

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧЕРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВАСТОПОЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

Институт информационных технологий
кафедра «Информационные системы»

Пояснительная записка

К расчетно-графической работе
по дисциплине «Управление данными»
на тему «Разработка базы данных предприятия
по производству молочной продукции»

Выполнил: студент III курса, группы: ИС/б-20-1-о

Направления подготовки (специальности) 09.03.02

Информационные системы и технологии

Профиль (специализация)

Хроменко Данил Алексеевич

Руководитель

Защита «__» ____ 2023 г.

Оценка _____

Руководитель

2023 г.

СОДЕРЖАНИЕ

| | |
|--|----|
| ВВЕДЕНИЕ | 3 |
| 1 АНАЛИТИЧЕСКАЯ ЧАСТЬ..... | 4 |
| 1.1 Анализ предметной области (концептуальное моделирование) | 4 |
| 1.2 Постановка задачи | 4 |
| 2 РАЗРАБОТКА ЛОГИЧЕСКОЙ МОДЕЛИ БАЗЫ ДАННЫХ | 6 |
| 2.1 Построение диаграммы «сущность-связь» в нотации П.Чена | 6 |
| 2.2 Построение модели, основанной на ключах (Key Based model)..... | 7 |
| 2.3 Построение полной атрибутивной модели в нотации IDEF1X..... | 8 |
| 3 РАЗРАБОТКА ФИЗИЧЕСКОЙ МОДЕЛИ БАЗЫ ДАННЫХ | 10 |
| 3.1 Выбор аппаратной и программной платформы для реализации БД.... | 10 |
| 3.2 Реализация базы данных | 10 |
| 3.3 Тестирование базы данных | 11 |
| 4 РАЗРАБОТКА КЛИЕНТСКОГО ПРИЛОЖЕНИЯ | 22 |
| 4.1 Обоснование выбора языков программирования | 22 |
| 4.2 Разработка интерфейса пользователя | 22 |
| 4.3 Алгоритм работы каждого из модулей..... | 23 |
| 4.4 Тестирование работы приложения | 23 |
| ЗАКЛЮЧЕНИЕ | 34 |
| СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ | 35 |
| ПРИЛОЖЕНИЕ А. | 36 |
| ПРИЛОЖЕНИЕ Б. | 38 |

ВВЕДЕНИЕ

В настоящее время информационные системы являются неотъемлемой частью управления производством и продажами во многих компаниях. Создание эффективной базы данных позволяет автоматизировать процессы управления, повысить эффективность работы и сократить время на выполнение рутинных задач. В рамках данной расчетно-графической работы была поставлена задача разработки базы данных для предприятия по производству молочной продукции.

Целью работы является проектирование и реализация базы данных, а также разработка десктопного приложения для ее использования. В базе данных будут храниться данные о клиентах, заказах, продуктах, поставщиках и поставках сырья. Благодаря этому предприятие сможет эффективнее управлять своими ресурсами и повысить качество обслуживания клиентов.

В ходе работы будут рассмотрены логическая и физическая модели базы данных, а также способы реализации связей между таблицами. Будет создано десктопное приложение для управления базой данных и предоставления доступа к ее функционалу. Результатом работы будет готовая база данных и работающее приложение, которые можно использовать для управления производством и продажами на предприятии по производству молочной продукции.

1 АНАЛИТИЧЕСКАЯ ЧАСТЬ

1.1 Анализ предметной области (концептуальное моделирование)

Для анализа предметной области «производство молочной продукции» были изучены документы, связанные с процессом производства и продажи продукции. Из этих документов были выявлены основные объекты (сущности ПрО) и их атрибуты, которые необходимо хранить в базе данных.

Кроме того, было проведено общение со специалистами данной предметной области, в результате которого были получены сведения об её особенностях. Эти сведения помогли установить ограничения целостности, зависимости и связи между объектами (субъектами) предметной области.

Специалисты также предоставили информацию о том, какие алгоритмы обработки данных используются в данной предметной области и какие задачи ставятся перед информационной системой. Эта информация будет использована для разработки логической и физической модели базы данных.

1.2 Постановка задачи

Была изучена предметная область предприятия по производству молочной продукции. Основными сущностями, информация о которых должна храниться в базе данных, являются: клиенты, продукты, поставщики, заказы и поставки сырья.

Были определены связи между сущностями: «клиент» может сделать заказ на интересующие его продукты, «поставщик» может осуществить поставку сырья для производства продукции предприятием, «заказ» содержит информацию о клиенте и заказываемом продукте, «поставка» сырья содержит информацию о

поставщике и поставляемом сырье, «продукт» содержит включает в себя единицы продукции, производимой и получаемой предприятием.

Будут использоваться две группы пользователей разрабатываемой базы данных: Администратор базы данных и начальник отдела продаж. Администратор базы данных будет иметь доступ к управлению базой данных, в том числе добавлению, изменению и удалению записей в таблицах. Начальник отдела продаж будет иметь доступ к просмотру и управлению заказами и поставками.

Требования к функциональности системы должны удовлетворять потребности обеих групп пользователей и обеспечивать эффективное и надежное управление информацией о производстве молочной продукции.

2 РАЗРАБОТКА ЛОГИЧЕСКОЙ МОДЕЛИ БАЗЫ ДАННЫХ

2.1 Построение диаграммы «сущность-связь» в нотации П.Чена

Для разработки логической модели базы данных предприятия по производству молочной продукции были изучены предметная область и основные бизнес-правила, а также был проведен анализ таблиц сущностей.

Для построения диаграммы «сущность-связь» были выбраны основные сущности и связи между ними, которые удовлетворяют основным требованиям. Исходя из анализа таблиц, в диаграмму были включены следующие сущности: «Клиент», «Продукт», «Поставщик», «Заказ», «Поставка сырья» и «Детали поставки».

Для удобства чтения диаграммы и ее понимания, были добавлены названия сущностей и связей, а также атрибуты каждой сущности. Все связи и сущности с их атрибутами изображены на рисунке 2.1.

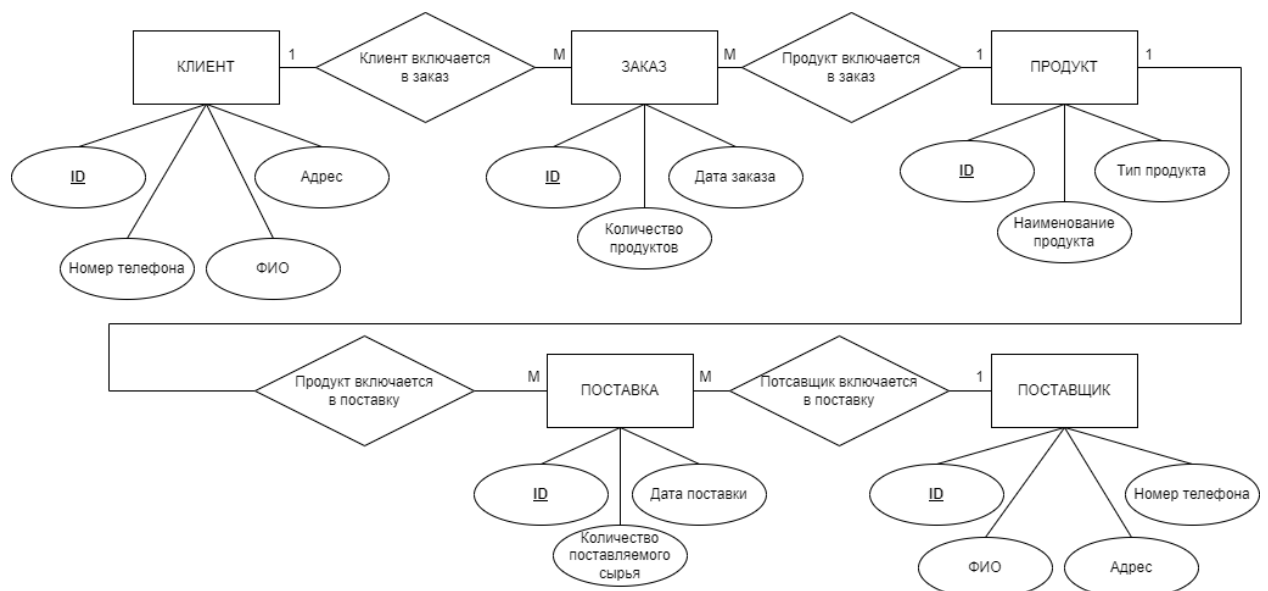


Рисунок 2.1 – Диаграмма сущность-связь

2.2 Построение модели, основанной на ключах (Key Based model)

В рамках раздела для разработки базы данных предприятия по производству молочной продукции были изучены сущности, которые охватывают область производства продукции, а именно: «Клиент», «Продукт», «Поставщик», «Заказ», «Детали поставки» и «Поставка». Целью модели, основанной на ключах, является предоставление обзора структур данных и ключей, необходимых для поддержки определенной области. Для каждой сущности были определены ее атрибуты, первичные и вторичные ключи.

Была построена модель, основанная на ключах (рисунок 2.2). Эта модель описывает основные структуры данных, необходимые для поддержки определенной области. В модели были включены все сущности, первичные и вторичные ключи вместе с примерами атрибутов. Она позволила определить контекст, в котором могут быть созданы подробные модели для конкретного воплощения.

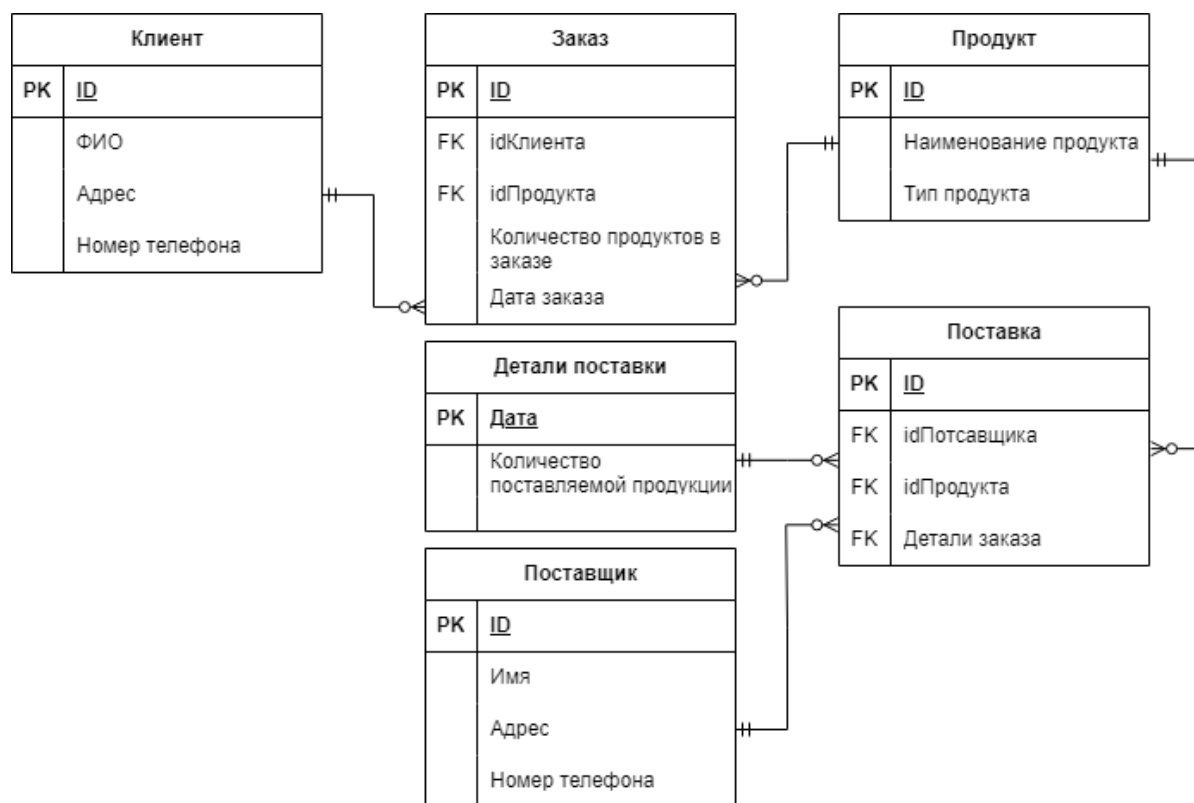


Рисунок 2.2 – Модель основанная на ключах

Таким образом, было проведено построение модели базы данных, основанной на ключах, которая будет использоваться в дальнейшем при разработке базы данных предприятия по производству молочной продукции.

2.3 Построение полной атрибутивной модели в нотации IDEF1X

Полученная в пункте 2.2 модель была подвергнута нормализации до 3НФ. Ниже описано приведение исходной модели базы данных к каждой из нормальных форм.

Первая нормальная форма требует атомарности каждого из атрибутов отношения. Так как заказ может иметь множество дополнительных требований, то отношение «Поставка» не находится в 1НФ. Для приведения к 1НФ требуется разбить это отношение на два: Поставка (ID Поставки, ID Поставщика, ID Продукта, Количество продуктов, Время поставки); Поставщик (ID Поставщика, Название, Номер телефона, Адрес). Теперь отношения находятся в 1НФ.

Вторая нормальная форма требует нахождения отношений в 1НФ и полной зависимости каждого не ключевого атрибута от каждого ключа. В каждом отношении каждый атрибут зависит от каждого ключа, значит отношения находятся в 2НФ.

Третья нормальная форма требует нахождения отношения в 2НФ и не транзитивной зависимости каждого не ключевого атрибута от первичного ключа. В каждом отношении отсутствуют транзитивные зависимости, значит они находятся в 3НФ.

После проведения нормализации была построена полная атрибутивная модель в нотации IDEF1X (рисунок 2.3).

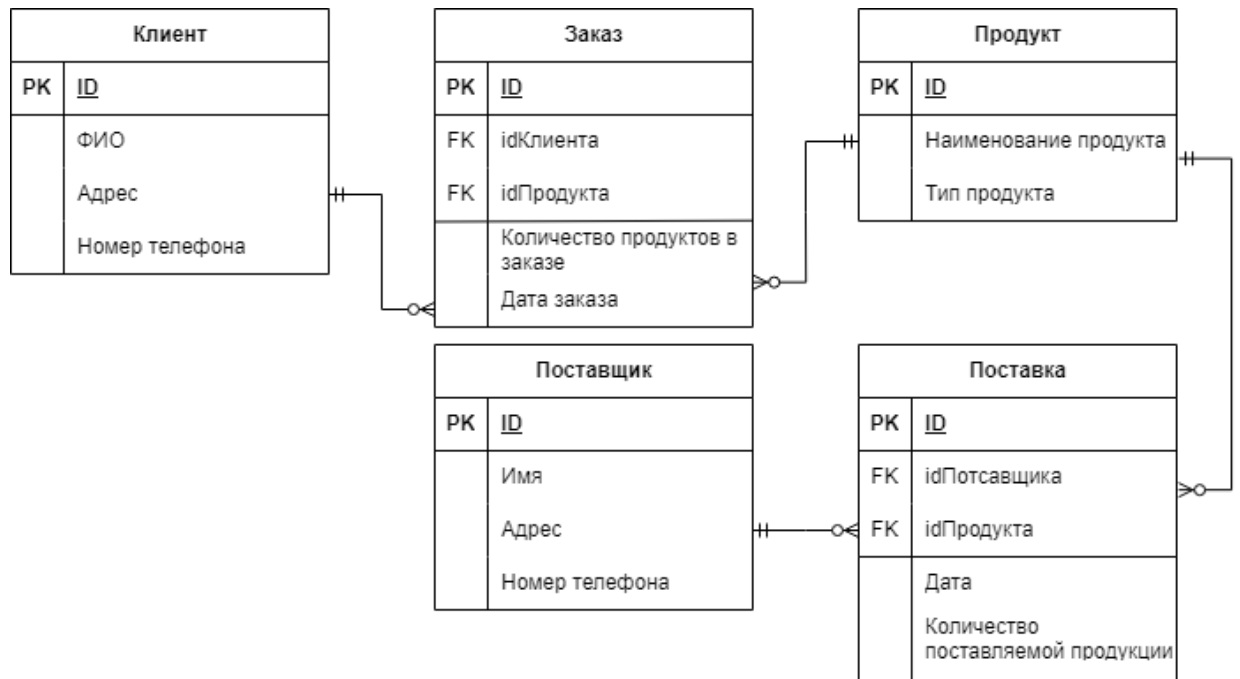


Рисунок 2.3 – Полная атрибутивная модель в нотации IDEF1X

Результатом проведенной нормализации является построение полной атрибутивной модели в нотации IDEF1X, которая включает в себя 5 таблиц, оптимизированных и приведенных к 3 нормальной форме. Данные таблицы будут использоваться в дальнейшем при проектировании физической модели базы данных и реализации десктопного приложения для неё.

3 РАЗРАБОТКА ФИЗИЧЕСКОЙ МОДЕЛИ БАЗЫ ДАННЫХ

3.1 Выбор аппаратной и программной платформы для реализации БД

Для реализации базы данных предприятия по производству молочной продукции была выбрана СУБД PostgreSQL. Этот выбор был сделан после сравнительного анализа нескольких вариантов, таких как firebirdSQL и MySQL.

