МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ автономное ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

(ФГБОУ ВПО МПУ)



Кафедра СМАРТ-технологии

Лабораторная работа № 4

«Работа с базами данных средствами в приложениях на C#»

По дисциплине «Программирование и алгоритмизация на языках высокого уровня»

Группа \_\_\_\_241-324 \_\_\_\_\_\_

№ группы

Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Сальников Л.В.

Подпись студента

Дата \_\_\_27.05.2025\_\_\_\_

Дата сдачи

Преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Крыжановская Т.Г.

Подпись преподавателя

Преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Клецкин А.А.

Подпись преподавателя

2025

Цель:

* Разработать приложение по обработке данных пользователя, размещенных во внешней БД Postgres. Создание структуры базы данных и первичное заполнение выполнить с помощью postgers (по 10-15 записей для каждой таблицы).
* Подготовить приложение на языке C# по работе с базой данных через ADO (OLE). Просмотр и редактирование данных таблиц реализовать с помощью отдельных форм приложения.

Выполнение задания:

**1. Подготовка базы данных PostgreSQL**

* Создал базу данных.
* Создал таблицы enterprises и products согласно заданию:
  + enterprises: enterprise\_code (PK), name, phone.
  + products: product\_id (PK), enterprise\_code (FK), product\_name, production\_jan, price\_jan, production\_dec, price\_dec.
* Заполнил таблицы тестовыми данными (5 предприятий, 20 продуктов).

Создание таблицы предприятий:

--Таблица предприятий

CREATE TABLE enterprises (

enterprise\_code SERIAL PRIMARY KEY, -- Код предприятия (автоинкремент)

name VARCHAR(100) NOT NULL, -- Название предприятия

phone VARCHAR(20) -- Телефон предприятия

);

Создание таблицы продуктов:

--Таблица продукции

CREATE TABLE products (

product\_id SERIAL PRIMARY KEY, -- Уникальный код продукции (автоинкремент)

enterprise\_code INT NOT NULL, -- Код предприятия (внешний ключ)

product\_name VARCHAR(100) NOT NULL, -- Название (вид) продукции

production\_jan INT, -- Выпуск в январе

price\_jan NUMERIC(10, 2), -- Цена за единицу в январе

production\_dec INT, -- Выпуск в декабре

price\_dec NUMERIC(10, 2), -- Цена за единицу в декабре

CONSTRAINT fk\_enterprise FOREIGN KEY (enterprise\_code) REFERENCES enterprises(enterprise\_code)

);

Заполнение таблицы предприятий:

INSERT INTO enterprises (name, phone) VALUES

('Предприятие А', '123-456-7890'),

('Предприятие Б', '234-567-8901'),

('Предприятие В', '345-678-9012'),

('Предприятие Г', '456-789-0123'),

('Предприятие Д', '567-890-1234');

Заполнение таблицы продуктов:

INSERT INTO products (enterprise\_code, product\_name, production\_jan, price\_jan, production\_dec, price\_dec) VALUES

(1, 'Продукт 1', 100, 10.00, 120, 12.00),

(1, 'Продукт 2', 200, 15.00, 210, 18.00),

(1, 'Продукт 3', 150, 20.00, 140, 22.00),

(1, 'Продукт 4', 300, 25.00, 310, 28.00),

(2, 'Продукт 1', 80, 9.00, 90, 10.50),

(2, 'Продукт 5', 100, 12.00, 110, 13.50),

(2, 'Продукт 6', 60, 18.00, 70, 20.00),

(2, 'Продукт 7', 200, 22.00, 190, 23.00),

(3, 'Продукт 2', 120, 15.50, 130, 17.00),

(3, 'Продукт 8', 50, 10.00, 60, 11.00),

(3, 'Продукт 9', 90, 8.00, 85, 8.50),

(3, 'Продукт 10', 70, 13.00, 75, 14.00),

(4, 'Продукт 3', 110, 19.00, 120, 20.00),

(4, 'Продукт 11', 130, 21.00, 140, 23.00),

(4, 'Продукт 12', 60, 17.00, 65, 18.00),

(4, 'Продукт 13', 80, 14.00, 85, 15.00),

(5, 'Продукт 1', 140, 11.00, 150, 12.50),

(5, 'Продукт 14', 100, 16.00, 110, 17.50),

(5, 'Продукт 15', 90, 20.00, 95, 21.00),

(5, 'Продукт 16', 70, 18.00, 75, 19.50);

