

КОМИТЕТ ПО ОБРАЗОВАНИЮ ПРАВИТЕЛЬСТВА САНКТ-  
ПЕТЕРБУРГА

Санкт-Петербургское государственное  
бюджетное профессиональное образовательное учреждение  
«КОЛЛЕДЖ ЭЛЕКТРОНИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ»

**«МДК.07.01 Управление и автоматизация баз данных»**

**О Т Ч Ё Т**

**п о л а б о р а т о р н о й р а б о т е № 3**

**«Подготовка и импорт данных»**

Работу выполнил студент 325гр.:

Шлычков И. Д.

Преподаватель: Фомин А.В.

Санкт-Петербург 2025

# Импорт данных

Для выполнения работы были использованы данные о береговой линии, представленные по ссылке <https://www.naturalearthdata.com/downloads/10m-physical-vectors/> . Рисунок 1



Рисунок 1 – сайт с данными

После скачивания архивы были распакованы в рабочую папку. Данные содержали координаты береговых линий. Рисунок 2

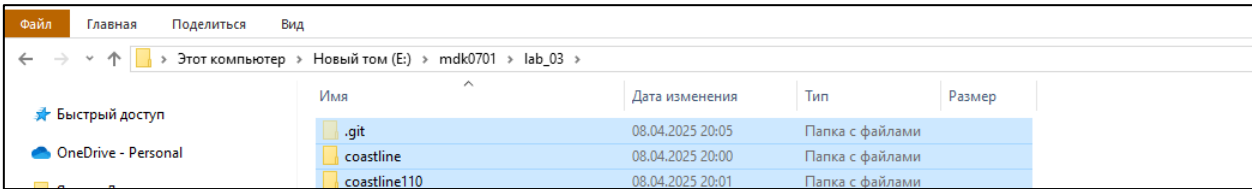


Рисунок 2 – рабочая папка

Для хранения данных ранее была создана база данных (lab02) в СУБД PostgreSQL. Структура базы данных включала координат береговых линий, а также дополнительных таблицы для метаданных. Рисунок 3