PostgreSQL был выбран, так как он имеет ряд преимуществ по сравнению с другими СУБД. В частности, PostgreSQL имеет более широкие возможности по работе с большими объемами данных, более высокую производительность и лучшую поддержку многопоточности. Кроме того, PostgreSQL обладает более продвинутым механизмом обработки транзакций, что обеспечивает высокую надежность и целостность данных.

Для реализации физической схемы базы данных были использованы стандартные типы данных, поддерживаемые PostgreSQL, такие как INTEGER, VARCHAR, DATE и др. Тип данных для каждого атрибута был выбран на основе его предназначения и возможностей, предоставляемых СУБД.

Таким образом, выбор PostgreSQL для реализации базы данных предприятия по производству молочной продукции был обоснован его преимуществами по сравнению с другими СУБД и соответствием требованиям предметной области.

3.2 Реализация базы данных

На основании спроектированной логической модели базы данных была построена физическая модель, которая была реализована в виде соответствующих таблиц в базе данных.

В таблице «Клиент» содержатся данные о покупателе, включая id, ФИО, адрес и номер телефона. Такая таблица была выбрана для того, чтобы иметь возможность отслеживать информацию о покупателе, его контактных данных и местоположении.

В таблице «Продукт» содержится информация о заказанном продукте, включая id, наименование продукта и тип продукта. Такая таблица была выбрана для хранения информации о продукции, которую производит и получает молокозавод.

В таблице «Поставщик» содержатся данные о поставщике, включая id поставщика, имя, адрес и номер телефона. Такая таблица была выбрана для того, чтобы иметь возможность отслеживать информацию о поставщике сырья.

В таблице «Заказ» содержатся данные о заказе продукции, включая номер заказа (ID), id клиента, id продукта, количество единиц заказанных продуктов и дата заказа, для хранения информации о заказах, которые поступают от клиентов.

В таблице «Поставка» содержатся данные о поставке товаров для молокозавода от поставщиков, включая id поставки, id поставщика, id продукта, количество продуктов и дата поставки, для хранения информации о поставках сырья для производства молочной продукции.

Все SQL запросы для создания базы данных приведены в приложении А.

3.3 Тестирование базы данных

Были составлены тестовые данные в соответствии с предметной областью и спроектированной базой данных, тестовые данные представлены в таблицах 3.1 – 3.3.

Таблица 3.1 – Тестовые данные для таблицы «Клиенты»

| ФИО | Адрес | Номер телефона |
|----------------------|-------------------------|------------------|
| Иванов Иван Иванович | Москва, ул. Ленина, д.1 | +7(495)123-45-67 |

| | | |
|---------------------------|--|------------------|
| Петров Петр Петрович | Санкт-Петербург, Невский проспект, д.20 | +7(812)987-65-43 |
| Сидорова Анна Васильевна | Екатеринбург, ул. Гагарина, д.5 | +7(343)535-55-55 |
| Козлова Елена Игоревна | Новосибирск, ул. Кирова, д.10 | +7(383)757-77-77 |
| Николаева Ольга Сергеевна | Красноярск, ул. Мира, д.15 | +7(391)868-88-88 |

Таблица 3.2 – Тестовые данные для таблицы «Продукты»

| Наименование продукта | Тип продукта |
|-----------------------|--------------|
| Молоко 3.2% | Сырьё |
| Творог | Продукт |
| Сливки 20% | Продукт |
| Йогурт стандартный | Продукт |
| Кефир 2% | Продукт |
| Сыр российский | Продукт |
| Масло 82% | Продукт |
| Ряженка 4% | Продукт |
| Простокваша | Продукт |
| Молоко | Сырьё |

Таблица 3.3 – Тестовые данные для таблицы «Поставщики»

| Имя | Адрес | Номер телефона |
|----------------------|--|------------------|
| ООО "Молоко Севера" | Санкт-Петербург, ул. Ломоносова, д.10 | +7(812)111-11-11 |
| ЗАО "Молоко-Продукт" | Москва, ул. Профсо- юзная, д.25 | +7(495)222-22-22 |
| ИП Иванов | Екатеринбург, ул. Пушкина, д.7 | +7(343)333-33-33 |

Было проведено тестирование базы данных посредством SQL запросов. Было создано 15 тестовых запросов, представленных ниже.

Запрос 1: вывод всех столбцов из таблицы «Product». Код запроса представлен в листинге 3.1. Результат выполнения – на рисунке 3.1.

Листинг 3.1 – Текст запроса №1

```
SELECT "ProductID", "ProductName", "Type"
FROM mf."Product";
```

| | ProductID [PK] integer | ProductName character varying (50) | Type character varying (50) |
|---|----------------------------------|--|---------------------------------------|
| 1 | 1 | Простоквашино | Молоко |
| 2 | 2 | Мазекино | Творог |
| 3 | 3 | БЮ Александров | Сырок |
| 4 | 5 | Молния | Айран |

Рисунок 3.1 – Результат выполнения запроса №1

Запрос 2: вывод количества продуктов по заданным параметрам. Код запроса представлен в листинге 3.2. Результат выполнения – на рисунке 3.2.

Листинг 3.2 – Текст запроса №2

```
SELECT COUNT(*)
FROM mf."Product"
WHERE "ProductName" = 'БЮ Александров' AND "Type" = 'Сырок'
```

| | count bigint |
|---|------------------------|
| 1 | 1 |

Рисунок 3.2 – Результат выполнения запроса №2

Запрос 3: вставка нового продукта в таблицу «Product». Код запроса представлен в листинге 3.3. Результат выполнения – на рисунке 3.3.

Листинг 3.3 – Текст запроса №3

```
INSERT INTO mf."Product" ("ProductName", "Type")
VALUES ('Домашкино', 'Молоко');
```

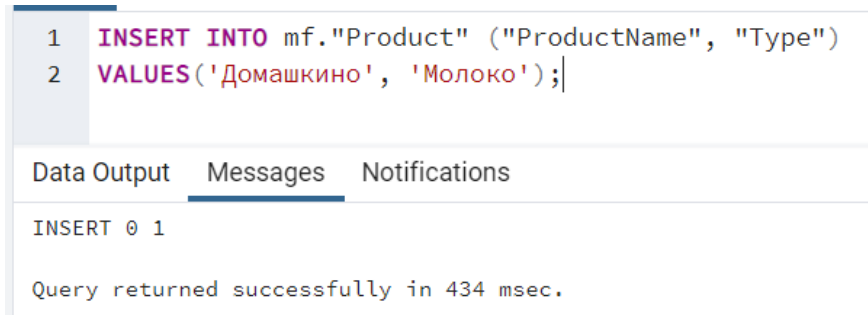


Рисунок 3.3 – Результат выполнения запроса №3

Запрос 4: удаление продукта из таблицы «Product». Код запроса представлен в листинге 3.4. Результат выполнения – на рисунке 3.4.

Листинг 3.4 – Текст запроса №4

```
DELETE FROM mf."Product"
WHERE "ProductName" = 'Домашкино' AND "Type" = 'Молоко';
```

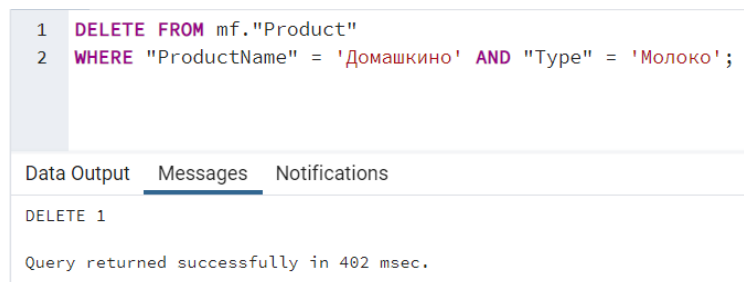


Рисунок 3.4 – Результат выполнения запроса №4

Запрос 5: обновление продукта в таблице «Product». Код запроса представлен в листинге 3.5. Результат выполнения – на рисунке 3.5.

Листинг 3.5 – Текст запроса №5

```
UPDATE mf."Product"
SET "ProductName" = 'Вкусново', "Type" = 'Творог'
WHERE "ProductName" = 'Мазекино' AND "Type" = 'Творог';
```

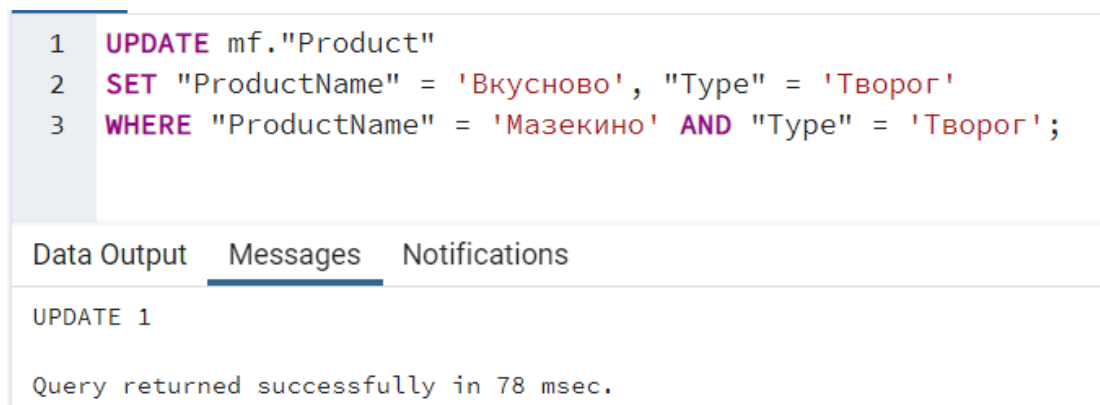


Рисунок 3.5 – Результат выполнения запроса №5

Запрос 6: вывод id клиента из таблицы «Клиент» по заданным параметрам. Код запроса представлен в листинге 3.6. Результат выполнения – на рисунке 3.6.

Листинг 3.6 – Текст запроса №6

```

SELECT "ClientID"
FROM mf."Client"
WHERE "FIO" = 'Фомин Александр Вадимович' AND "Address" = 'Тула' AND "PhoneNumber"
= '88881234567';

```

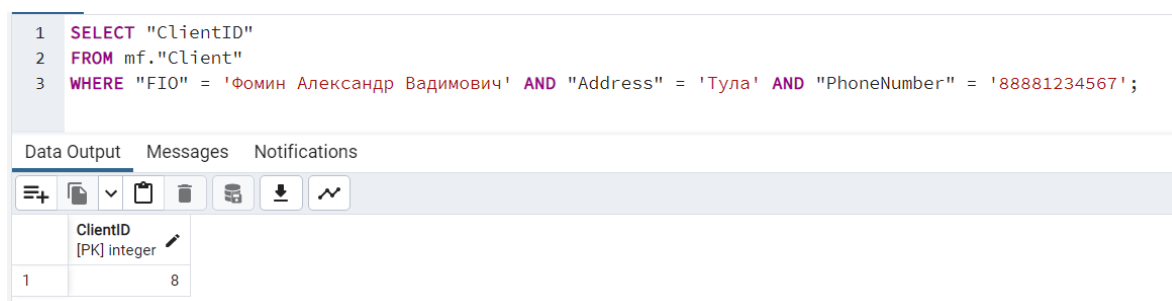


Рисунок 3.6 – Результат выполнения запроса №6

Запрос 7: вывод количества клиентов по заданным параметрам. Код запроса представлен в листинге 3.7. Результат выполнения – на рисунке 3.7.

Листинг 3.7 – Текст запроса №7

```

SELECT COUNT("FIO") AS count
FROM mf."Client"
WHERE "FIO" = 'Фомин Александр Вадимович' AND "Address" = 'Тула' AND "PhoneNumber"
= '88881234567'

```

```

1 SELECT COUNT("FIO") AS count
2 FROM mf."Client"
3 WHERE "FIO" = 'Фомин Александр Вадимович' AND "Address" = 'Тула' AND "PhoneNumber" = '88881234567'

```

| Data Output | | Messages | Notifications |
|-------------|-----------------|----------|---------------|
| | count bigint | | |
| 1 | 1 | | |

Рисунок 3.7 – Результат выполнения запроса №7

Запрос 8: вставка новой записи в таблицу «Заказ», где при этом в поле DateTime вставляется текущее время. Код запроса представлен в листинге 3.8. Результат выполнения – на рисунке 3.8.

Листинг 3.8 – Текст запроса №8

```

INSERT INTO mf."Order" ("ClientID", "ProductID", "ProductCount", "DateTime")
VALUES (1, 3, 10, date_trunc('second', now()::timestamp));

```

```

1 INSERT INTO mf."Order" ("ClientID", "ProductID", "ProductCount", "DateTime")
2 VALUES (1, 3, 10, date_trunc('second', now()::timestamp));

```

Data Output Messages Notifications

INSERT 0 1

Query returned successfully in 382 msec.

Рисунок 3.8 – Результат выполнения запроса №8

Запрос 9: удаление записи из таблицы Order по заданным параметрам, при этом записи сортируются по дате создания по возрастанию и удаляется самая новая из найденных записей. Код запроса представлен в листинге 3.9. Результат выполнения – на рисунке 3.9.

Листинг 3.9 – Текст запроса №9

```

DELETE FROM mf."Order"
WHERE "OrderID" IN (
SELECT "OrderID"

```



```

FROM mf."Order"
WHERE "ClientID" = 1 AND "ProductID" = 3 AND "ProductCount" = 10
ORDER BY "DateTime" DESC
LIMIT 1
)

```



Рисунок 3.9 – Результат выполнения запроса №9

Запрос 10: вывод всех записей из таблицы «Заказы», которые были сделаны за последние сутки. Код запроса представлен в листинге 3.10. Результат выполнения – на рисунке 3.10.

Листинг 3.10 – Текст запроса №10

```

SELECT "OrderID", "ClientID", "ProductID", "ProductCount", "DateTime"
FROM mf."Order"
WHERE "DateTime" > (date_trunc('second', now()::timestamp) - interval '24 hour')

```

The screenshot shows a SQL query editor with the following code:

```

1 SELECT "OrderID", "ClientID", "ProductID", "ProductCount", "DateTime"
2 FROM mf."Order"
3 WHERE "DateTime" > (date_trunc('second', now()::timestamp) - interval '24 hour')

```

Below the editor, the 'Data Output' tab is selected, showing a table with the following data:

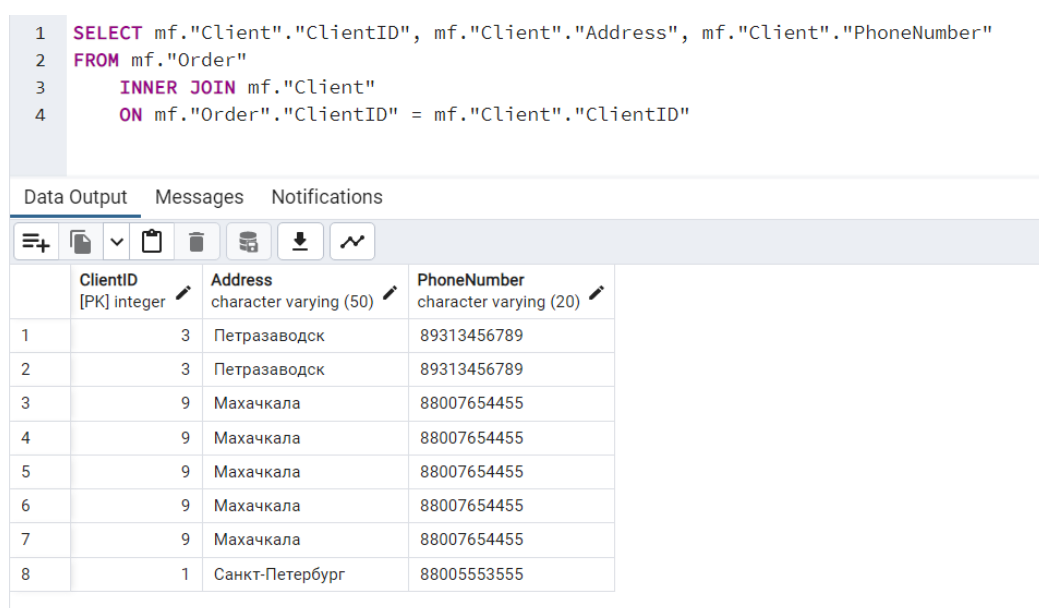
| | OrderID [PK] integer | ClientID integer | ProductID integer | ProductCount integer | DateTime timestamp without time zone |
|---|-------------------------|---------------------|----------------------|-------------------------|---|
| 1 | 15 | 1 | 3 | 10 | 2023-04-03 16:06:41 |

Рисунок 3.10 – Результат выполнения запроса №10

Запрос 11: вывод всех записей из таблицы «Клиент», для которых был создан заказ (запись в таблице «Запись»). Код запроса представлен в листинге 3.11. Результат выполнения – на рисунке 3.11.

