Министерство науки и высшего образования Российской Федерации

федеральное государственное бюджетное образовательное учреждение высшего образования

«Российский экономический университет имени Г.В. Плеханова»

**МОСКОВСКИЙ ПРИБОРОСТРОИТЕЛЬНЫЙ ТЕХНИКУМ**

специальность 09.02.07 «Информационные системы и программирование»

Квалификация: Программист

ПРАКТИЧЕСКАЯ РАБОТА №4

ПО ОП.04 «Основы алгоритмизации и программирования»

|  |  |
| --- | --- |
| Выполнила студентка  группы П50-2-22  Панкова К.А. | Проверил преподаватель  \_\_\_\_\_\_\_\_\_Т.Д. Артамонова  «\_\_\_» \_\_\_\_\_\_\_\_\_ 2023 года |

Москва 2023

Цель: разработка информационной системы для управления товарами, заказами, клиентами и сотрудниками. Система должна включать подсистемы для управления товарами, заказами, клиентами и сотрудниками, а также функционал авторизации и регистрации пользователей.

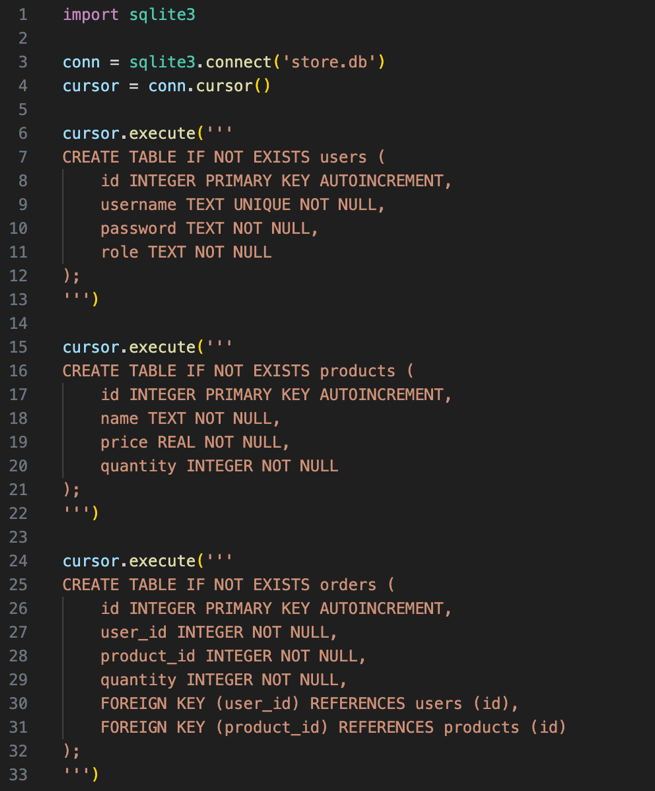
Ход работы:

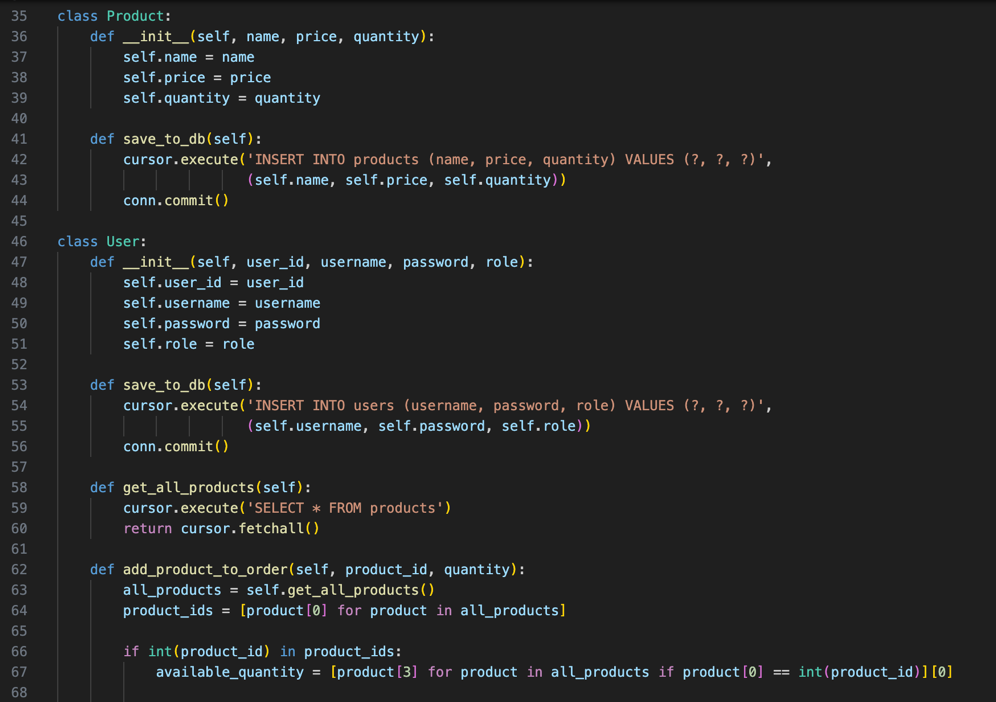
1. В коде сначала устанавливается подключение к базе данных store.db с помощью sqlite3.connect(). Затем создаются три таблицы: users, products, и orders, с использованием команд CREATE TABLE IF NOT EXISTS.

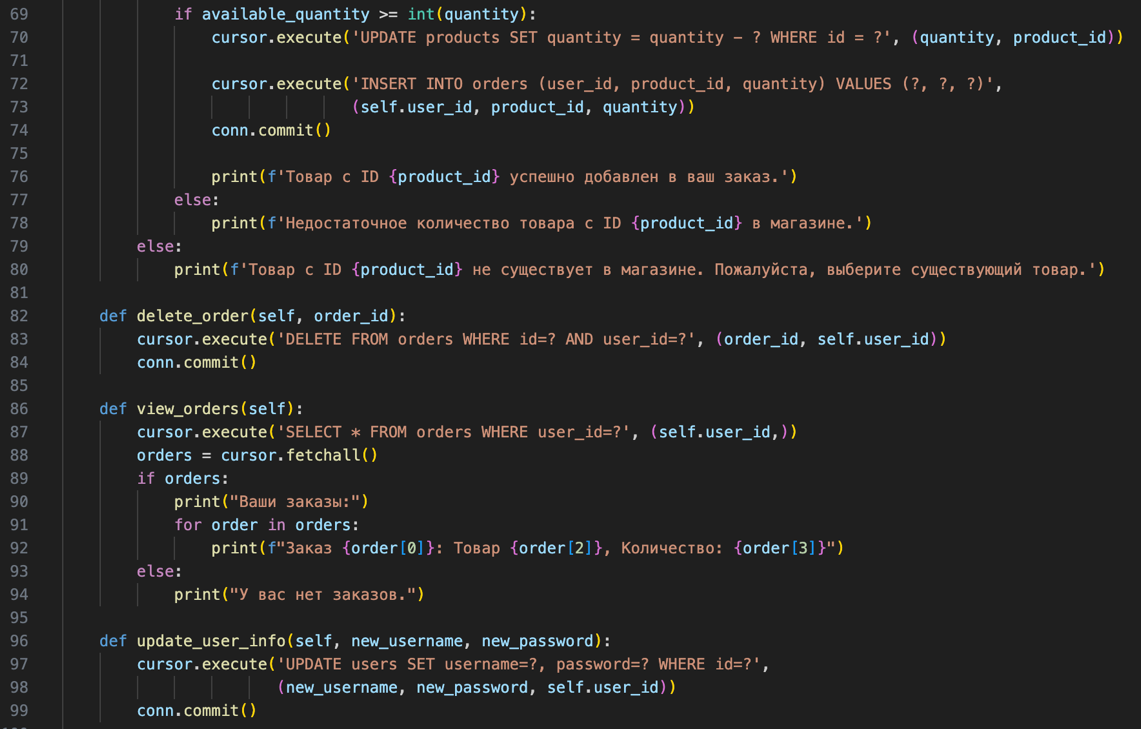
Далее в коде определены два класса: User и Product.