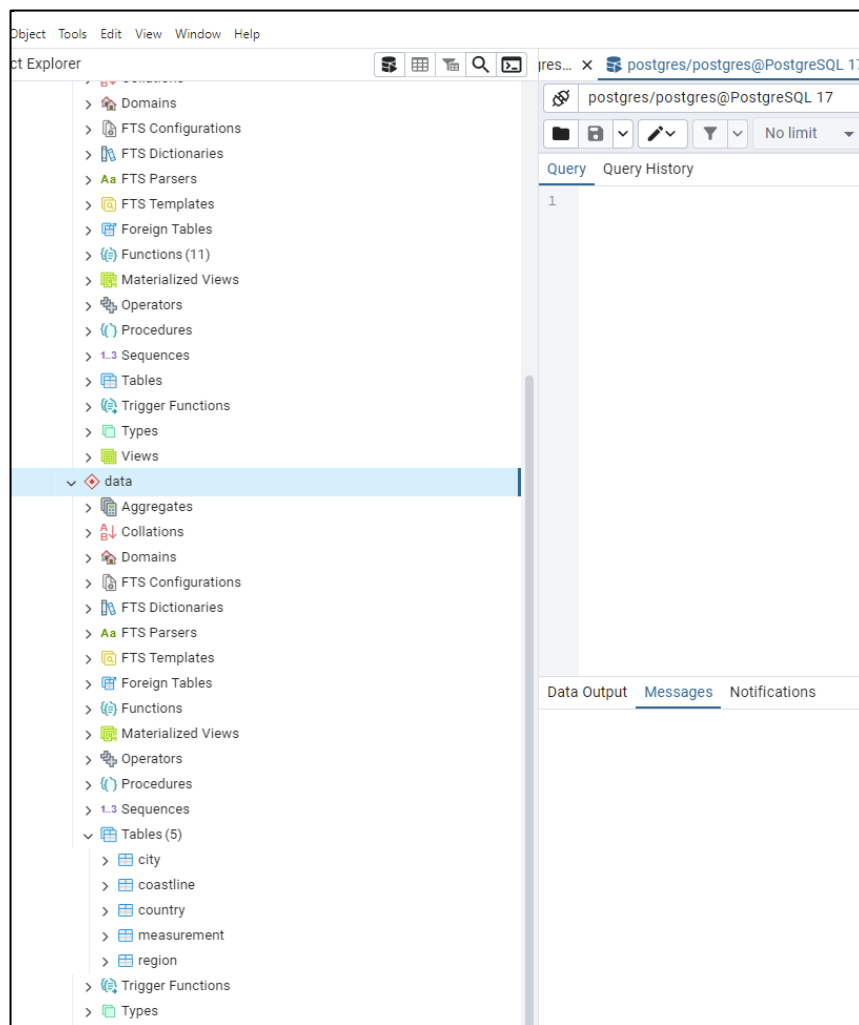


Рисунок 3 – база данных

## Создаем проекты

Для импорта данных был разработан скрипт на C#, который подключает