Листинг 3.11 – Текст запроса №11

```
SELECT mf."Client"."ClientID", mf."Client"."Address", mf."Client"."PhoneNumber"
FROM mf."Order"
INNER JOIN mf."Client"
ON mf."Order"."ClientID" = mf."Client"."ClientID"
```



| | ClientID [PK] integer | Address character varying (50) | PhoneNumber character varying (20) |
|---|--------------------------|-----------------------------------|---------------------------------------|
| 1 | 3 | Петразаводск | 89313456789 |
| 2 | 3 | Петразаводск | 89313456789 |
| 3 | 9 | Махачкала | 88007654455 |
| 4 | 9 | Махачкала | 88007654455 |
| 5 | 9 | Махачкала | 88007654455 |
| 6 | 9 | Махачкала | 88007654455 |
| 7 | 9 | Махачкала | 88007654455 |
| 8 | 1 | Санкт-Петербург | 88005553555 |

Рисунок 3.11 – Результат выполнения запроса №11

Запрос 12: удаление записи из таблицы «Поставка» по заданным параметрам, при этом записи сортируются по дате создания по возрастанию и удаляется самая новая из найденных записей. Код запроса представлен в листинге 3.12. Результат выполнения – на рисунке 3.12.

Листинг 3.12 – Текст запроса №12

```
DELETE FROM mf."Delivery"
WHERE "DeliveryID" IN (
SELECT "DeliveryID"
FROM mf."Delivery"
WHERE "VendorID" = 6 AND "ProductID" = 3 AND "ProductCount" = 7)
```

```
ORDER BY "DateTime" DESC
LIMIT 1
)
```

```

1 DELETE FROM mf."Delivery"
2 WHERE "DeliveryID" IN (
3     SELECT "DeliveryID"
4     FROM mf."Delivery"
5     WHERE "VendorID" = 6 AND "ProductID" = 3 AND "ProductCount" = 7
6     ORDER BY "DateTime" DESC
7     LIMIT 1
8 )

```

Data Output Messages Notifications

DELETE 1

Query returned successfully in 84 msec.

Рисунок 3.12 – Результат выполнения запроса №12

Запрос 13: вывод записей из таблицы «Заказ», где в заказе больше 5 продуктов. Код запроса представлен в листинге 3.13. Результат выполнения – на рисунке 3.13.

Листинг 3.13 – Текст запроса №13

```
SELECT "OrderID", "ClientID", "ProductID", "ProductCount", "DateTime"
FROM mf."Order"
WHERE "ProductCount" > 5
```

```

1 SELECT "OrderID", "ClientID", "ProductID", "ProductCount", "DateTime"
2 FROM mf."Order"
3 WHERE "ProductCount" > 5

```

Data Output Messages Notifications

| | OrderID [PK] integer | ClientID integer | ProductID integer | ProductCount integer | DateTime timestamp without time zone |
|---|-------------------------|---------------------|----------------------|-------------------------|---|
| 1 | 2 | 3 | 3 | 8 | 2023-03-29 12:33:16 |
| 2 | 3 | 9 | 3 | 9 | 2023-03-29 12:34:38 |
| 3 | 4 | 9 | 3 | 12 | 2023-03-29 12:58:49 |
| 4 | 11 | 9 | 3 | 9 | 2023-03-29 17:42:50 |
| 5 | 12 | 9 | 3 | 9 | 2023-03-29 17:42:51 |
| 6 | 13 | 9 | 3 | 9 | 2023-03-29 17:42:58 |
| 7 | 15 | 1 | 3 | 10 | 2023-04-03 16:06:41 |

Рисунок 3.13 – Результат выполнения запроса №13

Запрос 14: вывод всех записей из таблицы «Поставщик», для которых созданы поставки (записи в таблице «Поставки»). Код запроса представлен в листинге 3.14. Результат выполнения – на рисунке 3.14.

Листинг 3.14 – Текст запроса №14

```
SELECT v."VendorID", v."Name", v."Address", v."PhoneNumber"
FROM mf."Delivery" AS d
INNER JOIN mf."Vendor" AS v
ON d."VendorID" = v."VendorID"
```

| | | |
|---|------------|--|
| 1 | SELECT | v."VendorID", v."Name", v."Address", v."PhoneNumber" |
| 2 | FROM | mf."Delivery" AS d |
| 3 | INNER JOIN | mf."Vendor" AS v |
| 4 | ON | d."VendorID" = v."VendorID" |

| Data Output Messages Notifications | | | | |
|------------------------------------|--------------------------|--------------------------------|-----------------------------------|---------------------------------------|
| | VendorID [PK] integer | Name character varying (50) | Address character varying (50) | PhoneNumber character varying (50) |
| 1 | 1 | Молоково | Москва | 88005553555 |
| 2 | 5 | Кэфтеме | Анапа | 77777777777 |

Рисунок 3.14 – Результат выполнения запроса №14

Запрос 15: Вывод всех записей из таблицы «Клиент», где адрес клиента – Санкт-Петербург. Код запроса представлен в листинге 3.15. Результат выполнения – на рисунке 3.15.

Листинг 3.15 – Текст запроса №15

```
SELECT "ClientID", "FIO", "Address", "PhoneNumber"
FROM mf."Client"
WHERE "Address" = 'Санкт-Петербург'
```

```

1 SELECT "ClientID", "FIO", "Address", "PhoneNumber"
2 FROM mf."Client"
3 WHERE "Address" = 'Санкт-Петербург'|

```

Data Output

Messages

Notifications

≡+

📄

▼

📋

🗑️

🗄️

⬇️

📈





| | ClientID [PK] integer  | FIO character varying (50)  | Address character varying (50)  | PhoneNumber character varying (20)  |
|---|--|---|---|---|
| 1 | 1 | Конопалов Владимир Федорович | Санкт-Петербург | 88005553555 |

Рисунок 3.15 – Результат выполнения запроса №15

По результатам проведённых тестов можно сделать вывод, что как SQL-запросы, так и таблицы в базе данных, созданы правильно и работают корректно.

4 РАЗРАБОТКА КЛИЕНТСКОГО ПРИЛОЖЕНИЯ

4.1 Обоснование выбора языков программирования

При разработке клиентского приложения для базы данных предприятия по производству молочной продукции был выбран язык программирования C#. Этот язык отличается от аналогов, таких как Java тем, что он более прост в изучении и удобен в использовании при создании десктопных приложений. Кроме того, C# поддерживает платформу .NET, что позволяет легко использовать стандартные библиотеки для работы с базами данных.

Для разработки десктопного приложения была использована среда разработки Visual Studio. Эта среда предоставляет удобный интерфейс для создания десктопных приложений и позволяет быстро и легко создавать пользовательские интерфейсы. По сравнению с аналогами, такими как Eclipse или IntelliJ IDEA, Visual Studio обладает более широкими возможностями по созданию и отладке приложений, а также удобной интеграцией с платформой .NET.

Таким образом, выбор языка программирования C# и среды разработки Visual Studio был обоснован удобством использования, широкими возможностями и удобной интеграцией с платформой .NET.

4.2 Разработка интерфейса пользователя

Был разработан интерфейс пользователя на языке «C#» в IDE «Visual Studio». Для удобства работы было использовано интерактивное построение интерфейса пользователя. Код программы был представлен в листингах в приложении Б. Разработанный интерфейс пользователя включает следующие элементы:

- главное меню приложения с кнопками для доступа к различным разделам приложения;

- окно просмотра клиентов, содержащее таблицу со списком клиентов и кнопками для добавления, редактирования и удаления записей;
- окно просмотра продуктов, содержащее таблицу со списком продуктов и кнопками для добавления, редактирования и удаления записей;
- окно просмотра заказов, содержащее таблицу со списком заказов и кнопками для добавления, редактирования и удаления записей;
- окно просмотра поставщиков, содержащее таблицу со списком поставщиков и кнопками для добавления, редактирования и удаления записей;
- осуществлена автоматическая загрузка и сохранение базы данных.

Разработанный интерфейс пользователя был организован и реализован с учетом требований предметной области и удобства использования.

4.3 Алгоритм работы каждого из модулей

Алгоритм работы программы построен на основе архитектуры MVP. При получении текстовых данных объект класса передаёт полученные данные в базу данных, предварительно обработав их. Классы уровня модели обеспечивают взаимодействие программы с базой данных. Файл `Form1.cs` содержит в себе файл `from1.designer.cs`, посредством иерархии контейнеров и компонентов была разработана программа, представляющая собой систему управления базой данных.

4.4 Тестирование работы приложения

Для тестирования работы приложения были заранее созданы тестовые данные (таблицы 3.1 – 3.3 из раздела 3). На рисунке 4.1 можно наблюдать стартовое окно при запуске программы управления базой данных. На рисунке 4.2 изображен интерфейс для менеджера по купле/продаже продукции для предприятия, который может управлять таблицами «Заказы» и «Поставки». На рисунке 4.3

представлен интерфейс для администратора, у которого есть доступ ко всем таблицам в базе данных.

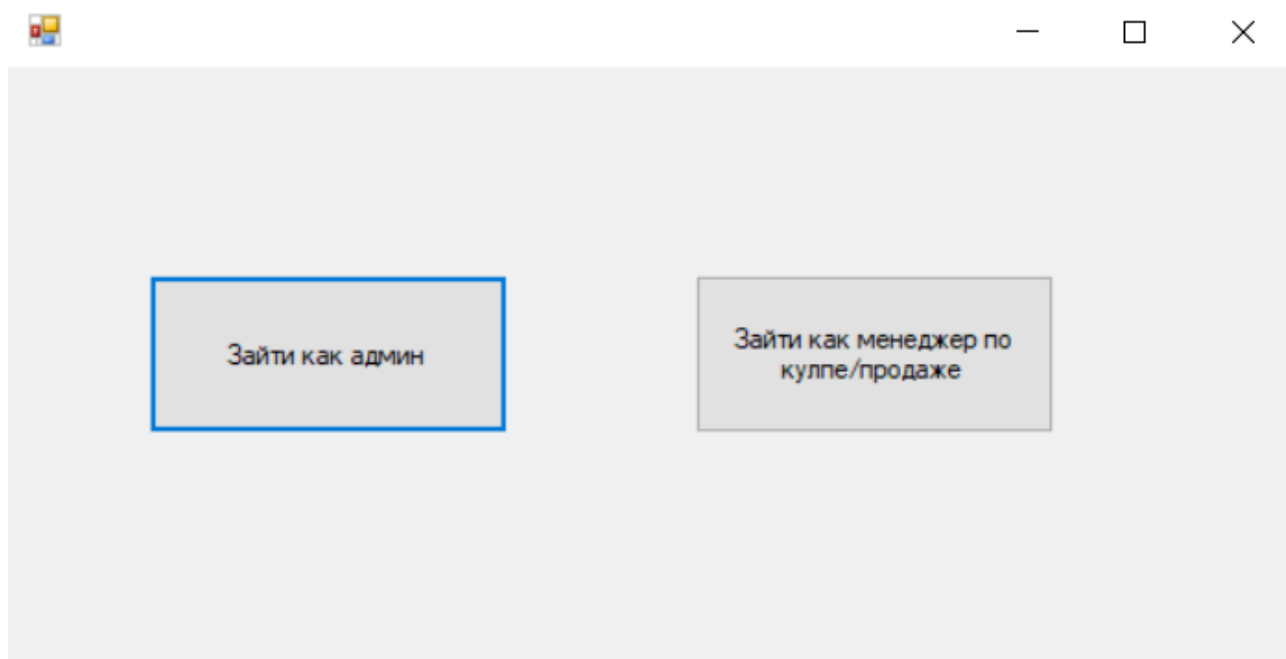


Рисунок 4.1 – Окно разграничения доступа к базе данных

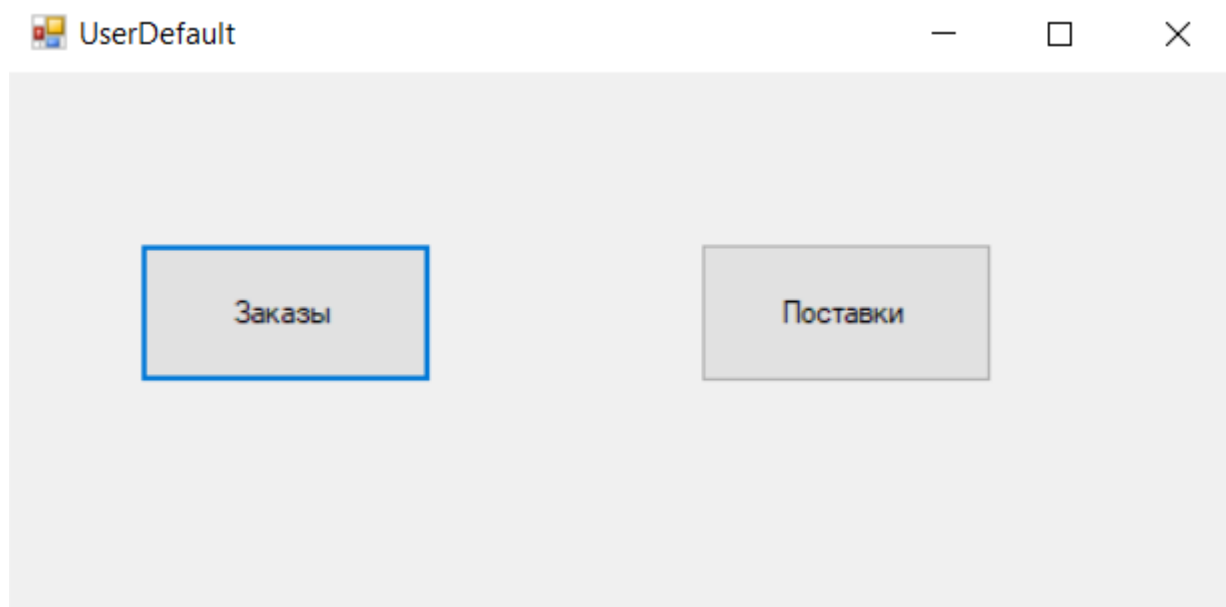


Рисунок 4.2 – Таблицы доступные для редактирования менеджером по закупкам

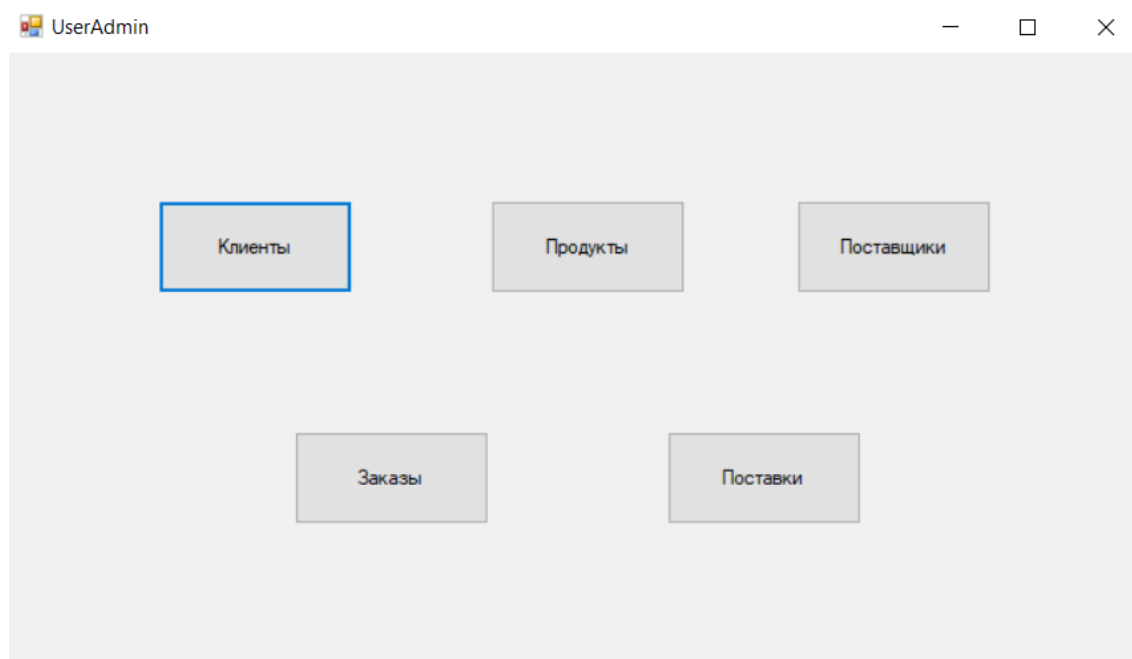


Рисунок 4.3 – Таблицы доступные для редактирования администратором БД

На рисунках 4.4 и 4.5 изображены примеры окна взаимодействия с определённой таблицей, выбранной пользователем, окно позволяет осуществить создание, редактирование, удаление и демонстрацию записей.

