Федеральное государственное автономное

образовательное учреждение

Высшего образования

«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

|  |
| --- |
| Институт космических и информационных технологий |
| институт |
| Информационных систем |
| кафедра |

**ОТЧЕТ О ПРАКТИЧЕСКОЙ РАБОТЕ**

|  |
| --- |
| Оконное приложение |
| тема |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Преподаватель | |  |  |  | И. А. Ефремов |
|  | |  | подпись, дата |  | инициалы, фамилия |
| Студент | КИ21-22Б, 032157041 |  |  |  | И. А. Дуденко |
|  | номер группы, зачётной книжки |  | подпись, дата |  | инициалы, фамилия |

Красноярск 2023

**Ход выполнения работы**

##### **Задание:**

##### Необходимо создать оконное приложение "Склад" (используя любой язык программирования), которое будет подключаться к серверу PostgreSQL и конкретно к базе Northwind. В оконном приложении необходимо вывести все данные из таблицы products, а также настроить возможность работы с этими данными.

##### Необходимо создать функцию добавления нового продукта, удаления старого продукта (по id номеру), поиска продукта (продуктов) по нескольким параметра:

##### по category\_id (номеру категории),

##### по product\_name (названию продукта),

##### по discontinued (продаваемые/приостановленные).

Для выполнения работы было принято решение разработать приложение WinForm на языке программирования C#.

Для работы с базой данных для начало необходимо подключиться к ней (листинг 1).

private string connectionString = String.Format("Server=localhost;Port=5432;" +

"User Id=postgres;Password=sudlenia;Database=Northwind");

private NpgsqlConnection connection;

NpgsqlCommand cmd = new NpgsqlCommand();

public Form1()

{

InitializeComponent();

connection = new NpgsqlConnection(connectionString);

connection.Open();

}

Листинг 1 – подключение к базе данных

Далее необходимо было разработать интерфейс (рисунок 1) и вывести все данные из таблицы products – вывод осуществлялся с помощью метода Select (листинг 2).

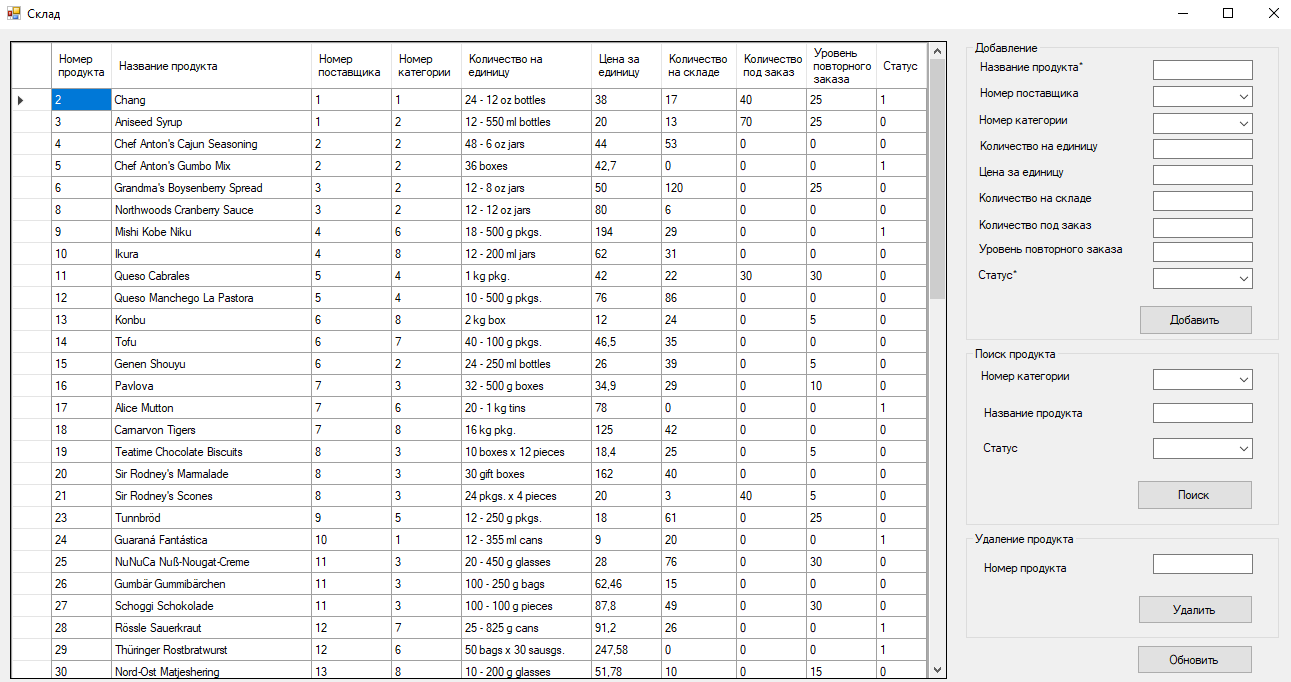


Рисунок 1 – интерфейс приложения

private void Select(string query)

{

cmd = new NpgsqlCommand(query, connection);

NpgsqlDataReader reader = cmd.ExecuteReader();

dataGridView1.Rows.Clear();

while (reader.Read())

{

var product\_id = reader[0];

var product\_name = reader[1];

var supplier\_id = reader[2];

var category\_id = reader[3];

var quantity\_per\_unit = reader[4];

var unit\_price = reader[5];

var units\_in\_stock = reader[6];

var units\_in\_order = reader[7];

var reorder\_lever = reader[8];

var discontinued = reader[9];

dataGridView1.Rows.Add(product\_id, product\_name, supplier\_id, category\_id, quantity\_per\_unit, unit\_price, units\_in\_stock, units\_in\_order,

reorder\_lever, discontinued);

}

reader.Close();

}

Листинг 2 – метод Select

Удаление продуктов осуществляется по полю product\_id. Удаление так же происходит и в таблице order\_details для того, чтобы избежать ошибки при удалении.

string query = $"delete from order\_details where product\_id = {parsedNumber};" +

$"delete from products where product\_id={parsedNumber}";

cmd = new NpgsqlCommand(query, connection);

cmd.ExecuteNonQuery();

Листинг 3 – запрос для удаления из таблицы

Поиск происходит по полям category\_id, product\_name, discontinued. Для удобства поиска вылезает список имеющихся данных в базе данных (рисунок 2).

