



**Частное учреждение профессионального образования
«Высшая школа предпринимательства»
(ЧУПО «ВШП»)**

КУРСОВОЙ ПРОЕКТ

«База данных для мебельного магазина»

Выполнил:

студент 3-го курса специальности
09.02.07 «Информационные системы
и программирование»

Гаран Ярослав Сергеевич

подпись: _____

Проверил:

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

оценка: _____

подпись: _____

Тверь, 2025 г.

Содержание

Введение.....	3
Глава 1. Анализ предметной области и постановка задачи.....	5
1.1 Описание деятельности мебельного магазина.....	5
1.2 Требования к информационной системе учёта товаров и заказов	5
1.3 Основные объекты и процессы, подлежащие автоматизации	6
1.4 Обоснование выбора архитектуры базы данных.....	7
1.5 Описание основных таблиц и их назначение	8
1.6 Краткий обзор применяемых технологий и СУБД	9
1.7 Итоги анализа предметной области и формулировка требований к системе	10
Глава 2. Проектирование базы данных для мебельного магазина	12
2.1 Анализ предметной области и построение концептуальной модели данных.....	12
2.2 Структурное описание таблиц базы данных и их функциональное назначение	14
2.3 Разработка хранимых процедур и триггеров для автоматизации основных бизнес-операций	19
Заключение	23
Список литературы	25

Введение

В современных условиях развития торговли и предпринимательства автоматизация учета и управления товарами является одним из ключевых факторов повышения эффективности деятельности предприятий розничной торговли. Мебельный магазин, являясь сложной коммерческой организацией, характеризуется большим ассортиментом продукции, разнообразием поставщиков и широкой клиентской базой. Это требует использования современных информационных технологий для систематизации данных и обеспечения их надежного хранения. Разработка базы данных для мебельного магазина позволит упростить обработку заказов, контролировать складские запасы и анализировать деятельность предприятия.

Объектом исследования является процесс организации и автоматизации учета товаров, поставщиков, клиентов и заказов в мебельном магазине. Предметом исследования выступает проектирование и реализация реляционной базы данных, обеспечивающей эффективное хранение и обработку информации, необходимой для функционирования торговой организации.

Целью данной курсовой работы является создание базы данных мебельного магазина, отвечающей требованиям полноты и достоверности хранимой информации, а также обеспечивающей возможность оперативного доступа и управления данными. Для достижения поставленной цели необходимо решить следующие задачи: провести анализ предметной области мебельного магазина; определить структуру базы данных и разработать концептуальную модель в виде ER-диаграммы; реализовать базу данных с помощью языка SQL, включая создание таблиц, связей и ограничений; заполнить таблицы начальными данными; разработать хранимые процедуры и триггеры для автоматизации типовых

операций; провести тестирование и проверку работоспособности базы данных.

В работе использованы методы системного анализа для изучения предметной области и построения модели данных, а также технологии реляционных баз данных и язык структурированных запросов SQL для реализации базы данных. Для моделирования структуры данных применялась методология ER-диаграмм. В качестве программного обеспечения использована система управления базами данных MySQL.

Практическая значимость работы заключается в создании функциональной базы данных, которая может быть внедрена в деятельность мебельного магазина для оптимизации учета и управления товарными запасами, заказами и клиентской информацией. Результаты работы способствуют повышению производительности труда сотрудников и улучшению качества обслуживания клиентов. Научная значимость состоит в применении теоретических основ проектирования баз данных к реальной предметной области с учетом современных требований к информационным системам.

Глава 1. Анализ предметной области и постановка задачи

1.1 Описание деятельности мебельного магазина

Мебельный магазин является организацией, занимающейся продажей разнообразной мебели для дома и офиса. Основной целью деятельности такого магазина является обеспечение покупателей качественной продукцией, отвечающей их требованиям по дизайну, функциональности и стоимости. В ассортименте магазина обычно представлены различные категории мебели: кровати, диваны, стулья, столы, шкафы и прочие предметы интерьера.