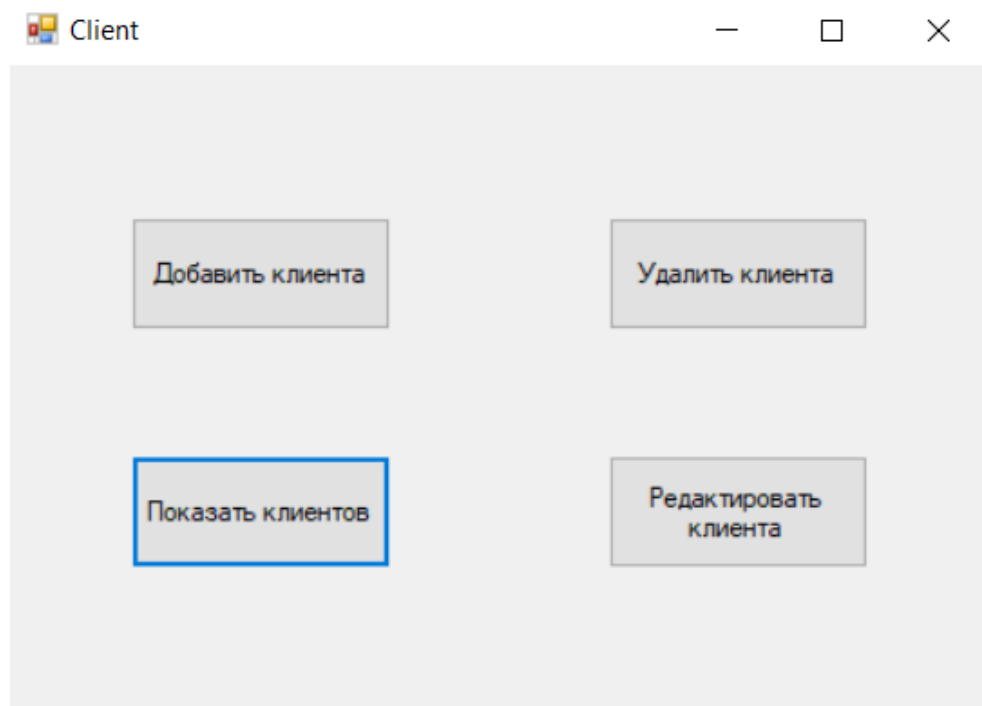


Рисунок 4.4 – Окно управления таблицей клиентов

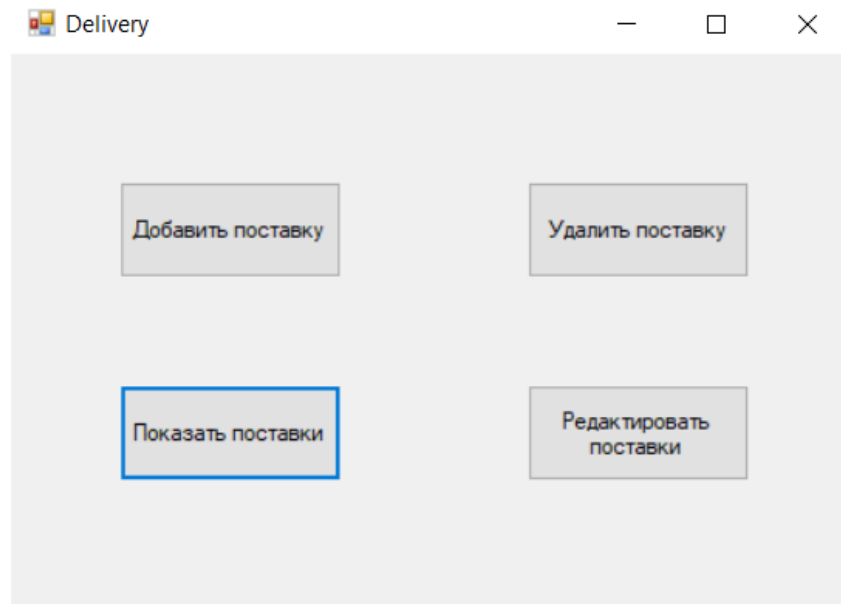


Рисунок 4.5 – Окно управления таблицей поставок

На рисунке 4.6 представлен пример добавления записи в таблицу «Клиент». На рисунке 4.7 пример удаления записи из таблицы «Клиент». На рисунке 4.8 происходит добавление «ошибочного» клиента. на рисунке 4.9 отображены записи с введённым «ошибочным» клиентом. На рисунке 4.10 происходит редактирование записи в таблице «Клиент» для исправления данных о клиенте. Результат редактирования таблицы изображен на рисунке 4.11, посредством нажатия кнопки «Показать клиентов».

 A screenshot of a software window titled "ClientAdd". It contains a form with three input fields. The first field is labeled "Фамилия Имя Отчество" and contains the text "Иванов Иван Иванович". The second field is labeled "Адрес" and contains the text "Москва, ул. Ленина, д.1". The third field is labeled "Номер телефона" and contains the text "+7 (945) 123-45-67". Below the fields is a button labeled "Добавить", which is highlighted with a blue rectangular border. At the bottom of the window, there is a message box that says "New entry successfully added".

Рисунок 4.6 – Добавление записи в таблицу «Клиент»

The screenshot shows a window titled "ClientDelete" with three input fields at the top: "Фамилия Имя Отчество" (Last Name, First Name, Patronymic), "Адрес" (Address), and "Номер телефона" (Phone Number). The first field contains "Хроменко Данил Алексеевич", the second contains "Анапа, ул. Владимирская, д. 114", and the third contains "+7 (988) 366-54-69". Below these fields is a button labeled "Удалить" (Delete). At the bottom, a message box displays "Entry successfully deleted".

Рисунок 4.7 – Удаление записи из таблицы

The screenshot shows a window titled "ClientAdd" with three input fields at the top: "Фамилия Имя Отчество" (Last Name, First Name, Patronymic), "Адрес" (Address), and "Номер телефона" (Phone Number). The first field contains "Ошибочный клиент" (Incorrect client), the second contains "Ошибочный адрес" (Incorrect address), and the third contains "+7(777)777-77-77". Below these fields is a button labeled "Добавить" (Add). At the bottom, a message box displays "New entry successfully added".

Рисунок 4.8 – Добавление ошибочного клиента

ClientShow

| | | | |
|----|--------------------------|---|------------------|
| 9 | Иванов Иван Иванович | Москва, ул. Ленина, д.1 | +7(495)123-45-67 |
| 10 | Петров Петр Петрович | Санкт-Петербург, Невский проспект, д.20 | +7(812)987-65-43 |
| 11 | Сидорова Анна Васильевна | Екатеринбург, ул. Гагарина, д.5 | +7(343)535-55-55 |
| 12 | Козлова Елена Игоревна | Новосибирск, ул. Кирова, д.10 | +7(383)757-77-77 |
| 13 | Ошибочный клиент | Ошибочный адрес | +7(777)777-77-77 |

Рисунок 4.9 – Отображение записей таблицы «Клиент» с ошибочным клиентом

ClientUpdate

| | Фамилия Имя Отчество | Адрес | Номер телефона |
|--------|---------------------------|----------------------------|------------------|
| Старые | Ошибочный клиент | Ошибочный адрес | +7(777)777-77-77 |
| Новые | Николаева Ольга Сергеевна | Красноярск, ул. Мира, д.15 | +7(391)868-88-88 |

Редактировать

Entry successfully updated

Рисунок 4.10 – Редактирование записи в таблице «Клиент»



Рисунок 4.11 – Отображение записей «Клиент» с исправленным клиентом

Рисунок 4.12 отображает панель добавления продуктов в таблицу «Продукты». На рисунке 4.13 представлен результат многократного добавления записей в таблицу продуктов.

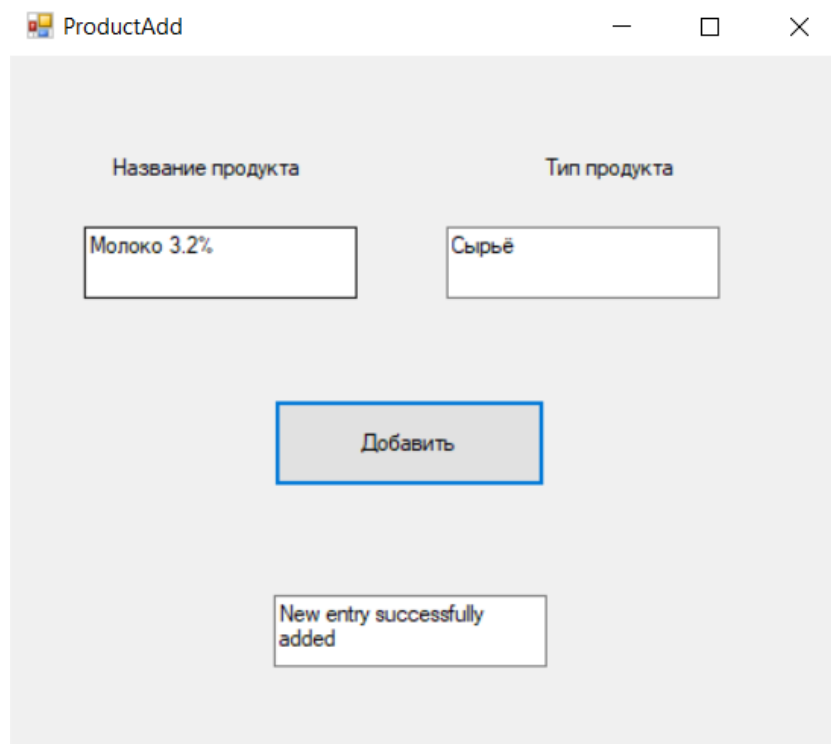


Рисунок 4.12 – Отображение панели с добавлением записи в «Продукт»

ProductShow

| | | |
|---|--------------------|---------|
| 1 | Молоко 3.2% | Сырьё |
| 2 | Творог | Продукт |
| 3 | Сливки 20% | Продукт |
| 4 | Йогурт стандартный | Продукт |
| 5 | Кефир 2% | Продукт |
| 6 | Сыр российский | Продукт |
| 7 | Масло 82% | Продукт |
| 8 | Ряженка 4% | Продукт |
| 9 | Простокваша | Продукт |

Рисунок 4.13 – Отображение записей в таблице «Продукт»

На рисунке 4.14 можно наблюдать интерфейс добавления поставщиков. На рисунке 4.15 отображён процесс отображения поставщиков, успешно добавленных в таблицу.

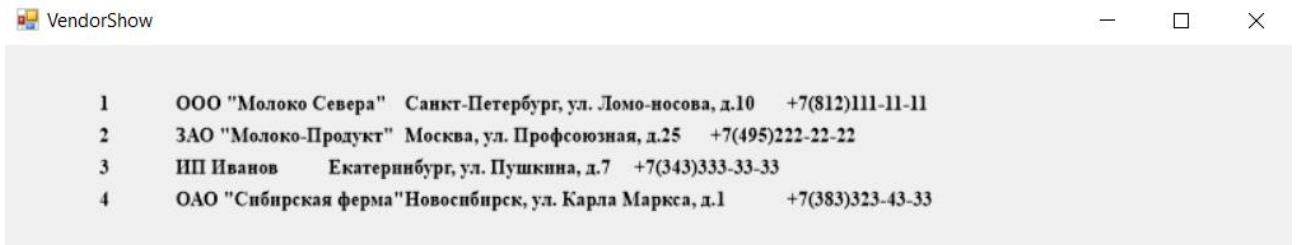
VendorAdd

| Название поставщика | Адрес | Номер телефона |
|---------------------|--------------------------------|------------------|
| ИП Иванов | Екатеринбург, ул. Пушкина, д.7 | +7(343)333-33-33 |

Добавить

New entry successfully added

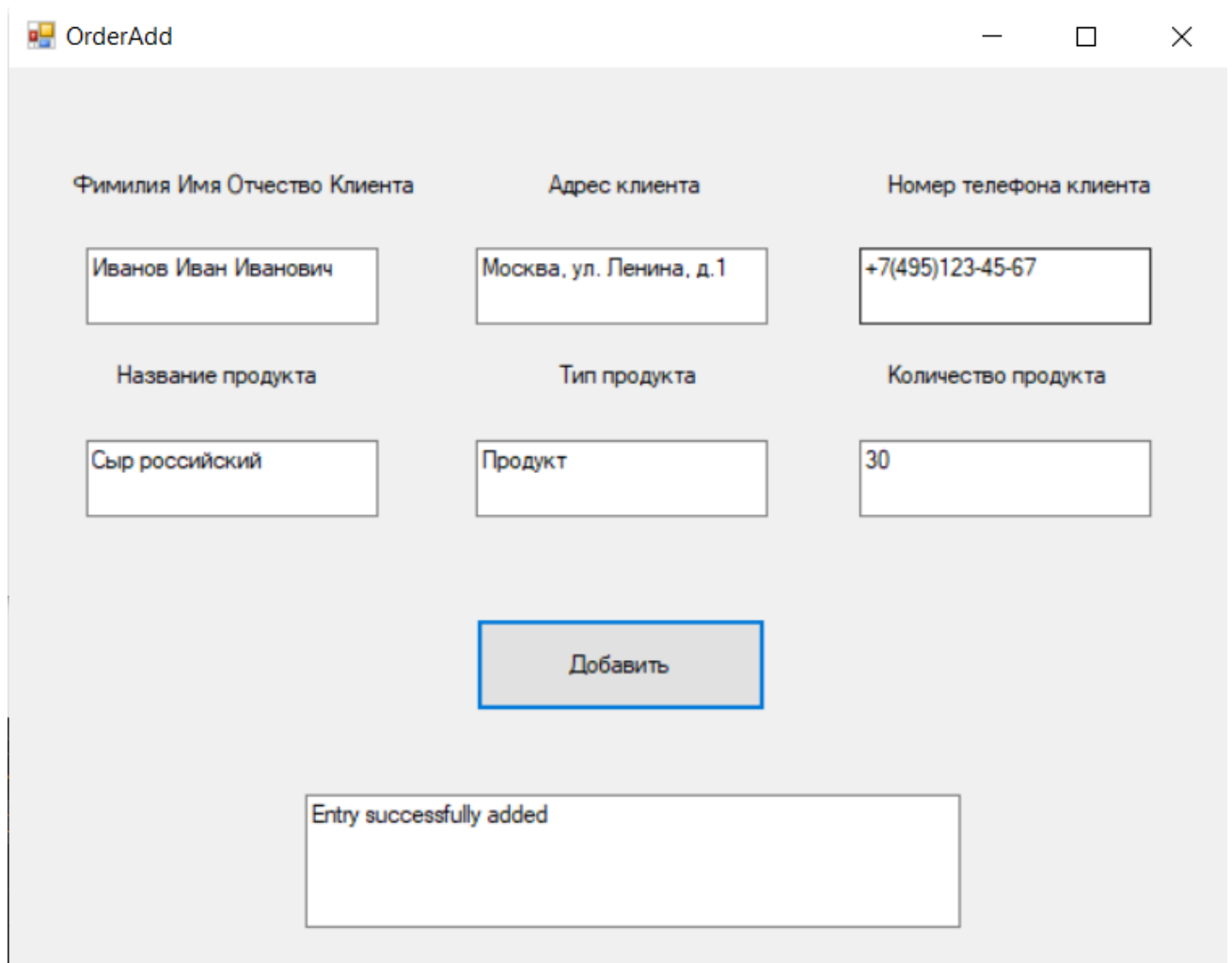
Рисунок 4.14 – Добавление записи в таблицу «Поставщик»



| | | | |
|---|-----------------------|---------------------------------------|------------------|
| 1 | ООО "Молоко Севера" | Санкт-Петербург, ул. Ломоносова, д.10 | +7(812)111-11-11 |
| 2 | ЗАО "Молоко-Продукт" | Москва, ул. Профсоюзная, д.25 | +7(495)222-22-22 |
| 3 | ИП Иванов | Екатеринбург, ул. Пушкина, д.7 | +7(343)333-33-33 |
| 4 | ОАО "Сибирская ферма" | Новосибирск, ул. Карла Маркса, д.1 | +7(383)323-43-33 |

Рисунок 4.15 – Отображение тестовых записей таблицы «Поставщик»

На рисунке 4.16 осуществляется пример добавления заказа в таблицу «Заказы». На рисунке 4.17 продемонстрирован результат добавления записей в таблицу.



