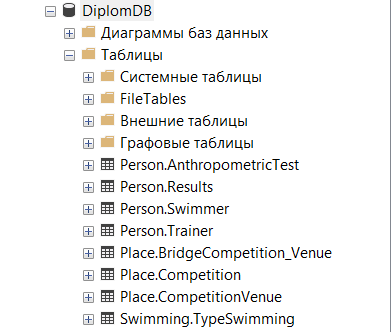
CompetitionId INT NOT NULL FOREIGN KEY REFERENCES Place.Сompetition(СompetitionId),

TypeId INT NOT NULL FOREIGN KEY REFERENCES Swimming.TypeSwimming(TypeId),

SwimmerId INT NOT NULL FOREIGN KEY REFERENCES Person.Swimmer(SwimmerId),

ResultTime DATETIME

)



1. **Напишите скрипт для получения 1 млн человек с различными именами и фамилиями.**

SELECT TOP 1000000

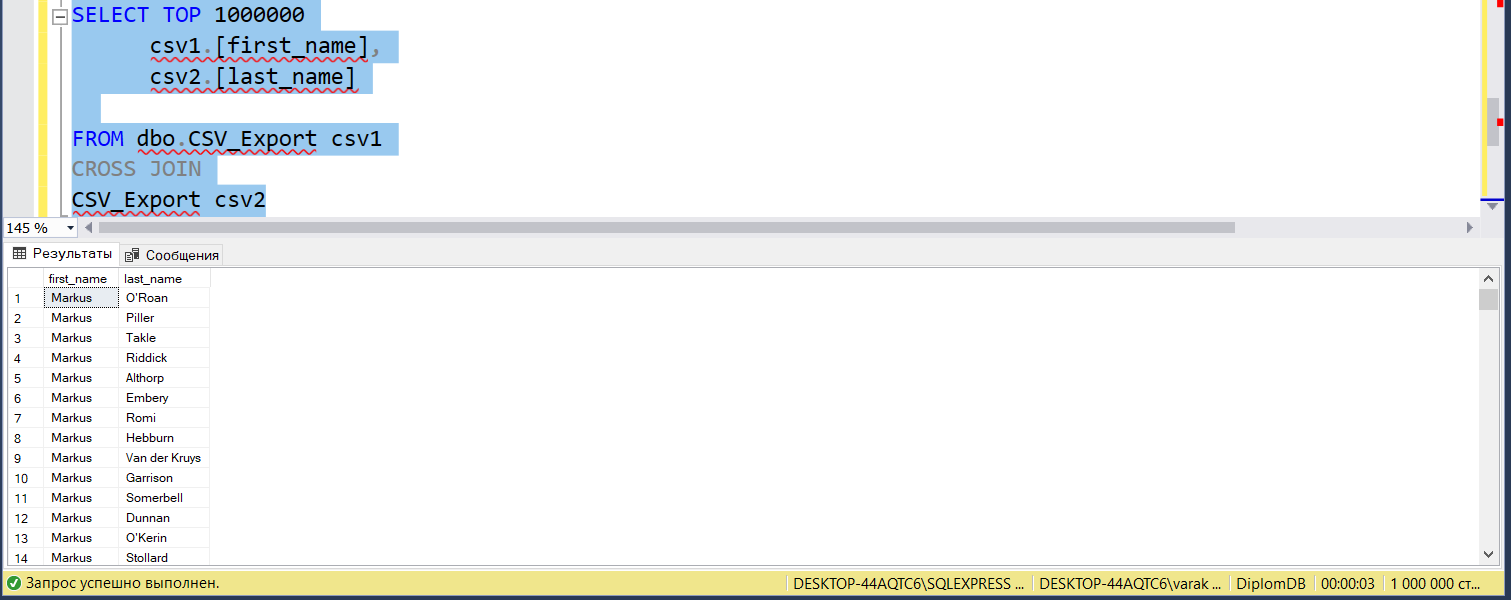
csv1.[first\_name],

csv2.[last\_name]

FROM dbo.CSV\_Export csv1

CROSS JOIN

CSV\_Export csv2



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?**

**Когда:** Для загрузки данных в таблицы из внешних источников; Для создания рекурсивных запросов; Для повышения читаемости кода.

**Чем отличается:** С помощью WITH возможно создавать рекурсивные запросы.