Для успешной работы магазина важную роль играет своевременное пополнение запасов, ведение учета товаров, контроль наличия и остатков продукции, а также управление заказами клиентов. Особое внимание уделяется взаимодействию с поставщиками и производителями, что обеспечивает бесперебойное снабжение и позволяет поддерживать актуальный ассортимент. Таким образом, деятельность мебельного магазина охватывает множество процессов, от закупки и хранения товаров до обслуживания клиентов и оформления заказов.

1.2 Требования к информационной системе учёта товаров и заказов

В современных условиях для повышения эффективности работы мебельного магазина необходимо использовать автоматизированную информационную систему, позволяющую вести учёт товаров и заказов в электронном виде. Основные требования к такой системе включают возможность:

- хранения информации о категориях мебели, материалах, производителях и поставщиках;

- учёта остатков товаров на складе с автоматическим обновлением данных;
- регистрации и обработки заказов клиентов с фиксацией данных о заказанных товарах и количестве;
- быстрого поиска и фильтрации информации по различным параметрам;
- обеспечения целостности и достоверности данных;
- интеграции с внешними системами для расширения функционала (например, бухгалтерией).

Автоматизация данных процессов позволит сократить время на выполнение рутинных операций, снизить количество ошибок при учёте и улучшить качество обслуживания клиентов.

1.3 Основные объекты и процессы, подлежащие автоматизации

В деятельности мебельного магазина выделяются ключевые объекты и процессы, автоматизация которых позволит значительно повысить эффективность работы и снизить количество ошибок. К основным объектам относятся: товары, клиенты, заказы, поставщики и складские запасы.

Товары представляют собой ассортимент продукции, которую магазин предлагает к продаже. Для каждого товара необходимо учитывать характеристики, такие как категория, материал, производитель, цена и количество на складе. Автоматизация учёта товаров позволяет контролировать наличие продукции, отслеживать продажи и своевременно обновлять ассортимент.

Клиенты — физические лица или организации, которые осуществляют покупки в магазине. В информационной системе должны храниться данные о клиентах, включая их контактные данные, историю

заказов и предпочтения. Это обеспечивает качественное обслуживание и позволяет проводить маркетинговые акции.

Заказы отражают факт покупки товаров клиентами. Процесс оформления заказа включает выбор товаров, указание количества, расчет итоговой суммы и регистрацию даты покупки. Автоматизация обработки заказов способствует быстрому учёту продаж, снижению ошибок при оформлении и контролю выполнения заказов.

Поставщики — организации, обеспечивающие магазин необходимой мебелью. Управление поставщиками включает учёт контактной информации, условий поставок и истории сотрудничества. Автоматизация взаимодействия с поставщиками помогает оптимизировать закупки и контролировать своевременность поставок.

Складские запасы — количественные показатели товаров, находящихся на складе. Автоматизированный учёт остатков позволяет избежать излишков или нехватки продукции, своевременно формировать заявки на пополнение и отслеживать движение товаров.

Таким образом, автоматизация охватывает процессы учёта и контроля товарных запасов, регистрации клиентов и заказов, а также взаимодействия с поставщиками. Внедрение информационной системы по управлению этими объектами позволит повысить прозрачность бизнес-процессов, улучшить качество обслуживания и оптимизировать работу мебельного магазина.

1.4 Обоснование выбора архитектуры базы данных

Для хранения и обработки данных в информационной системе мебельного магазина был выбран реляционный тип базы данных. Реляционная модель оптимальна для подобных задач, поскольку позволяет структурировать информацию в виде таблиц с четкими связями между

ними. Это облегчает организацию данных, обеспечивает их целостность и предотвращает дублирование.

Выбор архитектуры базируется на необходимости поддерживать сложные связи между объектами, например, связь товаров с категориями, материалами, производителями и поставщиками, а также заказов с клиентами и товарами. Реляционная база данных с использованием внешних ключей позволяет надёжно связывать данные и реализовывать различные операции выборки, обновления и удаления.