Фамилия Имя Отчество Клиента: Иванов Иван Иванович

Адрес клиента: Москва, ул. Ленина, д.1

Номер телефона клиента: +7(495)123-45-67

Название продукта: Сыр российский

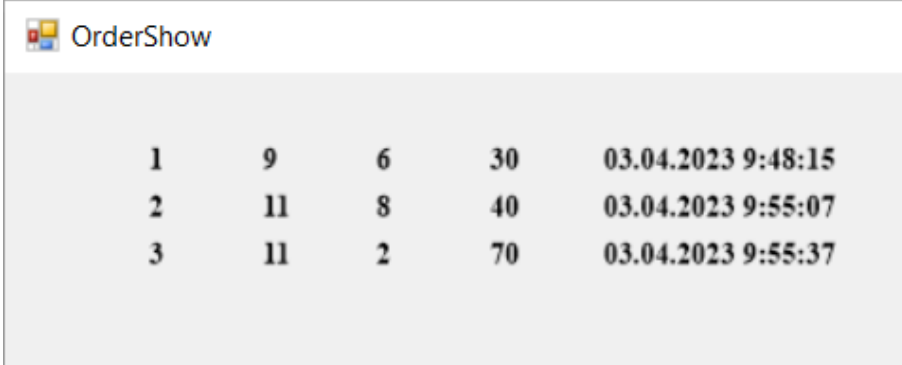
Тип продукта: Продукт

Количество продукта: 30

Добавить

Entry successfully added

Рисунок 4.16 – Добавление заказа в таблицу «Заказы»

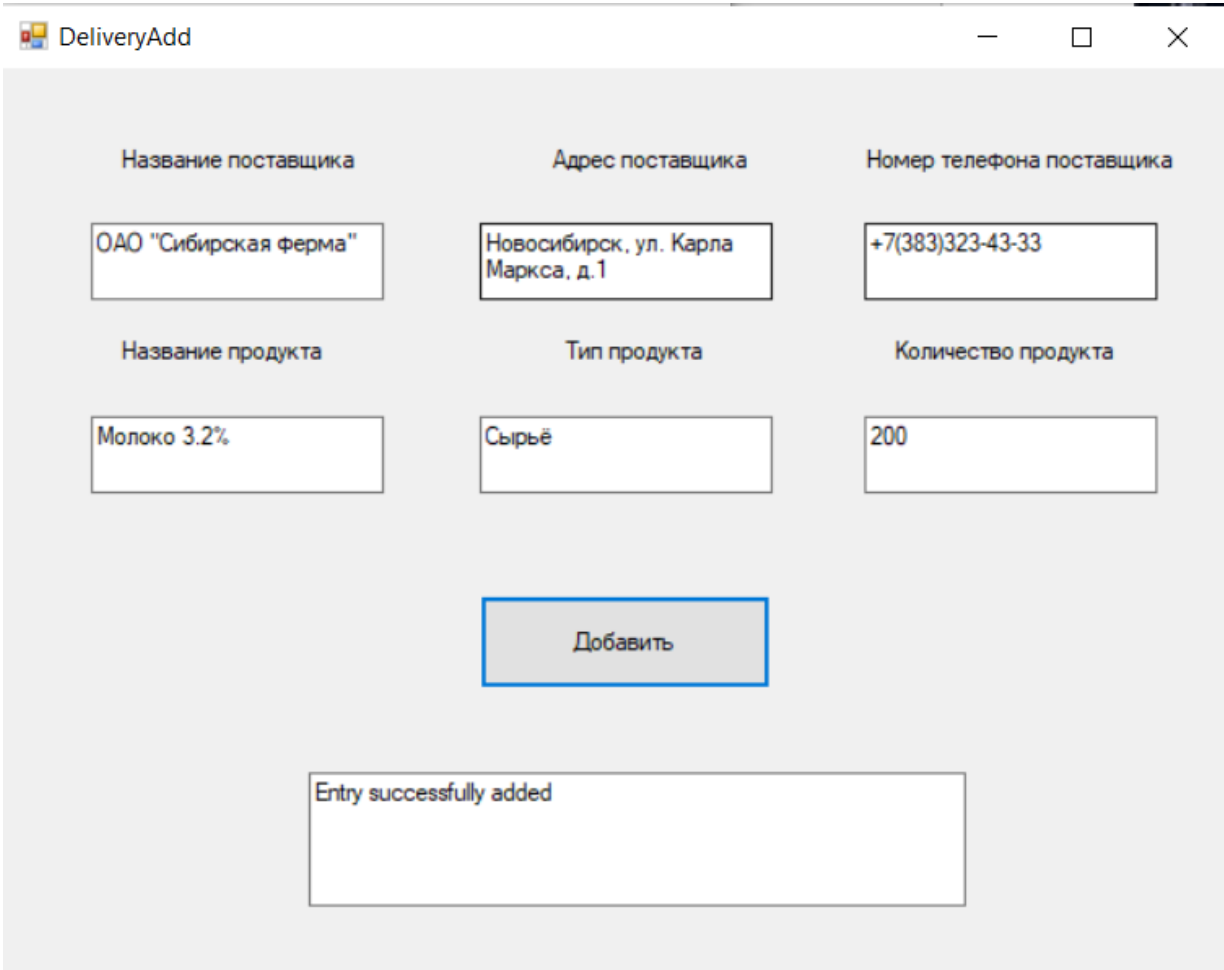


OrderShow

| | | | | |
|---|----|---|----|--------------------|
| 1 | 9 | 6 | 30 | 03.04.2023 9:48:15 |
| 2 | 11 | 8 | 40 | 03.04.2023 9:55:07 |
| 3 | 11 | 2 | 70 | 03.04.2023 9:55:37 |

Рисунок 4.17 – Отображение заказов в таблице «Заказы»

На рисунке 4.18 осуществляется пример добавления заказа в таблицу «Поставщики». На рисунке 4.19 продемонстрирован результат добавления записей в таблицу.



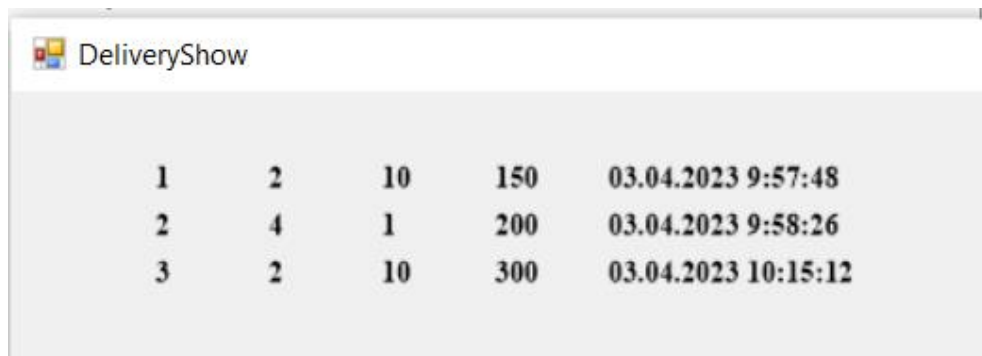
DeliveryAdd

| | | |
|-----------------------|------------------------------------|---------------------------|
| Название поставщика | Адрес поставщика | Номер телефона поставщика |
| ОАО "Сибирская ферма" | Новосибирск, ул. Карла Маркса, д.1 | +7(383)323-43-33 |
| Название продукта | Тип продукта | Количество продукта |
| Молоко 3.2% | Сырьё | 200 |

Добавить

Entry successfully added

Рисунок 4.18 – Добавление поставки в таблицу «Поставки»



The screenshot shows a window titled "DeliveryShow" with a table containing three rows of data. Each row consists of five columns: an ID, two numerical values, a third numerical value, and a timestamp.

| | | | | |
|---|---|----|-----|---------------------|
| 1 | 2 | 10 | 150 | 03.04.2023 9:57:48 |
| 2 | 4 | 1 | 200 | 03.04.2023 9:58:26 |
| 3 | 2 | 10 | 300 | 03.04.2023 10:15:12 |

Рисунок 4.19 – Демонстрация поставок из таблицы «Поставки»

По результатам тестирования можно сделать вывод, что программа работает так, как задумывалось на этапе изучения предметной области.

ЗАКЛЮЧЕНИЕ

В ходе выполнения работы была разработана база данных для предприятия по производству молочной продукции. Была проведена аналитическая часть работы, включающая анализ предметной области и постановку задачи. Затем была выполнена разработка логической и физической моделей базы данных, включая построение диаграммы «сущность-связь» в нотации П.Чена, модели основанной на ключах и полной атрибутивной модели в нотации IDEF1X. Была выбрана аппаратная и программная платформа для реализации базы данных, выполнена ее реализация и тестирование, а также было проведено разграничение прав доступа.

Кроме того, было выполнено дополнительное задание - разработка десктопного приложения на языке C# в VisualStudio. Были обоснованы выбор языка программирования, разработан интерфейс пользователя, определен алгоритм работы каждого из модулей и проведено тестирование работы приложения.

В результате проделанной работы была создана функциональная база данных и соответствующее приложение для управления ею. Таким образом, была достигнута цель расчетно-графической работы – были получены и закреплены практические и теоретические навыки концептуального, логического и физического проектирования баз данных, а также использования внутри разрабатываемых приложений СУБД, использующих принципы SQL.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Коннолли Т., Бегг К. Системы управления базами данных: проектирование, реализация и управление. – М.: Издательство «Вильямс», 2011. – 1104 с. (ISBN 978-5-8459-1740-2)
2. Эльмасри Р., Наватхе Ш. Фундаментальные базы данных: Пер. с англ. – М.: Издательство «Вильямс», 2015. – 1296 с. (ISBN 978-5-8459-1994-9)
3. Дейт К. Дж. Введение в системы баз данных. – М.: Вильямс, 2012. – 1360 с. (ISBN 978-5-8459-1845-4)
4. C. J. Date. Database Design and Relational Theory: Normal Forms and All That Jazz. – O'Reilly Media, 2012. – 278 p. (ISBN 978-1449328019)
5. Официальный сайт PostgreSQL. <https://www.postgresql.org/>
6. Официальный сайт Microsoft Visual Studio. <https://visualstudio.microsoft.com/>
7. Oracle. ER-моделирование. <https://www.oracle.com/ru/database/what-is-data-modeling/>
8. Redgate. IDEF1X. <https://www.red-gate.com/simple-talk/sql/database-administration/data-modelling-idef1x-notation-basics/>
9. Хабр. Разработка приложений на C# с использованием Visual Studio. <https://habr.com/ru/post/268821/>
10. Хабр. Нормализация баз данных: описание методов и примеры на PostgreSQL. <https://habr.com/ru/company/postgrespro/blog/348986>
11. Codeacademy. Нормализация баз данных. <https://www.codecademy.com/articles/what-is-normalization>

ПРИЛОЖЕНИЕ А.**Создание таблиц для базы данных****Листинг А.1 – PostgreSQL запрос для создания таблицы «Клиент»**

```
CREATE TABLE mf."Client"
(
    "ClientID" integer NOT NULL GENERATED ALWAYS AS IDENTITY ( INCREMENT 1 START
1 ),
    "FIO" character varying NOT NULL,
    "Address" character varying NOT NULL,
    "PhoneNumber" character varying NOT NULL,
    PRIMARY KEY ("ClientID")
);

ALTER TABLE IF EXISTS mf."Client"
    OWNER to postgres;
```

Листинг А.2 – PostgreSQL запрос для создания таблицы «Продукт»

```
CREATE TABLE mf."Product"
(
    "ProductID" integer NOT NULL GENERATED ALWAYS AS IDENTITY ( INCREMENT 1 START
1 ),
    "ProductName" character varying NOT NULL,
    "Type" character varying NOT NULL,
    PRIMARY KEY ("ProductID")
);

ALTER TABLE IF EXISTS mf."Product"
    OWNER to postgres;
```

Листинг А.3 – PostgreSQL запрос для создания таблицы «Поставщик»

```
CREATE TABLE mf."Vendor"
(
    "VendorID" integer NOT NULL GENERATED ALWAYS AS IDENTITY ( INCREMENT 1 START
1 ),
    "Name" character varying NOT NULL,
    "Adress" character varying NOT NULL,
    "PhoneNumber" character varying NOT NULL,
    PRIMARY KEY ("VendorID")
);

ALTER TABLE IF EXISTS mf."Vendor"
    OWNER to postgres;
```

Листинг А.4 – PostgreSQL запрос для создания таблицы «Поставка»

```
CREATE TABLE mf."Delivery"
(
    "DeliveryID" integer NOT NULL GENERATED ALWAYS AS IDENTITY ( INCREMENT 1 START
1 ),
    "VendorID" integer NOT NULL,
    "ProductID" integer NOT NULL,
    "ProductCount" integer NOT NULL,
```

```

    "Date" timestamp without time zone NOT NULL,
    PRIMARY KEY ("DeliveryID"),
    CONSTRAINT "Vendor" FOREIGN KEY ("VendorID")
        REFERENCES mf."Vendor" ("VendorID") MATCH SIMPLE
        ON UPDATE NO ACTION
        ON DELETE NO ACTION
        NOT VALID,
    CONSTRAINT "Product" FOREIGN KEY ("ProductID")
        REFERENCES mf."Product" ("ProductID") MATCH SIMPLE
        ON UPDATE NO ACTION
        ON DELETE NO ACTION
        NOT VALID
);

```

Листинг A.5 – PostgreSQL запрос для создания таблицы «Заказ»

```

CREATE TABLE mf."Order"
(
    "OrderID" integer NOT NULL GENERATED ALWAYS AS IDENTITY ( INCREMENT 1 START 1
),
    "ClientID" integer NOT NULL,
    "ProductID" integer NOT NULL,
    "ProductCount" integer NOT NULL,
    "Date" timestamp without time zone NOT NULL,
    PRIMARY KEY ("OrderID"),
    CONSTRAINT "Client" FOREIGN KEY ("ClientID")
        REFERENCES mf."Client" ("ClientID") MATCH SIMPLE
        ON UPDATE NO ACTION
        ON DELETE NO ACTION
        NOT VALID,
    CONSTRAINT "Product" FOREIGN KEY ("ProductID")
        REFERENCES mf."Product" ("ProductID") MATCH SIMPLE
        ON UPDATE NO ACTION
        ON DELETE NO ACTION
        NOT VALID
);

ALTER TABLE IF EXISTS mf."Order"
    OWNER to postgres;

```

ПРИЛОЖЕНИЕ Б.**Основные классы разработанные на C#****Листинг Б.1 – Код класса Client (форма)**

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

using Npgsql;

namespace MilkFactory
{
    public partial class Client : Form
    {
        private string Username_;
        private string Password_;
        public Client(string Username, string Password)
        {
            Username_ = Username;
            Password_ = Password;
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            ClientShow clientShow = new ClientShow(Username_, Password_);
            clientShow.Show();
        }

        private void button2_Click(object sender, EventArgs e)
        {
            ClientAdd clientAdd = new ClientAdd(Username_, Password_);
            clientAdd.Show();
        }

        private void button3_Click(object sender, EventArgs e)
        {
            ClientDelete clientDelete = new ClientDelete(Username_, Password_);
            clientDelete.Show();
        }

        private void button4_Click(object sender, EventArgs e)
        {
            ClientUpdate clientUpdate = new ClientUpdate(Username_, Password_);
            clientUpdate.Show();
        }
    }
}

```

Листинг Б.2 – Код класса Client

```

namespace MilkFactory
{
    partial class Client
    {
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        /// <param name="disposing">true if managed resources should be disposed;
otherwise, false.</param>
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        #region Windows Form Designer generated code

        /// <summary>
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        private void InitializeComponent()
        {
            this.button1 = new System.Windows.Forms.Button();
            this.button2 = new System.Windows.Forms.Button();
            this.button3 = new System.Windows.Forms.Button();
            this.button4 = new System.Windows.Forms.Button();
            this.SuspendLayout();
            //
            // button1
            //
            this.button1.Location = new System.Drawing.Point(73, 212);
            this.button1.Name = "button1";
            this.button1.Size = new System.Drawing.Size(153, 62);
            this.button1.TabIndex = 0;
        }
    }
}

```

```

this.button1.Text = "Показать клиентов";
this.button1.UseVisualStyleBackColor = true;
this.button1.Click += new System.EventHandler(this.button1_Click);
//
// button2
//
this.button2.Location = new System.Drawing.Point(73, 83);
this.button2.Name = "button2";
this.button2.Size = new System.Drawing.Size(153, 62);
this.button2.TabIndex = 1;
this.button2.Text = "Добавить клиента";
this.button2.UseVisualStyleBackColor = true;
this.button2.Click += new System.EventHandler(this.button2_Click);
//
// button3
//
this.button3.Location = new System.Drawing.Point(355, 83);
this.button3.Name = "button3";
this.button3.Size = new System.Drawing.Size(153, 62);
this.button3.TabIndex = 2;
this.button3.Text = "Удалить клиента";
this.button3.UseVisualStyleBackColor = true;
this.button3.Click += new System.EventHandler(this.button3_Click);
//
// button4
//
this.button4.Location = new System.Drawing.Point(355, 212);
this.button4.Name = "button4";
this.button4.Size = new System.Drawing.Size(153, 62);
this.button4.TabIndex = 3;
this.button4.Text = "Редактировать клиента";
this.button4.UseVisualStyleBackColor = true;
this.button4.Click += new System.EventHandler(this.button4_Click);
//
// Client
//
this.AutoScaleDimensions = new System.Drawing.SizeF(8F, 16F);
this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
this.ClientSize = new System.Drawing.Size(582, 353);
this.Controls.Add(this.button4);
this.Controls.Add(this.button3);
this.Controls.Add(this.button2);
this.Controls.Add(this.button1);

```



```

        this.Name = "Client";
        this.Text = "Client";
        this.ResumeLayout(false);

    }

#endregion

private System.Windows.Forms.Button button1;
private System.Windows.Forms.Button button2;
private System.Windows.Forms.Button button3;
private System.Windows.Forms.Button button4;
    }
}

```

Листинг Б.3 – Код класса ClientAdd (форма)

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

using Npgsql;

namespace MilkFactory
{
    public partial class ClientAdd : Form
    {
        private string Username_;
        private string Password_;
        public ClientAdd(string Username, string Password)
        {
            Username_ = Username;
            Password_ = Password;
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)

```

```

{
    try
    {
        string fio = textBox1.Text.ToString();
        string address = textBox2.Text.ToString();
        string phoneNumber = textBox3.Text.ToString();

        if (!string.IsNullOrEmpty(fio) && !string.IsNullOrEmpty(address)
        && !string.IsNullOrEmpty(phoneNumber))
        {
            string connectionString = "Host=localhost;Username=" +
Username_ + ";Password=" + Password_ + ";Database=MilkFactory";

            var con = new NpgsqlConnection(connectionString);
            con.Open();

            string sql = "INSERT INTO mf.\"Client\"( \"FIO\", \"Address\",
\"PhoneNumber\") VALUES('\" + fio + \"', '\" + address + \"', '\" + phoneNumber +
\"')\";";

            var cmd = new NpgsqlCommand(sql, con);
            cmd.ExecuteNonQuery();

            con.Close();
            textBox4.Text = "New entry successfully added";
        } else
        {
            textBox4.Text = "The data must not be empty";
        }

    } catch (Exception ex)
    {
        Console.WriteLine(ex.Message);
        textBox4.Text = "Error, something wrong";
    }
}
}

