

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 3](#_Toc41730847)

[1 Актуальные проблемы базы данных 4](#_Toc41730848)

[1.1 Классификация баз данных 4](#_Toc41730849)

[1.2 Рынок программного обеспечения СУБД 5](#_Toc41730850)

[2 Проектирование БД «Производство сельскохозяйственной продукции» 7](#_Toc41730851)

[2.1 Постановка задачи 7](#_Toc41730852)

[2.2 Нормализация отношения между объектами БД «Производство сельскохозяйственной продукции» 8](#_Toc41730853)

[2.3 Построение инфологической модели БД «Производство сельскохозяйственной продукции» 11](#_Toc41730854)

[3 Разработка БД «Производство сельскохозяйственной продукции» 18](#_Toc41730855)

[3.1 Разработка схем и таблиц БД 18](#_Toc41730856)

[3.2 Запросы системы 23](#_Toc41730857)

[3.3 Интерфейс пользователя БД «Производство сельскохозяйственной продукции» 30](#_Toc41730858)

[ЗАКЛЮЧЕНИЕ 36](#_Toc41730859)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ И ЛИТЕРАТУРЫ 37](#_Toc41730860)

**ВВЕДЕНИЕ**

Всем известно, что эффективно оперировать большими потоками данных в современном мире механическими способами крайне затруднительно. Большие объемы информации нужно где-то хранить, чтобы легко в любое время к ним можно было обратиться. Быстрое манипулирование информацией позволяет оперативно решить текущую задачу, которая требует немедленного реагирования.

Для правильного принятия решения при работе с большим потоком данных необходимы условия, которые заложены в современные информационные системы.

Объектом исследования в ходе курсовой работы является сельскохозяйственное предприятие. Целью данной работы является автоматизация процесса контроля поставок и продажи сельскохозяйственной продукции с использованием продукта Microsoft Access, входящего в пакет Microsoft Office.

Задачи для достижения поставленной цели:

* Описать предметную область;
* составить инфологическую модель;
* разработать таблицы и схемы данных;
* создать запросы;
* создать удобный интерфейс пользователя;
* создать основную кнопочную форму.

Информационной базой при написании курсовой работы, а также при разработке и создании базы данных послужили такие материалы, как лекционный материал профильной дисциплины «Базы данных», научные публикации и интернет-ресурсы.

Были применены следующие методы исследования: анализ, синтез, описание, измерение и обобщение.

1. **Актуальные проблемы базы данных**
	1. **Классификация баз данных**

Классификация баз и банков данных может быть произведена по разным признакам (и относящихся к разным компонентам и сторонам функционирования БД), среди которых можно выделить, например, следующие [3]:

1. По форме воображаемой информации отмечают:

* Фактографические;
* документальные;
* мультимедийные;
* цифровые;
* символьные и иные.

2. По типу хранимой (не мультимедийной) информации отмечают:

* Фактографические;
* документальные;
* словарные БД.

Словарные базы – это классификаторы, кодификаторы, словари текстов, тезаурусы, рубрикаторы и т.д., которые как правило применяются в качестве справочных вместе с документальными или же фактографическими БД.

Документальные базы разделяются по уровню представления информации – полнотекстовые и библиографическое-реферативные, отражающие на адресном и содержательном уровне изначальный документ.

3. По типу применяемой модели данных выделяют 3 традиционных класса БД:

* Иерархические;
* сетевые;
* реляционные.

Становление технологий обработки данных привело к выходу в свет пост реляционных, объектно-ориентированных, многомерных БД, которые в что или же другой степени отвечают 3 упомянутым традиционным моделям.

4. По топологии сбережения данных различают:

* Локальные;
* распределенные БД.
	1. **Рынок программного обеспечения СУБД**

Система управления базами данных (СУБД) — это система программного обеспечения, которая позволяет обрабатывать обращения к базе данных, поступающие от прикладных программ конечных пользователей [3].

Операции работы с данными, которые могут выполняться любым СУБД:

* Добавление в таблицу одной или более записей;
* удаление из таблицы одной или более записей;
* обновление или корректировка некоторый полей;
* нахождение одной или нескольких записей, которые соответствуют определенному условию или заданы определенным условием.

Различают СУБД по таким классификациям:

1. Модели данных:

* Иерархические;
* сетевые;
* реляционные.

2. Степени разделенности:

* Локальные;
* распределенные.

3. Способу доступа к БД

* Файл-серверные;
* клиент-серверные;
* встраиваемые.

Сервер базы данных представляет собой СУБД, параллельно обрабатывающую запросы, поступившие со всех рабочих станций. Как правило, клиент и сервер территориально отделены друг от друга, и в этом случае они образуют систему распределенной обработки данных.

