



Лабораторные работы по курсу:
«Разработка Интернет Приложений»

Введение в Python

Исполнитель:
Студентка группы ИУ5-52

Корякин Д.А.

Преподаватель:

Гапанюк Ю.Е,

«___» _____



Лабораторная работа 3

Цель: Знакомство с модулями, ООП в Python а также сетью и api vk

ЛИСТИНГ:

MyClients.py

```
from BaseClient import BaseClient
from datetime import datetime
import re
import sys

class UserIdClient(BaseClient):

    def __init__(self, username):
        super().__init__(username)

    method = "users.get"

    def response_handler(self, response):
        response_json = response.json()
        uid = 0
        try:
            uid = response_json.get("response")[0].get("uid")
        except (KeyError, TypeError, IndexError):
            print("No such user")
            sys.exit(-1)
        if uid == 0:
            print("Значение id не было получено")
            sys.exit(-1)
        return uid

    def get_params(self):
        return {"user_ids": self.username}

class FriendsBirthdaysClient(BaseClient):

    method = "friends.get"

    def __init__(self, uid):
        super().__init__(uid)

    def response_handler(self, response):
        response_json = response.json()
        return self.get_ages(response_json)

    def get_params(self):
        return {"user_id": self.username, "fields": "bdate"}

    @staticmethod
    def get_ages(response_json):
        ages = []
        today = datetime.today()
        pattern = re.compile("^\\d{1,2}\\d{1,2}\\d{4}$")
        try:
            users = response_json.get("response", [])
            for user in users:
                birthday_date = user.get("bdate")
                if birthday_date:
                    if pattern.search(birthday_date):
                        date = datetime.strptime(birthday_date, "%d.%m.%Y")
                        ages.append(int(round((today - date).days / 365, 0)))
        except (KeyError, TypeError):
            print("Incorrect json response")
        if len(ages) == 0:
            print("Нет данных о днях рождения")
            sys.exit(-1)
        return ages
```

BaseClient.py

```
import requests
import sys

class BaseClient:
    username = ""

    def __init__(self, username):
        self.username = username

    # URL vk api
    BASE_URL = "https://api.vk.com/method/"
    # метод vk api
    method = ""

    # Получение GET параметров запроса
    def get_params(self):
        return None

    # Склейка url
    def generate_url(self, method):
        url = '{0}{1}'.format(self.BASE_URL, method)
        return url

    # Отправка запроса к VK API
    def _get_data(self, method):
        response = None
        try:
            response = requests.get(self.generate_url(self.method), params=self.get_params())
        except requests.RequestException as re:
            print("Ошибка при выполнении запроса ", re)
            sys.exit(-1)
        if response is None:
            print("Пустой ответ")
            sys.exit(-1)
        return self.response_handler(response)

    # Обработка ответа от VK API
    def response_handler(self, response):
        print(response.text)
        return response

    # Запуск клиента
    def execute(self):
        return self._get_data(self.method)
```

ProcessAges.py

```
import matplotlib.pyplot as plt
import matplotlib as mpl
from tkinter import *
```

```
def print_gist(ages):
    fig = plt.figure("Гистограмма")
    plt.hist(ages)
    plt.title("Распределение возрастов")
    plt.ylabel("Количество людей, шт")
    plt.xlabel("Возраст, лет")
    plt.grid(True)
    plt.show()
```

Main.py

```
from MyClients import UserIdClient
from MyClients import FriendsBirthdaysClient
from ProcessAges import print_gist
```

```
def main():
    print("Введите username:", end=" ")
    user_id = input()
    uid = UserIdClient(user_id).execute()
    print("User id: ", uid)

    friends_ages = FriendsBirthdaysClient(uid).execute()

    print_gist(friends_ages)
```

```
if __name__ == "__main__":
    main()
```