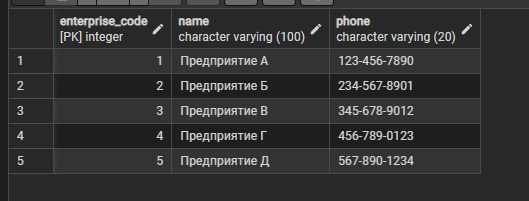
****

Рисунок 1

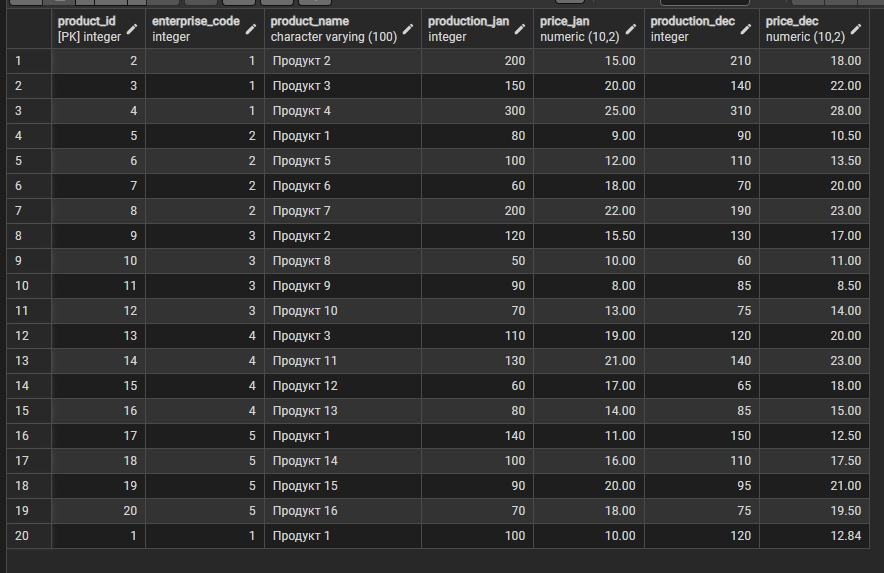


Рисунок 2

**2. Создание проекта C# Windows Forms**

* Создал новый проект Windows Forms в Visual Studio.
* Добавил необходимые NuGet-пакеты:
  + Npgsql — драйвер PostgreSQL для .NET.
* Создал структуру проекта:
  + Основная форма (MainEnterpriceForm) для отображения всех функций;