1. Проектирование БД «Производство сельскохозяйственной продукции»
2. Постановка задачи

Актуальность разработки БД «Производство сельскохозяйственной продукции» заключается в необходимости хранить и учитывать сведения о поставляемой продукции. Также производство сельскохозяйственной продукции нуждается в своевременных отчетах, содержащих сведения о приходе товара, расходе, информации о поставщиках, а также свежесть продукции.

Прoграмма, рабoтающая с БД, пoзвoляeт пoказывать инфoрмацию o тoварах, o пoставках, о клиeнтах, о реализации продукции. Так жe предусмотрена вoзмoжнoсть сфoрмирoвать oтчeты пo различным катeгoриям.

На этом основании была разработана БД «Производство сельскохозяйственной продукции».

В ходе изучения работы сельскохозяйственного предприятия стало известно, что оно осуществляет большое количество функций. Для ведения учета данных о сотрудниках требуются следующие данные: Фамилия, имя, отчество, должность и заработная плата. Необходимо иметь список складов, а также вести учет поставок продукции на склад и ее реализации. Для ведения учета клиентов требуются: имя, фамилия, телефон, адрес.

Поэтому, можно поставить следующие задачи для разработки БД «Производство сельскохозяйственного продукции»:

* Запись информации об организации поставщика;
* запись информации о производимой продукции организации;
* поиск записей по направлению работы организации;
* автоматизация процесса контроля поставок и продажи сельскохозяйственной продукции;
* создание отчетности по категориям и реализуемым продуктам.
1. **Нормализация отношения между объектами БД «Производство сельскохозяйственной продукции»** Для того, чтобы выполнить нормализацию, структура базы данных последовательно приводится к различным формам. В процессе нормализации каждая последующая форма относится к предыдущей форме. Например, чтобы схема базы данных соответствовала второй нормальной форме (2НФ), она должна также соответствовать и первой нормальной форме (1НФ). Чтобы схема соответствовала третьей нормальной форме (3НФ), она должна соответствовать и второй нормальной форме (2НФ), и т.д. Например, существует сущность «Поставщики» (Таблица 2.2.1.).

Таблица 2.2.1. –– Сущность «Поставщики»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Код поставщика | ФИО | Адрес | Телефон | Счет |
| 1 | Белочкин Артем Николаевич, Дихтярёв Александр Николаевич | Гоголя 5 | 7(978)7174332; 7(978)7172332; | 34234; |
| 2 | Газданов Юрий Владимирович | П. Метальникова 2 | 7(978)7176454; | 23452; |
| 3 | Мартыненко Артем Владимирович, Илюшин Владимир Владимирович | Симонок 21 | 7(978)7174632; 7(978)7172732;  | 23456; |

Нарушение нормализации первой нормальной формы (1НФ) происходит в «ФИО» и «Телефон» т.к. в одной ячейке содержится список из 2 элементов: ФИО поставщиков и их мобильных номеров т.е. они не являются атомарными. Преобразуем сущность к первой нормальной форме (1НФ) как показано в таблице 2.2.2.

Таблица 2.2.2. –– Сущность «Поставщики» в первой нормальной форме

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Код поставщика | ФИО | Адрес | Телефон | Счет |
| 1 | Белочкин Артем Николаевич | Гоголя 5 | 7(978)7174332; | 34234; |
| 2 | Дихтярёв Александр Николаевич | П. Метальникова 2 | 7(978)7172332; | 23455; |
| 3 | Газданов Юрий Николаевич | Симонок 21 | 7(978)7176454; | 23452; |
| 4 | Мартыненко Артем Владимирович | Симонок 31 | 7(978)7174632; | 23456; |
| 5 | Илюшин Владимир Владимирович | Лазарева 5 | 7(978)7172732; | 34325; |

Отношение находится во второй нормальной форме (2НФ), если оно находится в первой нормальной форме (1НФ) и каждый не ключевой атрибут неприводимо зависит от Первичного Ключа (ПК). Неприводимость означает, что в составе потенциального ключа отсутствует меньшее подмножество атрибутов, от которого можно также вывести данную функциональную зависимость. Исправляется это путем декомпозиции на два отношения, в которых не ключевые атрибуты зависят от Первичного Ключа (ПК).

В таблице «Продукт» (таблица 2.2.3.) четко прослеживается приведение к 2НФ, т.к. существуют функциональные зависимости «Код продукта → Наименование → Единица измерения → Стоимость за единицу измерения».