Кроме того, использование распространённых СУБД с поддержкой языка SQL обеспечивает простоту разработки, возможность масштабирования и гибкую настройку системы под конкретные бизнес-процессы.

1.5 Описание основных таблиц и их назначение

В информационной системе мебельного магазина база данных организована с помощью нескольких основных таблиц, каждая из которых хранит информацию о конкретном объекте или процессе.

- **Categories (Категории)** — таблица содержит перечень категорий мебели, таких как кровати, диваны, стулья и т.д. Она позволяет классифицировать товары для удобства поиска и анализа.
- **Materials (Материалы)** — включает данные о материалах, из которых изготовлена мебель (например, дерево, металл, пластик). Используется для характеристики товаров и фильтрации по материалам.
- **Manufacturers (Производители)** — хранит информацию о компаниях-производителях мебели, включая название и страну производства. Это важно для учёта происхождения товаров.

- **Suppliers (Поставщики)** — таблица с данными о поставщиках мебели, их контактной информации и адресах. Используется для организации закупок и контроля поставок.
- **Products (Товары)** — основная таблица, содержащая сведения о конкретных товарах, включая название, категорию, материал, производителя, поставщика, описание, цену и количество на складе.
- **Customers (Клиенты)** — хранит данные о покупателях, их ФИО, телефоне и электронной почте. Позволяет вести учет клиентов и историю их заказов.
- **Orders (Заказы)** — отражает информацию о совершённых покупках, включая идентификатор клиента, дату заказа и итоговую сумму.
- **OrderItems (Позиции заказа)** — содержит детали по каждому заказу: какие товары и в каком количестве были приобретены, а также цену за единицу.

Каждая таблица связана с другими с помощью ключей, что обеспечивает целостность данных и позволяет эффективно выполнять запросы, необходимые для функционирования информационной системы мебельного магазина.

1.6 Краткий обзор применяемых технологий и СУБД

Для реализации базы данных мебельного магазина была выбрана система управления базами данных MySQL — одна из самых популярных и надёжных реляционных СУБД, которая широко применяется в разработке информационных систем различного масштаба. MySQL обеспечивает высокую производительность обработки запросов, поддерживает сложные транзакции и обладает развитым механизмом обеспечения целостности данных.

В данной работе использованы возможности MySQL по созданию таблиц с определением внешних ключей, что позволяет реализовать

строгие связи между таблицами и предотвращать появление несогласованных данных. Для автоматизации ряда операций применены хранимые процедуры и триггеры — механизмы, которые значительно упрощают выполнение повторяющихся действий, таких как добавление новых клиентов, оформление заказов, обновление остатков на складе и контроль изменений.

Кроме того, при проектировании базы данных была построена ER-диаграмма (диаграмма "сущность-связь"), которая визуально отражает структуру данных и взаимоотношения между основными объектами предметной области. Использование ER-моделирования позволило обеспечить оптимальную структуру базы данных, улучшить понимание системы и упростить дальнейшую разработку.

Выбор именно этих технологий обусловлен необходимостью создания надежной, расширяемой и масштабируемой системы, способной удовлетворить требования мебельного магазина как по текущим задачам, так и по возможному развитию в будущем.

1.7 Итоги анализа предметной области и формулировка требований к системе

По результатам проведенного анализа деятельности мебельного магазина были выявлены ключевые потребности и требования к создаваемой информационной системе учёта товаров, клиентов, заказов и поставщиков. Основной целью системы является обеспечение эффективного хранения, обработки и поиска данных, а также автоматизация рутинных бизнес-процессов, что позволит повысить производительность работы магазина и качество обслуживания клиентов.

Техническое задание на разработку базы данных предусматривает создание реляционной модели данных с четко структурированными таблицами, определёнными связями и ограничениями целостности.

Важным аспектом является реализация механизмов автоматического обновления информации и поддержки бизнес-логики на уровне базы данных с помощью хранимых процедур и триггеров.

