****

**Частное учреждение профессионального образования**

**«Высшая школа предпринимательства»**

**(ЧУПО «ВШП»)**

КУРСОВОЙ ПРОЕКТ

«Создание базы данных по теме склад фруктов»

Выполнил:

студент 3-го курса специальности

09.02.07 «Информационные системы и программирование»  
Смирнова Валерия Олеговна

подпись: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Проверил:

преподаватель дисциплины,  
преподаватель ЧУПО «ВШП»,  
к.ф.н. Ткачев П.С.

оценка: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Оглавление**

[Введение 2](#_Toc11094)

[Анализ предметной области 4](#_Toc23557)

[Проектирование БД 7](#_Toc20258)

[Концептуальная модель (ER-диаграмма): 7](#_Toc22396)

[Логическая модель (Схема данных): 7](#_Toc1484)

[Разработка основных запросов 8](#_Toc8049)

[SQL-запросы: 8](#_Toc13495)

[Транзакция: 9](#_Toc7796)

[Локальные переменные: 9](#_Toc13318)

[Условие: 10](#_Toc19622)

[Хранимая процедура: 10](#_Toc16820)

[Триггер: 11](#_Toc1263)

[Пользовательская функция: 12](#_Toc30332)

[Заключение 13](#_Toc18633)

[Список литературы 14](#_Toc4853)

# Введение

Актуальность: Автоматизация складского учёта является ключевой задачей для любого торгового или логистического предприятия. Эффективное управление остатками фруктов, которые часто являются скоропортящимся товаром, позволяет минимизировать потери, оптимизировать закупки и повысить общую рентабельность. База данных предоставляет для этого надёжный и структурированный инструмент.

Цель проекта: Разработать и реализовать реляционную базу данных для автоматизации учета операций на складе фруктов.

Задачи:

· Провести анализ предметной области.

· Спроектировать концептуальную, логическую и физическую модели БД.

· Реализовать БД в выбранной СУБД (например, MySQL или PostgreSQL).

· Наполнить БД тестовыми данными.

· Реализовать набор типовых запросов для работы с данными.

· (Опционально) Разработать простой графический интерфейс для управления БД.

Объект исследования: Процессы складского учета фруктовой продукции. Предмет исследования:Модель данных, адекватно отражающая предметную область.

Методы исследования: анализ, проектирование, моделирование, программная реализация.

# Анализ предметной области

1. Поступление товара: Оформление поставки от конкретного поставщика. Товар приходит партиями, у каждой партии есть дата поставки, срок

годности и закупочная цена.

1. Хранение товара: Учет текущего количества каждого товара на складе. Важно учитывать остатки по партиям (FIFO/FEFO - первый пришел, первый ушел / первый истекает, первый уходит).
2. Реализация (продажа) товара:

Оформление продажи или списания товара. При продаже количество товара на складе уменьшается.

1. Учёт контрагентов: Ведение списка поставщиков и клиентов.
2. Формирование отчетов: Остатки товаров на складе, движение товара за период, просроченные товары, прибыльность.

Сущности и их атрибуты:

1. Таблица "Фрукты"

ID\_фрукта (первичный ключ)

Название (строка)

Тип (строка)

Цена\_за\_кг (число)

Единица\_измерения (строка)

1. Таблица "Поставщики"

ID\_поставщика (первичный ключ)

Название (строка)

Контактное\_лицо (строка)

Телефон (строка)

Адрес (строка)

1. Таблица "Клиенты"

ID\_клиента (первичный ключ)

Имя (строка)

Контактное\_лицо (строка)

Телефон (строка)

Адрес (строка)

1. Таблица "Поступления"

ID\_поступления (первичный ключ)

ID\_фрукта (внешний ключ на "Фрукты")

ID\_поставщика (внешний ключ на "Поставщики")

Количество\_кг (число)

Дата\_поступления (дата)

1. Таблица "Отгрузки

ID\_отгрузки (первичный ключ)

ID\_фрукта (внешний ключ на "Фрукты")

ID\_клиента (внешний ключ на "Клиенты")

Количество\_кг (число)