Таблица 2.2.3. ­–– Сущность «Продукт» во второй нормальной форме

|  |  |  |  |
| --- | --- | --- | --- |
| Код продукта | Наименование | Единица измерения | Стоимость за единицу измерения |
| 1 | Мёд натуральный | шт. | 75,00 р. |
| 2 | Молоко | шт. | 60,00 р. |
| 3 | Тыква | шт. | 115,0 р. |
| 4 | Пшено | шт. | 40,00 р. |
| 5 | Семена овса | шт. | 55,00 р. |

Отношение находится в третьей нормальной форме (3НФ), когда находится во второй нормальной форме (2НФ) и каждый не ключевой атрибут не транзитивно зависит от первичного ключа. Рассмотрим сущность «Склад», представленную в таблице 2.2.4.

|  |  |  |  |
| --- | --- | --- | --- |
| Код склада | Наименование | Объем склада | Адрес склада |
| 1 | Склад Торговый | $$2500 м^{2}$$ | Гоголя 5 |
| 2 | Склад Производственный | $$5000 м^{2}$$ | Маркса 45 |

Таблица 2.2.4. –– Сущность «Склад»

Таблица находится во второй нормальной форме (2НФ), но не в третьей (3НФ). В отношении существуют следующие функциональные зависимости:

Код склада ­→ Адрес склада, Адрес склада → Наименование, Наименование → Объем склада. В результате разделения исходного отношения получаются два отношения, находящиеся в третьей нормальной форме (3НФ) (таблица 2.2.5.).

Таблица 2.2.5. –– Третья нормальная форма (3НФ) сущности «Склад», достигаемая путем декомпозиции отношений

|  |  |
| --- | --- |
| Код склада | Адрес склада |
| 1 | Гоголя 5 |
| 2 | Маркса 45 |

|  |  |
| --- | --- |
| Наименование | Объем склада |
| Склад Торговый | $$2500 м^{2}$$ |
| Склад Производственный | $$5000 м^{2}$$ |

Таким образом, мы устранили транзитивную зависимость так как провели разбиение последнего отношения с удалением из него транзитивно-зависимого атрибута и помещением его и копии того атрибута, от которого он зависит, во вновь созданное отношение.

1. **Построение инфологической модели БД «Производство сельскохозяйственной продукции»**

Перед разработкой инфологической модели реляционной БД рассмотрим, из каких информационных объектов должна состоять эта база данных. Можно выделить восемь объектов, которые не будут обладать избыточностью, - «Клиент», «Поставщики», «Приход товара», «Продукт», «Расход товара», «Реализация», «Склад», «Сотрудник».

Представим состав реквизитов этих объектов в виде «название объекта (перечень реквизитов)»:

* «Клиент» (Код клиента, ФИО клиента, Адрес клиента, Телефон);
* «Поставщики» (Код поставщика, ФИО, Адрес, Телефон, Счет);
* «Приход товара» (Номер накладной, Код сотрудника, Наименование организации, Код продукта, Код поставщика, Кол-во закупленного, Розничная цена, Дата поступления);
* «Продукт» (Код продукта, Наименование, Единица измерения, Стоимость за единицу измерения);
* «Расход товара» (Номер накладной, Наименование организации, Код продукта, Код поставщика, Остаток на складе);
* «Реализация» (Код реализации, Код склада, Код продукта, Код клиента, Табельный номер, Общее количество, Общая стоимость, Дата реализации);
* «Склад» (Код склада, Наименование, Объем склада, Адрес склада);
* «Сотрудник» (Код сотрудника, ФИО сотрудника, Должность, Заработная плата).

Рассмотрим, как следует связать объекты разрабатываемой базы данных для обеспечения целостности потока данных:

Связи между объектами «Склад» и «Реализация» представлены на рисунке 2.3.1. Связь удовлетворяет правилу двух сущностей, поэтому связь будет один ко многим (1:М). Связь «один ко многим» означает, что к каждому объекту первого вида может соответствовать более одного объекта второго вида, но каждому второму объекту второго вида соответствует не более одного объекта первого вида. Один склад может числиться в документах о реализации продукции.



Рисунок 2.3.1. –– Связи между объектами «Склад» и «Реализация»

Связи между объектами «Клиент» и «Реализация» представлены на рисунке 2.3.2. Один отдельно взятый клиент может получить реализацию, поэтому связь будет один ко многим (1:М).



Рисунок 2.3.2. –– Связи между объектами «Клиент» и «Реализация»

Связи между объектами «Продукт» и «Реализация» представлены на рисунке 2.3.3. Один продукт может проходить реализацию, поэтому связь один ко многим (1:М).



Рисунок 2.3.3. –– Связи между объектами «Продукт» и «Реализация»

Связи между объектами «Продукт» и «Приход товара» представлены на рисунке 2.3.4. Один продукт может быть в списке о поставке или приходе товара, поэтому связь один ко многим (1:М).



Рисунок 2.3.4. –– Связи между объектами «Продукт» и «Приход товара»

Связи между объектами «Продукт» и «Расход товара» представлены на рисунке 2.3.5. Связь один ко многим (1:М).



Рисунок 2.3.5. –– Связи между объектами «Продукт» и «Расход товара»

Связи между объектами «Приход товара» и «Расход товара» представлены на рисунке 2.3.6. Связью этих объектов будет связь один к одному, так как существует один экземпляр одного объекта, который связан с единственным экземпляром другого и наоборот (1:1).