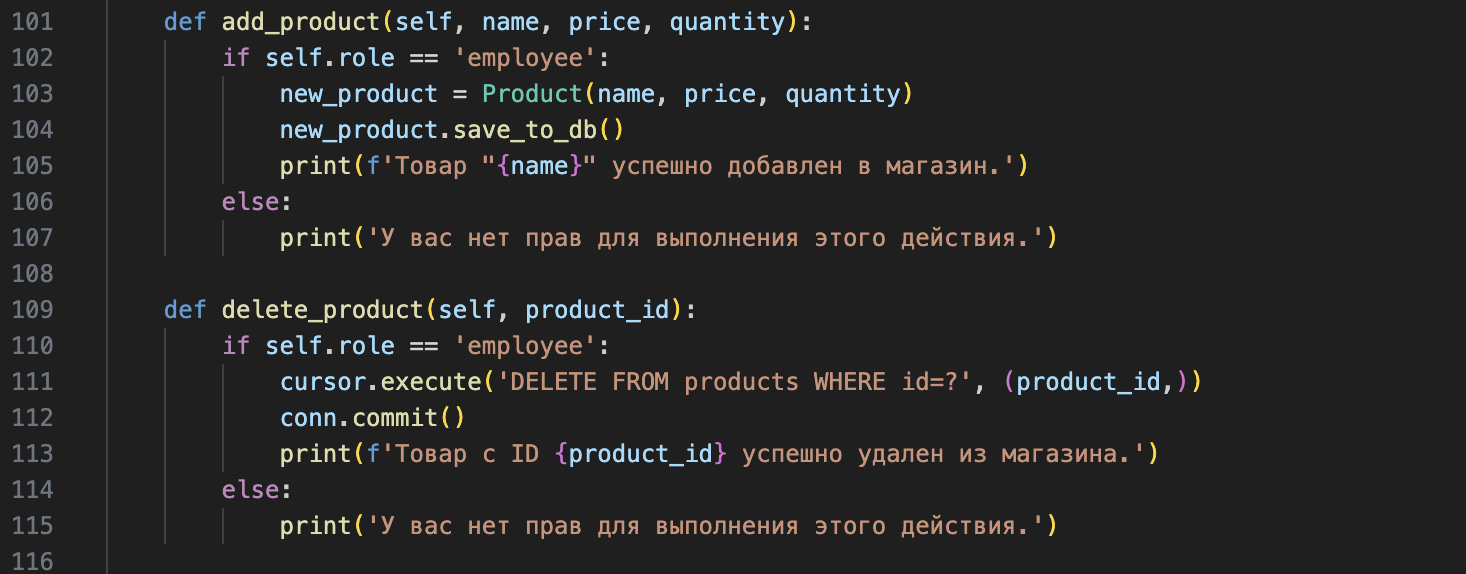
Класс Product представляет товар и содержит метод save\_to\_db, который выполняет операцию вставки новой записи в таблицу products с использованием cursor.execute() и сохраняет изменения в базу данных с помощью conn.commit().

Класс User представляет пользователя и содержит методы для работы с базой данных: save\_to\_db для сохранения пользователя, get\_all\_products для получения всех товаров из базы данных, add\_product\_to\_order для добавления товара в заказ, delete\_order для удаления заказа, view\_orders для просмотра заказов и т.д. Эти методы также используют cursor.execute() для выполнения SQL-запросов и conn.commit() для сохранения изменений в базе данных.