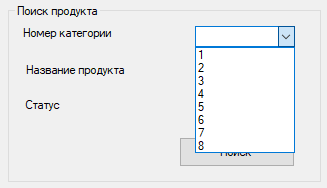


Рисунок 2 – демонстрация списка данных

Поиск данных осуществляется по всем введённым пользователем данных. В листинге 4 показана часть как собирается запрос для поиска.

string query = "select \* from products where (";

if (comboBox2.SelectedItem != null)

{

query += $"(category\_id='{comboBox2.Text}')";

}

Листинг 4 – часть запроса для поиска данных

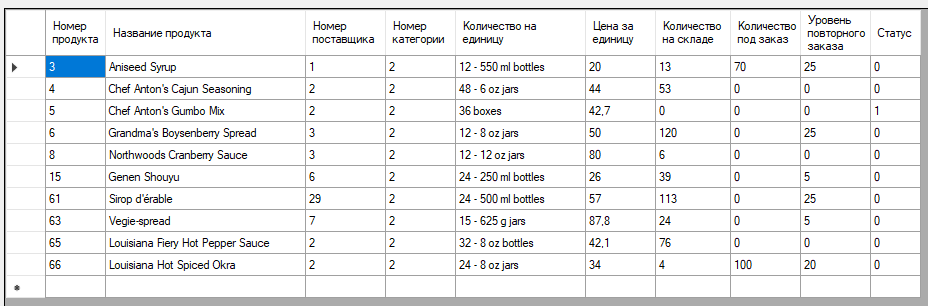


Рисунок 3 – результат поиска по полю category\_id = 2

Добавление товара осуществляется аналогично, как и в функции поиска, т.е. добавляются только те данные, которые ввёл пользователь. Присутствует проверка на то, что некоторые значения в базе данных не должны иметь null значение (рисунок 4), а также что в числовых полях не может быть текст (рисунок 5 и листинг 5). На рисунке 6 можно увидеть демонстрацию функции добавления в таблицу products.

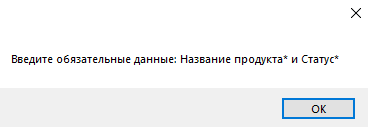


Рисунок 4 – проверка на null

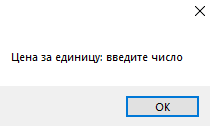


Рисунок 5 – проверка на числовое значение

if (textBox6.Text != "")

{

if (double.TryParse(textBox6.Text.Replace('.', ','), out double unit))

{

string \_unit = unit.ToString().Replace(',', '.');

values += $"{\_unit}, ";

query += "unit\_price, ";

}

else

{

flag = false;

MessageBox.Show("Цена за единицу: введите число");

}

}

Листинг 5 – проверка на числовое значение

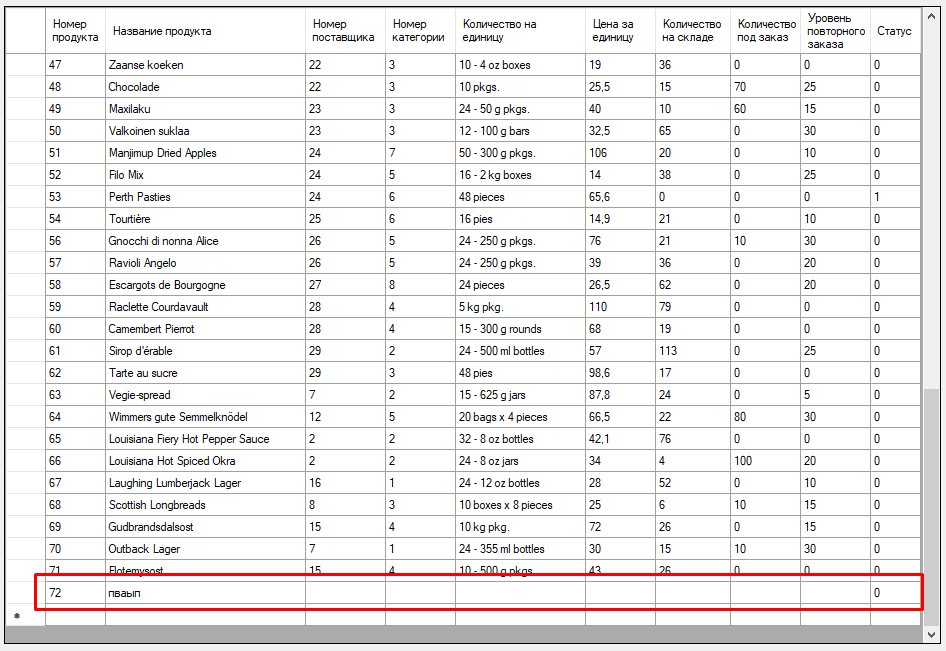


Рисунок 6 – добавление в таблицу products

Так же в приложении присутствует кнопка «Обновить». Она была предусмотрена для сброса поиска и на случай, если приложением «Склад» будут пользоваться два человека для того, чтобы видеть актуальные данные.