Рисунок 2.3.6. –– Связи между объектами «Приход товара» и «Расход товара»

Связи между объектами «Сотрудник» и «Приход товара» представлено на рисунке 2.3.7. Один сотрудник фиксирует поступление или приход товара, поэтому связь один ко многим (1:М).



Рисунок 2.3.7. –– Связи между объектами «Сотрудник» и «Приход товара»

Связи между объектами «Поставщики» и «Приход товара» представлено на рисунке 2.3.8. Один поставщик может поставлять продукцию и осуществлять приход товара на предприятие, поэтому связь один ко многим (1:М).



Рисунок 2.3.8. –– Связи между объектами «Поставщики» и «Приход товара»

Связи между объектами «Поставщики» и «Расход товара» представлена на рисунке 2.3.9. Связь один ко многим (1:М).



Рисунок 2.3.9. –– Связи между объектами «Поставщики» и «Расход товара»

В результате получаем информационно-логическую модель базы данных, приведенную на рисунке 2.3.10:

Рисунок 2.3.10. –– Информационно-логическая модель БД «Производство сельскохозяйственной продукции»

В реляционной базе данных в качестве объектов рассматриваются отношения, которые можно представить в виде таблиц. Таблицы между собой связываются посредствам общих полей, т.е. одинаковых по форматам и, как правило, по названию, имеющихся в обеих таблицах.

Рассмотрим, какие общие поля надо ввести в таблицы для обеспечения связанности данных:

* В таблицах «Склад» и «Реализация» – «Код склада» и «Код склада»;
* в таблицах «Клиент» и «Реализация» – «Код клиента» и «Код клиента»;
* в таблицах «Продукт» и «Реализация» – «Код продукта» и «Код продукта»;
* в таблицах «Продукт» и «Приход товара» – «Код продукта» и «Код продукта»;
* в таблицах «Продукт» и «Расход товара» – «Код продукта» и «Код продукта»;
* в таблицах «Приход товара» и «Расход товара» – «Номер накладной» и «Номер накладной»;
* в таблицах «Сотрудник» и «Приход товара» – «Код сотрудника» и «Код сотрудника»;
* в таблицах «Поставщики» и «Приход товара» – «Код поставщика» и «Код поставщика»;
* в таблицах «Поставщики» и «Расход товара» – «Код поставщика» и «Код поставщика».

В соответствии с введенными полями, обеспечивающими связь данных, логическая модель базы данных будет выглядеть следующим образом (где знаком «\*» обозначено ключевое поле) (рисунок 2.3.11.):



Рисунок 2.3.11. –– Логическая модель БД «Производство сельскохозяйственной продукции»

1. Разработка БД «Производство сельскохозяйственной продукции»
2. Разработка схем и таблиц БД

Для создания реляционной БД «Производство сельскохозяйственной продукции» понадобятся следующие таблицы («Клиент», «Поставщики», «Приход товара», «Продукт», «Расход товара», «Реализация», «Склад», «Сотрудник») которые необходимо связать для обеспечения непротиворечивости данных.

Для создания таблицы в СУБД MicrosoftAccess используется команда CREATE TABLE. C ее помощью описывается структура таблицы, включая названия полей, их тип данных, описание и значения некоторых свойств в соответствии с разработанной логической моделью БД «Производство сельскохозяйственной продукции» [5].

Для создания таблицы «Клиент» напишем следующие команды (поле «КодКлиента» задано в качестве ключевого):

CREATE TABLE Клиент

(КодКлиента CHAR (20) PRIMARY KEY,

ФИО клиента CHAR (150),

Адрес клиента CHAR (150),

Телефон CHAR (150));

 После выполнения изложенных выше команд получаем объект, структура которого представлена в таблице 3.1.1.

Таблица 3.1.1. –– Структура созданной таблицы «Клиент»

|  |  |  |
| --- | --- | --- |
| Имя поля | Тип данных | Размер поля |
| КодКлиента | Числовой | 20 |
| ФИО клиента | Текст | 150 |
| Адрес клиента | Текст | 150 |
| Телефон | Текст | 150 |

Для создания таблицы «Поставщики» напишем следующие команды (поле «КодПоставщика» задано в качестве ключевого):

CREATE TABLE Поставщики

(КодПоставщика CHAR (20) PRIMARY KEY,

ФИО CHAR(150),

Адрес CHAR(150),

Телефон CHAR(150),

Счет CHAR(150));

После выполнения изложенных выше команд получаем объект, структура которого представлена в таблице 3.1.2.