проект к базе данных и загружает данные из распакованных файлов. Рисунок 4

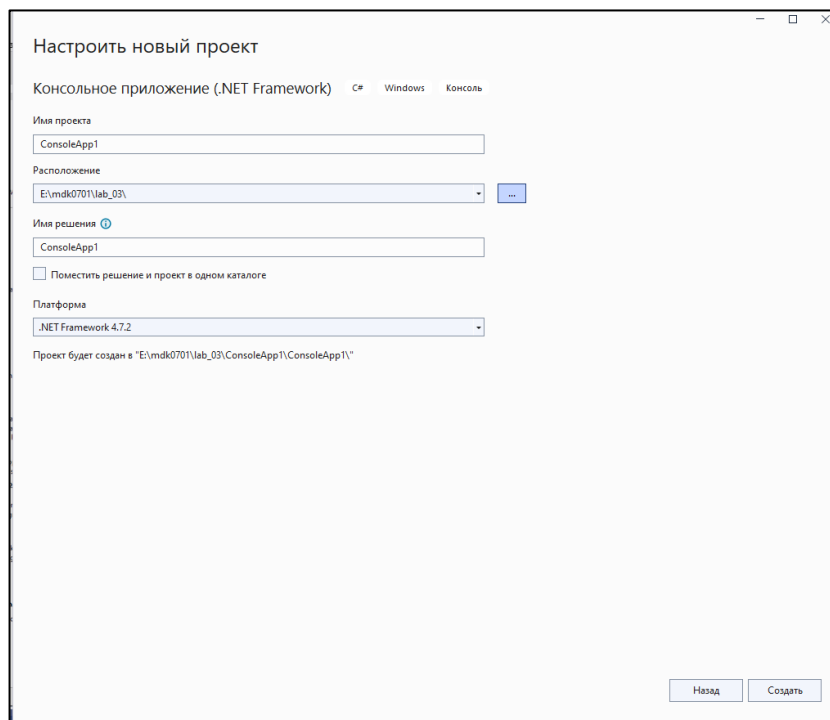
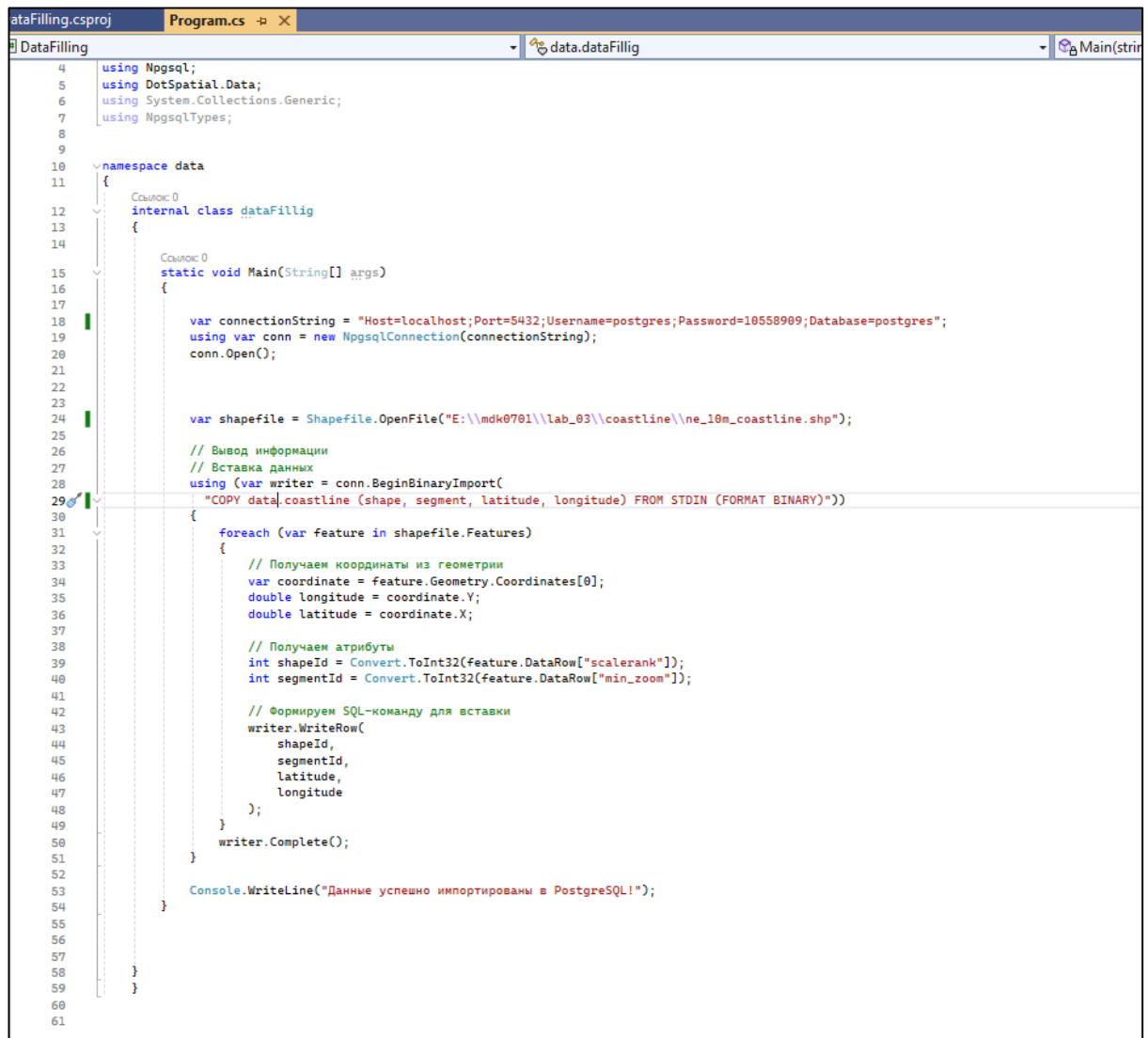