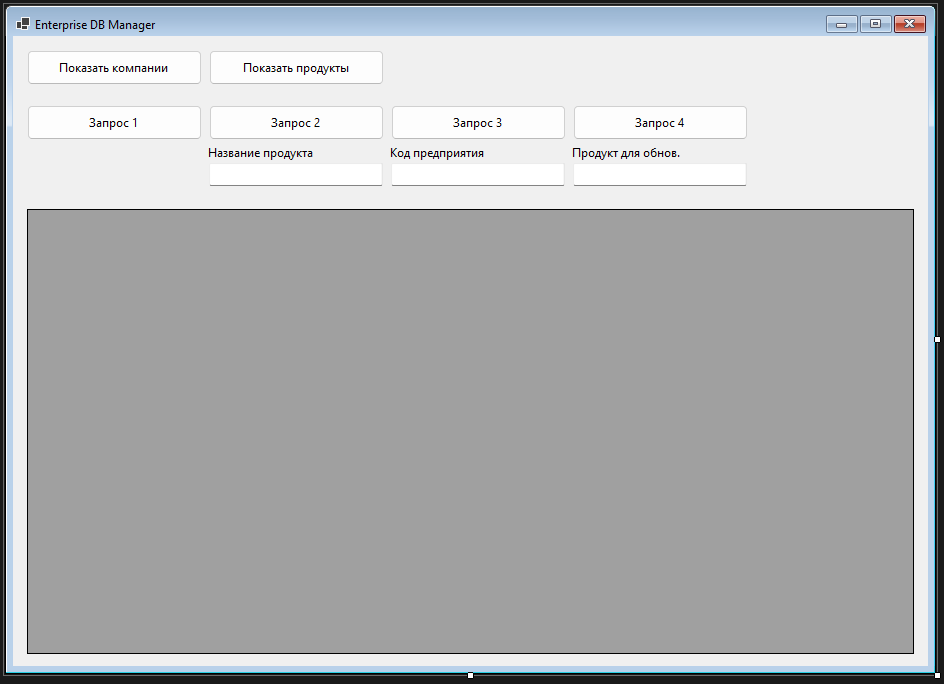


Рисунок 3

* + Форма управления компаниями (CompanyForm) для работы с компаниями;

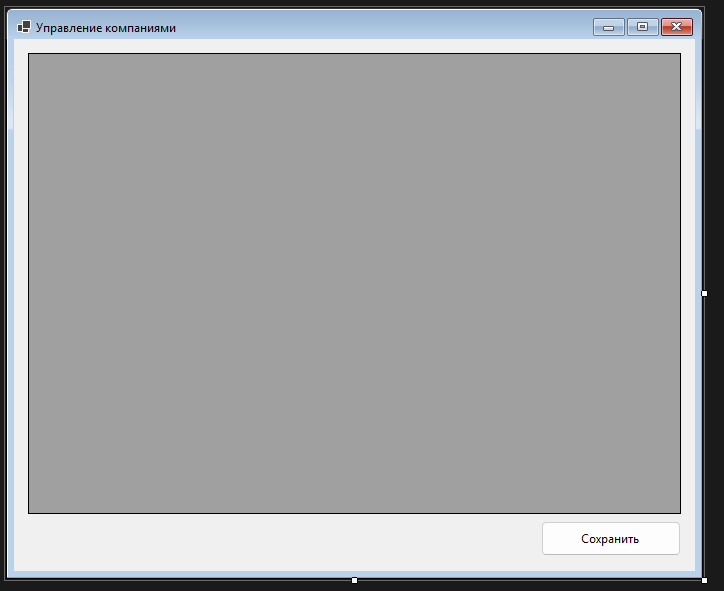


Рисунок 4

* + Форма управления продуктами (ProductForm) для работы с продуктами.