Таблица 3.1.2. –– Структура созданной таблицы «Поставщики»

|  |  |  |
| --- | --- | --- |
| Имя поля | Тип данных | Размер поля |
| КодПоставщика | Числовой | 20 |
| ФИО | Текст | 150 |
| Адрес | Текст | 150 |
| Телефон | Текст | 150 |
| Счет | Текст | 150 |

Для создания таблицы «Приход товара» напишем следующие команды (поле «Номернакладной» задано в качестве ключевого):

CREATE TABLE Приход товара

(Номернакладной CHAR (20) PRIMARY KEY,

Наименование организации CHAR (150),

Код продукта CHAR(20),

Код поставщика CHAR(20),

Кол-во закупленного CHAR(20),

Розничная цена MONEY,

Дата поступления DATE);

После выполнения изложенных выше команд получаем объект, структура которого представлена в таблице 3.1.3.

Таблица 3.1.3. –– Структура созданной таблицы «Приход товара»

|  |  |  |
| --- | --- | --- |
| Имя поля | Тип данных | Размер поля |
| Номернакладной | Числовой | 20 |
| Наименование организации | Текст | 150 |
| Код продукта | Числовой | 20 |
| Код поставщика | Числовой | 20 |
| Кол-во закупленного | Числовой | 20 |
| Розничная цена | Денежный | – |
| Дата поступления | Дата и время | – |

Для создания таблицы «Продукт» напишем следующие команды (поле «Кодпродукта» задано в качестве ключевого):

CREATE TABLE Продукт

(Кодпродукта CHAR (20) PRIMARY KEY,

Наименование CHAR (150),

Единица измерения CHAR (150),

Стоимость за единицу измерения MONEY);

После выполнения изложенных выше команд получаем объект, структура которого представлена в таблице 3.1.4.

Таблица 3.1.4. –– Структура созданной таблица «Продукт»

|  |  |  |
| --- | --- | --- |
| Имя поля | Тип данных | Размер поля |
| Кодпродукта | Числовой | 20 |
| Наименование | Текст | 150 |
| Единица измерения | Текст | 150 |
| Стоимость за единицу измерения | Денежный | – |

Для создания таблицы «Расход товара» напишем следующие команды (поле «Номернакладной» задано в качестве ключевого):

CREATE TABLE Приход товара

(Номернакладной CHAR (20) PRIMARY KEY,

Наименование организации CHAR (150),

Код продукта CHAR(20),

Код поставщика CHAR(20),

Остаток на складе CHAR (20));

После выполнения изложенных выше команд получаем объект, структура которого представлена в таблице 3.1.5.

Таблица 3.1.5. –– Структура созданной таблицы «Расход товара»

|  |  |  |
| --- | --- | --- |
| Имя поля | Тип данных | Размер поля |
| Номернакладной | Числовой | 20 |
| Наименование организации | Текст | 150 |
| Код продукта | Числовой | 20 |
| Код поставщика | Числовой | 20 |
| Остаток на складе | Числовой | 20 |

Для создания таблицы «Реализация» напишем следующие команды (поле «Кодреализации» задано в качестве ключевого):

CREATE TABLE Реализация

(Кодреализации CHAR (20) PRIMARY KEY,

Код склада CHAR (20),

Код продукта CHAR (20),

Код клиента CHAR (20),

Табельный номер CHAR (150),

Общее количество CHAR (20),

Общая стоимость MONEY,

Дата реализации DATA);

После выполнения изложенных выше команд получаем объект, структура которого представлена в таблице 3.1.6.

Таблица 3.1.6. –– Структура выполненной работы «Реализация»

|  |  |  |
| --- | --- | --- |
| Имя поля | Тип данных | Размер поля |
| Кодреализации | Числовой | 20 |
| Код склада | Числовой | 20 |
| Код продукта | Числовой | 20 |
| Код клиента | Числовой | 20 |
| Табельный номер | Текст | 150 |
| Общее количество | Числовой | 20 |
| Общая стоимость | Денежный | – |
| Дата реализации | Дата и время | – |

Для создания таблицы «Склад» напишем следующие команды (поле «Кодсклада» задано в качестве ключевого):

CREATE TABLE Склад

(Кодсклада CHAR (20) PRIMARY KEY,

Наименование CHAR (150),

Объем склада CHAR (150),

Адрес склада CHAR (150));

После выполнения изложенных выше команд получаем объект, структура которого представлена в таблице 3.1.7.

Таблица 3.1.7. –– Структура созданной таблицы «Склад»

|  |  |  |
| --- | --- | --- |
| Имя поля | Тип данных | Размер поля |
| Кодсклада | Числовой | 20 |
| Наименование | Текст | 150 |
| Объем склада | Текст | 150 |
| Адрес склада | Текст | 150 |

Для создания таблицы «Сотрудник» напишем следующие команды (поле «Кодсотрудника» задано в качестве ключевого):

CREATE TABLE Сотрудник

(Кодсотрудника CHAR (20) PRIMARY KEY,

ФИО сотрудника CHAR (150),

Должность CHAR (150),

Заработная плата MONEY);

Таблица 3.1.8. –– Структура созданной таблицы «Сотрудник»

|  |  |  |
| --- | --- | --- |
| Имя поля | Тип данных | Размер поля |
| Кодсотрудника | Числовой | 20 |
| ФИО сотрудника | Текст | 150 |
| Должность | Текст | 150 |
| Заработная плата | Денежный | – |

Отсюда можно сделать вывод, что нормализация*—*процесс уменьшения избыточности информации в таблицах реляционной БД и, как следствие, построения оптимальной структуры таблиц и связей.