Дата\_отгрузки (дата)

1. Таблица "Остатки"

ID\_остатка (первичный ключ)

ID\_фрукта (внешний ключ на "Фрукты")

Количество\_кг (число)

1. Таблица "Склада"

ID\_склада (первичный ключ)

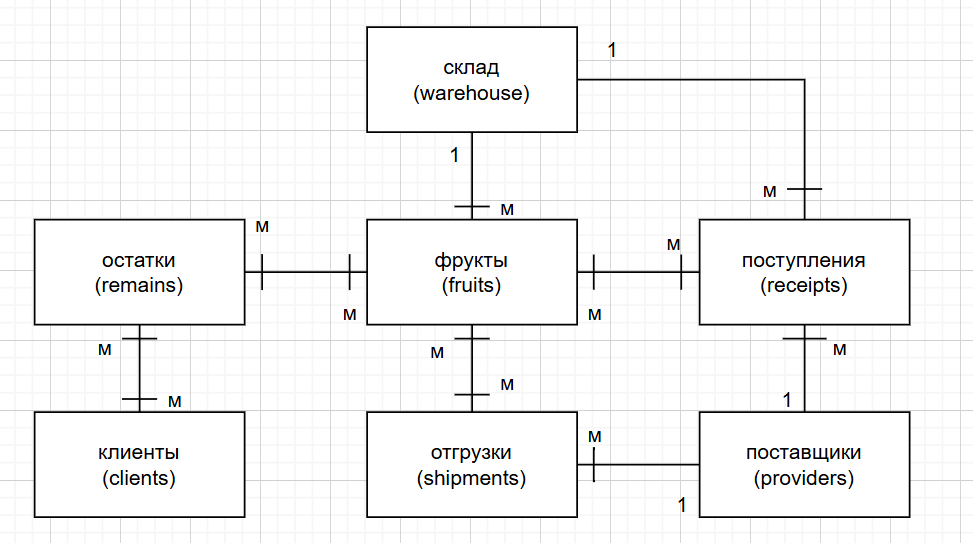
ID\_фрукта (внешний ключ на "Фрукты")

Количество\_кг (число)

