**Министерство транспорта Российской Федерации**

**Федеральное государственное автономное образовательное  
учреждение высшего образования  
«РУТ (МИИТ)»**

**Институт транспортной техники и систем управления**

**Кафедра «Управление и защита информации»**

КУРСОВОЙ ПРОЕКТ

по дисциплине

**«Основы построения защищенных баз данных»**

**на тему  
«Хлебопекарня»**

Выполнил: ст. гр. ТКИ-411  
Анисимов Д.М.  
Проверил: Васильева М.А.

Москва 2022

Оглавление

[ЦЕЛЬ КУРСОВОГО ПРОЕКТА 5](#_Toc104306270)

[ЗАДАНИЕ НА КУРСОВОВОЙ ПРОЕКТ 5](#_Toc104306271)

[ПРОЕКТИРОВАНИЕ РЕЛЯЦИОННОЙ БАЗЫ ДАННЫХ 6](#_Toc104306272)

[1 Инфологическое проектирование 6](#_Toc104306273)

[1.1 Анализ предметной области 6](#_Toc104306274)

[1.2 Анализ информационных задач и круга пользователей системы 7](#_Toc104306275)

[2 Логическое проектирование реляционной БД 7](#_Toc104306276)

[2.1 Составление реляционных отношений 7](#_Toc104306277)

[2.2 Нормализация полученных отношений 11](#_Toc104306278)

[3 Физическое проектирование БД 11](#_Toc104306279)

[3.1 Разработка скриптов на создание базы данных и таблиц 11](#_Toc104306280)

[3.2 Разработка скриптов на добавление данных в таблицы 13](#_Toc104306281)

[3.3 Разработка необходимых запросов, представлений (view), процедур, функций и триггеров 16](#_Toc104306282)

[Список литературы 18](#_Toc104306283)

# ЦЕЛЬ КУРСОВОГО ПРОЕКТА

Целью курсового проекта является изучение методов и закрепление знаний в проектировании реляционных баз данных (РБД) в системе управления базами данных (СУБД) Microsoft SQL Server.

# ЗАДАНИЕ НА КУРСОВОВОЙ ПРОЕКТ

В данном курсовом проекте ставится задача разработать РБД в СУБД Microsoft SQL Server по заданной теме «Кондитерская фабрика». Проектирование РБД проводится с помощью метода «Сущность-связь». Проверка построенной модели РБД осуществляется с помощью метода нормализации отношений.

Описание предметной области.

БД создается для информационного обслуживания сотрудников хлебопекарни. Хлебопекарня выпекает изделия на заказ. Продукты заказывают у поставщиков, выпечку отправляют в магазины города.

Готовые запросы:

1. Показывать ассортимент товара и его цену.

2. Показывать сумму (количество) заказываемого продукта у данного поставщика за отчетный период.

3. Показывать список товара (общую сумму заказа, суму заказа данного товара), заказанного данным магазином.

4. Показывать количество произведенного товара (за отчетный период, по названию).

# ПРОЕКТИРОВАНИЕ РЕЛЯЦИОННОЙ БАЗЫ ДАННЫХ

## Инфологическое проектирование

### Анализ предметной области

База данных создаётся для информационного обслуживания администрации фабрики

Выделим базовые сущности этой предметной области:

* bakery\_menu
* bakery\_produced
* bakery\_suplier
* bakery\_store

ER-диаграмма БД, выполненная в среде визуального построения диаграмм PHPMyAdmin приведена на Рисунке 1. На данной ER-диаграмме присутствуют связи многие - ко - многим.