```

Листинг Б.4 – Код класса ClientAdd

```
namespace MilkFactory
```

```

{
    partial class ClientAdd
    {
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        /// <param name="disposing">true if managed resources should be disposed;
otherwise, false.</param>
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        #region Windows Form Designer generated code

        /// <summary>
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        private void InitializeComponent()
        {
            this.textBox1 = new System.Windows.Forms.TextBox();
            this.textBox2 = new System.Windows.Forms.TextBox();
            this.textBox3 = new System.Windows.Forms.TextBox();
            this.label1 = new System.Windows.Forms.Label();
            this.label2 = new System.Windows.Forms.Label();
            this.label3 = new System.Windows.Forms.Label();
            this.button1 = new System.Windows.Forms.Button();
            this.textBox4 = new System.Windows.Forms.TextBox();
            this.SuspendLayout();
            //
            // textBox1
            //
            this.textBox1.Location = new System.Drawing.Point(40, 120);

```

```
this.textBox1.Multiline = true;
this.textBox1.Name = "textBox1";
this.textBox1.Size = new System.Drawing.Size(191, 46);
this.textBox1.TabIndex = 0;
//
// textBox2
//
this.textBox2.Location = new System.Drawing.Point(295, 120);
this.textBox2.Multiline = true;
this.textBox2.Name = "textBox2";
this.textBox2.Size = new System.Drawing.Size(191, 46);
this.textBox2.TabIndex = 1;
//
// textBox3
//
this.textBox3.Location = new System.Drawing.Point(548, 120);
this.textBox3.Multiline = true;
this.textBox3.Name = "textBox3";
this.textBox3.Size = new System.Drawing.Size(191, 46);
this.textBox3.TabIndex = 2;
//
// label1
//
this.label1.AutoSize = true;
this.label1.Location = new System.Drawing.Point(56, 73);
this.label1.Name = "label1";
this.label1.Size = new System.Drawing.Size(161, 16);
this.label1.TabIndex = 3;
this.label1.Text = "Фамилия Имя Отчество";
//
// label2
//
this.label2.AutoSize = true;
this.label2.Location = new System.Drawing.Point(360, 73);
this.label2.Name = "label2";
this.label2.Size = new System.Drawing.Size(47, 16);
this.label2.TabIndex = 4;
this.label2.Text = "Адрес";
//
// label3
//
this.label3.AutoSize = true;
this.label3.Location = new System.Drawing.Point(581, 73);
```

```

this.label3.Name = "label3";
this.label3.Size = new System.Drawing.Size(119, 16);
this.label3.TabIndex = 5;
this.label3.Text = "Номер телефона";
//
// button1
//
this.button1.Location = new System.Drawing.Point(295, 241);
this.button1.Name = "button1";
this.button1.Size = new System.Drawing.Size(191, 56);
this.button1.TabIndex = 6;
this.button1.Text = "Добавить";
this.button1.UseVisualStyleBackColor = true;
this.button1.Click += new System.EventHandler(this.button1_Click);
//
// textBox4
//
this.textBox4.Location = new System.Drawing.Point(295, 352);
this.textBox4.Multiline = true;
this.textBox4.Name = "textBox4";
this.textBox4.Size = new System.Drawing.Size(191, 46);
this.textBox4.TabIndex = 7;
//
// ClientAdd
//
this.AutoScaleDimensions = new System.Drawing.SizeF(8F, 16F);
this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
this.ClientSize = new System.Drawing.Size(802, 453);
this.Controls.Add(this.textBox4);
this.Controls.Add(this.button1);
this.Controls.Add(this.label3);
this.Controls.Add(this.label2);
this.Controls.Add(this.label1);
this.Controls.Add(this.textBox3);
this.Controls.Add(this.textBox2);
this.Controls.Add(this.textBox1);
this.Name = "ClientAdd";
this.Text = "ClientAdd";
this.ResumeLayout(false);
this.PerformLayout();

```

```

}

```

```

#endregion

private System.Windows.Forms.TextBox textBox1;
private System.Windows.Forms.TextBox textBox2;
private System.Windows.Forms.TextBox textBox3;
private System.Windows.Forms.Label label1;
private System.Windows.Forms.Label label2;
private System.Windows.Forms.Label label3;
private System.Windows.Forms.Button button1;
private System.Windows.Forms.TextBox textBox4;
}
}

```

Листинг Б.5 – Код класса ClientDelete (форма)

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

using Npgsql;

namespace MilkFactory
{
    public partial class ClientDelete : Form
    {
        private string Username_;
        private string Password_;
        public ClientDelete(string Username, string Password)
        {
            Username_ = Username;
            Password_ = Password;
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            try

```

```

{
    string fio = textBox1.Text.ToString();
    string address = textBox2.Text.ToString();
    string phoneNumber = textBox3.Text.ToString();

    if (!string.IsNullOrEmpty(fio) && !string.IsNullOrEmpty(address)
    && !string.IsNullOrEmpty(phoneNumber))
    {
        string connectionString = "Host=localhost;Username=" +
Username_ + ";Password=" + Password_ + ";Database=MilkFactory";

        var con = new NpgsqlConnection(connectionString);
        con.Open();

        string sql = "DELETE FROM mf.\"Client\" WHERE \"FIO\" = '" +
fio + "' AND \"Address\" = '" + address + "' AND \"PhoneNumber\" = '" + phoneNumber
+ "'; ";

        var cmd = new NpgsqlCommand(sql, con);
        cmd.ExecuteNonQuery();

        con.Close();
        textBox4.Text = "Entry successfully deleted";
    }
    else
    {
        textBox4.Text = "The data must not be empty";
    }

}
catch (Exception ex)
{
    Console.WriteLine(ex.Message);
    textBox4.Text = "Error, something wrong";
}
}
}

```

Листинг Б.6 – Код класса ClientDelete

```

namespace MilkFactory
{
    partial class ClientDelete
    {
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        /// <param name="disposing">true if managed resources should be disposed;
otherwise, false.</param>
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        #region Windows Form Designer generated code

        /// <summary>
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        private void InitializeComponent()
        {
            this.textBox4 = new System.Windows.Forms.TextBox();
            this.button1 = new System.Windows.Forms.Button();
            this.label3 = new System.Windows.Forms.Label();
            this.label2 = new System.Windows.Forms.Label();
            this.label1 = new System.Windows.Forms.Label();
            this.textBox3 = new System.Windows.Forms.TextBox();
            this.textBox2 = new System.Windows.Forms.TextBox();
            this.textBox1 = new System.Windows.Forms.TextBox();
            this.SuspendLayout();
            //
            // textBox4
            //

```



```

this.textBox4.Location = new System.Drawing.Point(306, 342);
this.textBox4.Multiline = true;
this.textBox4.Name = "textBox4";
this.textBox4.Size = new System.Drawing.Size(191, 46);
this.textBox4.TabIndex = 15;
//
// button1
//
this.button1.Location = new System.Drawing.Point(306, 231);
this.button1.Name = "button1";
this.button1.Size = new System.Drawing.Size(191, 56);
this.button1.TabIndex = 14;
this.button1.Text = "Удалить";
this.button1.UseVisualStyleBackColor = true;
this.button1.Click += new System.EventHandler(this.button1_Click);
//
// label3
//
this.label3.AutoSize = true;
this.label3.Location = new System.Drawing.Point(592, 63);
this.label3.Name = "label3";
this.label3.Size = new System.Drawing.Size(119, 16);
this.label3.TabIndex = 13;
this.label3.Text = "Номер телефона";
//
// label2
//
this.label2.AutoSize = true;
this.label2.Location = new System.Drawing.Point(371, 63);
this.label2.Name = "label2";
this.label2.Size = new System.Drawing.Size(47, 16);
this.label2.TabIndex = 12;
this.label2.Text = "Адрес";
//
// label1
//
this.label1.AutoSize = true;
this.label1.Location = new System.Drawing.Point(67, 63);
this.label1.Name = "label1";
this.label1.Size = new System.Drawing.Size(161, 16);
this.label1.TabIndex = 11;
this.label1.Text = "Фамилия Имя Отчество";
//

```

```

// textBox3
//
this.textBox3.Location = new System.Drawing.Point(559, 110);
this.textBox3.Multiline = true;
this.textBox3.Name = "textBox3";
this.textBox3.Size = new System.Drawing.Size(191, 46);
this.textBox3.TabIndex = 10;
//
// textBox2
//
this.textBox2.Location = new System.Drawing.Point(306, 110);
this.textBox2.Multiline = true;
this.textBox2.Name = "textBox2";
this.textBox2.Size = new System.Drawing.Size(191, 46);
this.textBox2.TabIndex = 9;
//
// textBox1
//
this.textBox1.Location = new System.Drawing.Point(51, 110);
this.textBox1.Multiline = true;
this.textBox1.Name = "textBox1";
this.textBox1.Size = new System.Drawing.Size(191, 46);
this.textBox1.TabIndex = 8;
//
// ClientDelete
//
this.AutoScaleDimensions = new System.Drawing.SizeF(8F, 16F);
this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
this.ClientSize = new System.Drawing.Size(800, 450);
this.Controls.Add(this.textBox4);
this.Controls.Add(this.button1);
this.Controls.Add(this.label3);
this.Controls.Add(this.label2);
this.Controls.Add(this.label1);
this.Controls.Add(this.textBox3);
this.Controls.Add(this.textBox2);
this.Controls.Add(this.textBox1);
this.Name = "ClientDelete";
this.Text = "ClientDelete";
this.ResumeLayout(false);
this.PerformLayout();

```

```

}

```

```

#endregion

private System.Windows.Forms.TextBox textBox4;
private System.Windows.Forms.Button button1;
private System.Windows.Forms.Label label3;
private System.Windows.Forms.Label label2;
private System.Windows.Forms.Label label1;
private System.Windows.Forms.TextBox textBox3;
private System.Windows.Forms.TextBox textBox2;
private System.Windows.Forms.TextBox textBox1;
}
}

```

Листинг Б.7 – Код класса ClientShow (форма)

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

using Npgsql;

namespace MilkFactory
{
    public partial class ClientShow : Form
    {
        private string Username_;
        private string Password_;
        public ClientShow(string Username, string Password)
        {
            Username_ = Username;
            Password_ = Password;
            InitializeComponent();
        }

        private void pictureBox1_Click(object sender, EventArgs e)
        {
            try
            {
                string connectionString = "Host=localhost;Username=" + Username_
+ ";Password=" + Password_ + ";Database=MilkFactory";

                var con = new NpgsqlConnection(connectionString);
                con.Open();

                string sql = "SELECT \"ClientID\", \"FIO\", \"Address\", \"Pho-
neNumber\" " +
                    "FROM mf.\"Client\"; ";
            }
            catch { }
        }
    }
}

```

```

var cmd = new NpgsqlCommand(sql, con);
NpgsqlDataReader dr = cmd.ExecuteReader();

int i = 1;
while (dr.Read())
{
    string text = dr[0].ToString();
    text += "\t" + dr[1].ToString() + "\t" + dr[2].ToString() +
"\t" + dr[3].ToString();
    Brush brush = new SolidBrush(Color.Black);
    Font font = new System.Drawing.Font("Times New Roman", 9,
FontStyle.Bold);
    Graphics graphics = pictureBox1.CreateGraphics();
    graphics.TextRenderingHint = System.Drawing.Text.TextRender-
ingHint.AntiAlias;
    graphics.DrawString(text, font, brush, 50, 20 * i);
    i++;
}

con.Close();
}
catch(Exception ex)
{
    Console.WriteLine(ex.Message);
}
}
}
}

```

Листинг Б.8 – Код класса ClientShow

```

namespace MilkFactory
{
    partial class ClientShow
    {
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        /// <param name="disposing">true if managed resources should be disposed;
        otherwise, false.</param>
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        #region Windows Form Designer generated code

        /// <summary>
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        private void InitializeComponent()
        {

```

```

        this.pictureBox1 = new System.Windows.Forms.PictureBox();
        ((System.ComponentModel.ISupportInitialize)(this.pictureBox1)).BeginInit();
        this.SuspendLayout();
        //
        // pictureBox1
        //
        this.pictureBox1.Location = new System.Drawing.Point(12, 12);
        this.pictureBox1.Name = "pictureBox1";
        this.pictureBox1.Size = new System.Drawing.Size(1058, 629);
        this.pictureBox1.TabIndex = 0;
        this.pictureBox1.TabStop = false;
        this.pictureBox1.Click += new System.EventHandler(this.pictureBox1_Click);
        //
        // ClientShow
        //
        this.AutoScaleMode = new System.Drawing.SizeF(8F, 16F);
        this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
        this.ClientSize = new System.Drawing.Size(1082, 653);
        this.Controls.Add(this.pictureBox1);
        this.Name = "ClientShow";
        this.Text = "ClientShow";
        ((System.ComponentModel.ISupportInitialize)(this.pictureBox1)).EndInit();
        this.ResumeLayout(false);

    }

    #endregion

    private System.Windows.Forms.PictureBox pictureBox1;
}

```

Листинг Б.9 – Код класса ClientUpdate (форма)

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

using Npgsql;

namespace MilkFactory
{
    public partial class ClientUpdate : Form
    {
        private string Username_;
        private string Password_;
        public ClientUpdate(string Username, string Password)
        {
            Username_ = Username;
            Password_ = Password;
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {

```

```

try
{
    string fioOld = textBox1.Text.ToString();
    string addressOld = textBox2.Text.ToString();
    string phoneNumberOld = textBox3.Text.ToString();

    string fioNew = textBox5.Text.ToString();
    string addressNew = textBox6.Text.ToString();
    string phoneNumberNew = textBox7.Text.ToString();

    if (!string.IsNullOrEmpty(fioOld) && !string.IsNullOrEmpty(ad-
dressOld) && !string.IsNullOrEmpty(phoneNumberOld) &&
        !string.IsNullOrEmpty(fioNew) && !string.IsNullOrEmpty(ad-
dressNew) && !string.IsNullOrEmpty(phoneNumberNew))
    {
        string connectionString = "Host=localhost;Username=" +
Username_ + ";Password=" + Password_ + ";Database=MilkFactory";

        var con = new NpgsqlConnection(connectionString);
        con.Open();

        string sql = "UPDATE mf.\"Client\" SET \"FIO\"='" + fioNew +
        "', \"Address\"='" + addressNew + "', \"PhoneNumber\"='" + phoneNumberNew + "' "
        +
        "WHERE \"FIO\" = '" + fioOld + "' AND \"Address\" = '" +
addressOld + "' AND \"PhoneNumber\" = '" + phoneNumberOld + "'; ";

        var cmd = new NpgsqlCommand(sql, con);
        cmd.ExecuteNonQuery();

        con.Close();
        textBox4.Text = "Entry successfully updated";
    }
    else
    {
        textBox4.Text = "The data must not be empty";
    }
}
catch (Exception ex)
{
    Console.WriteLine(ex.Message);
    textBox4.Text = "Error, something wrong";
}
}
}

```

Листинг Б.10 – Код класса ClientUpdate

```

namespace MilkFactory
{
    partial class ClientUpdate
    {
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
    }
}

```

```

    /// <param name="disposing">true if managed resources should be disposed;
    otherwise, false.</param>
    protected override void Dispose(bool disposing)
    {
        if (disposing && (components != null))
        {
            components.Dispose();
        }
        base.Dispose(disposing);
    }

    #region Windows Form Designer generated code

    /// <summary>
    /// Required method for Designer support - do not modify
    /// the contents of this method with the code editor.
    /// </summary>
    private void InitializeComponent()
    {
        this.textBox4 = new System.Windows.Forms.TextBox();
        this.button1 = new System.Windows.Forms.Button();
        this.label3 = new System.Windows.Forms.Label();
        this.label2 = new System.Windows.Forms.Label();
        this.label1 = new System.Windows.Forms.Label();
        this.textBox3 = new System.Windows.Forms.TextBox();
        this.textBox2 = new System.Windows.Forms.TextBox();
        this.textBox1 = new System.Windows.Forms.TextBox();
        this.textBox5 = new System.Windows.Forms.TextBox();
        this.textBox6 = new System.Windows.Forms.TextBox();
        this.textBox7 = new System.Windows.Forms.TextBox();
        this.label4 = new System.Windows.Forms.Label();
        this.label5 = new System.Windows.Forms.Label();
        this.label6 = new System.Windows.Forms.Label();
        this.label7 = new System.Windows.Forms.Label();
        this.label8 = new System.Windows.Forms.Label();
        this.SuspendLayout();
        //
        // textBox4
        //
        this.textBox4.Location = new System.Drawing.Point(445, 459);
        this.textBox4.Multiline = true;
        this.textBox4.Name = "textBox4";
        this.textBox4.Size = new System.Drawing.Size(191, 46);
        this.textBox4.TabIndex = 23;
        //
        // button1
        //
        this.button1.Location = new System.Drawing.Point(445, 348);
        this.button1.Name = "button1";
        this.button1.Size = new System.Drawing.Size(191, 56);
        this.button1.TabIndex = 22;
        this.button1.Text = "Редактировать";
        this.button1.UseVisualStyleBackColor = true;
        this.button1.Click += new System.EventHandler(this.button1_Click);
        //
        // label3
        //
        this.label3.AutoSize = true;
        this.label3.Location = new System.Drawing.Point(731, 56);
        this.label3.Name = "label3";
        this.label3.Size = new System.Drawing.Size(119, 16);
        this.label3.TabIndex = 21;
        this.label3.Text = "Номер телефона";
        //

```

