**Семинарское задание 14**

**Основы подключения и создания базы данных**

1. **Подключение к SQLite**: Напишите скрипт на Python, который подключается к базе данных SQLite (создаёт её, если она не существует), и выводит сообщение "Подключение успешно". Используйте библиотеку sqlite3.
2. **Создание таблицы**: Подключитесь к базе данных SQLite и создайте таблицу users с колонками id, name, и age. Убедитесь, что таблица создаётся только один раз.
3. **Вставка данных**: Подключитесь к базе данных SQLite, создайте таблицу products с колонками id, name, и price. Вставьте несколько записей в таблицу и сохраните изменения.
4. **Получение данных из таблицы**: Напишите скрипт, который подключается к базе данных и выполняет SQL-запрос для выборки всех записей из таблицы users. Выведите результаты на экран.
5. **Обновление данных**: Подключитесь к базе данных и обновите возраст конкретного пользователя в таблице users. Выведите сообщение, указывающее, что данные успешно обновлены.

**Продвинутые операции с базой данных**

1. **Удаление данных**: Напишите скрипт для удаления записи из таблицы products, основываясь на значении id продукта. После удаления выведите оставшиеся записи, чтобы убедиться, что удаление прошло успешно.
2. **Исключение при подключении к базе данных**: Напишите скрипт, который подключается к базе данных и использует try-except для обработки ошибок подключения. В случае ошибки выведите сообщение "Ошибка подключения".
3. **Использование параметризованных запросов**: Подключитесь к базе данных и выполните параметризованный запрос для добавления данных в таблицу users, чтобы избежать SQL-инъекций. Запрос должен принимать параметры от пользователя.
4. **Создание соединения с PostgreSQL/MySQL**: Используйте библиотеку psycopg2 для PostgreSQL или mysql-connector-python для MySQL. Подключитесь к базе данных PostgreSQL или MySQL, создайте таблицу и вставьте несколько записей.
5. **Создание функции для выполнения SQL-запросов**: Напишите функцию execute\_query, которая принимает SQL-запрос и параметры, подключается к базе данных, выполняет запрос и возвращает результаты. Проверьте функцию, выполнив запрос для выборки данных из таблицы.