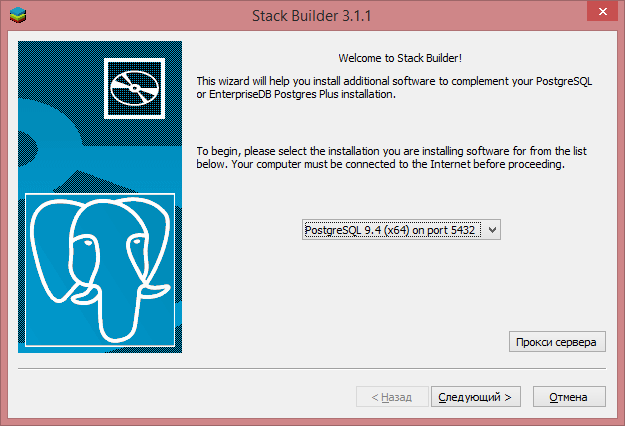
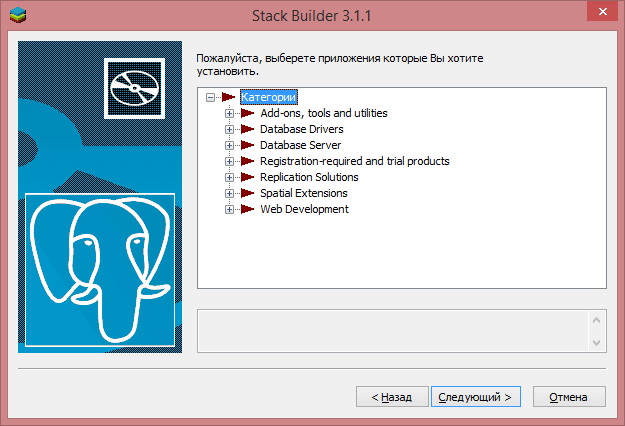
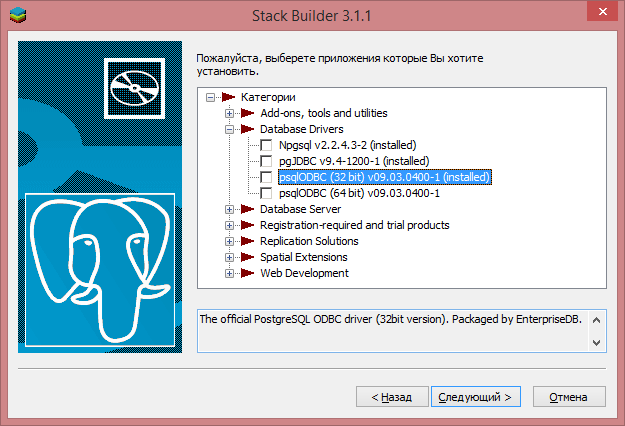
1. Установка драйвера для PostgreSQL
2. Запустить Application Stack Builder из меню пуск (папка PostgreSQL 9.4)



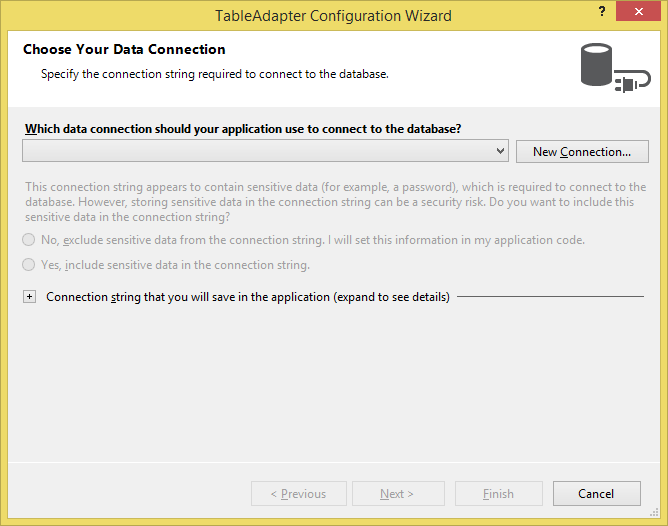
1. В появившемся окне выбрать нужный север и нажать кнопку далее.



