**Урок 12: Работа с библиотекой requests**

**Повторение прошлого материала (10 минут):**

“Прежде чем мы начнем, давайте вспомним, что мы изучали на прошлом уроке. Кто может объяснить, что такое tkinter и для чего она используется?”

“Правильно, tkinter — это библиотека для создания графических интерфейсов в Python.”

“Кто может рассказать, какие элементы интерфейса мы создавали и как обрабатывали события в tkinter?”

“Верно, мы создавали окна, кнопки, метки и текстовые поля, а также обрабатывали нажатия кнопок.”

**Цели и задачи урока:**

“Сегодня мы будем изучать библиотеку requests, которая позволяет взаимодействовать с веб-сервисами и получать данные из интернета.”

**Познание нового (25 минут):**

Основные понятия:

Что такое библиотека requests:

1. Библиотека requests — это простая и удобная библиотека для выполнения HTTP-запросов в Python. С ее помощью можно отправлять запросы на сервер и получать ответы.”

2. Установка библиотеки requests:

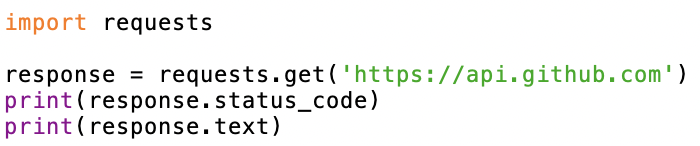
“Для начала нам нужно установить библиотеку requests. Откройте терминал и выполните команду pip install requests.”

3. Отправка GET-запроса:

“Самый простой тип запроса — это GET-запрос. Он используется для получения данных с сервера.”

GET — метод для получения данных с сервера. Он передаётся с URL, поэтому виден в адресной строке браузера или истории посещений сайтов. Применяется в фильтрах поисковиков и онлайн-магазинов.

Пример кода:



4. Работа с JSON-данными:

Часто серверы возвращают данные в формате JSON. Мы можем легко работать с этими данными с помощью библиотеки requests.”

Пример кода:



5. Отправка POST-запроса:

“POST-запрос используется для отправки данных на сервер.”

POST — метод отправки данных на сервер, например, после заполнения формы регистрации или авторизации на сайте. Информация передается в теле запроса и не может быть перехвачена злоумышленниками.

Пример кода:



**Некомпьютерная активность (10 минут):**

Обсуждение:

Примеры использования библиотеки requests:

“Библиотеку requests можно использовать для различных задач, таких как получение данных с веб-сервисов, отправка форм и взаимодействие с API.”

Вопросы для размышления:

“Почему важно уметь взаимодействовать с веб-сервисами? Какие задачи можно решать с помощью библиотеки requests?”

Активность:

Ученики делятся на небольшие группы:

Учитель делит класс на группы по 3-4 человека и дает задание:

1. “Обсудите в группах, какие приложения вы могли бы создать с использованием библиотеки requests. У вас есть 5 минут.”

Каждая группа представляет свои идеи классу:

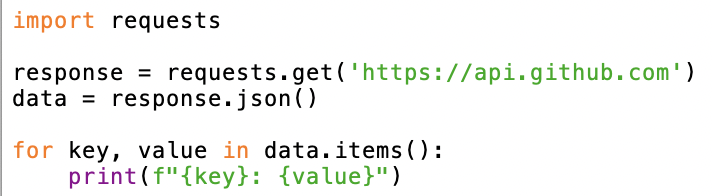
2. Учитель вызывает каждую группу по очереди для представления своих идей. После каждого выступления учитель обсуждает идеи с классом и приводит дополнительные примеры.

**Работа над проектом (25 минут):**

Практическое задание: Получение данных с веб-сервиса

“Давайте создадим простую программу, которая получает данные с веб-сервиса. Откройте IDLE, выберите ‘File’ -> ‘New File’.”

Написание кода для получения данных:



“Сохраните файл как requests\_example.py, затем выберите ‘Run’ -> ‘Run Module’ или нажмите F5. Вы должны увидеть данные, полученные с веб-сервиса.”

Структура программы:

“Мы создали программу, которая отправляет GET-запрос на сервер и выводит полученные данные. Это показывает, как легко можно взаимодействовать с веб-сервисами с помощью библиотеки requests.”

**Дополнительное задание:**

Задание: Изменить программу для получения данных с другого веб-сервиса:

“Теперь давайте изменим нашу программу, чтобы она получала данные с другого веб-сервиса. Попробуйте написать следующий код:”



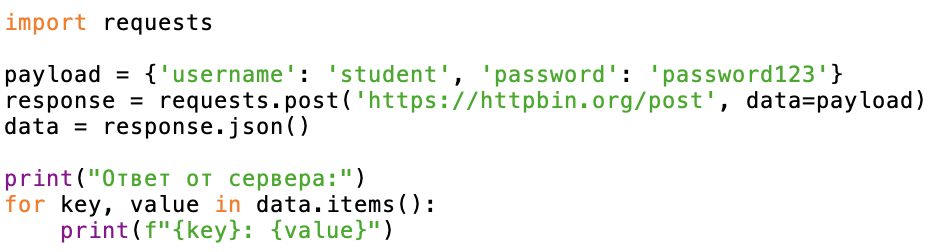
**Проблемная задача (10 минут)**

Задание: Написание программы с использованием библиотеки requests

Пример выполнения:

“Теперь давайте создадим программу, которая будет отправлять POST-запрос с данными и получать ответ от сервера.”

“Напишите следующий код:”



“Запустите программу и посмотрите, как она работает. Какие данные были отправлены и получены?”

**Рефлексия (10 минут):**

“Сегодня мы узнали о библиотеке requests и как выполнять HTTP-запросы в Python. Кто может объяснить, как отправить GET- и POST-запросы?”