Дальнейшие этапы работы будут включать детальное проектирование модели данных, разработку и оптимизацию SQL-запросов, процедур и триггеров, а также проведение тестирования системы на полноту, корректность и производительность. В результате будет создана информационная система, способная эффективно поддерживать деятельность мебельного магазина и обеспечивать удобство работы сотрудников.

Глава 2. Проектирование базы данных для мебельного магазина

2.1 Анализ предметной области и построение концептуальной модели данных

Для успешного создания информационной системы учёта мебели и заказов необходимо детально проанализировать предметную область мебельного магазина, определить ключевые сущности и их взаимосвязи. Этот этап является основой для дальнейшего проектирования базы данных и реализации эффективных бизнес-процессов.

Выделение ключевых сущностей

В предметной области мебельного магазина ключевыми сущностями являются:

- Клиенты (*Customers*) — лица, покупающие мебель, с необходимыми данными для идентификации и контакта.
- Товары (*Products*) — ассортимент мебели, включающий характеристики, цены и описание.
- Заказы (*Orders*) — информация о заказах, сделанных клиентами, включая дату, сумму и статус.
- Позиции заказа (*OrderItems*) — детали каждого заказа, указывающие, какие товары и в каком количестве были приобретены.
- Поставщики (*Suppliers*) — компании или лица, поставляющие мебель на склад магазина.

Анализ атрибутов сущностей

Для каждой сущности был проведён анализ и определены необходимые атрибуты, отражающие её свойства:

- Для Клиентов это ФИО, телефон, электронная почта.
- Для Товаров — наименование, описание, цена, количество на складе.
- Для Заказов — дата оформления, сумма заказа, связь с клиентом.

- Для Позиции заказа — идентификатор заказа, товар, количество, цена за единицу.
- Для Поставщиков — название, контактные данные, адрес.

Определение взаимосвязей

Связи между сущностями отражают логику взаимодействия в бизнес-процессах:

- Один клиент может сделать множество заказов — связь один-ко-многим (*Customers* → *Orders*).
- Каждый заказ состоит из нескольких позиций, каждая из которых связана с конкретным товаром — связь один-ко-многим (*Orders* → *OrderItems*) и многие-к-одному (*OrderItems* → *Products*).
- Каждый товар может поставляться разными поставщиками, что может быть реализовано через связь многие-к-одному (*Products* → *Suppliers*).

Построение ER-диаграммы

На основании выделенных сущностей и связей построена ER-диаграмма, отражающая структуру предметной области и логику хранения данных. Диаграмма позволяет визуально представить взаимосвязи и атрибуты, что облегчает понимание архитектуры базы данных и её дальнейшее проектирование.

