**НЕГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ ЧАСТНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«МОСКОВСКИЙ ФИНАНСОВО-ПРОМЫШЛЕННЫЙ УНИВЕРСИТЕТ «СИНЕРГИЯ»**

|  |  |  |
| --- | --- | --- |
| **Факультет/Институт** |  | Информационных технологий |
|  |  | (наименование факультета/ Института) |
| **Направление/специальность** |  | Информационные системы и технологии |
| **подготовки:** |  | (код и наименование направления /специальности подготовки) |
| **Форма обучения:** |  | очная |
|  |  | (очная, очно-заочная, заочная) |
|  |  |  |

**Лабораторный практикум № 9**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **по дисциплине** |  | Администрирование информационных систем | | |
|  | | |  | (наименование дисциплины) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Обучающийся** |  | **Евстигнеев Владимир Викторович** |  |  |
|  |  | (ФИО) |  | (подпись) |
| **Группа** |  | **группа ВБИо-301рсоб** |  |
|  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Преподаватель** |  | Сибирев Иван Валерьевич |  |  |
|  |  | (ФИО) |  | (подпись) |

**Москва 2024 г.**

### **1. Анализ предметной области «Интернет-магазин»**

Для интернет-магазина можно выделить следующие сущности:

1. **Товары (Products):** информация о продуктах, которые продаются.
2. **Клиенты (Customers):** информация о зарегистрированных покупателях.
3. **Заказы (Orders):** данные о покупках, сделанных клиентами.
4. **Категории (Categories):** классификация товаров.

#### Связи между сущностями:

* Один товар принадлежит одной категории, но одна категория может содержать много товаров (**один-ко-многим**).
* Один клиент может сделать много заказов, но каждый заказ принадлежит одному клиенту (**один-ко-многим**).
* Один заказ может содержать много товаров, а один товар может быть в разных заказах (**многие-ко-многим**). Для этого потребуется промежуточная таблица **OrderDetails**.

### **2. Построение ER-диаграммы**

#### Таблицы и атрибуты:

1. **Categories**:
   * CategoryID (PK)
   * CategoryName
   * Description
2. **Products**:
   * ProductID (PK)
   * ProductName
   * CategoryID (FK → Categories)
   * Price
   * Stock
3. **Customers**:
   * CustomerID (PK)
   * FullName
   * Email
   * PhoneNumber
   * Address
4. **Orders**:
   * OrderID (PK)
   * CustomerID (FK → Customers)
   * OrderDate
   * TotalAmount
5. **OrderDetails**:
   * OrderDetailID (PK)
   * OrderID (FK → Orders)
   * ProductID (FK → Products)
   * Quantity
   * Price

+-----------------+ +-----------------+

| Categories | | Customers |

+-----------------+ +-----------------+

| \* CategoryID |<--------+ | \* CustomerID |

| CategoryName | | | FullName |

| Description | | | Email |

+-----------------+ | | PhoneNumber |

| | Address |

| +-----------------+

|

+-| +-----------------+

+-----------------+ | +-<| Orders |

| Products |>------+ +-----------------+

+-----------------+ +-->| \* OrderID |

| \* ProductID |<--------+ | CustomerID |

| ProductName | | | OrderDate |

| CategoryID | | | TotalAmount |

| Price | | +-----------------+

| Stock | |

+-----------------+ |

|

v

+-------------------------+

| OrderDetails |

+-------------------------+

| \* OrderDetailID |

| OrderID FK |

| ProductID FK |

| Quantity |

| Price |

+-------------------------+

### 3. Связи:

1. **Categories → Products:** Один ко многим.
2. **Customers → Orders:** Один ко многим.
3. **Orders ↔ Products через OrderDetails:** Многие ко многим.

### **4. Реализация в SQL**

#### 4.1. Создание таблиц

-- Таблица категорий

CREATE TABLE Categories (

CategoryID INT PRIMARY KEY AUTO\_INCREMENT,

CategoryName VARCHAR(255) NOT NULL,

Description TEXT

);

-- Таблица товаров

CREATE TABLE Products (

ProductID INT PRIMARY KEY AUTO\_INCREMENT,

ProductName VARCHAR(255) NOT NULL,

CategoryID INT,

Price DECIMAL(10, 2) NOT NULL,

Stock INT NOT NULL,

FOREIGN KEY (CategoryID) REFERENCES Categories(CategoryID)

);

-- Таблица клиентов

CREATE TABLE Customers (

CustomerID INT PRIMARY KEY AUTO\_INCREMENT,

FullName VARCHAR(255) NOT NULL,

Email VARCHAR(255) NOT NULL UNIQUE,

PhoneNumber VARCHAR(15),

Address TEXT

);

-- Таблица заказов

CREATE TABLE Orders (

OrderID INT PRIMARY KEY AUTO\_INCREMENT,

CustomerID INT,

OrderDate DATETIME DEFAULT CURRENT\_TIMESTAMP,

TotalAmount DECIMAL(10, 2),

FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)

);

-- Таблица деталей заказа

CREATE TABLE OrderDetails (

OrderDetailID INT PRIMARY KEY AUTO\_INCREMENT,

OrderID INT,

ProductID INT,

Quantity INT NOT NULL,

Price DECIMAL(10, 2) NOT NULL,

FOREIGN KEY (OrderID) REFERENCES Orders(OrderID),

FOREIGN KEY (ProductID) REFERENCES Products(ProductID)

);