1. Этот код представляет собой набор функций, которые позволяют управлять базой данных магазина.

Функция update\_product обновляет информацию о продукте в базе данных по указанному идентификатору. Проверяет роль пользователя и выдает сообщение об успешном обновлении или отсутствии прав.

Функция filter\_products фильтрует продукты по ключевому слову в их названии. Проверяет роль пользователя и выводит результаты фильтрации или сообщение об отсутствии прав.

Функция view\_all\_products выводит все продукты из базы данных.

Функция view\_all\_employees выводит всех сотрудников из базы данных, проверяя при этом роль пользователя.

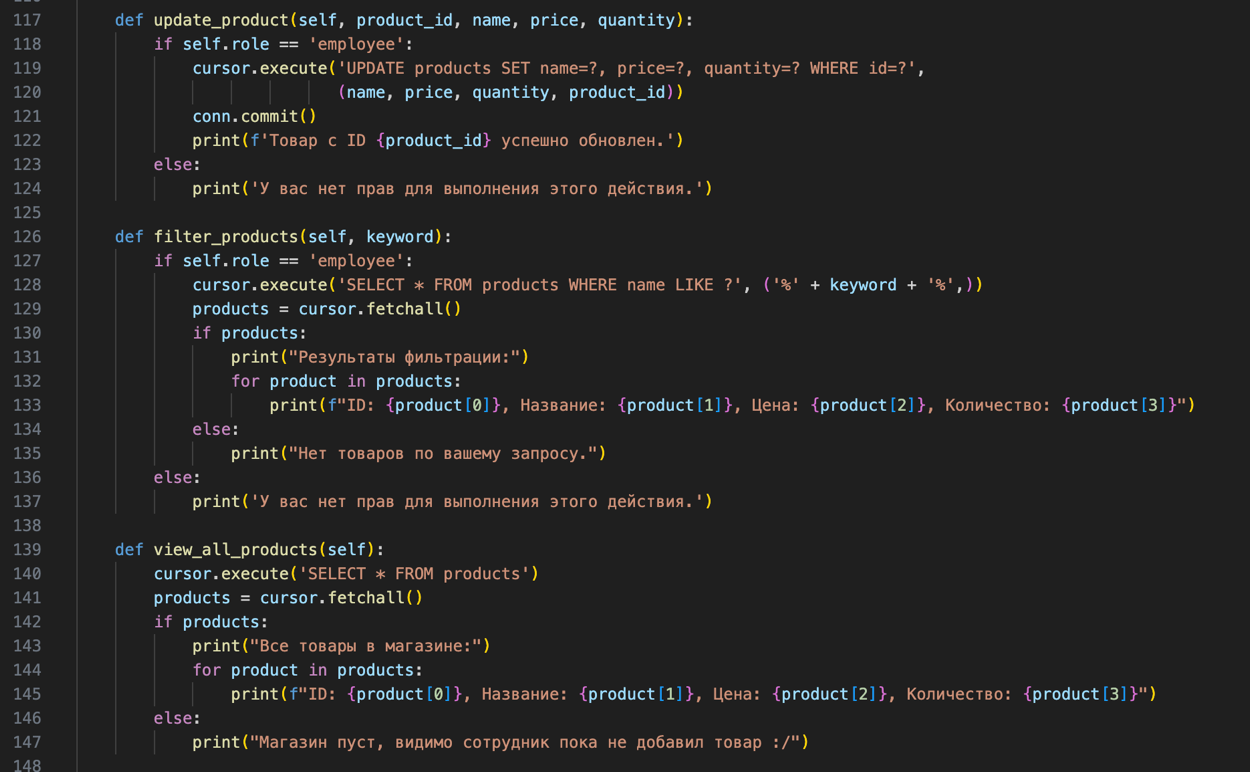
Функция add\_employee добавляет нового сотрудника в базу данных, проверяя при этом роль пользователя.