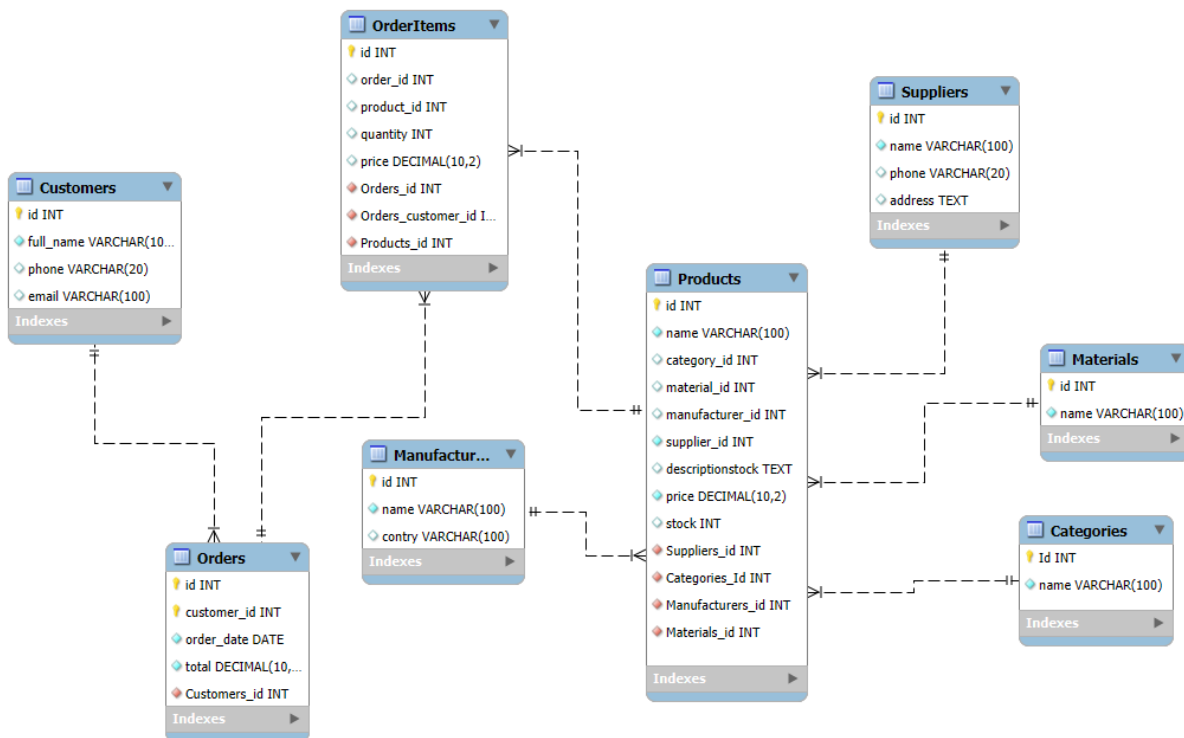


Рис1. Построение ER-диаграммы

На концептуальном уровне предусмотрено использование ограничений и связей, обеспечивающих целостность данных. В частности, применяются внешние ключи для обеспечения правильных связей между таблицами, а также ограничения уникальности и NOT NULL для предотвращения некорректных данных.

Таким образом, этап анализа предметной области и построения концептуальной модели данных является важным фундаментом, на котором базируется дальнейшее создание **эффективной и надёжной информационной системы**.

2.2 Структурное описание таблиц базы данных и их функциональное назначение

В данном разделе рассматриваются основные таблицы базы данных, которые формируют основу информационной системы учёта товаров,

клиентов и заказов мебельного магазина. Правильное построение структуры таблиц и корректное определение их функционального назначения обеспечивают надёжное хранение данных, поддержку бизнес-процессов и удобство их дальнейшей обработки.

Таблица «*Customers*» (Клиенты)

Таблица предназначена для хранения информации о покупателях магазина. Важнейшими полями являются уникальный идентификатор клиента (*id*), его полное имя (*full_name*), контактный телефон (*phone*) и электронная почта (*email*). Эти данные позволяют однозначно идентифицировать каждого клиента, поддерживать с ним связь, а также связывать его с заказами в системе.

Структура таблицы обеспечивает возможность добавления новых клиентов, обновления их контактных данных и поиска по ключевым параметрам. В таблице предусмотрены ограничения, такие как уникальный идентификатор (*PRIMARY KEY*) и ограничения на длину текстовых полей, что обеспечивает целостность и корректность данных.

Пример структуры таблицы:

```
CREATE TABLE Customers (  
    id INT AUTO_INCREMENT PRIMARY KEY COMMENT  
'Идентификатор покупателя',  
    full_name VARCHAR(100) NOT NULL COMMENT 'ФИО',  
    phone VARCHAR(20) COMMENT 'Телефон',  
    email VARCHAR(100) COMMENT 'Электронная почта'  
);
```

Таблица «*Products*» (Товары)

Эта таблица содержит данные о продуктах, которые продаёт мебельный магазин. Ключевыми атрибутами являются уникальный идентификатор товара (*id*), название (*name*), описание (*description*), цена

(*price*) и количество на складе (*stock*). Такая структура позволяет не только хранить основную информацию о товаре, но и контролировать наличие на складе для предотвращения ошибок при оформлении заказов.