Рисунок 4 – создание проекта

Тут представлен фрагмент кода, в котором указаны порт БД, пароль и название самой БД. Также был добавлен путь к папке с данными, чтобы скрипт смог их перенести в нашу схему. Рисунок 5



```
1  DataFilling.csproj | Program.cs | data.dataFillig | Main(strin
2
3  4 using Npgsql;
4  5 using DotSpatial.Data;
5  6 using System.Collections.Generic;
6  7 using NpgsqlTypes;
7  8
8  9
9  10 namespace data
10 11 {
11 12     Ссылка: 0
12 13     internal class dataFillig
13 14     {
14 15
15 16     Ссылка: 0
16 17     static void Main(String[] args)
17 18     {
18 19         var connectionString = "Host=localhost;Port=5432;Username=postgres;Password=18558909;Database=postgres";
19 20         using var conn = new NpgsqlConnection(connectionString);
20 21         conn.Open();
21 22
22 23
23 24         var shapefile = Shapefile.OpenFile("E:\\mdk0701\\lab_03\\coastline\\ne_10m_coastline.shp");
24 25
25 26         // Вывод информации
26 27         // Вставка данных
27 28         using (var writer = conn.BeginBinaryImport(
28 29             "COPY data.coastline (shape, segment, latitude, longitude) FROM STDIN (FORMAT BINARY)"))
29 30         {
30 31             foreach (var feature in shapefile.Features)
31 32             {
32 33                 // Получаем координаты из геометрии
33 34                 var coordinate = feature.Geometry.Coordinates[0];
34 35                 double longitude = coordinate.Y;
35 36                 double latitude = coordinate.X;
36 37
37 38                 // Получаем атрибуты
38 39                 int shapeId = Convert.ToInt32(feature.DataRow["scalerank"]);
39 40                 int segmentId = Convert.ToInt32(feature.DataRow["min_zoom"]);
40 41
41 42                 // Формируем SQL-команду для вставки
42 43                 writer.WriteRow(
43 44                     shapeId,
44 45                     segmentId,
45 46                     latitude,
46 47                     longitude
47 48                 );
48 49             }
49 50             writer.Complete();
50 51         }
51 52         Console.WriteLine("Данные успешно импортированы в PostgreSQL!");
52 53     }
53 54 }
54 55
55 56
56 57
57 58 }
58 59
59 60
60 61
```