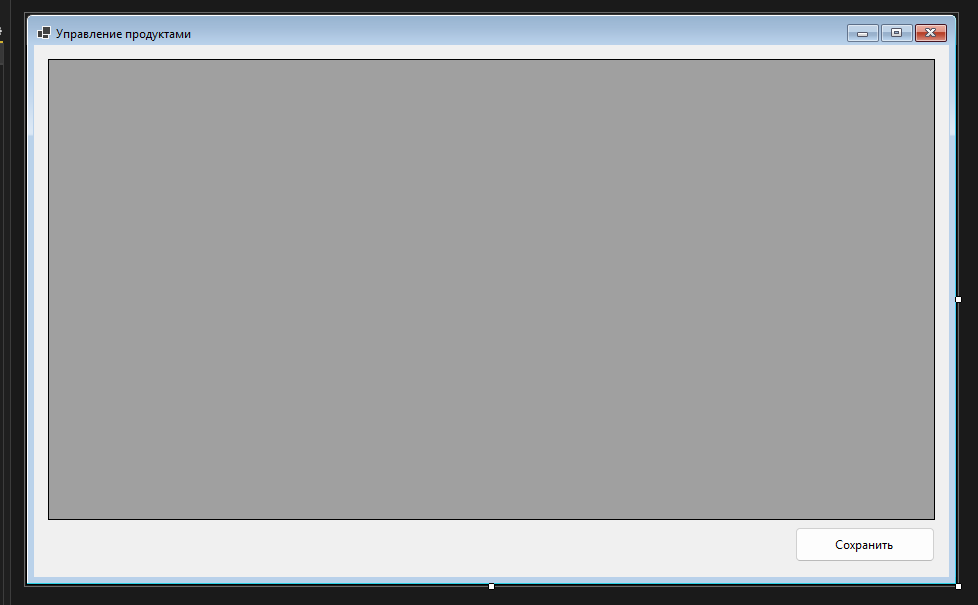


Рисунок 5

**3. Настройка подключения к базе данных**

* Добавил класс DatabaseHelper для управления подключениями и выполнением запросов.
* В DatabaseHelper прописал строку подключения с параметрами к базе PostgreSQL.
* Реализовал методы для выполнения SELECT-запросов и обновлений.

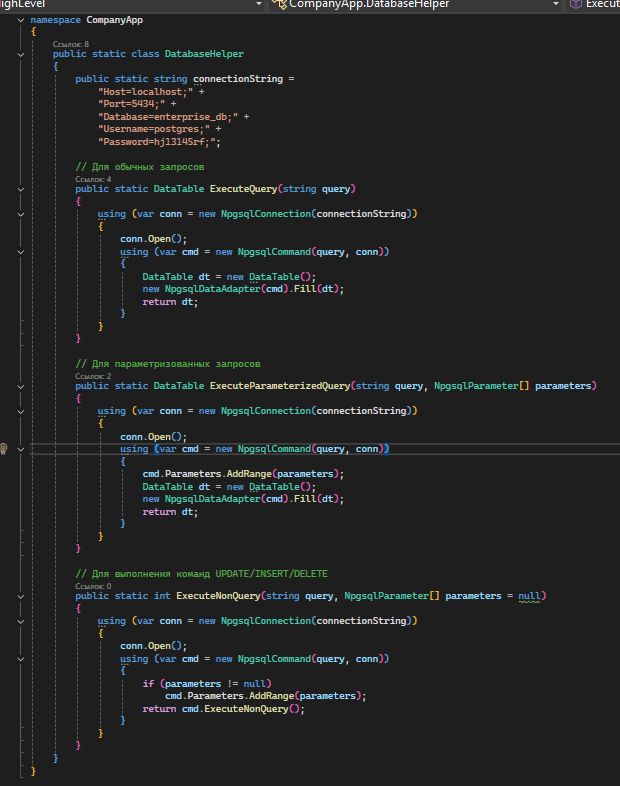


Рисунок 6

**4. Загрузка и отображение данных**

* При загрузке формы загружаются данные из таблицы enterprises и отображаются в DataGridView.
* Позволяет пользователю редактировать данные прямо в таблице.
* Осуществляется проверка ввода (валидация), например, формат телефона.

**5. Сохранение изменений в базе данных**

* При нажатии кнопки "Сохранить" фиксируются изменения из DataGridView в базе через NpgsqlDataAdapter и NpgsqlCommandBuilder.
* Обрабатываются возможные ошибки при сохранении.

**6. Реализация специальных запросов из задания**

* Добавил отдельные кнопки или формы для выполнения запросов:
  + Предприятия с увеличением выпуска и цены в декабре
  + Предприятия с ограниченным ростом цены на конкретный продукт
  + Количество видов продукции и средняя цена декабря по предприятиям
  + Увеличение цены на 7% для заданного продукта и предприятия
* Выводится результаты запросов в DataGridView или отдельном окне.

Вывод:

* Разработал приложение по обработке данных пользователя, размещенных во внешней БД Postgres. Создание структуры базы данных и первичное заполнение выполнил с помощьюpg Admin (по 10-15 записей для каждой таблицы).
* Подготовил приложение на языке C# по работе с базой данных через ADO (OLE). Просмотр и редактирование данных таблиц реализовал с помощью отдельных форм приложения.