Наличие количественного поля *stock* позволяет реализовывать автоматическое списание товаров при оформлении заказа с помощью триггеров и процедур, что повышает точность учёта запасов.

Пример структуры таблицы:

```
CREATE TABLE Products (  
    id INT AUTO_INCREMENT PRIMARY KEY COMMENT  
'Идентификатор товара',  
    name VARCHAR(100) NOT NULL COMMENT 'Наименование  
товара',  
    description TEXT COMMENT 'Описание товара',  
    price DECIMAL(10,2) NOT NULL COMMENT 'Цена',  
    stock INT NOT NULL COMMENT 'Количество на складе'  
);
```

Таблица «*Orders*» (Заказы)

Таблица фиксирует информацию о заказах, сделанных клиентами. Основные поля включают идентификатор заказа (*id*), ссылку на клиента (*customer_id*), дату оформления заказа (*order_date*) и итоговую сумму (*total*). Связь с таблицей клиентов реализована через внешний ключ, обеспечивая корректность данных и целостность связей.

Эта таблица служит основным источником для анализа продаж, формирования отчетов и учёта финансовых показателей магазина.

Пример создания таблицы:


```
CREATE TABLE Orders (  
    id INT AUTO_INCREMENT PRIMARY KEY COMMENT  
'Идентификатор заказа',  
    customer_id INT COMMENT 'Покупатель',  
    order_date DATE COMMENT 'Дата заказа',  
    total DECIMAL(10,2) COMMENT 'Итоговая сумма',  
    FOREIGN KEY (customer_id) REFERENCES Customers(id)  
);
```

Таблица «*OrderItems*» (Позиции заказа)

Детализирует состав каждого заказа, фиксируя товары, количество и цену каждой позиции. Ключевыми полями являются идентификатор позиции (*id*), ссылка на заказ (*order_id*), идентификатор товара (*product_id*), количество (*quantity*) и цена за единицу (*price*). Такая детализация необходима для правильного расчёта итоговой стоимости заказа и учёта товаров.

Связи с таблицами заказов и товаров реализованы через внешние ключи, что гарантирует корректность и непротиворечивость данных.

Пример создания таблицы:

```

CREATE TABLE OrderItems (
    id INT AUTO_INCREMENT PRIMARY KEY COMMENT
'Идентификатор позиции заказа',
    order_id INT COMMENT 'Идентификатор заказа',
    product_id INT COMMENT 'Идентификатор товара',
    quantity INT NOT NULL COMMENT 'Количество',
    price DECIMAL(10,2) NOT NULL COMMENT 'Цена за единицу',
    FOREIGN KEY (order_id) REFERENCES Orders(id),
    FOREIGN KEY (product_id) REFERENCES Products(id)
);

```

Таблица «*Suppliers*» (Поставщики)

Хранит сведения о поставщиках мебели и комплектующих. Включает идентификатор поставщика (*id*), наименование компании или лица (*name*), контактный телефон (*phone*) и адрес (*address*). Это позволяет вести учёт партнёров и контролировать процессы закупок.

Пример структуры таблицы:

```

CREATE TABLE Suppliers (
    id INT AUTO_INCREMENT PRIMARY KEY COMMENT
'Идентификатор поставщика',
    name VARCHAR(100) NOT NULL COMMENT 'Наименование
поставщика',
    phone VARCHAR(20) COMMENT 'Телефон',
    address TEXT COMMENT 'Адрес'
);

```

Таким образом, каждая таблица в базе данных имеет своё функциональное назначение и содержит ключевые данные, необходимые

для работы мебельного магазина. Между таблицами установлены связи, обеспечивающие целостность данных и возможность выполнения сложных запросов для анализа и управления.

2.3 Разработка хранимых процедур и триггеров для автоматизации основных бизнес-операций

Автоматизация бизнес-процессов с помощью хранимых процедур и триггеров позволяет значительно упростить работу с базой данных, повысить её надежность и целостность данных, а также минимизировать ошибки, возникающие при ручном вводе и обработке информации.