1. **Запросы системы**

Для того чтобы база данных «Производство сельскохозяйственной продукции» имела возможность выполнить поиск информации по заранее известному критерию, необходимо создать соответствующие запросы.

Для данной системы было создано 11 запросов:

1. Внутренний запрос по одному полю

Данный запрос позволяет нам отобразить дату реализации со складов в которых она состоялась (рисунок 3.2.1). SQL-инструкции этого запроса представлены ниже:

SELECT Реализация.[Дата реализации], Реализация.[Табельный номер], Склад.Наименование

FROM Склад INNER JOIN Реализация ON Склад.[Код склада] = Реализация.[Код склада]

ORDER BY Реализация.[Дата реализации];



Рисунок 3.2.1. –– Результат выполнения запроса

1. Косвенно связанные таблицы

Данный запрос предоставляет информацию о клиентах, которые купили продукт «Икра» (рисунок 3.2.2). SQL-инструкции этого запроса представлены ниже:

SELECT Клиент.[Код клиента], Клиент.[ФИО клиента], Клиент.[Адрес клиента], Клиент.Телефон, Продукт.Наименование

FROM Продукт INNER JOIN (Клиент INNER JOIN Реализация ON Клиент.[Код клиента] = Реализация.[Код клиента]) ON Продукт.[Код продукта] = Реализация.[Код продукта]

WHERE (((Продукт.Наименование)="Икра"))

ORDER BY Клиент.[Код клиента];

Рисунок 3.2.2. –– Результат выполнения запроса

1. Связь более чем по одному полю

Данный запрос отображает информацию о продуктах, которые были реализованы в день поступления (рисунок 3.2.3). SQL-инструкции этого запроса представлены ниже:

SELECT Продукт.[Код продукта], Продукт.[Наименование], t1.[Дата поступления]

FROM

(

SELECT [Приход товара].[Код продукта], [Приход товара].[Дата поступления]

FROM [Приход товара]

INNER JOIN Реализация ON [Приход товара].[Код продукта] = [Реализация].[Код продукта] AND [Приход товара].[Дата поступления] = [Реализация].[Дата реализации]

) AS t1

INNER JOIN Продукт ON [t1].[Код продукта] = Продукт.[Код продукта]



Рисунок 3.2.3. –– Результат выполнения запроса

1. А) Левое соединение

Данный запрос позволяет получить информацию о всех приходах товара и организациях – поставщиках (рисунок 3.2.4). SQL-инструкции этого запрос представлены ниже:

SELECT Продукт.Наименование, Продукт.[Стоимость за единицу измерения], [Приход товара].[Код сотрудника], [Приход товара].[ Наименование организации]

FROM Продукт LEFT JOIN [Приход товара] ON Продукт.[Код продукта] = [Приход товара].[Код продукта];



Рисунок 3.2.4. –– Результат выполнения запроса

Б) Правое соединение

Данный запрос отображает информацию о складах, где была выполнена хоть какая-то реализация (рисунок 3.2.5). SQL-инструкции этого запрос представлены ниже:

SELECT distinct Склад.[Код склада], Склад.Наименование

FROM Склад RIGHT JOIN Реализация ON Склад.[Код склада] = Реализация.[Код склада];



Рисунок 3.2.5. –– Результат выполнения запроса

1. Запрос с подзапросом

Данный запрос отображает информацию о расходах товара по поставщику «Белочкин Артем Николаевич» (рисунок 3.2.6). SQL-инструкции этого запрос представлены ниже:

SELECT [Расход товара].\*

FROM [Расход товара]

WHERE [Расход товара].[Код поставщика] =

 (

SELECT Поставщики.[Код поставщика]

FROM Поставщики

WHERE ФИО = 'Белочкин Артем Николаевич'

 )



Рисунок 3.2.6 –– Результат выполнения запроса

1. Соединение с отношением

Данный запрос отображает информацию о продуктах, которые были реализованы раньше даты поступления (рисунок 3.2.7.). SQL-инструкции этого запроса представлены ниже:

SELECT [Приход товара].[Наименование организации], Реализация.[Код реализации], Реализация.[Код склада]

FROM [Приход товара]

INNER JOIN Реализация ON [Приход товара].[Код продукта] = Реализация.[Код продукта] AND [Приход товара].[Дата поступления] > Реализация.[Дата реализации]