Рисунок 5 – фрагмент кода

Но прежде, чем подключить наш проект к БД, мы должны перейти во вкладку проект. Рисунок 6

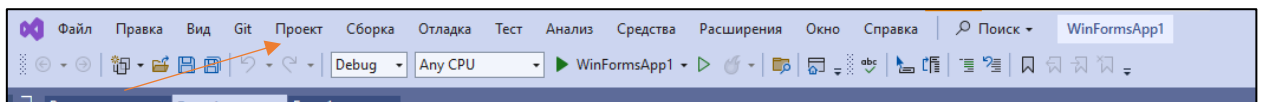


Рисунок 6

И затем скачать дополнительные расширения представленные на Рисунке 7, чтобы всё заработало

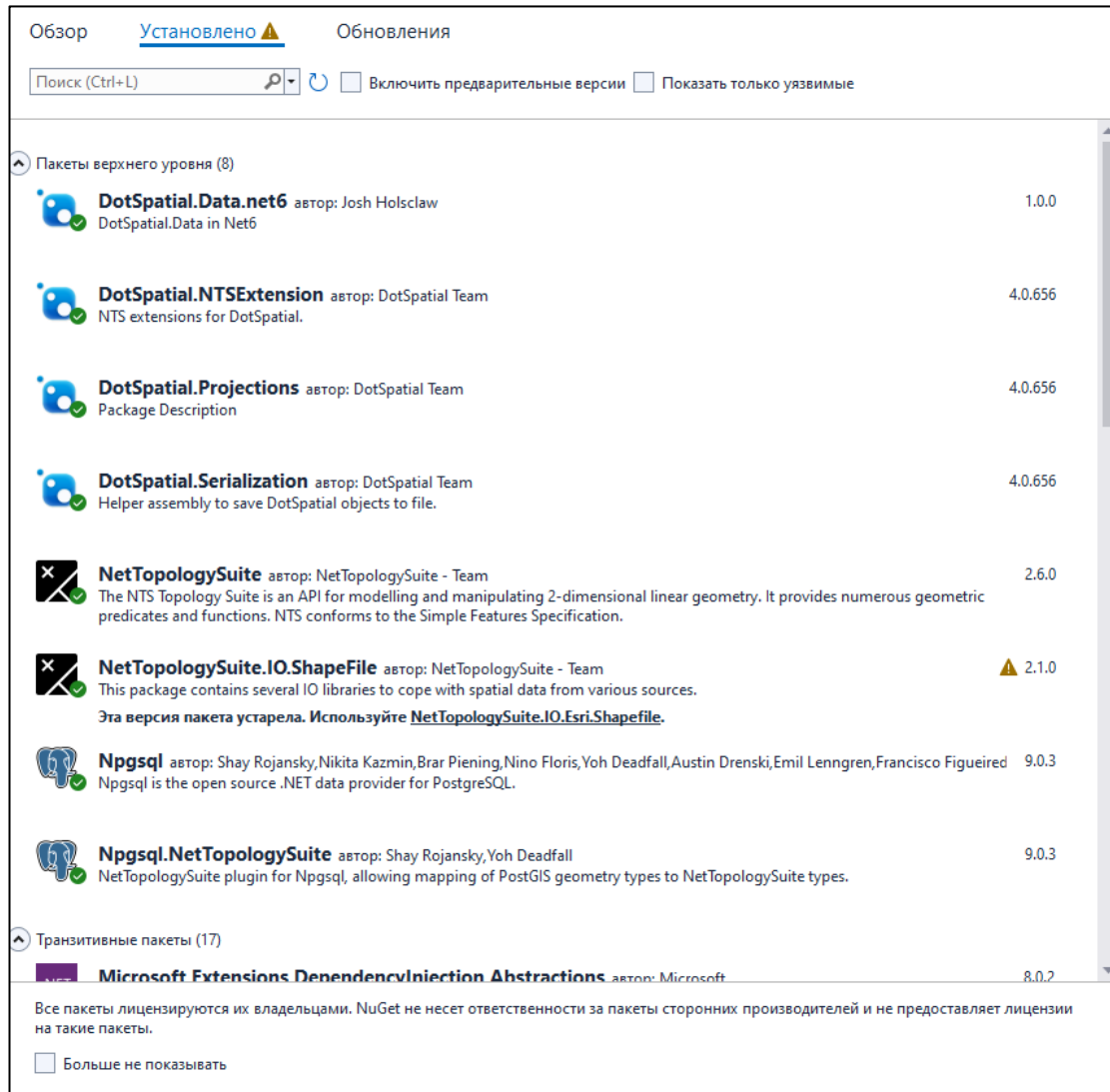


Рисунок 7 – дополнительные расширения

После чего мы запускаем нашу программу. И убеждаемся что всё работает.  