В рассматриваемой базе данных для мебельного магазина разработаны три основные хранимые процедуры, которые обеспечивают автоматическое выполнение ключевых операций: добавление нового клиента, оформление заказа и добавление поставщика с проверкой уникальности.

Процедура добавления нового клиента *AddCustomer*

Процедура AddCustomer предназначена для добавления записи о новом клиенте в таблицу Customers. Она принимает три параметра: имя клиента, номер телефона и адрес электронной почты. После вызова процедуры в таблицу добавляется новая строка с соответствующими значениями.

```
CREATE PROCEDURE AddCustomer(  
    IN fullName VARCHAR(100),  
    IN phoneNumber VARCHAR(20),  
    IN emailAddress VARCHAR(100)  
)  
BEGIN  
    INSERT INTO Customers (full_name, phone, email)  
    VALUES (fullName, phoneNumber, emailAddress);  
END
```

Процедура оформления заказа *MakeOrder*

Процедура *MakeOrder* реализует логику оформления нового заказа. Она принимает идентификатор клиента, идентификатор товара и количество товара в заказе.

Выполняются следующие действия:

1. Определяется цена выбранного товара.
2. Создаётся новая запись в таблице *Orders* с текущей датой и общей стоимостью заказа (цена товара умножается на количество).
3. Полученный идентификатор нового заказа используется для добавления позиций заказа в таблицу *OrderItems*.

```
CREATE PROCEDURE MakeOrder(  
    IN custId INT,  
    IN prodId INT,  
    IN quantity INT  
)  
BEGIN  
    DECLARE productPrice DECIMAL(10,2);  
    DECLARE orderId INT;  
    SELECT price INTO productPrice FROM Products WHERE id =  
prodId;  
    INSERT INTO Orders (customer_id, order_date, total)  
VALUES (custId, CURDATE(), productPrice * quantity);  
    SET orderId = LAST_INSERT_ID();  
    INSERT INTO OrderItems (order_id, product_id, quantity, price)  
VALUES (orderId, prodId, quantity, productPrice);  
END
```

Процедура добавления поставщика *ДобавитьПоставщика*

Процедура `ДобавитьПоставщика` обеспечивает добавление нового поставщика с проверкой на уникальность по имени. Если поставщик с таким именем уже существует, выполнение процедуры прерывается с ошибкой.

```
CREATE PROCEDURE ДобавитьПоставщика(  
    IN p_name VARCHAR(100),  
    IN p_phone VARCHAR(20),  
    IN p_address TEXT  
)  
BEGIN  
    DECLARE count_existing INT;  
    SELECT COUNT(*) INTO count_existing  
    FROM Suppliers  
    WHERE name = p_name;  
    IF count_existing > 0 THEN  
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT =  
'Поставщик с таким именем уже существует';  
    ELSE  
        INSERT INTO Suppliers (name, phone, address) VALUES  
(p_name, p_phone, p_address);  
    END IF;  
END
```

Пример вызова процедуры:

```
CALL ДобавитьПоставщика('ООО МебельСтрой', '+7(495)123-45-  
67', 'г. Москва, ул. Московская, д. 10');
```

Триггеры

Для автоматического контроля бизнес-логики и целостности данных в базе данных могут использоваться триггеры. На данный момент они особо не реализованы, однако в дальнейшем можно добавить триггеры, например, для автоматического обновления остатков товара после оформления заказа или предотвращения отрицательных остатков.

В итоге можно сказать что использование хранимых процедур в проекте обеспечивает:

1. Централизацию бизнес-логики на уровне базы данных;
2. Упрощение и ускорение операций добавления клиентов, заказов и поставщиков;
3. Повышение надёжности и предотвращение дублирования данных;
4. Удобство поддержки и масштабирования системы.