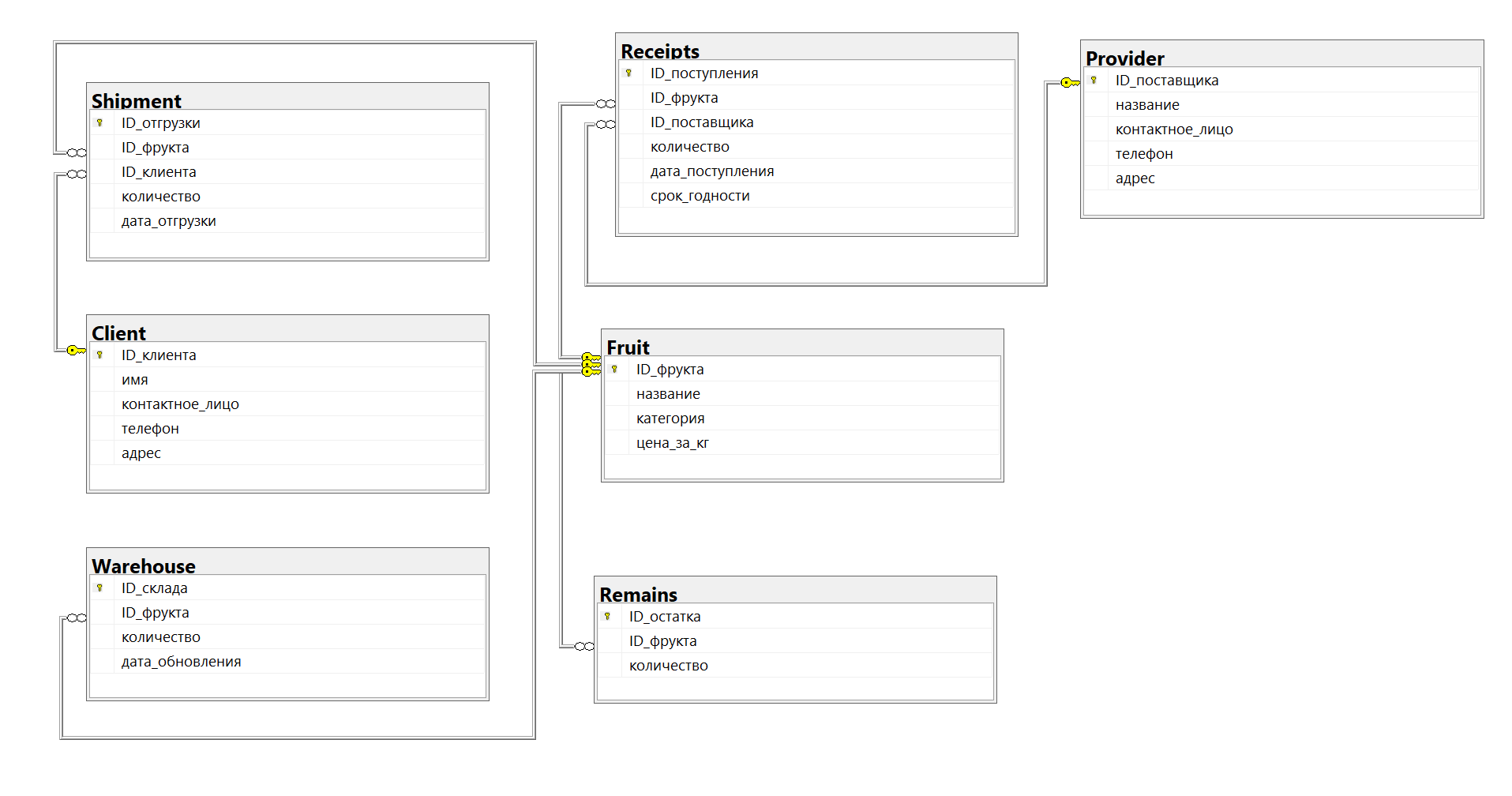
Дата (дата)

# Проектирование БД

## Концептуальная модель (ER-диаграмма):



## Логическая модель (Схема данных):



# Разработка основных запросов

## SQL-запросы:

--1 запрос

SELECT название, категория

FROM Fruit

WHERE категория = 'фрукты'

--2 запрос

INSERT INTO Client (ID\_клиента, имя, контактное\_лицо, телефон, адрес)

VALUES ('5', 'фруктики', 'дмитреева ульяна', '89083946356', 'Москва')

--3 запрос

UPDATE Shipment

SET количество = 80

WHERE ID\_отгрузки = 3

--4 запрос

SELECT Fruit.название,

Remains.количество

FROM Remains

JOIN Fruit ON Remains.ID\_фрукта = Fruit.ID\_фрукта

--5 запрос

SELECT Warehouse.ID\_склада, Fruit.название, Warehouse.количество, Warehouse.дата\_обновления

FROM Warehouse

JOIN Fruit ON Warehouse.ID\_фрукта = Fruit.ID\_фрукта

ORDER BY Warehouse.дата\_обновления DESC;

## Транзакция:

BEGIN TRANSACTION;

-- 1. Вставляем новое поступление

INSERT INTO Receipts (ID\_фрукта, ID\_поставщика, ID\_поступления, количество, дата\_поступления)

VALUES (1, 1, 4, 100, CURRENT\_TIMESTAMP);

-- 2. Обновляем остатки на складе

UPDATE Remains

SET количество = количество + 100

WHERE ID\_фрукта = 1;

-- 3. Обновляем склад

UPDATE Warehouse

SET количество = количество + 100, дата\_обновления = CURRENT\_TIMESTAMP

WHERE ID\_фрукта = 1;

-- Проверка на ошибки

IF @@ERROR <> 0

BEGIN

ROLLBACK TRANSACTION;

PRINT 'Ошибка при выполнении транзакции. Все изменения откатились.';

END

ELSE

BEGIN

COMMIT TRANSACTION;

PRINT 'Транзакция успешно завершена.';

