

## Лабораторная работа № 11.

**Тема:** Основы программирования (Python). Работа с библиотекой requests, http-запросы.

**Цель:** закрепить теоретические знания и применить их на практике.

**Задание:**

### Вариант 1

1. Поиск информации о супергерое.

Используйте Superhero API для получения информации о конкретном супергерое по его имени. Напишите программу, которая:

Выводит список всех супергероев.

Принимает от пользователя имя супергероя (например, Batman).

Отправляет HTTP-запрос к API для получения данных о супергерое.

Выводит на экран следующую информацию:

Имя супергероя.

Интеллект, силу и скорость (из раздела powerstats).

Список его псевдонимов (если есть).

Пример работы программы:

Введите имя супергероя: Batman

Информация о супергерое:

- Имя: Batman

- Интеллект: 100

- Сила: 26

- Скорость: 27

- Псевдонимы: ['Dark Knight', 'Caped Crusader']

Для выполнения заданий используйте библиотеку requests для отправки HTTP-запросов.

Документация API доступна по ссылке: <https://akabab.github.io/superhero-api/api/>

2. Получение случайного факта о числе с Numbers API <http://numbersapi.com>

Используйте Numbers API для получения случайного факта о числе (например, число 42).

Выведите факт на экран.

3. Получение курсов валют с ExchangeRate-API <https://www.exchangerate-api.com/>

Используйте ExchangeRate-API (без токена) для получения курсов валют. Выведите курс доллара США к евро.

4. Получение случайного совета с Advice Slip API <https://api.adviceslip.com>

Используйте Advice Slip API для получения случайного совета. Выведите текст совета на экран.

5. Анекдоты с Chuck Norris API

Сделайте запрос к <https://api.chucknorris.io/jokes/random> и выведите текст шутки (value).

6. POST-запрос на JSONPlaceholder.

Отправьте POST-запрос к <https://jsonplaceholder.typicode.com/posts> с телом:

```
{"title": "Hello", "body": "This is a test post", "userId": 1}
```

Выведите статус-код и ID созданного поста (из ответа сервера).

Для всех заданий выводите статус-код ответа (200, 404 и т.д.).

Добавьте обработку ошибок (try/except) для случаев, если API недоступно.

Усложнение: сохраните результаты запросов в файл (posts.json, weather.txt и т.д.).

### Вариант 2

### 1. Сравнение силы двух супергероев.

Используйте Superhero API для сравнения силы двух супергероев. Напишите программу, которая:

Выводит список всех супергероев.

Принимает от пользователя имена двух супергероев (например, Hulk и Thor).

Отправляет HTTP-запросы к API для получения данных о каждом супергерое.

Сравнивает их показатель strength (сила) из раздела powerstats.

Выводит на экран имя супергероя с большей силой.

Пример работы программы:

Введите имя первого супергероя: Hulk

Введите имя второго супергероя: Thor

Сравнение силы:

- Hulk: 80

- Thor: 100

Победитель: Thor

Для выполнения заданий используйте библиотеку requests для отправки HTTP-запросов.

Документация API доступна по ссылке: <https://akabab.github.io/superhero-api/api/>

### 2. Список национальностей по имени.

Используя API <https://api.nationalize.io/?name=alex>, определите, у какого имени (name=ваше\_имя) наибольшая вероятность национальности. Выведите страну (country\_id) и вероятность (probability).

### 3. Получение списка видов котов.

Сделай запрос к <https://api.thecatapi.com/v1/breeds> и выведи названия первых 5 пород кошек.

### 4. Информация о стране по названию.

Используй <https://restcountries.com/v3.1/name/Kazakhstan> для получения данных о стране. Выведи её столицу, площадь и население.

5. Используйте IP Geolocation API (<https://ip-api.com/>) для определения географического положения по IP-адресу. Создайте программу, которая принимает IP-адрес и выводит информацию о стране, городе, регионе, почтовом индексе и часовом поясе.

### 6. Отправка данных формы на тестовый API

Используйте тестовый API <https://httpbin.org/post>, который возвращает отправленные данные.

Отправьте POST-запрос с данными формы (ключ-значение):

username — ваш никнейм (например, "python\_student"),

message — произвольный текст (например, "Hello API!").

Выведите:

Статус-код ответа,

Ответ сервера в формате JSON (поле form с переданными данными).

Для всех заданий выводите статус-код ответа (200, 404 и т.д.).

Добавьте обработку ошибок (try/except) для случаев, если API недоступно.

Усложнение: сохраните результаты запросов в файл (posts.json, weather.txt и т.д.).

Дополнительные задания:

#### 1. Яндекс API.

У Яндекс.Диска есть удобное и простое API. Для описания всех его методов существует [Полигон](#). Нужно написать программу, которая принимает на вход путь до файла на компьютере и сохраняет на Яндекс.Диск с таким же именем.

1. Все ответы приходят в формате json;
2. Загрузка файла по ссылке происходит с помощью метода put и передачи туда данных;
3. Токен можно получить, кликнув на полигоне на кнопку "Получить OAuth-токен".

Важно: Токен публиковать не нужно, переменную для токена нужно оставить пустой!  
Шаблон для программы

```
1 class YaUploader:
2     def __init__(self, token: str):
3         self.token = token
4
5     def upload(self, file_path: str):
6         """Метод загружает файлы по списку file_list на яндекс диск"""
7         # Тут ваша логика
8         # Функция может ничего не возвращать
9
10
11 if __name__ == '__main__':
12     # Получить путь к загружаемому файлу и токен от пользователя
13     path_to_file = ...
14     token = ...
15     uploader = YaUploader(token)
16     result = uploader.upload(path_to_file)
17
```

## 2. stackoverflow API.

Самый важный сайт для программистов это [stackoverflow](https://stackoverflow.com). И у него тоже есть [API](#) Нужно написать программу, которая выводит все вопросы за последние два дня и содержит тэг 'Python'. Для этого задания токен не требуется.

### Отчет должен содержать (см. образец):

- номер и тему лабораторной работы;
- фамилию, номер группы студента и вариант задания;
- скриншоты с исходным кодом программ и комментариями;
- скриншоты с результатами выполнения программ;
- пояснения, если необходимо;
- выводы.

Отчеты в формате **pdf** отправлять на email: **colledge20education23@gmail.com**