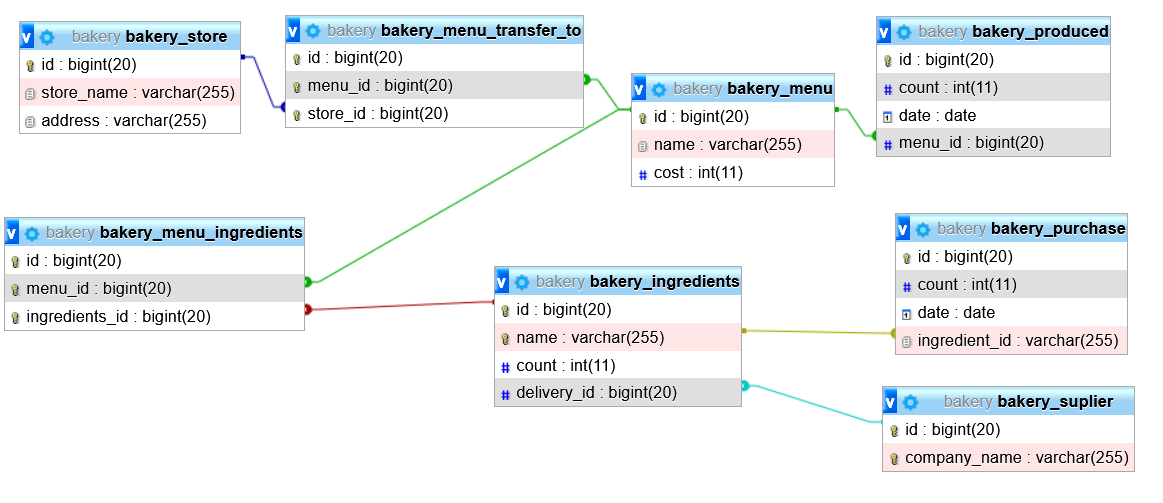


Рисунок 1 – Уточненная ER-диаграмма

В полученной ER-диаграмме присутствуют только связи один-ко-многим. Такие связи легко реализуются средствами СУБД.

### Анализ информационных задач и круга пользователей системы

Система создаётся для обслуживания следующих групп пользователей:

* Администраторы фабрики
* Менеджеры продаж

Определим границы информационной поддержки пользователей:

**1) Функциональные возможности:**

* Ведение БД (запись, чтение, модификация, удаление в архив);
* Обеспечение логической непротиворечивости БД;
* Обеспечение защиты данных от несанкционированного или случайного доступа (определение прав доступа);
* Реализация наиболее часто встречающихся запросов в готовом виде.

**2) Готовые запросы:**

* Показывать ассортимент товара и его цену.
* Показывать сумму (количество) заказываемого продукта у данного поставщика за отчетный период.
* Показывать список товара (общую сумму заказа, суму заказа данного товара), заказанного данным магазином.
* Показывать количество произведенного товара (за отчетный период, по названию).

## Логическое проектирование реляционной БД

### Составление реляционных отношений

Каждое реляционное отношение соответствует одной сущности (объекту ПрО), и в него вносятся все атрибуты сущности. Для каждого отношения необходимо определить первичный ключ и внешние ключи (если они есть). В том случае, если базовое отношение не имеет потенциальных ключей, вводится *суррогатный первичный ключ*, который не несёт смысловой нагрузки и служит только для идентификации записей.

1. Для сущности **Меню фабрики** (bakery\_menu) потенциальным ключом является атрибут «id», т.к. его значение является уникальным. Но значениями этого атрибута будут, числа. Введем для этой сущности суррогатный ключ ID.
2. Для сущности **Произведенные товары** (bakery\_ingredients) атрибут menu\_id является потенциальным ключом. Для единообразия именования для этой сущности в качестве ключевого атрибута введем суррогатный ключ ID.
3. Для сущности **Необходимые ингредиенты** (bakery\_ingredients) атрибут menu\_id является потенциальным ключом. Для единообразия именования для этой сущности в качестве ключевого атрибута введем суррогатный ключ ID.
4. Для сущности **Связь Ингедиенты Меню** (bakery\_menu\_ingredients) совокупность атрибутов menu\_id и ingredients\_id является уникальной и является потенциальным ключом.
5. Для сущности **Магазин** (bakery\_store) атрибут id является уникальным и является потенциальным ключом.
6. Для сущности **Поставщик** (bakery\_suplier) атрибут id является уникальным и является потенциальным ключом.
7. Для сущности **Покупатель** (bakery\_purchase) атрибут id является уникальным и является потенциальным ключом.

**Таблица 1 – Схема отношения** **Меню фабрики** **(Bakery\_menu)**

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| Первичный ключ | id | Целое число | первичный ключ |
| Наименование товара | name | Строка (255) | уникальное, обязательное поле |
| Стоимость | cost | Целое число | обязательное поле |

Таблица 2 – Схема отношения Произведенные товары (bakery\_produced)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| Первичный ключ | id | Целое число | первичный ключ |
| Дата производства товара | date | Дата | обязательное поле |
| Количество | count | Целое число | обязательное поле |
| Вторичный ключ | menu\_id | Целое число | обязательное поле |

Таблица 3 – Схема отношения Необходимые ингредиенты (bakery\_ingredients)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| Первичный ключ | id | Целое число | первичный ключ |
| Имя | name | Строка(255) | обязательное поле |
| Количество | count | Целое число | обязательное поле |
| Вторичный ключ | delivery\_id | Целое число | обязательное поле |

Таблица 4 – Схема отношения Связь Ингедиенты Меню (bakery\_menu\_ingredients)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| Первичный ключ | id | Целое число | первичный ключ |
| ID товара | menu\_id | Целое число | обязательное поле |
| ID ингредиента | ingredients\_id | Целое число | обязательное поле |

Таблица 5 – Схема отношения Связь Магазин Меню (bakery\_menu\_transfer)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| Первичный ключ | id | Целое число | первичный ключ |
| ID товара | menu\_id | Целое число | обязательное поле |
| ID магазина | store\_id | Целое число | обязательное поле |

Таблица 6 – Схема отношения Магазин (bakery\_store)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| ID магазина | id | Целое число | первичный ключ |
| Наименование магазина | store\_name | Строка(255) | обязательное поле |
| Адрес доставки | address | Целое число | обязательное поле |

Таблица 7 – Схема отношения Поставщик (bakery\_suplier)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| ID поставщика | id | Целое число | первичный ключ |
| Наименование поставщика | company\_name | Строка(255) | обязательное поле |

Таблица 8 – Схема отношения Покупатель (bakery\_purchase)

|  |  |  |  |
| --- | --- | --- | --- |
| *Содержание поля* | *Имя поля* | *Тип, длина* | *Примечания* |
| Первичный ключ | id | Целое число | первичный ключ |
| Стоимость | cost | Целое число | обязательное поле |
| Количество | count | Целое число | обязательное поле |
| Вторичный ключ | ingredient\_id | Целое число | обязательное поле |
| Дата производства товара | date | Дата | обязательное поле |

### Нормализация полученных отношений

**1НФ.** Для приведения таблиц к 1НФ требуется составить таблицы (один атрибут – один столбец) и разбить сложные атрибуты на простые, а многозначные атрибуты вынести в отдельные отношения.  
В наших таблицах все атрибуты простые.

**2НФ.** Все отношения находятся во 2НФ, так как для каждого из них введен уникальный ключ.

**3НФ**. Все неключевые атрибуты зависят только от первичного ключа и не зависят друг от друга. Поэтому спроектированная **база данных находится в третьей нормальной форме**.

## Физическое проектирование БД

### Разработка скриптов на создание базы данных и таблиц

БД «Хлебопекарня» проектируется в среде Microsoft SQL Server Management Studio.

*Создание базы данных «*кондитерская фабрика*»:*

USE bakery;

*Создание таблицы Ингредиенты*

CREATE TABLE Ingredients (

id bigint NOT NULL PRIMARY KEY IDENTITY(1,1),

[name] varchar(255) NOT NULL UNIQUE([name]),

[count] int NOT NULL,

[delivery\_id] bigint NOT NULL

)

*Создание таблицы Меню*

CREATE TABLE Menu (

[id] bigint NOT NULL PRIMARY KEY IDENTITY(1,1),

[name] varchar(255) NOT NULL,

[cost] int NOT NULL

)

*Создание таблицы связь Меню Ингредиенты*

CREATE TABLE Menu\_ingredients (

id bigint NOT NULL PRIMARY KEY IDENTITY(1,1),

menu\_id bigint NOT NULL,

ingredients\_id bigint NOT NULL

)

*Создание таблицы связь Меню Поставки*

CREATE TABLE Menu\_transfer (

id bigint NOT NULL PRIMARY KEY IDENTITY(1,1),

menu\_id bigint NOT NULL,

store\_id bigint NOT NULL

)

*Создание таблицы связь Магазин*

CREATE TABLE Store (

id bigint NOT NULL PRIMARY KEY IDENTITY(1,1),

store\_name varchar(255) NOT NULL,

[address] varchar(255) NOT NULL

)

*Создание таблицы связь Поставщик*

CREATE TABLE Suplier (

id bigint NOT NULL PRIMARY KEY IDENTITY(1,1),

company\_name varchar(255) NOT NULL

)

*Создание таблицы связь Произведенные товары*

CREATE TABLE Produced (

id bigint NOT NULL PRIMARY KEY IDENTITY(1,1),

[count] int NOT NULL,

[date] date NOT NULL,

menu\_id bigint NOT NULL

)

*Создание таблицы связь Покупатель*

CREATE TABLE Purchase (

id bigint NOT NULL PRIMARY KEY IDENTITY(1,1),

[count] int NOT NULL,

[date] date NOT NULL,

[cost] int NOT NULL,

ingredient\_id varchar(255) NOT NULL

)

### Разработка скриптов на добавление данных в таблицы

*Заполнение таблицы* *Поставщик:*

INSERT INTO conditer\_suplier (company\_name) VALUES

('Русский Сахар'),

('Макфа'),

('Синявинское'),

('ЭкоМилк'),

('Домик в Деревне'),

('Красный октябрь'),

('Башкирская содовая компания'),

('Мозырьсоль'),

('Dr.Oetker'),

('Коптевский рынок'),

('Лафа'),

('Олейна'),

('FruitOnline'),

('Haas');

*Заполнение таблицы* *Магазин:*

INSERT INTO Store (store\_name, [address]) VALUES

('Пекарня', 'г.Москва Кожевническая ул., 8, стр. 2'),

('Старинные пироги', 'Г.Москва 1-я ул. Машиностроения, 10');

*Заполнение таблицы* *Ингредиенты:*

INSERT INTO Ingredients ([name], [count], delivery\_id) VALUES

('Сахар', 1000, 1),

('Мука пшеничная', 400, 2),

('Яйца', 100, 3),

('Масло сливочное', 500, 4),

('Сметана', 600, 5),

('Какао порошок', 9999, 6),

('Сода пищевая', 400, 7),

('Молоко', 3000, 5),

('Соль', 99999, 8),

('Дрожжи сухие', 1780, 9),

('Шея свинная', 1900, 10),

('Картофель', 4000, 11),

('Лук репчатый', 538, 10),

('Масло растительное', 125, 12),

('Перец черный молотый', 70, 11),

('Вишня', 253, 13),

('Разрыхлитель', 228, 9),

('Ванильный сахар', 9999999, 1),

('Крахмал', 439, 14),

('Мука ржаная', 400, 2),

('Мед', 120, 10),

('Молоко сухое', 384, 5),

('Солод', 64, 11),

('Тмин', 22, 10);

*Заполнение таблицы* *Меню:*

INSERT INTO Menu ([name], [cost]) VALUES

('пирог \"Зебра\"', 600),

('пирог с мясом и картошкой', 1100),

('Пирог Вишневый', 650),

('Хлеб белый', 25),

('Шоколадный пирог', 650),

('Хлеб ржаной', 23);

*Заполнение таблицы* *Связь Меню Ингредиенты:*

INSERT INTO Menu\_ingredients (menu\_id, ingredients\_id) VALUES

(1, 1),

(1, 2),

(1, 3),

(1, 4),

(1, 5),

(1, 6),

(1, 7),

(2, 1),

(2, 2),

(2, 3),

(2, 4),

(2, 8),

(2, 9),

(2, 10),

(2, 11),

(2, 12),

(2, 13),

(2, 14),

(2, 15),

(3, 1),

(3, 2),

(3, 3),

(3, 4),

(3, 5),

(3, 16),

(3, 17),

(3, 18),

(3, 19),

(4, 1),

(4, 2),

(4, 9),

(4, 10),

(4, 14),

(5, 1),

(5, 2),

(5, 3),

(5, 6),

(5, 8),

(5, 14),

(5, 17),

(6, 2),

(6, 4),

(6, 9),

(6, 10),

(6, 20),

(6, 21),

(6, 22),

(6, 23),

(6, 24);

*Заполнение таблицы* *Связь Меню Магазин:*

INSERT INTO Menu\_transfer (menu\_id, store\_id) VALUES

(1, 1),

(2, 1),

(2, 2),

(3, 2),

(4, 1),

(5, 1),

(5, 2),

(6, 1);

*Заполнение таблицы* *Связь Произведено:*

INSERT INTO Produced ([count], [date], menu\_id) VALUES

(20, '2022-05-22', 1),

(19, '2022-05-23', 1),

(15, '2022-05-22', 2),

(16, '2022-05-23', 2),

(12, '2022-05-20', 3),

(10, '2022-05-19', 3),

(120, '2022-05-22', 4),

(120, '2022-05-23', 4),

(23, '2022-05-22', 5),

(29, '2022-05-23', 5),

(48, '2022-05-22', 6),

(50, '2022-05-23', 6);

*Заполнение таблицы* *Связь Покупки:*

INSERT INTO Purchase ([count], [date], ingredient\_id, cost) VALUES

(400, '2022-05-22', 'Сахар', 3040),

(100, '2022-05-22', 'Мука пшеничная', 1000),

(380, '2022-05-23', 'Мука пшеничная', 3800),

(80, '2022-05-22', 'Яйца', 632),

(80, '2022-05-21', 'Яйца', 632),

(100, '2022-05-23', 'Яйца', 790),

(254, '2022-05-22', 'Масло сливочное', 165),

(600, '2022-05-21', 'Сметана', 465),

(800, '2022-05-22', 'Молоко', 734),

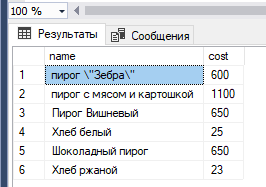
(1200, '2022-05-23', 'Шея свинная', 1200);

### Разработка необходимых запросов, представлений (view), процедур, функций и триггеров

***1. Показывать ассортимент товара и его цену:***

SELECT [name], cost

FROM Menu



**Рисунок 3**

***2. Показывать сумму (количество) заказываемого продукта у данного поставщика за отчетный период:***

SELECT p1.ingredient\_id as ingredient, SUM(p1.[count]) as [Sum]

FROM Purchase as p1

WHERE ingredient\_id IN

(

SELECT i1.[name]

FROM Ingredients as i1

INNER JOIN Suplier as s1

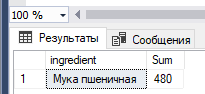
ON i1.delivery\_id = s1.id

WHERE s1.company\_name = 'Макфа'

)

and (p1.[date] >'2022.05.21' and p1.[date] <= '2022.05.24')

GROUP BY p1.ingredient\_id



**Рисунок 4**

***3. Показывать список товара (общую сумму заказа, суму заказа данного товара), заказанного данным магазином.***

SELECT SUM(m1.cost\*p1.[count]) as [Sum], m1.[name]

FROM Produced as p1

INNER JOIN Menu as m1

ON p1.menu\_id = m1.id

WHERE menu\_id IN (

SELECT m2.menu\_id

FROM Menu\_transfer as m2

INNER JOIN Store as s1

ON s1.id = m2.store\_id

WHERE s1.store\_name = 'Пекарня'

)

GROUP BY m1.[name]

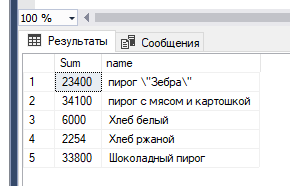


Рисунок 5

***4. Показывать количество произведенного товара (за отчетный период, по названию).***

SELECT SUM(p1.[count]) as [Sum], m1.[name]

FROM Produced as p1

INNER JOIN Menu as m1

ON p1.menu\_id = m1.id

WHERE m1.[name] = 'Пирог Вишневый'

AND (p1.[date] > '2022-05-18' AND p1.[date] < '2022-05-21')

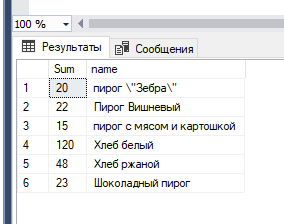


Рисунок 6

Построим диаграмму спроектированной базы данных (Рисунок 7).