Рисунок 8

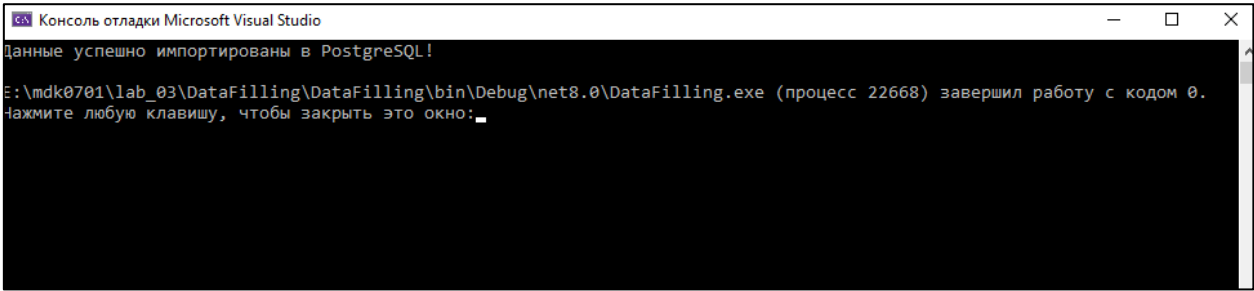


Рисунок 8

Выводим через select, чтобы посмотреть результат. Рисунок 9

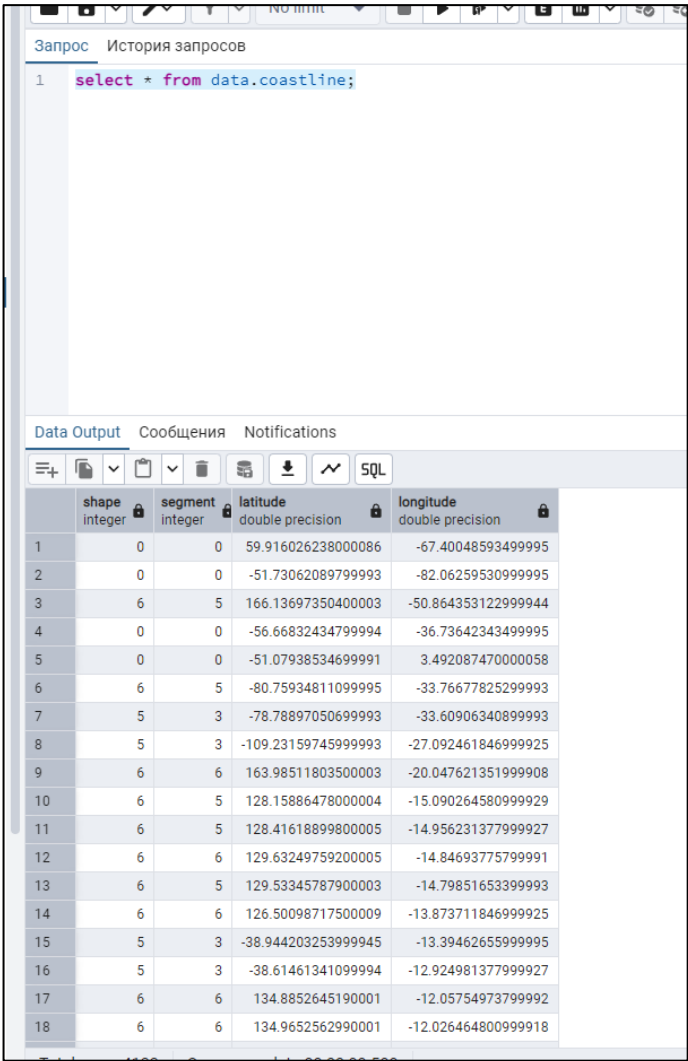


Рисунок 9

Для визуализации данных был создан второй проект на C#. Приложение подключалось к БД, извлекало координаты береговых линий и отображало их на карте мира. Рисунок 10-11