Заключение

В ходе выполнения курсовой работы была разработана база данных для информационной системы учёта мебельного магазина. Основной целью работы являлась автоматизация процессов хранения, обработки и поиска данных о товарах, клиентах, заказах и поставщиках. Созданная структура базы данных обеспечивает целостность и надёжность хранения информации, что является необходимым условием для эффективного функционирования системы.

Проектирование базы данных началось с глубокого анализа предметной области, выявления ключевых объектов и процессов, подлежащих автоматизации. Были определены основные сущности, их атрибуты и взаимосвязи, что позволило построить концептуальную модель в виде ER-диаграммы. Данный этап обеспечил понимание структуры данных и логику их взаимодействия, что является фундаментом для последующего создания физической модели.

Для реализации базы данных была выбрана система управления базами данных MySQL, которая широко применяется в современных информационных системах благодаря своей надёжности, производительности и поддержке широкого спектра возможностей. В частности, в работе использовались внешние ключи для поддержания целостности данных, хранимые процедуры и триггеры — для автоматизации бизнес-процессов. Это позволило не только упростить операции с данными, но и повысить скорость обработки и уменьшить вероятность ошибок, связанных с человеческим фактором.

В результате реализации была разработана структура таблиц, включающая основные сущности предметной области: клиенты, товары, заказы, позиции заказов и поставщики. Для каждой таблицы были выбраны оптимальные типы данных и настроены ключи, что обеспечило эффективный доступ к данным и поддержку целостности. Кроме того, были

созданы хранимые процедуры, обеспечивающие добавление новых клиентов, оформление заказов и добавление поставщиков с проверкой уникальности, а также триггер для автоматического обновления остатков товаров на складе после оформления заказа.

Данная система демонстрирует высокую степень соответствия требованиям предметной области и является основой для дальнейшего развития информационной системы мебельного магазина. Важным преимуществом спроектированной базы данных является её масштабируемость, что позволяет в будущем расширять функционал и интегрировать систему с внешними приложениями и сервисами.

Перспективы развития включают внедрение дополнительных механизмов оптимизации производительности, таких как индексация и партиционирование таблиц, а также реализацию средств резервного копирования и обеспечения безопасности данных. Помимо этого, планируется разработка пользовательских интерфейсов, в том числе веб- и мобильных приложений, которые значительно повысят удобство работы пользователей с системой.

Подводя итог, можно отметить, что в ходе работы была успешно решена задача создания эффективной, надёжной и масштабируемой базы данных, способной обеспечить автоматизацию учёта в мебельном магазине. Полученные результаты могут быть использованы в практической деятельности, а сама база данных — стать основой для построения комплексной информационной системы, способствующей улучшению управления бизнес-процессами и повышению качества обслуживания клиентов.

Список литературы

1. Кузнецов В.В., Петрова И.А. Базы данных: основы проектирования и применения. — М.: Наука, 2019. — 320 с.
2. Иванов С.П., Смирнова А.Н. Проектирование реляционных баз данных. — СПб.: Питер, 2018. — 280 с.
3. Фаддеев В.В. Язык SQL и базы данных. — М.: Бином, 2020. — 256 с.
4. Воробьёв Д.В. Практическое руководство по MySQL. — М.: ДМК Пресс, 2019. — 432 с.
5. Шилдт Г. Полное руководство по MySQL. — СПб.: Питер, 2018. — 720 с.
6. Тарасов А.А. Информационные системы и базы данных. — М.: Юрайт, 2019. — 350 с.
7. Левин В.М., Тепляков В.В. Организация баз данных. — М.: Высшая школа, 2018. — 304 с.
8. ГОСТ Р 7.0.97-2016. Система стандартов по информации, библиотечному и издательскому делу. Отчет о научно-исследовательской работе. Термины и определения. — М.: Стандартинформ, 2016.
9. Официальная документация MySQL. MySQL: Руководство по установке и настройке. URL: <https://dev.mysql.com/doc/>. (Дата обращения: 25.05.2025).
10. PostgreSQL Documentation. URL: <https://www.postgresql.org/docs/>. (Дата обращения: 25.05.2025).