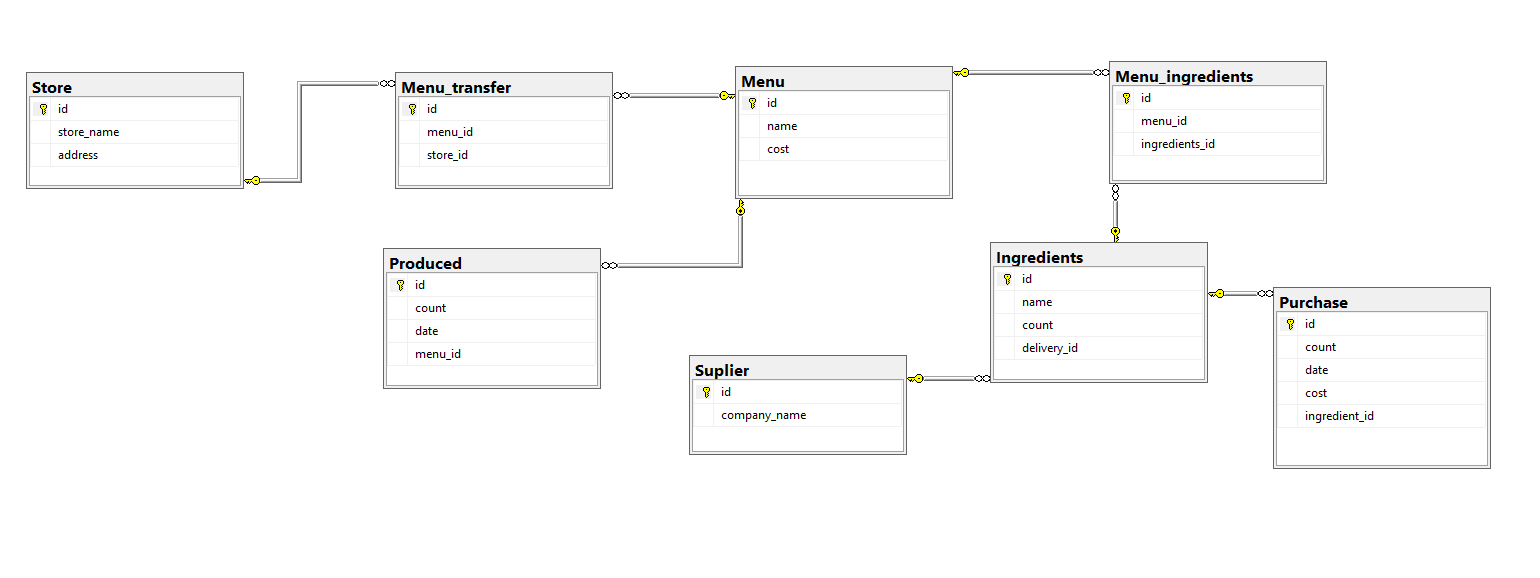


Рисунок 7

# Список литературы

**Visual Paradigm** Entity Relation [Online] // Visual Paradigm Online. - @2020 by Visual Paradigm, 2020. - 2 2, 2020. - https://online.visual-paradigm.com/app/diagrams/#diagram:proj=0&type=ERDiagram.

ГОСТ 7.32-2017. Межгосударственный стандарт. Система стандартов по информации, библиотечному и издательскому делу. Отчет о научно-исследовательской работе. Структура и правила оформления [В Интернете] // Официальный сайт компании «КонсультантПлюс». - 24 Март 2020 г.. - http://www.consultant.ru/document/cons\_doc\_LAW\_292293/.