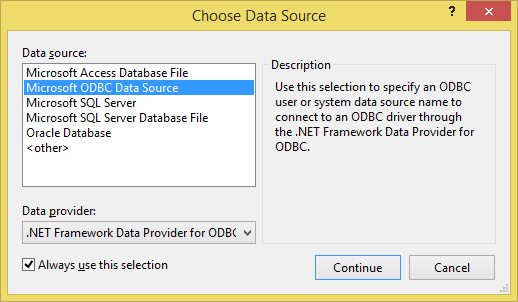
1. Открыть закладку Database Drivers



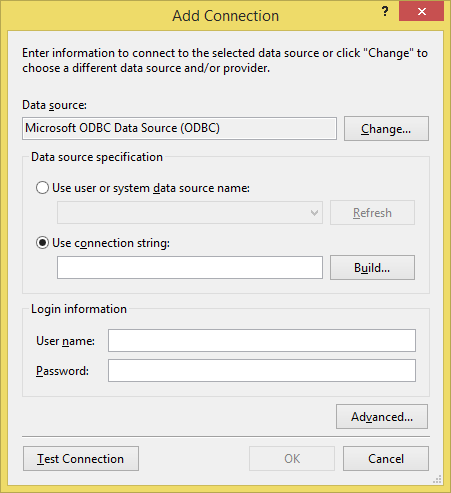
1. Список драйверов:
2. Npgsql – драйвер для .Net (нам нужен этот) (пункт III)
3. pgJDBC – драйвер для Java
4. psglODBC (32 bit) – ODBC драйвер (или этот) (пункт II)
5. psglODBC (64 bit) – ODBC драйвер
6. Выбираем нужные драйвера и нажимаем далее до конца установки.
7. Используя ODBC
   1. Добавить в проект DataSet
   2. Добавить в DataSet TableAdapter



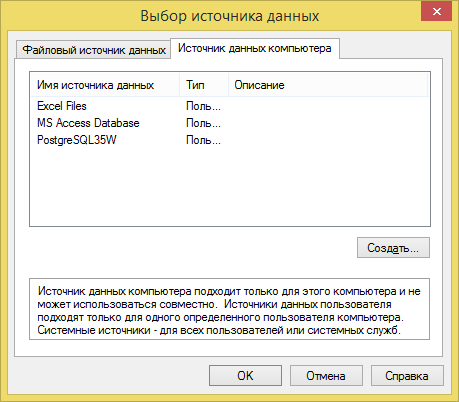
* 1. Нажать на кнопку New Connection



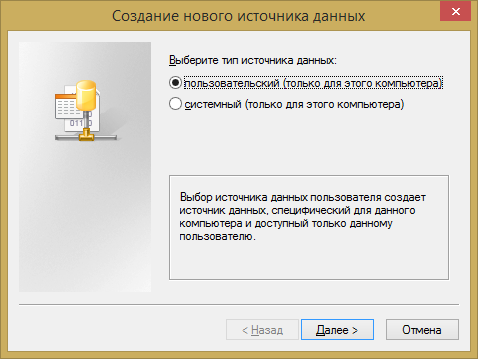
* 1. Выбрать Microsoft ODBC Data Source и нажать Continue



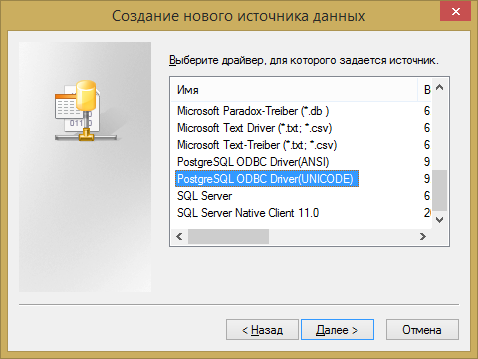
* 1. Выбрать Use connection string и нажать на Build