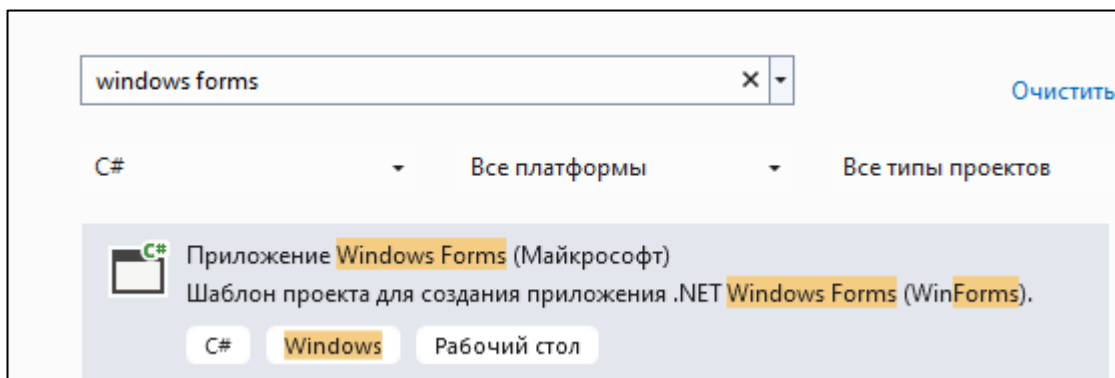


Рисунок 10 -создание проекта

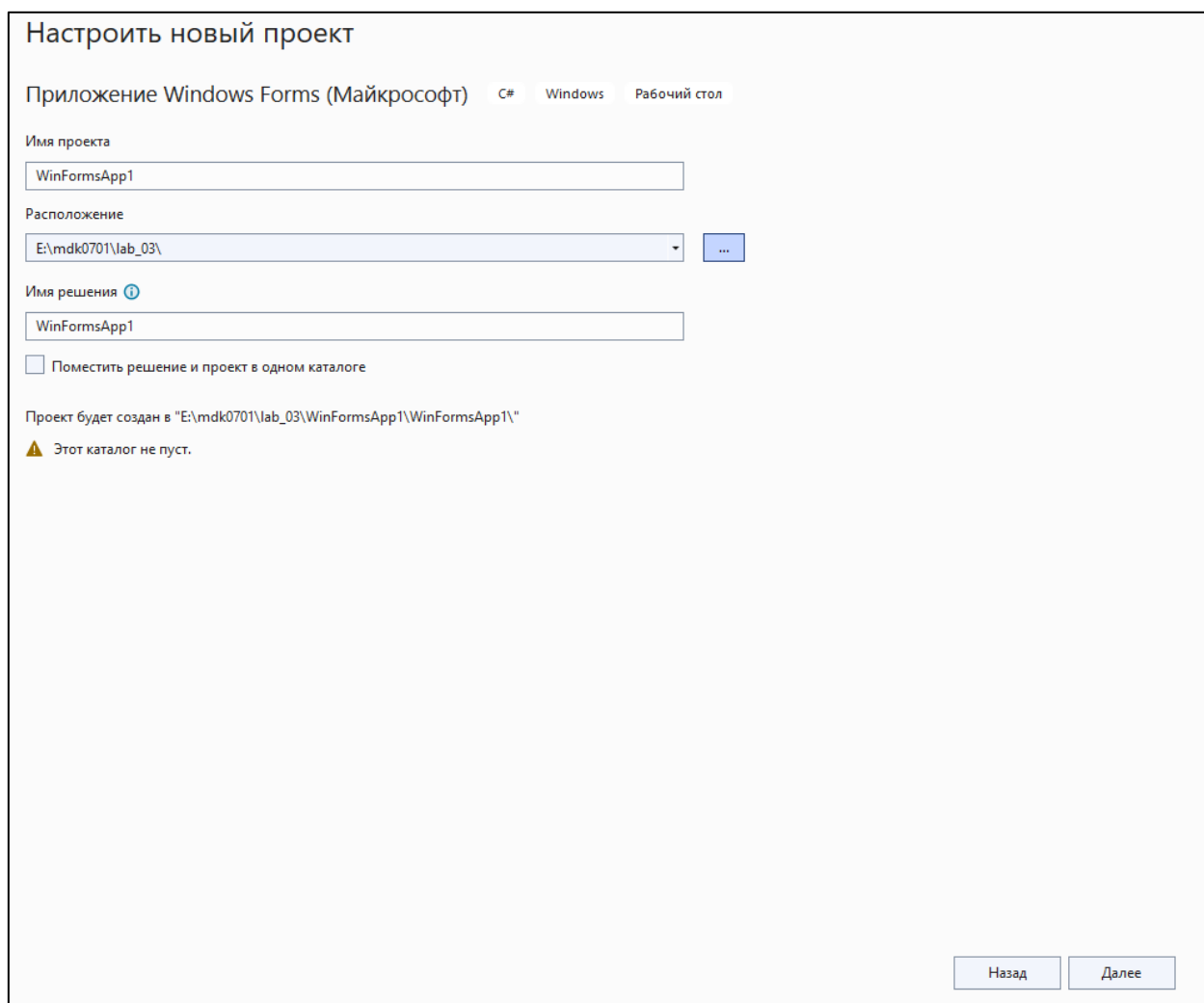


Рисунок 11 -создание проекта



Тут представлен фрагмент кода, в котором указаны порт БД, пароль и название самой БД. С помощью этого сценария приложение выводит карту мира с помощью данных, которые были занесены прошлым скриптом.