```

// label2
//
this.label2.AutoSize = true;
this.label2.Location = new System.Drawing.Point(510, 56);
this.label2.Name = "label2";
this.label2.Size = new System.Drawing.Size(47, 16);
this.label2.TabIndex = 20;
this.label2.Text = "Адрес";
//
// label1
//
this.label1.AutoSize = true;
this.label1.Location = new System.Drawing.Point(206, 56);
this.label1.Name = "label1";
this.label1.Size = new System.Drawing.Size(161, 16);
this.label1.TabIndex = 19;
this.label1.Text = "ФИМИЛИЯ Имя Отчество";
//
// textBox3
//
this.textBox3.Location = new System.Drawing.Point(698, 103);
this.textBox3.Multiline = true;
this.textBox3.Name = "textBox3";
this.textBox3.Size = new System.Drawing.Size(191, 46);
this.textBox3.TabIndex = 18;
//
// textBox2
//
this.textBox2.Location = new System.Drawing.Point(445, 103);
this.textBox2.Multiline = true;
this.textBox2.Name = "textBox2";
this.textBox2.Size = new System.Drawing.Size(191, 46);
this.textBox2.TabIndex = 17;
//
// textBox1
//
this.textBox1.Location = new System.Drawing.Point(190, 103);
this.textBox1.Multiline = true;
this.textBox1.Name = "textBox1";
this.textBox1.Size = new System.Drawing.Size(191, 46);
this.textBox1.TabIndex = 16;
//
// textBox5
//
this.textBox5.Location = new System.Drawing.Point(190, 243);
this.textBox5.Multiline = true;
this.textBox5.Name = "textBox5";
this.textBox5.Size = new System.Drawing.Size(191, 46);
this.textBox5.TabIndex = 26;
//
// textBox6
//
this.textBox6.Location = new System.Drawing.Point(445, 240);
this.textBox6.Multiline = true;
this.textBox6.Name = "textBox6";
this.textBox6.Size = new System.Drawing.Size(191, 46);
this.textBox6.TabIndex = 25;
//
// textBox7
//
this.textBox7.Location = new System.Drawing.Point(698, 243);
this.textBox7.Multiline = true;
this.textBox7.Name = "textBox7";
this.textBox7.Size = new System.Drawing.Size(191, 46);

```



```

this.textBox7.TabIndex = 24;
//
// label4
//
this.label4.AutoSize = true;
this.label4.Location = new System.Drawing.Point(743, 202);
this.label4.Name = "label4";
this.label4.Size = new System.Drawing.Size(119, 16);
this.label4.TabIndex = 29;
this.label4.Text = "Номер телефона";
//
// label5
//
this.label5.AutoSize = true;
this.label5.Location = new System.Drawing.Point(522, 202);
this.label5.Name = "label5";
this.label5.Size = new System.Drawing.Size(47, 16);
this.label5.TabIndex = 28;
this.label5.Text = "Адрес";
//
// label6
//
this.label6.AutoSize = true;
this.label6.Location = new System.Drawing.Point(218, 202);
this.label6.Name = "label6";
this.label6.Size = new System.Drawing.Size(161, 16);
this.label6.TabIndex = 27;
this.label6.Text = "Фамилия Имя Отчество";
//
// label7
//
this.label7.AutoSize = true;
this.label7.Location = new System.Drawing.Point(49, 118);
this.label7.Name = "label7";
this.label7.Size = new System.Drawing.Size(56, 16);
this.label7.TabIndex = 30;
this.label7.Text = "Старые";
//
// label8
//
this.label8.AutoSize = true;
this.label8.Location = new System.Drawing.Point(49, 243);
this.label8.Name = "label8";
this.label8.Size = new System.Drawing.Size(50, 16);
this.label8.TabIndex = 31;
this.label8.Text = "Новые";
//
// ClientUpdate
//
this.AutoScaleDimensions = new System.Drawing.SizeF(8F, 16F);
this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
this.ClientSize = new System.Drawing.Size(1082, 553);
this.Controls.Add(this.label8);
this.Controls.Add(this.label7);
this.Controls.Add(this.label4);
this.Controls.Add(this.label5);
this.Controls.Add(this.label6);
this.Controls.Add(this.textBox5);
this.Controls.Add(this.textBox6);
this.Controls.Add(this.textBox7);
this.Controls.Add(this.textBox4);
this.Controls.Add(this.button1);
this.Controls.Add(this.label3);
this.Controls.Add(this.label2);

```

```

        this.Controls.Add(this.label1);
        this.Controls.Add(this.textBox3);
        this.Controls.Add(this.textBox2);
        this.Controls.Add(this.textBox1);
        this.Name = "ClientUpdate";
        this.Text = "ClientUpdate";
        this.ResumeLayout(false);
        this.PerformLayout();

    }

    #endregion

    private System.Windows.Forms.TextBox textBox4;
    private System.Windows.Forms.Button button1;
    private System.Windows.Forms.Label label3;
    private System.Windows.Forms.Label label2;
    private System.Windows.Forms.Label label1;
    private System.Windows.Forms.TextBox textBox3;
    private System.Windows.Forms.TextBox textBox2;
    private System.Windows.Forms.TextBox textBox1;
    private System.Windows.Forms.TextBox textBox5;
    private System.Windows.Forms.TextBox textBox6;
    private System.Windows.Forms.TextBox textBox7;
    private System.Windows.Forms.Label label4;
    private System.Windows.Forms.Label label5;
    private System.Windows.Forms.Label label6;
    private System.Windows.Forms.Label label7;
    private System.Windows.Forms.Label label8;
}
}

```

Листинг Б.11 – Код класса Delivery (форма)

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace MilkFactory
{
    public partial class Delivery : Form
    {
        private string Username_;
        private string Password_;
        public Delivery(string Username, string Password)
        {
            Username_ = Username;
            Password_ = Password;
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            DeliveryShow deliveryShow = new DeliveryShow(Username_, Password_);
            deliveryShow.Show();
        }

        private void button2_Click(object sender, EventArgs e)

```

```

    {
        DeliveryAdd deliveryAdd = new DeliveryAdd(Username_, Password_);
        deliveryAdd.Show();
    }

    private void button3_Click(object sender, EventArgs e)
    {
        DeliveryDelete deliveryDelete = new DeliveryDelete(Username_, Pass-
word_);
        deliveryDelete.Show();
    }
}

```

Листинг Б.12 – Код класса Delivery

```

namespace MilkFactory
{
    partial class Delivery
    {
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        /// <param name="disposing">true if managed resources should be disposed;
otherwise, false.</param>
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        #region Windows Form Designer generated code

        /// <summary>
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        private void InitializeComponent()
        {
            this.button4 = new System.Windows.Forms.Button();
            this.button3 = new System.Windows.Forms.Button();
            this.button2 = new System.Windows.Forms.Button();
            this.button1 = new System.Windows.Forms.Button();
            this.SuspendLayout();
            //
            // button4
            //
            this.button4.Location = new System.Drawing.Point(356, 210);
            this.button4.Name = "button4";
            this.button4.Size = new System.Drawing.Size(153, 62);
            this.button4.TabIndex = 7;
            this.button4.Text = "Редактировать поставки";
            this.button4.UseVisualStyleBackColor = true;
            //
            // button3
            //

```

```

this.button3.Location = new System.Drawing.Point(356, 81);
this.button3.Name = "button3";
this.button3.Size = new System.Drawing.Size(153, 62);
this.button3.TabIndex = 6;
this.button3.Text = "Удалить поставку";
this.button3.UseVisualStyleBackColor = true;
this.button3.Click += new System.EventHandler(this.button3_Click);
//
// button2
//
this.button2.Location = new System.Drawing.Point(74, 81);
this.button2.Name = "button2";
this.button2.Size = new System.Drawing.Size(153, 62);
this.button2.TabIndex = 5;
this.button2.Text = "Добавить поставку";
this.button2.UseVisualStyleBackColor = true;
this.button2.Click += new System.EventHandler(this.button2_Click);
//
// button1
//
this.button1.Location = new System.Drawing.Point(74, 210);
this.button1.Name = "button1";
this.button1.Size = new System.Drawing.Size(153, 62);
this.button1.TabIndex = 4;
this.button1.Text = "Показать поставки";
this.button1.UseVisualStyleBackColor = true;
this.button1.Click += new System.EventHandler(this.button1_Click);
//
// Delivery
//
this.AutoScaleDimensions = new System.Drawing.SizeF(8F, 16F);
this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
this.ClientSize = new System.Drawing.Size(582, 353);
this.Controls.Add(this.button4);
this.Controls.Add(this.button3);
this.Controls.Add(this.button2);
this.Controls.Add(this.button1);
this.Name = "Delivery";
this.Text = "Delivery";
this.ResumeLayout(false);

}

#endregion

private System.Windows.Forms.Button button4;
private System.Windows.Forms.Button button3;
private System.Windows.Forms.Button button2;
private System.Windows.Forms.Button button1;
}
}

```

Листинг Б.13 – Код класса DeliveryAdd (форма)

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

```

```

using Npgsql;

namespace MilkFactory
{
    public partial class DeliveryAdd : Form
    {
        private string Username_;
        private string Password_;
        public DeliveryAdd(string Username, string Password)
        {
            Username_ = Username;
            Password_ = Password;
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            try
            {
                string name = textBox1.Text.ToString();
                string address = textBox2.Text.ToString();
                string phoneNumber = textBox3.Text.ToString();

                string productName = textBox5.Text.ToString();
                string productType = textBox6.Text.ToString();
                string productCount = textBox7.Text.ToString();

                if (!string.IsNullOrEmpty(name) && !string.IsNullOrEmpty(address)
                    && !string.IsNullOrEmpty(phoneNumber) &&
                        !string.IsNullOrEmpty(productName) && !string.IsNul-
1NullOrEmpty(productType) && !string.IsNullOrEmpty(productCount))
                {
                    string connectionString = "Host=localhost;Username=" +
Username_ + ";Password=" + Password_ + ";Database=MilkFactory";

                    var con = new NpgsqlConnection(connectionString);
                    con.Open();

                    var cmd = new NpgsqlCommand();
                    cmd.Connection = con;

                    // check count products
                    string sql = "SELECT COUNT(*) " +
                                "FROM mf.\"Product\" " +
                                "WHERE \"ProductName\" = '" + productName + "'
AND \"Type\" = '" + productType + "'";
                    cmd.CommandText = sql;
                    NpgsqlDataReader dr = cmd.ExecuteReader();

                    dr.Read();
                    int countProducts = dr.GetInt32(0);

                    dr.Close();

                    // product with selected parameters is exist
                    if (countProducts > 0)
                    {

                        // get product id
                        sql = "SELECT \"ProductID\" " +
                            "FROM mf.\"Product\" " +
                            "WHERE \"ProductName\" = '" + productName + "' AND
\"Type\" = '" + productType + "'";
                        cmd.CommandText = sql;

```

```

dr = cmd.ExecuteReader();
dr.Read();
int productID = dr.GetInt32(0);
dr.Close();

// check count vendors with selected parameters
sql = "SELECT COUNT(\"Name\") AS count " +
      "FROM mf.\"Vendor\" " +
      "WHERE \"Name\" = '" + name + "' AND \"Ad-
dress\" = '" + address + "' " +
      "AND \"PhoneNumber\" = '" + phoneNumber +
      "'";

cmd.CommandText = sql;
dr = cmd.ExecuteReader();
dr.Read();
int countVendors = dr.GetInt32(0);
dr.Close();

// if vendor with selected parameters is exist
if (countVendors > 0)
{
    // get vendor id
    sql = "SELECT \"VendorID\" " +
          "FROM mf.\"Vendor\" " +
          "WHERE \"Name\" = '" + name + "' AND \"Address\"
= '" + address + "' " +
          "AND \"PhoneNumber\" = '" + phoneNumber + "'";

    cmd.CommandText = sql;
    dr = cmd.ExecuteReader();
    dr.Read();
    int vendorID = dr.GetInt32(0);
    dr.Close();

    // insert new delivery
    sql = "INSERT INTO mf.\"Delivery\" (" +
          "\"VendorID\", \"ProductID\",
\"ProductCount\", \"DateTime\") " +
          "VALUES(" + vendorID + ", " + productID + ", "
+ productCount + ", date_trunc('second', now()::timestamp)); ";
    cmd.CommandText = sql;
    cmd.ExecuteNonQuery();
}
else
{
    // insert new vendor with selected parameters
    sql = "INSERT INTO mf.\"Vendor\"(" +
          "\"Name\", \"Address\", \"PhoneNumber\") " +
          "VALUES('" + name + "', '" + address + "', '" +
phoneNumber + "'); ";

    cmd.CommandText = sql;
    cmd.ExecuteNonQuery();

    // get vendor id
    sql = "SELECT \"VendorID\" " +
          "FROM mf.\"Vendor\" " +
          "WHERE \"Name\" = '" + name + "' AND \"Address\"
= '" + address + "' " +
          "AND \"PhoneNumber\" = '" + phoneNumber + "'";
    cmd.CommandText = sql;

```

```

        dr = cmd.ExecuteReader();
        dr.Read();
        int vendorID = dr.GetInt32(0);
        dr.Close();

        // insert new delivery
        sql = "INSERT INTO mf.\"Delivery\" (" +
            "\"VendorID\",                \"ProductID\",
        \"ProductCount\", \"DateTime\") " +
            "VALUES(" + vendorID + ", " + productID + ", "
        + productCount + ", date_trunc('second', now()::timestamp)); ";
        cmd.CommandText = sql;
        cmd.ExecuteNonQuery();
    }
    textBox4.Text = "Entry successfully added";

    }
    else
    {
        textBox4.Text = "Cannot add an delivery because the se-
lected product does not exist";
    }
    con.Close();
}
else
{
    textBox4.Text = "The data must not be empty";
}

}
catch (Exception ex)
{
    Console.WriteLine(ex.Message);
    textBox4.Text = "Error, something wrong";
}
}
}
}

```

Листинг Б.14 – Код класса DeliveryAdd

```

namespace MilkFactory
{
    partial class DeliveryAdd
    {
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        /// <param name="disposing">true if managed resources should be disposed;
otherwise, false.</param>
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }
    }
}

```

```
#region Windows Form Designer generated code
```

```
/// <summary>
/// Required method for Designer support - do not modify
/// the contents of this method with the code editor.
/// </summary>
private void InitializeComponent()
{
    this.label6 = new System.Windows.Forms.Label();
    this.textBox7 = new System.Windows.Forms.TextBox();
    this.label4 = new System.Windows.Forms.Label();
    this.label5 = new System.Windows.Forms.Label();
    this.textBox5 = new System.Windows.Forms.TextBox();
    this.textBox6 = new System.Windows.Forms.TextBox();
    this.textBox4 = new System.Windows.Forms.TextBox();
    this.button1 = new System.Windows.Forms.Button();
    this.label3 = new System.Windows.Forms.Label();
    this.label2 = new System.Windows.Forms.Label();
    this.label1 = new System.Windows.Forms.Label();
    this.textBox3 = new System.Windows.Forms.TextBox();
    this.textBox2 = new System.Windows.Forms.TextBox();
    this.textBox1 = new System.Windows.Forms.TextBox();
    this.SuspendLayout();
    //
    // label6
    //
    this.label6.AutoSize = true;
    this.label6.Location = new System.Drawing.Point(581, 163);
    this.label6.Name = "label6";
    this.label6.Size = new System.Drawing.Size(150, 16);
    this.label6.TabIndex = 35;
    this.label6.Text = "Количество продукта";
    //
    // textBox7
    //
    this.textBox7.Location = new System.Drawing.Point(566, 210);
    this.textBox7.Multiline = true;
    this.textBox7.Name = "textBox7";
    this.textBox7.Size = new System.Drawing.Size(191, 46);
    this.textBox7.TabIndex = 34;
    //
    // label4
    //
    this.label4.AutoSize = true;
    this.label4.Location = new System.Drawing.Point(365, 163);
    this.label4.Name = "label4";
    this.label4.Size = new System.Drawing.Size(97, 16);
    this.label4.TabIndex = 33;
    this.label4.Text = "Тип продукта";
    //
    // label5
    //
    this.label5.AutoSize = true;
    this.label5.Location = new System.Drawing.Point(74, 163);
    this.label5.Name = "label5";
    this.label5.Size = new System.Drawing.Size(138, 16);
    this.label5.TabIndex = 32;
    this.label5.Text = "Название продукта";
    //
    // textBox5
    //
    this.textBox5.Location = new System.Drawing.Point(58, 210);
    this.textBox5.Multiline = true;
    this.textBox5.Name = "textBox5";
```