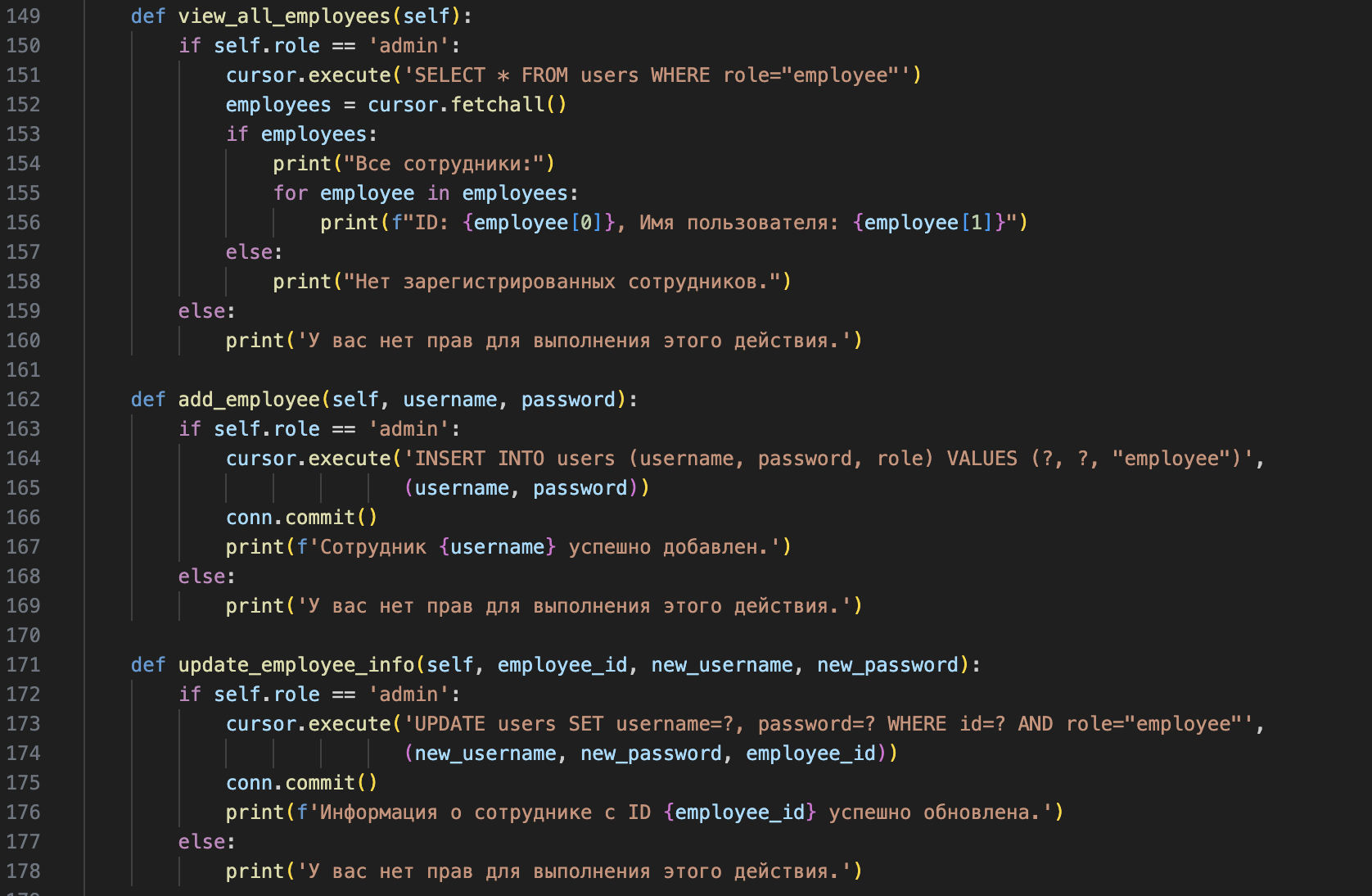
Функция update\_employee\_info обновляет информацию о сотруднике в базе данных, проверяя при этом роль пользователя.