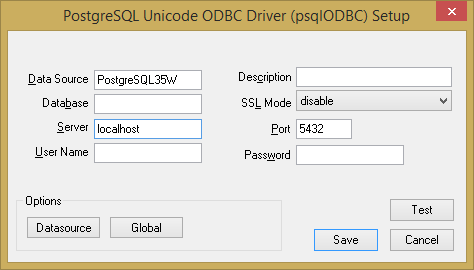
* 1. В открывшемся окне выбираем закладку «Источник данных компьютера» и нажимаем кнопку «Создать»



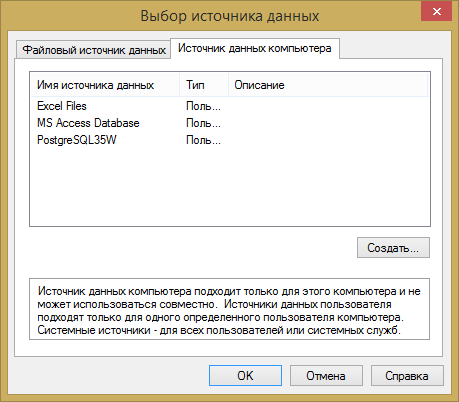
* 1. Выбираем «пользовательский» и нажимаем «Далее»



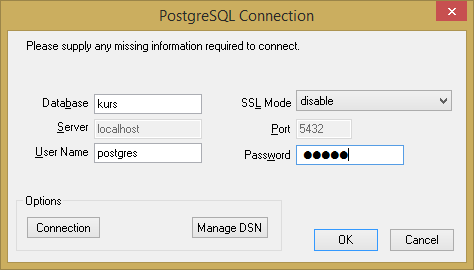
* 1. Выбираем «PostgreSQL ODBC Driver(UNICODE)» и нажимаем «Далее» и в следующем окне «Готово»



* 1. В появившемся окне выставляем параметры:
     1. Data Source – название соединения
     2. Server – адрес сервера
     3. Port – порт сервера (5432 – стандартный порт PostgreSQL)
     4. Остальные параметры не заполняем
  2. Нажимаем кнопку «Save»