END

## Локальные переменные:

DECLARE @FruitID INT;

DECLARE @ProviderID INT;

DECLARE @ClientID INT;

SET @FruitID = 1;

SET @ProviderID = 2;

SET @ClientID = 3;

SELECT \* FROM Fruit WHERE ID\_фрукта = @FruitID;

SELECT \* FROM Provider WHERE ID\_поставщика = @ProviderID;

SELECT \* FROM Client WHERE ID\_клиента = @ClientID;

## Условие:

SELECT \* FROM Fruit

WHERE цена\_за\_кг > 100;

## Хранимая процедура:

CREATE PROCEDURE RegisterFruitArrival

@ID\_фрукта INT,

@ID\_поставщика INT,

@Количество\_кг INT,

@Дата\_поступления DATE

AS

BEGIN

SET NOCOUNT ON;

-- Добавление новой записи в таблицу Поступления

INSERT INTO Receipts (ID\_фрукта, ID\_поставщика, количество, дата\_поступления)

VALUES (@ID\_фрукта, @ID\_поставщика, @количество\_кг, @дата\_поступления);

-- Обновление остатков

IF EXISTS (SELECT \* FROM Remains WHERE ID\_фрукта = @ID\_фрукта)

BEGIN

UPDATE Remains

SET количество = количество + @Количество\_кг

WHERE ID\_фрукта = @ID\_фрукта;

END

ELSE

BEGIN

INSERT INTO Remains(ID\_фрукта, количество)

VALUES (@ID\_фрукта, @Количество\_кг);

END

END;

## Триггер:

CREATE TRIGGER trg\_ReceiptWarehouse

ON Receipts

AFTER INSERT

AS

BEGIN

SET NOCOUNT ON;

-- Обновляем количество по фруктам, если запись есть

UPDATE s

SET s.количество = s.количество + i.количество,

s.дата\_обновления = GETDATE()

FROM Warehouse s

INNER JOIN inserted i ON s.ID\_фрукта = i.ID\_фрукта;

-- Вставляем новые записи, если фрукта нет в складе

INSERT INTO Warehouse(ID\_фрукта, количество, дата\_обновления)

SELECT i.ID\_фрукта, i.количество, GETDATE()

FROM inserted i

WHERE NOT EXISTS (

SELECT 1 FROM Warehouse s WHERE s.ID\_фрукта = i.ID\_фрукта

);

END

## Пользовательская функция:

CREATE FUNCTION dbo.fn\_GetCurrentNumber

(

@ID\_фрукта INT

)

RETURNS INT

AS

BEGIN

DECLARE @Receipts INT = 0;

DECLARE @Shipment INT = 0;

DECLARE @Remains INT = 0;

SELECT @Receipts = ISNULL(SUM(количество), 0)

FROM Receipts

WHERE ID\_фрукта = @ID\_фрукта;

SELECT @Shipment = ISNULL(SUM(количество), 0)

FROM Shipment

WHERE ID\_фрукта = @ID\_фрукта;

SET @Remains = @Shipment - @Receipts;

RETURN @Remains;

END

# Заключение

В ходе курсовой работы была проанализирована предметная область "Склад фруктов", выделены ключевые сущности и бизнес-процессы. Была разработана полноценная реляционная модель данных, реализованная в СУБД MySQL. Модель позволяет эффективно хранить и обрабатывать информацию о поставщиках, партиях товара, продажах И клиентах.

Разработанная база данных решает поставленные задачи: обеспечивает учет остатков с привязкой к партиям и срокам годности, формирует основные отчёты и предоставляет надёжный инструмент для автоматизации складского учёта.

# Список литературы

1. К. Дж. Дейт. "Введение в системы баз данных".
2. Томас Коннолли, Каролин Бегг. "Базы данных: проектирование, реализация и сопровождение. Теория и практика".
3. С. Д. Кузнецов. "Основы баз данных".
4. Энтони Молинаро. "SQL. Сборник рецептов".
5. Алан Бьюли. "Изучаем SQL".
6. Документация к СУБД: Официальная документация по MySQL / PostgreSQL / MS SQL.