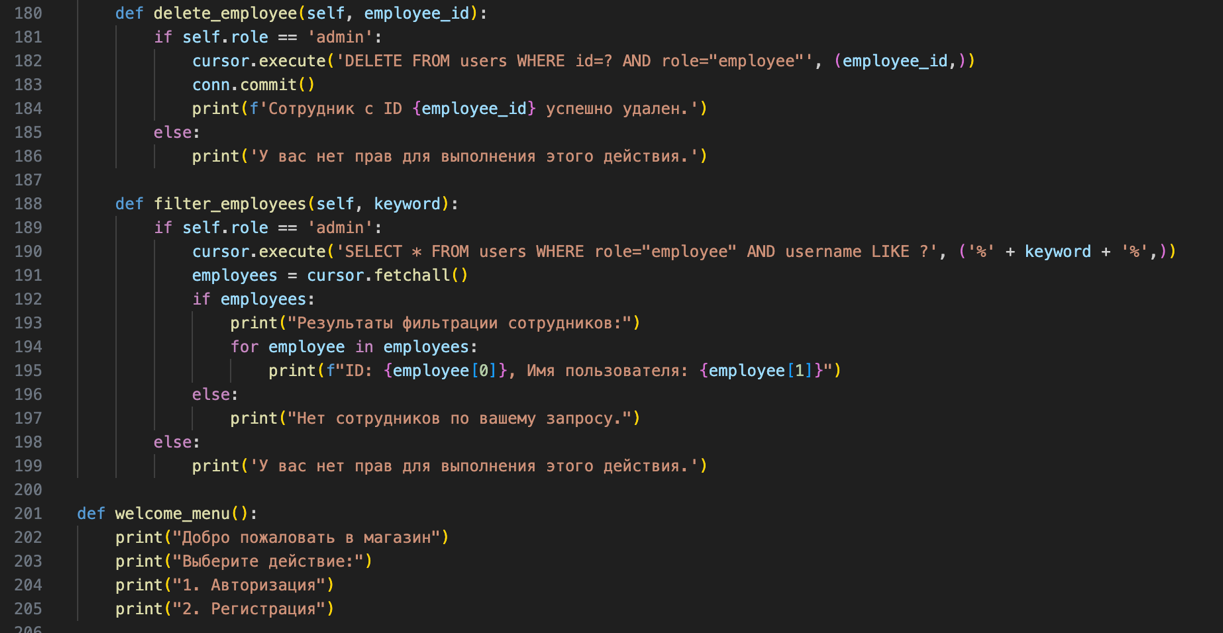
Функция delete\_employee удаляет сотрудника из базы данных.

Функция filter\_employees фильтрует сотрудников по ключевому слову в их имени.

Функция welcome\_menu - это вспомогательная функция, которая выводит приветственное сообщение и меню для пользователя.







1. Этот код представляет собой основной блок программы, который задает пользовательский интерфейс через консольное взаимодействие.