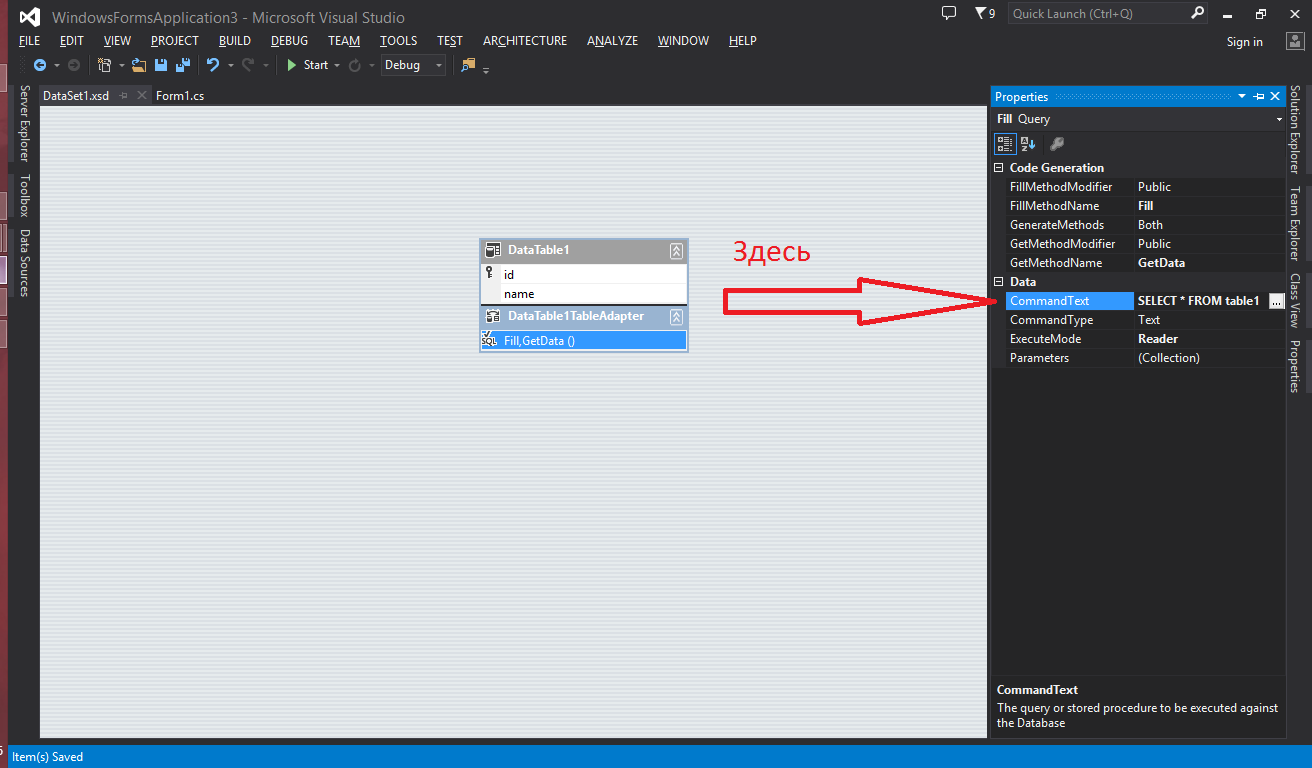


* 1. После этого жмём кнопку «Save», в списке источников данных выбираем созданный нами и жмём «ОК»

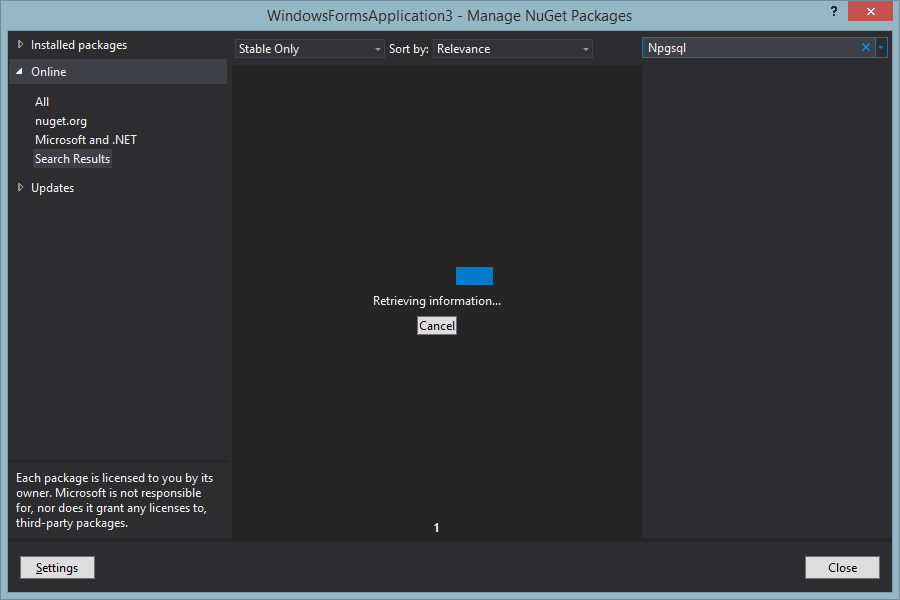


* 1. Этом окне вводим название базы(Database), имя пользователя (User Name) и пароль (Password) и нажимаем кнопку «ОК»