Рисунок 12

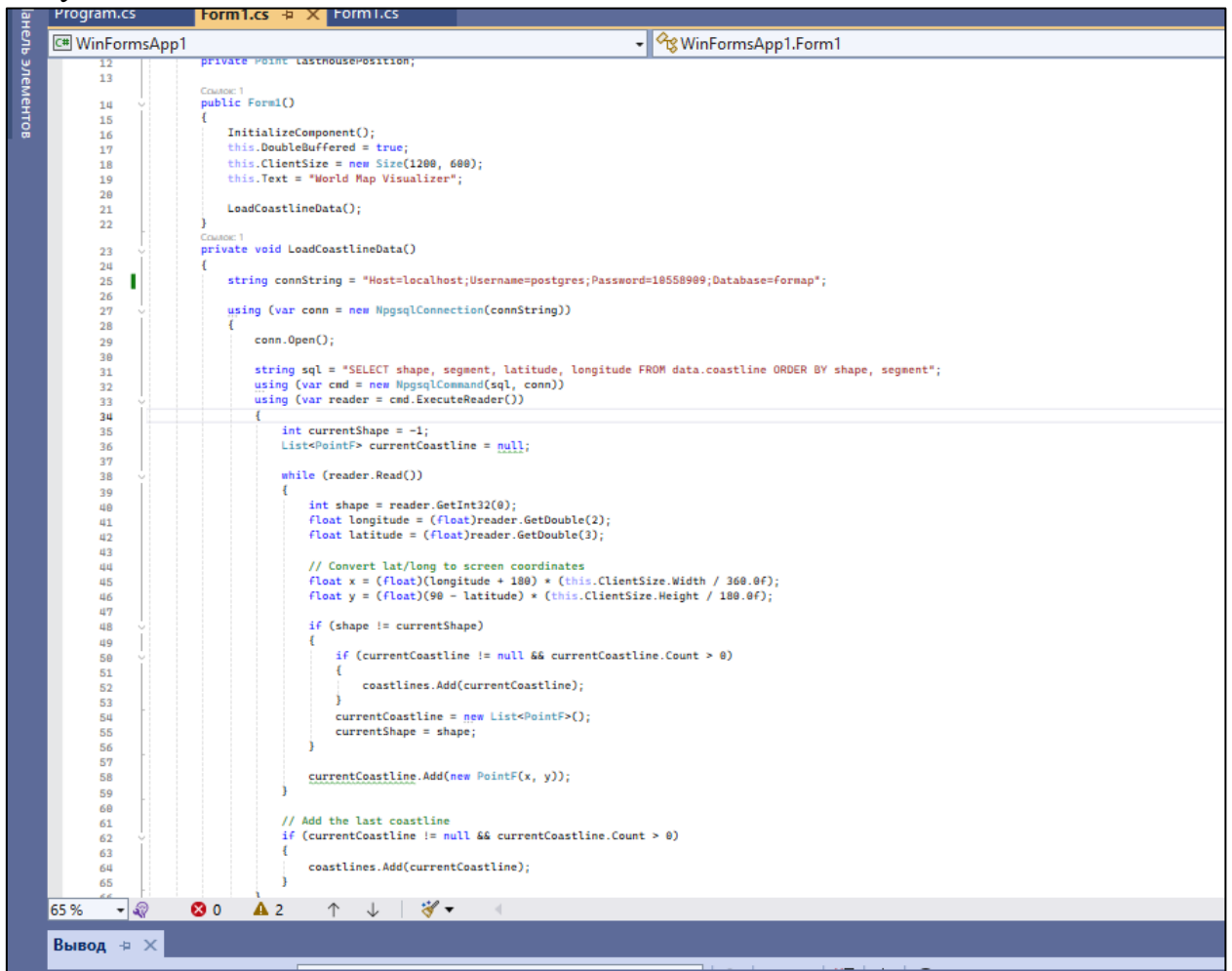


Рисунок 12 -фрагмент кода

Здесь представлен результат работы приложения. Рисунок 13

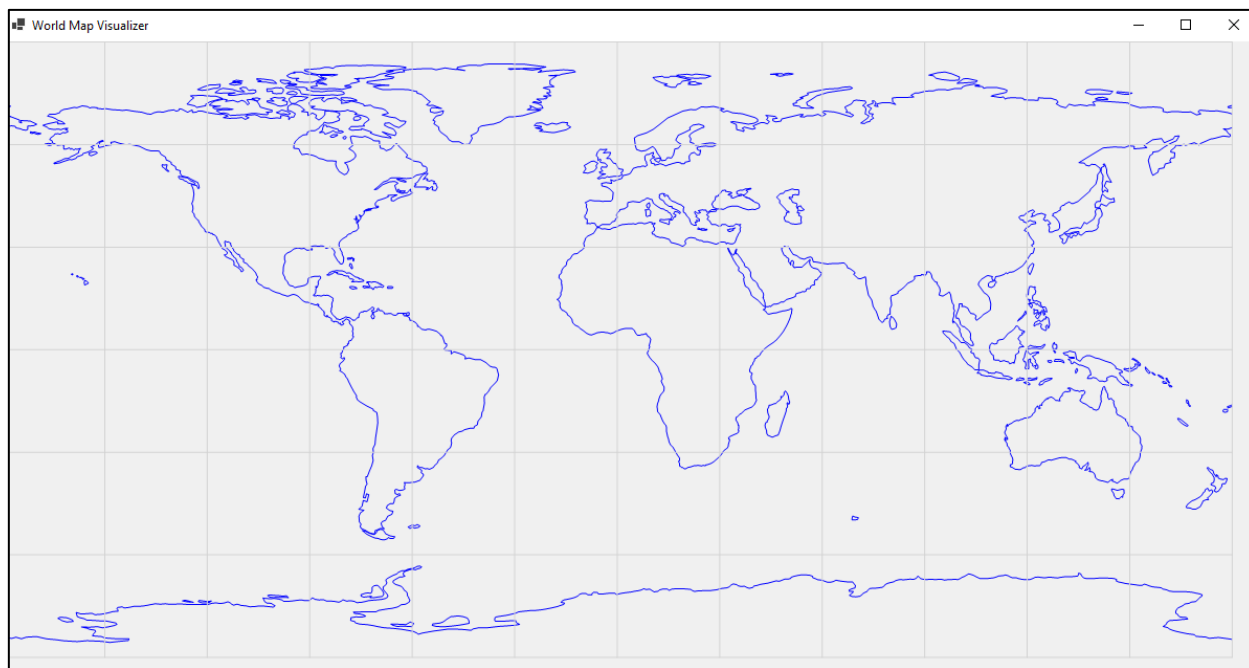


Рисунок 13 -результат

## **Заключение**

В ходе работы была успешно спроектирована база данных для хранения данных береговой линии, выполнены операции импорта данных и разработано приложение для их визуализации. Работа позволила закрепить навыки работы с базами данных, и с программой как Visual Studio.