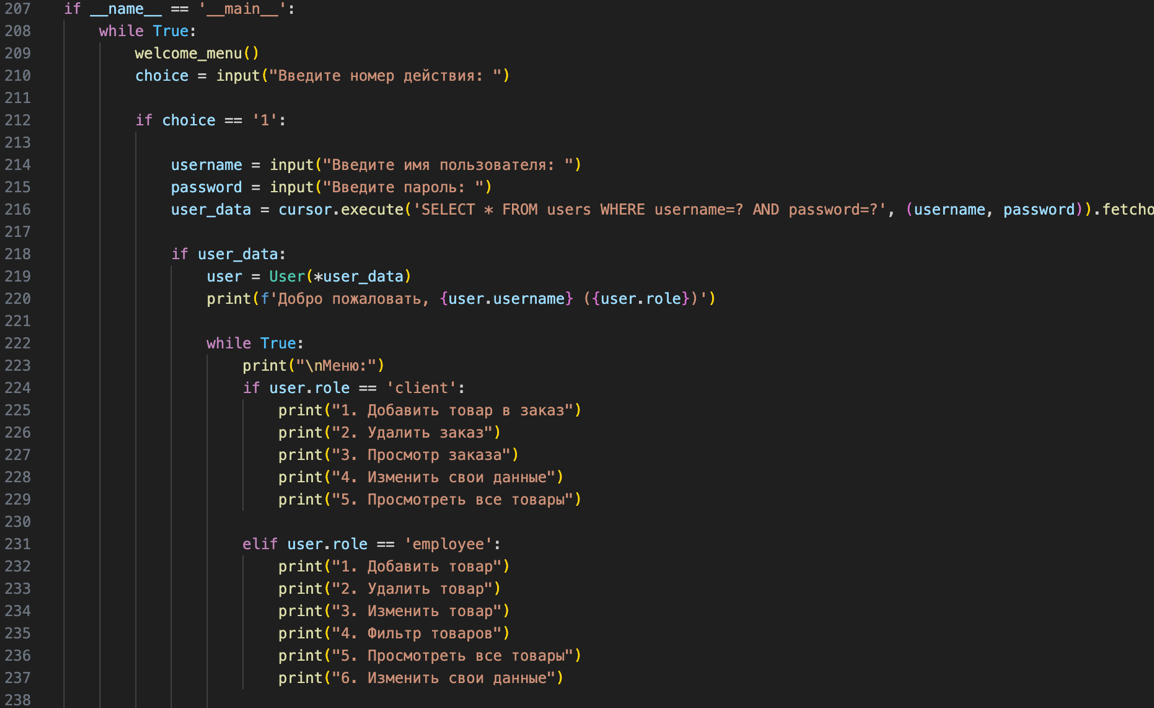
Основная идея программы заключается в том, что пользователь должен войти в систему и, в зависимости от его роли (клиент, сотрудник или администратор), ему доступны различные функции.

Когда программа запускается, она начинает с вывода приветственного меню с помощью функции welcome\_menu(). Затем программа ждет, пока пользователь введет номер действия. В зависимости от выбора пользователя программа предлагает определенный набор действий, который отличается в зависимости от роли, после чего пользователь вводит номер действия из этого набора.

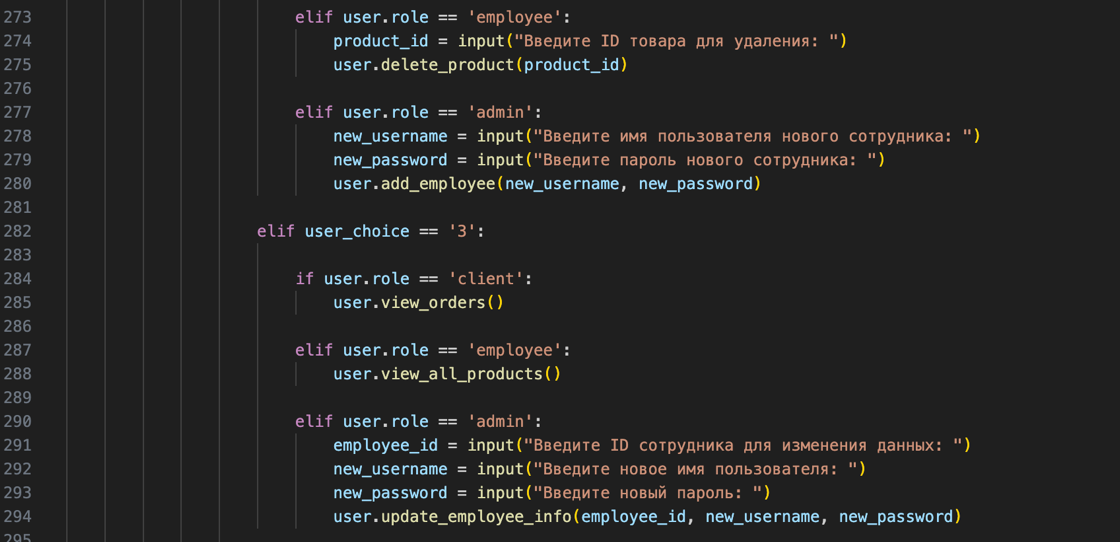
Далее, в зависимости от выбора пользователя, программа выполняет различные функции, такие как: добавление товара в заказ, удаление заказа, просмотр заказа, добавление товара, удаление товара, просмотр всех товаров, просмотр всех сотрудников, добавление сотрудника, изменение данных сотрудника, удаление сотрудника и фильтрация сотрудников.