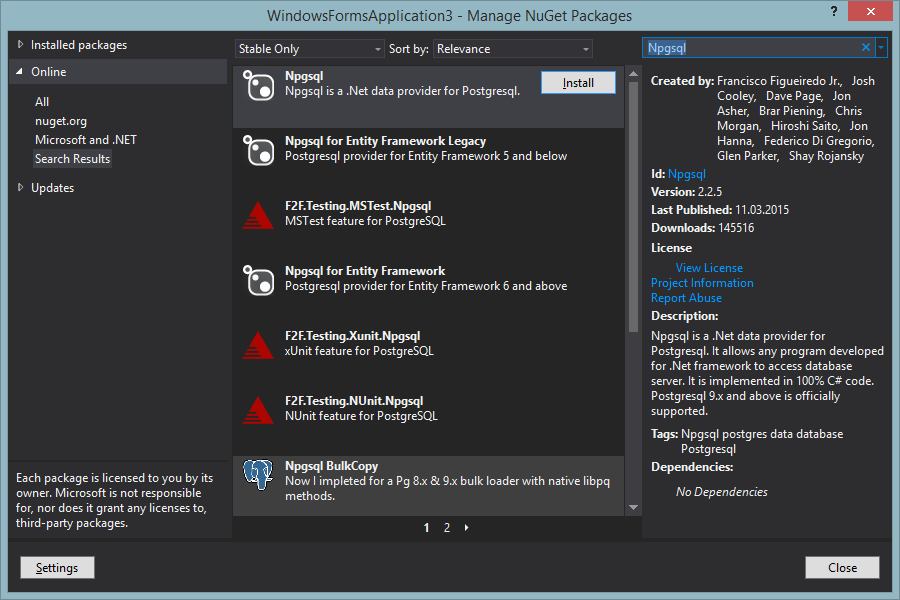
Будьте внимательны: если вы при создании базы имена таблиц, колонок и прочего указывали в двойных кавычках, то мастер запросов будет удалять двойные кавычки, и их нужно будет прописывать самому в свойствах запроса



1. Используя Npgsql
   1. В Solution Explorer нажать правой кнопкой мышки на названии проекта и выбрать Manage NuGet Packages



* 1. Слева выбрать Online и справа вверху в поиске ввести Npgsql



* 1. Установить Npgsql (на скриншоте выше)
  2. Работа с Npgsql

Подключение:

using Npgsql;

using NpgsqlTypes;

Строка соединения для Npgsql

String connectionString = "Server=127.0.0.1;Port=5432;User Id=postgres;Password=admin;Database=test;";

Для Соединения использовать NpgsqlConnection

NpgsqlConnection conn = new NpgsqlConnection(connectionString);

conn.Open();

conn.Close();

Для выполнения запросов использовать NpgsqlCommand:

NpgsqlCommand command = new NpgsqlCommand("SELECT \* FROM table1", conn);

Для выполнения запроса Insert, Update, Create или Delete использовать

command.ExecuteNonQuery();

Для выполнения запроса Select использовать

NpgsqlDataReader dr = command.ExecuteReader();

while (dr.Read())

{

for (int i = 0; i < dr.FieldCount; i++)

{

Console.Write("{0} \t", dr[i]);

}

Console.WriteLine();

}

Выполнение параметрического запроса

NpgsqlCommand command = new NpgsqlCommand("SELECT \* FROM table1 WHERE \"id\" > :val1", conn);

command.Parameters.Add(new NpgsqlParameter("val1", NpgsqlDbType.Integer));

command.Parameters[0].Value = 1;

Вывод данных в таблицу DataGridView

String connectionString = "Server=127.0.0.1;Port=5432;User Id=postgres;Password=admin;Database=test;";

NpgsqlConnection conn = new NpgsqlConnection(connectionString);

conn.Open();

NpgsqlCommand command = new NpgsqlCommand("SELECT \* FROM table1 WHERE \"id\" > :val1", conn);

try

{

command.Parameters.Add(new NpgsqlParameter("val1", NpgsqlDbType.Integer));

command.Parameters[0].Value = 1;

NpgsqlDataAdapter da = new NpgsqlDataAdapter(command);

DataSet ds = new DataSet();

da.Fill(ds);

this.dataGridView1.DataSource = ds.Tables[0];

}

finally

{

conn.Close();

}