#### 4.2. Добавление данных

-- Пример вставки данных

INSERT INTO Categories (CategoryName, Description) VALUES

('Electronics', 'Устройства и техника'),

('Books', 'Все виды книг'),

('Clothing', 'Мужская и женская одежда');

INSERT INTO Products (ProductName, CategoryID, Price, Stock) VALUES

('Smartphone', 1, 99999.99, 50),

('Laptop', 1, 99999.99, 30),

('Cracking the coding interview', 2, 1199.99, 100),

('Футболка мужская', 3, 799.99, 200);

INSERT INTO Customers (FullName, Email, PhoneNumber, Address) VALUES

('Vasya Pupkin', 'vasya.pupkin@mail.com', '+71234567890', 'Lenina st., 1'),

('Elena Golovach', 'golovach.lena@mail.com', '+79876543210', 'Mira St., 2');

INSERT INTO Orders (CustomerID, TotalAmount) VALUES

(1, 100799.98),

(2, 1199.98);

INSERT INTO OrderDetails (OrderID, ProductID, Quantity, Price) VALUES

(1, 1, 1, 99999.99),

(1, 4, 2, 799.99),

(2, 3, 1, 1199.99),

(2, 4, 1, 799.99);

#### 4.3. Настройка ролей и доступа

-- Создание ролей

CREATE ROLE Reader;

CREATE ROLE Editor;

-- Права доступа

GRANT SELECT ON Categories TO Reader;

GRANT SELECT ON Products TO Reader;

GRANT SELECT, INSERT, UPDATE, DELETE ON Orders TO Editor;

GRANT SELECT, INSERT, UPDATE, DELETE ON OrderDetails TO Editor;

### **5. CRUD-запросы**

#### Получение данных:

-- Все товары и их категории

SELECT p.ProductName, c.CategoryName, p.Price, p.Stock

FROM Products p

JOIN Categories c ON p.CategoryID = c.CategoryID;

#### Добавление данных:

-- Добавление нового товара

INSERT INTO Products (ProductName, CategoryID, Price, Stock)

VALUES ('Headphones', 1, 49999.99, 100);

#### Обновление данных:

-- Обновление цены товара

UPDATE Products SET Price = 59999.99 WHERE ProductID = 5;

#### Удаление данных:

-- Удаление товара

DELETE FROM Products WHERE ProductID = 5;

### **Создание пользовательских ролей и доступов**

Для выполнения задачи создадим две роли:

* **Reader**: доступ только к чтению (SELECT) для определённых таблиц.
* **Editor**: полный доступ к операциям (SELECT, INSERT, UPDATE, DELETE) для определённых таблиц.

#### SQL-запрос для создания ролей:

-- Создание ролей

CREATE ROLE Reader;

CREATE ROLE Editor;

-- Назначение прав доступа для роли Reader

GRANT SELECT ON Categories TO Reader;

GRANT SELECT ON Products TO Reader;

-- Назначение прав доступа для роли Editor

GRANT SELECT, INSERT, UPDATE, DELETE ON Orders TO Editor;

GRANT SELECT, INSERT, UPDATE, DELETE ON OrderDetails TO Editor;

### **1. Определение объектов базы данных для чтения**

Объекты базы данных с доступом **только для чтения (SELECT)**:

1. **Categories**: справочник категорий (редко изменяется).
2. **Products**: справочник товаров (редко изменяется).

### **2. Определение изменяемых объектов базы данных**

Объекты базы данных с доступом **к изменению (INSERT, UPDATE, DELETE)**:

1. **Orders**: данные о заказах. Частота корректировки: средняя.
2. **OrderDetails**: детали заказов. Частота корректировки: высокая.

Причина: заказы и их детали часто добавляются и иногда обновляются (например, при изменении статуса заказа).

### **4. Создание пользователей и привязка к ролям**

Создадим двух пользователей:

* **user\_reader**: доступ только для чтения.
* **user\_editor**: доступ для чтения и редактирования заказов и деталей заказов.

#### SQL-запрос для создания пользователей и привязки к ролям:

-- Создание пользователей

CREATE USER 'user\_reader'@'localhost' IDENTIFIED BY 'reader\_password';

CREATE USER 'user\_editor'@'localhost' IDENTIFIED BY 'editor\_password';

-- Привязка пользователей к ролям

GRANT Reader TO 'user\_reader'@'localhost';

GRANT Editor TO 'user\_editor'@'localhost';

### **Проверка и демонстрация доступов**

#### Для роли **Reader**:

Пользователь user\_reader сможет выполнить запросы:

-- Пример запроса на чтение (Reader)

SELECT \* FROM Products;

SELECT \* FROM Categories;

Если пользователь попытается изменить данные, он получит ошибку:

-- Ошибка: нет прав на изменение

INSERT INTO Products (ProductName, CategoryID, Price, Stock)

VALUES ('Tablet', 1, 199.99, 50);

#### Для роли **Editor**:

Пользователь user\_editor сможет выполнить следующие запросы:

-- Добавление заказа

INSERT INTO Orders (CustomerID, TotalAmount) VALUES (1, 14999.98);

-- Обновление данных о заказе

UPDATE Orders SET TotalAmount = 159999.98 WHERE OrderID = 1;

-- Удаление детали заказа

DELETE FROM OrderDetails WHERE OrderDetailID = 2;

Но он не сможет получить доступ к таблицам, к которым роль **Editor** не имеет прав:

-- Ошибка: нет прав на чтение таблицы Categories

SELECT \* FROM Categories;