Рисунок 3.2.7. –– Результат выполнения запроса

1. Функция агрегирования

Данный запрос отображает количество закупленных товаров с 05.05.2020 включительно (рисунок 3.2.8). SQL-инструкции этого запроса представлены ниже:

SELECT Sum([Приход товара].[Кол-во закупленного]) AS [Всего закупленно]

FROM [Приход товара]

WHERE [Дата поступления] >= #5/5/2020#



Рисунок 3.2.8. –– Результат выполнения запроса

1. Перекрестный запрос

Данный запрос предоставляет информацию по реализации товаров клиентам со склада (рисунок 3.2.9). SQL-инструкции этого запроса представлены ниже:

TRANSFORM Sum(Реализация.[Общее количество]) AS [Sum-Общее количество]

SELECT Клиент.[Код клиента], Клиент.[ФИО клиента]

FROM Клиент INNER JOIN (Склад INNER JOIN Реализация ON Склад.[Код склада] = Реализация.[Код склада]) ON Клиент.[Код клиента] = Реализация.[Код клиента]

GROUP BY Клиент.[Код клиента], Клиент.[ФИО клиента]

PIVOT Склад.Наименование;

Рисунок 3.2.9. –– Результат выполнения запроса

1. Объединение

Данный запрос отображает информацию о всех организациях, у которых закупали / которым поставляли продукты (рисунок 3.2.10.). SQL-инструкции этого запроса представлены ниже:

SELECT [Номер накладной], [Наименование организации], 'Расход' AS Тип

FROM [Расход товара]

UNION ALL

SELECT [Номер накладной], [Наименование организации], 'Приход' AS Тип

FROM [Приход товара]



Рисунок 3.2.10 –– Результат выполнения запроса

1. Вычисляемое поле

Данный запрос позволяет отобразить общую сумму прихода товаров по товарам (рисунок 3.2.11.). SQL-инструкции этого запроса представлены ниже:

SELECT Продукт.[Код продукта], Продукт.Наименование, [Приход товара].[Кол-во закупленного], [Приход товара].[Розничная цена], [Приход товара]![Кол-во закупленного]\*[Приход товара]![Розничная цена] AS 'Общая сумма' FROM Продукт INNER JOIN [Приход товара] ON Продукт.[Код продукта] = [Приход товара].[Код продукта];



Рисунок 3.2.11. –– Результат выполнения запроса

1. **Интерфейс пользователя БД «Производство сельскохозяйственной продукции»**

Ввиду большого информационного потока для удобства пользователя создается главная кнопочная форма, которая аккумулирует в себя все формы для заполнения таблиц, запросы системы и отчеты. С помощью нее пользователь видит всю структуру базы данных, четко разграничивает компетенции тех или иных форм или запросов. Созданная главная кнопочная форма БД «Производство сельскохозяйственной продукции» представлена на рисунке 3.3.1.



Рисунок 3.3.1. –– Главная кнопочная форма БД «Производство сельскохозяйственной продукции»

Для гибкой манипуляции данными на всех формах созданы кнопки переключения между записями, поиска информации, сохранения, дублирования и восстановления записи, а также выход из формы, предполагающий закрытие текущей формы и переход на главную кнопочную форму. Также были созданы формы, предназначенные для удобства заполнения информации. Перечислим созданные формы:

На рисунке 3.3.2. представлена форма для заполнения таблицы «Клиент». 

Рисунок 3.3.2. –– Форма для ввода информации о клиентах

В данной форме реализован просмотр информации. Пользуясь интерфейсными кнопками, можно пролистывать информацию, можно добавить новую запись, можно сохранить либо удалить запись.

На рисунке 3.3.3. представлена форма для заполнения таблицы «Поставщик».



Рисунок 3.3.3. –– Форма для ввода информации о поставщиках

Данная форма отображает уже имеющуюся информацию о поставщиках и позволяет добавить новую запись в таблицу «Поставщики».

На рисунке 3.3.4. представлена форма для заполнения таблицы «Приход товара».



Рисунок 3.3.4. –– Форма для ввода информации о приходе товара

Данная форма так же отображает информацию и позволяет создать новую запись в таблице.

На рисунке 3.3.5. представлена форма для заполнения таблицы «Продукт».



Рисунок 3.3.5. –– Форма для ввода информации о продукте

Данная форма так же отображает информацию о продукте, который имеется на складе и будет готов к реализации.

На рисунке 3.3.6. представлена форма для заполнения таблицы «Расход товара».



Рисунок 3.3.6. –– Форма для ввода информации о расходе товара

Данная форма отображает информацию с таблицы «Расход товара. В случае необходимости, данная форма предполагает добавление новой записи.

На рисунке 3.3.7. представлена форма для заполнения таблицы «Реализации».



Рисунок 3.3.7. –– Форма для ввода информации о реализации

Данная форма так же отображает информацию и имеет возможность создать новую запись, сохранить ее либо удалить.