Для выполнения всех этих функций используется методы класса User, которые зависят от роли пользователя.

Как только пользователь завершает определенное действие, программа возвращается обратно к выводу меню и ожидает следующего выбора.

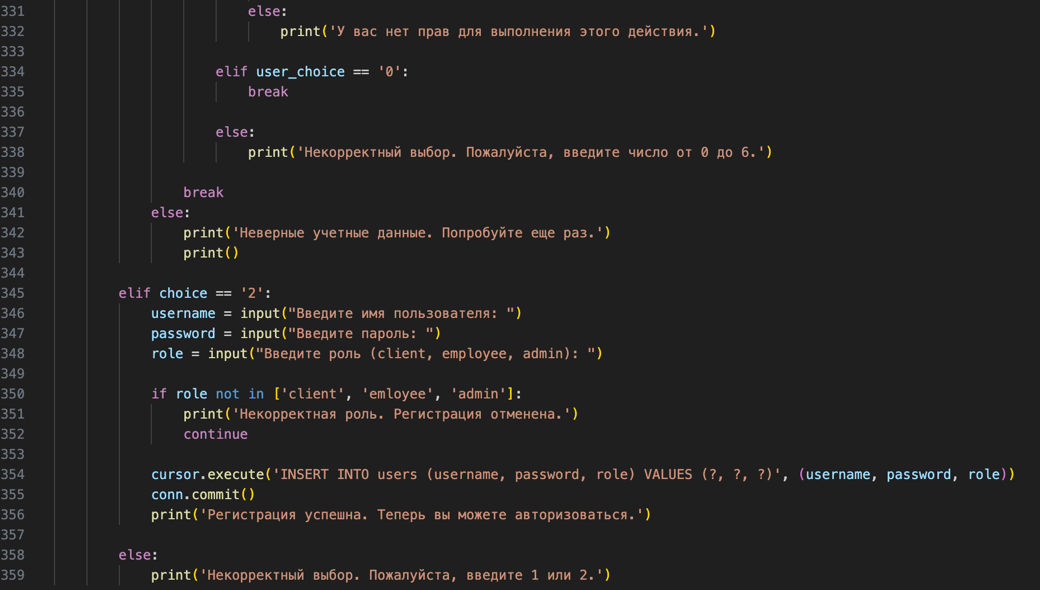






1. Данный код представляет собой текстовое меню, в котором пользователь может вводить различные числа для выполнения определенных действий. В зависимости от выбранного варианта и роли пользователя выполняются различные операции, такие как обновление информации о пользователе, просмотр продуктов или пользователей, фильтрация содержимого или удаление сотрудника. Встроенные проверки обеспечивают контроль доступа к выполнению различных действий в зависимости от роли пользователя.





Вывод: в ходе практической работы была разработана информационная система для управления товарами, заказами, клиентами и сотрудниками. Также система включает в себя подсистемы для управления товарами, заказами, клиентами и сотрудниками, а также функционал авторизации и регистрации пользователей.