```

this.textBox5.Size = new System.Drawing.Size(191, 46);
this.textBox5.TabIndex = 31;
//
// textBox6
//
this.textBox6.Location = new System.Drawing.Point(313, 210);
this.textBox6.Multiline = true;
this.textBox6.Name = "textBox6";
this.textBox6.Size = new System.Drawing.Size(191, 46);
this.textBox6.TabIndex = 30;
//
// textBox4
//
this.textBox4.Location = new System.Drawing.Point(201, 426);
this.textBox4.Multiline = true;
this.textBox4.Name = "textBox4";
this.textBox4.Size = new System.Drawing.Size(429, 80);
this.textBox4.TabIndex = 29;
//
// button1
//
this.button1.Location = new System.Drawing.Point(313, 319);
this.button1.Name = "button1";
this.button1.Size = new System.Drawing.Size(191, 56);
this.button1.TabIndex = 28;
this.button1.Text = "Добавить";
this.button1.UseVisualStyleBackColor = true;
this.button1.Click += new System.EventHandler(this.button1_Click);
//
// label3
//
this.label3.AutoSize = true;
this.label3.Location = new System.Drawing.Point(563, 47);
this.label3.Name = "label3";
this.label3.Size = new System.Drawing.Size(200, 16);
this.label3.TabIndex = 27;
this.label3.Text = "Номер телефона поставщика";
//
// label2
//
this.label2.AutoSize = true;
this.label2.Location = new System.Drawing.Point(358, 47);
this.label2.Name = "label2";
this.label2.Size = new System.Drawing.Size(128, 16);
this.label2.TabIndex = 26;
this.label2.Text = "Адрес поставщика";
//
// label1
//
this.label1.AutoSize = true;
this.label1.Location = new System.Drawing.Point(74, 47);
this.label1.Name = "label1";
this.label1.Size = new System.Drawing.Size(154, 16);
this.label1.TabIndex = 25;
this.label1.Text = "Название поставщика";
//
// textBox3
//
this.textBox3.Location = new System.Drawing.Point(566, 94);
this.textBox3.Multiline = true;
this.textBox3.Name = "textBox3";
this.textBox3.Size = new System.Drawing.Size(191, 46);
this.textBox3.TabIndex = 24;
//

```

```

        // textBox2
        //
        this.textBox2.Location = new System.Drawing.Point(313, 94);
        this.textBox2.Multiline = true;
        this.textBox2.Name = "textBox2";
        this.textBox2.Size = new System.Drawing.Size(191, 46);
        this.textBox2.TabIndex = 23;
        //
        // textBox1
        //
        this.textBox1.Location = new System.Drawing.Point(58, 94);
        this.textBox1.Multiline = true;
        this.textBox1.Name = "textBox1";
        this.textBox1.Size = new System.Drawing.Size(191, 46);
        this.textBox1.TabIndex = 22;
        //
        // DeliveryAdd
        //
        this.AutoScaleDimensions = new System.Drawing.SizeF(8F, 16F);
        this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
        this.ClientSize = new System.Drawing.Size(802, 553);
        this.Controls.Add(this.label6);
        this.Controls.Add(this.textBox7);
        this.Controls.Add(this.label4);
        this.Controls.Add(this.label5);
        this.Controls.Add(this.textBox5);
        this.Controls.Add(this.textBox6);
        this.Controls.Add(this.textBox4);
        this.Controls.Add(this.button1);
        this.Controls.Add(this.label3);
        this.Controls.Add(this.label2);
        this.Controls.Add(this.label1);
        this.Controls.Add(this.textBox3);
        this.Controls.Add(this.textBox2);
        this.Controls.Add(this.textBox1);
        this.Name = "DeliveryAdd";
        this.Text = "DeliveryAdd";
        this.ResumeLayout(false);
        this.PerformLayout();

    }

    #endregion

    private System.Windows.Forms.Label label6;
    private System.Windows.Forms.TextBox textBox7;
    private System.Windows.Forms.Label label4;
    private System.Windows.Forms.Label label5;
    private System.Windows.Forms.TextBox textBox5;
    private System.Windows.Forms.TextBox textBox6;
    private System.Windows.Forms.TextBox textBox4;
    private System.Windows.Forms.Button button1;
    private System.Windows.Forms.Label label3;
    private System.Windows.Forms.Label label2;
    private System.Windows.Forms.Label label1;
    private System.Windows.Forms.TextBox textBox3;
    private System.Windows.Forms.TextBox textBox2;
    private System.Windows.Forms.TextBox textBox1;
}
}

```

Листинг Б.15 – Код класса DeliveryDelete (форма)

```
using System;
```

```

using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

using Npgsql;

namespace MilkFactory
{
    public partial class DeliveryDelete : Form
    {
        private string Username_;
        private string Password_;
        public DeliveryDelete(string Username, string Password)
        {
            Username_ = Username;
            Password_ = Password;
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            try
            {
                string name = textBox1.Text.ToString();
                string address = textBox2.Text.ToString();
                string phoneNumber = textBox3.Text.ToString();

                string productName = textBox5.Text.ToString();
                string productType = textBox6.Text.ToString();
                string productCount = textBox7.Text.ToString();

                if (!string.IsNullOrEmpty(name) && !string.IsNullOrEmpty(address)
                    && !string.IsNullOrEmpty(phoneNumber) &&
                        !string.IsNullOrEmpty(productName) && !string.IsNul-
1NullOrEmpty(productType) && !string.IsNullOrEmpty(productCount))
                {
                    string connectionString = "Host=localhost;Username=" +
Username_ + ";Password=" + Password_ + ";Database=MilkFactory";

                    var con = new NpgsqlConnection(connectionString);
                    con.Open();

                    var cmd = new NpgsqlCommand();
                    cmd.Connection = con;

                    // check count products
                    string sql = "SELECT COUNT(*) " +
                                "FROM mf.\"Product\" " +
                                "WHERE \"ProductName\" = '" + productName + "'
AND \"Type\" = '" + productType + "'";
                    cmd.CommandText = sql;
                    NpgsqlDataReader dr = cmd.ExecuteReader();

                    dr.Read();
                    int countProducts = dr.GetInt32(0);

                    dr.Close();

                    // product with selected parameters is exist

```

```

if (countProducts > 0)
{
    // get product id
    sql = "SELECT \"ProductID\" " +
        "FROM mf.\"Product\" " +
        "WHERE \"ProductName\" = '" + productName + "' AND
\"Type\" = '" + productType + "'";
    cmd.CommandText = sql;
    dr = cmd.ExecuteReader();
    dr.Read();
    int productID = dr.GetInt32(0);
    dr.Close();

    // check count vendors with selected parameters
    sql = "SELECT COUNT(\"Name\") AS count " +
        "FROM mf.\"Vendor\" " +
        "WHERE \"Name\" = '" + name + "' AND \"Ad-
dress\" = '" + address + "' " +
        "AND \"PhoneNumber\" = '" + phoneNumber +
        "'";

    cmd.CommandText = sql;
    dr = cmd.ExecuteReader();
    dr.Read();
    int countVendors = dr.GetInt32(0);
    dr.Close();

    // if vendor with selected parameters is exist
    if (countVendors > 0)
    {
        // get vendor id
        sql = "SELECT \"VendorID\" " +
            "FROM mf.\"Vendor\" " +
            "WHERE \"Name\" = '" + name + "' AND \"Address\"
= '" + address + "' " +
            "AND \"PhoneNumber\" = '" + phoneNumber + "'";

        cmd.CommandText = sql;
        dr = cmd.ExecuteReader();
        dr.Read();
        int vendorID = dr.GetInt32(0);
        dr.Close();

        // delete delivery
        sql = "DELETE FROM mf.\"Delivery\" " +
            "WHERE \"DeliveryID\" IN( " +
            "SELECT \"DeliveryID\" " +
            "FROM mf.\"Delivery\" " +
            "WHERE \"VendorID\" = " + vendorID + " AND
\"ProductID\" = " + productID + " AND \"ProductCount\" = " + productCount +
            " ORDER BY \"DateTime\" DESC " +
            "LIMIT 1 )";

        Console.WriteLine(sql);
        cmd.CommandText = sql;
        cmd.ExecuteNonQuery();

        textBox4.Text = "Entry successfully deleted";
    }
    else
    {

```

```

        textBox4.Text = "Cannot delete a delivery because the
selected vendor does not exist";
    }

    }
    else
    {
        textBox4.Text = "Cannot delete a delivery because the
selected product does not exist";
    }
    con.Close();
}
else
{
    textBox4.Text = "The data must not be empty";
}

}
catch (Exception ex)
{
    Console.WriteLine(ex.Message);
    textBox4.Text = "Error, something wrong";
}
}
}
}

```

Листинг Б.16 – Код класса DeliveryDelete

```

namespace MilkFactory
{
    partial class DeliveryDelete
    {
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        /// <param name="disposing">true if managed resources should be disposed;
otherwise, false.</param>
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        #region Windows Form Designer generated code

        /// <summary>
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        private void InitializeComponent()
        {
            this.label6 = new System.Windows.Forms.Label();
            this.textBox7 = new System.Windows.Forms.TextBox();
            this.label4 = new System.Windows.Forms.Label();

```

```

this.label5 = new System.Windows.Forms.Label();
this.textBox5 = new System.Windows.Forms.TextBox();
this.textBox6 = new System.Windows.Forms.TextBox();
this.textBox4 = new System.Windows.Forms.TextBox();
this.button1 = new System.Windows.Forms.Button();
this.label3 = new System.Windows.Forms.Label();
this.label2 = new System.Windows.Forms.Label();
this.label1 = new System.Windows.Forms.Label();
this.textBox3 = new System.Windows.Forms.TextBox();
this.textBox2 = new System.Windows.Forms.TextBox();
this.textBox1 = new System.Windows.Forms.TextBox();
this.SuspendLayout();
//
// label6
//
this.label6.AutoSize = true;
this.label6.Location = new System.Drawing.Point(572, 163);
this.label6.Name = "label6";
this.label6.Size = new System.Drawing.Size(150, 16);
this.label6.TabIndex = 49;
this.label6.Text = "Количество продукта";
//
// textBox7
//
this.textBox7.Location = new System.Drawing.Point(557, 210);
this.textBox7.Multiline = true;
this.textBox7.Name = "textBox7";
this.textBox7.Size = new System.Drawing.Size(191, 46);
this.textBox7.TabIndex = 48;
//
// label4
//
this.label4.AutoSize = true;
this.label4.Location = new System.Drawing.Point(356, 163);
this.label4.Name = "label4";
this.label4.Size = new System.Drawing.Size(97, 16);
this.label4.TabIndex = 47;
this.label4.Text = "Тип продукта";
//
// label5
//
this.label5.AutoSize = true;
this.label5.Location = new System.Drawing.Point(65, 163);
this.label5.Name = "label5";
this.label5.Size = new System.Drawing.Size(138, 16);
this.label5.TabIndex = 46;
this.label5.Text = "Название продукта";
//
// textBox5
//
this.textBox5.Location = new System.Drawing.Point(49, 210);
this.textBox5.Multiline = true;
this.textBox5.Name = "textBox5";
this.textBox5.Size = new System.Drawing.Size(191, 46);
this.textBox5.TabIndex = 45;
//
// textBox6
//
this.textBox6.Location = new System.Drawing.Point(304, 210);
this.textBox6.Multiline = true;
this.textBox6.Name = "textBox6";
this.textBox6.Size = new System.Drawing.Size(191, 46);
this.textBox6.TabIndex = 44;
//

```

```

// textBox4
//
this.textBox4.Location = new System.Drawing.Point(192, 426);
this.textBox4.Multiline = true;
this.textBox4.Name = "textBox4";
this.textBox4.Size = new System.Drawing.Size(429, 80);
this.textBox4.TabIndex = 43;
//
// button1
//
this.button1.Location = new System.Drawing.Point(304, 319);
this.button1.Name = "button1";
this.button1.Size = new System.Drawing.Size(191, 56);
this.button1.TabIndex = 42;
this.button1.Text = "Удалить";
this.button1.UseVisualStyleBackColor = true;
this.button1.Click += new System.EventHandler(this.button1_Click);
//
// label3
//
this.label3.AutoSize = true;
this.label3.Location = new System.Drawing.Point(554, 47);
this.label3.Name = "label3";
this.label3.Size = new System.Drawing.Size(200, 16);
this.label3.TabIndex = 41;
this.label3.Text = "Номер телефона поставщика";
//
// label2
//
this.label2.AutoSize = true;
this.label2.Location = new System.Drawing.Point(349, 47);
this.label2.Name = "label2";
this.label2.Size = new System.Drawing.Size(128, 16);
this.label2.TabIndex = 40;
this.label2.Text = "Адрес поставщика";
//
// label1
//
this.label1.AutoSize = true;
this.label1.Location = new System.Drawing.Point(65, 47);
this.label1.Name = "label1";
this.label1.Size = new System.Drawing.Size(154, 16);
this.label1.TabIndex = 39;
this.label1.Text = "Название поставщика";
//
// textBox3
//
this.textBox3.Location = new System.Drawing.Point(557, 94);
this.textBox3.Multiline = true;
this.textBox3.Name = "textBox3";
this.textBox3.Size = new System.Drawing.Size(191, 46);
this.textBox3.TabIndex = 38;
//
// textBox2
//
this.textBox2.Location = new System.Drawing.Point(304, 94);
this.textBox2.Multiline = true;
this.textBox2.Name = "textBox2";
this.textBox2.Size = new System.Drawing.Size(191, 46);
this.textBox2.TabIndex = 37;
//
// textBox1
//
this.textBox1.Location = new System.Drawing.Point(49, 94);

```

```

        this.textBox1.Multiline = true;
        this.textBox1.Name = "textBox1";
        this.textBox1.Size = new System.Drawing.Size(191, 46);
        this.textBox1.TabIndex = 36;
        //
        // DeliveryDelete
        //
        this.AutoScaleDimensions = new System.Drawing.SizeF(8F, 16F);
        this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
        this.ClientSize = new System.Drawing.Size(802, 553);
        this.Controls.Add(this.label6);
        this.Controls.Add(this.textBox7);
        this.Controls.Add(this.label4);
        this.Controls.Add(this.label5);
        this.Controls.Add(this.textBox5);
        this.Controls.Add(this.textBox6);
        this.Controls.Add(this.textBox4);
        this.Controls.Add(this.button1);
        this.Controls.Add(this.label3);
        this.Controls.Add(this.label2);
        this.Controls.Add(this.label1);
        this.Controls.Add(this.textBox3);
        this.Controls.Add(this.textBox2);
        this.Controls.Add(this.textBox1);
        this.Name = "DeliveryDelete";
        this.Text = "DeliveryDelete";
        this.ResumeLayout(false);
        this.PerformLayout();

    }

    #endregion

    private System.Windows.Forms.Label label6;
    private System.Windows.Forms.TextBox textBox7;
    private System.Windows.Forms.Label label4;
    private System.Windows.Forms.Label label5;
    private System.Windows.Forms.TextBox textBox5;
    private System.Windows.Forms.TextBox textBox6;
    private System.Windows.Forms.TextBox textBox4;
    private System.Windows.Forms.Button button1;
    private System.Windows.Forms.Label label3;
    private System.Windows.Forms.Label label2;
    private System.Windows.Forms.Label label1;
    private System.Windows.Forms.TextBox textBox3;
    private System.Windows.Forms.TextBox textBox2;
    private System.Windows.Forms.TextBox textBox1;
}
}

```

Листинг Б.17 – Код класса DeliveryShow (форма)

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

using Npgsql;

```



```

namespace MilkFactory
{
    public partial class DeliveryShow : Form
    {
        private string Username_;
        private string Password_;
        public DeliveryShow(string Username, string Password)
        {
            Username_ = Username;
            Password_ = Password;
            InitializeComponent();
        }

        private void pictureBox1_Click(object sender, EventArgs e)
        {
            try
            {
                string connectionString = "Host=localhost;Username=" + Username_
+ ";Password=" + Password_ + ";Database=MilkFactory";

                var con = new NpgsqlConnection(connectionString);
                con.Open();

                string sql = "SELECT \"DeliveryID\", \"VendorID\", \"ProductID\",
\"ProductCount\", \"DateTime\" " +
                    "FROM mf.\"Delivery\"; ";

                var cmd = new NpgsqlCommand(sql, con);
                NpgsqlDataReader dr = cmd.ExecuteReader();

                int i = 1;
                while (dr.Read())
                {
                    string text = dr[0].ToString() + "\t" + dr[1].ToString() +
"\t" +
                                dr[2].ToString() + "\t" + dr[3].ToString() +
"\t" + dr[4].ToString();
                    Brush brush = new SolidBrush(Color.Black);
                    Font font = new System.Drawing.Font("Times New Roman", 9,
FontStyle.Bold);
                    Graphics graphics = pictureBox1.CreateGraphics();
                    graphics.TextRenderingHint = System.Drawing.Text.TextRender-
ingHint.AntiAlias;
                    graphics.DrawString(text, font, brush, 50, 20 * i);
                    i++;
                }

                con.Close();
            }
            catch (Exception ex)
            {
                Console.WriteLine(ex.Message);
            }
        }
    }
}

```