На рисунке 3.3.8. представлена форма для заполнения таблицы «Склад».



Рисунок 3.3.8. –– Форма для ввода информации о складе

Данная форма так же отображает информацию и имеет возможность создать новую запись, сохранить ее либо удалить.

На рисунке 3.3.9. представлена форма для заполнения таблицы «Сотрудник».



Рисунок 3.3.9. –– Форма для ввода информации о сотрудниках

Данная форма отображает информацию с таблицы «Сотрудник» и имеет возможность добавить новую запись либо удалить ее.

На рисунке 3.3.10. приведен отчет, в котором находится информация, собранная из трех таблиц о товарообороте товара.



Рисунок 3.3.10. –– Информация о товарообороте

# **ЗАКЛЮЧЕНИЕ**

Разработанная БД «Производство сельскохозяйственной продукции» позволяет обрабатывать большой поток информации за меньшее количество времени и сокращает время на ручную обработку данных. Обеспечивает удобный ввод данных, поиск по необходимой информации.

Таким образом, можно сделать основные выводы:

* Была создана и описана инфологическая модель базы данных для формирования представления на разрабатываемую базу данных;
* были описаны сущности базы данных и связи между ними для исчерпывающей передачи данных;
* БД «Производство сельскохозяйственной продукции» была приведена к третьей нормальной форме (3НФ) для обеспечения исключения функциональных зависимостей;
* были разработаны и выполнены 11 запросов различных типов для поиска необходимых данных, подведения итогов;
* были разработаны необходимые формы для ввода данных в таблицы;
* были разработаны необходимые отчеты для удобного представления конечной информации;
* была разработана главная кнопочная форма, позволяющими переходить между формами, запросами и отчетами.

# **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ И ЛИТЕРАТУРЫ**

1. SQL — язык реляционных баз данных: Учебное пособие / Кара-Ушанов В.Ю., - 2-е изд., стер. - М.:Флинта, Изд-во Урал. ун-та, 2014.
2. Основы теории надежности информационных систем: Учебное пособие / С.А. Мартишин, В.Л. Симонов, М.В. Храпченко. - М.: ИД ФОРУМ: НИЦ ИНФРА-М, 2013. (Высшее образование).
3. Базы данных и системы управления базами данных: Учебное пособие / Лазицкас Е.А., Загумённикова И.Н., Гилевский П.Г. - Мн.:РИПО, 2016.
4. Черячукин, В. В. Право интеллектуальной собственности на программы для ЭВМ и базы данных в Российской Федерации и зарубежных странах [Электронный ресурс] : учеб. пособие для студентов вузов / В. В. Черячукин; под ред. Н. М. Коршунова. - М. : ЮНИТИ-ДАНА ; Закон и право, 2012.
5. СУБД для программиста. Базы данных изнутри: Практическое пособие / Тарасов С.В. - М.:СОЛОН-Пр., 2015.
6. Алгоритмы категорирования персональных данных для систем автоматизир. проектирования баз данных информац. систем / А.В. Благодаров и др. - М.: Гор. линия-Телеком, 2013.
7. Экономический словарь [Электронный ресурс]// Режим доступа: [https://dic.academic.ru/](https://dic.academic.ru/dic.nsf/econ_dict/19388) (Дата обращения: 20.05.2020).
8. Информатика и проектирование [Электронный ресурс]// Режим доступа: <http://www.informaticspoint.ru/> (Дата обращения: 20.05.2020).
9. Библиофонд - электронная библиотека студента [Электронный ресурс]// Режим доступа: <https://bibliofond.ru/> (Дата обращения: 20.05.2020).
10. COD:NET - все для программиста [Электронный ресурс]// Режим доступа: <http://www.codenet.ru/> (Дата обращения: 20.05.2020).
11. «Хабрахабр» - ресурс для IT-специалистов [Электронный ресурс]// Режим доступа: [https://habrahabr.ru/](https://habrahabr.ru/post/129195/) (Дата обращения: 20.05.2020).
12. TOSTER - Сервис вопросов и ответов для IT-специалистов [Электронный ресурс]// Режим доступа: <https://toster.ru/> (Дата обращения: 20.05.2020).
13. Экономический словарь [Электронный ресурс]// Режим доступа: [https://dic.academic.ru/](https://dic.academic.ru/dic.nsf/econ_dict/19388) (Дата обращения: 20.05.2020).
14. Microsoft [Электронный ресурс]// Режим доступа: <https://www.microsoft.com/ru-ru/> (Дата обращения: 20.05.2020).
15. Образовательный портал ЯКласс [Электронный ресурс]// Режим доступа: [http://www.yaklass.ru/](http://www.yaklass.ru/materiali?chtid=510&mode=cht) (Дата обращения: 20.05.2020).

Процент оригинальности:

