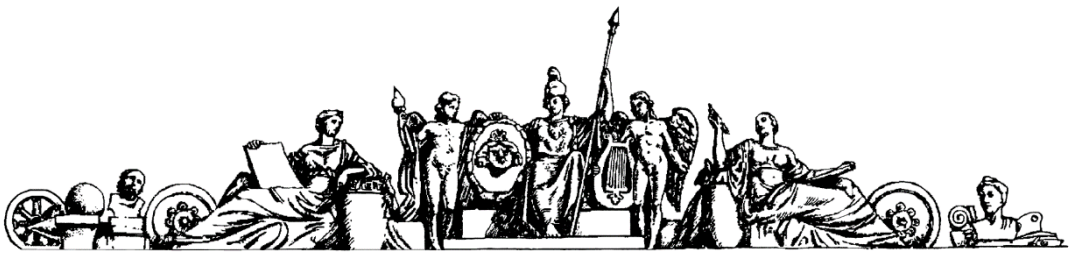


Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ им. Н. Э. БАУМАНА

Лабораторной работы по курсу:

«Разработка Интернет Приложений»

ЛР4. Python. Функциональные возможности

Выполнил: Житенев В.Г.

Группа РТ5-41

Преподаватель: Гапанюк Ю.Е.

Москва 2017 г.

Цель работы:

В этой лабораторной работе необходимо познакомиться с модулями и ООП в Python, а также освоить работу с сетью. Кроме того, необходимо создать набор классов для реализации работы с VK API.

Листинг:

baseclient.py

```
import requests


class BaseClient:
    # URL vk api
    BASE_URL = "https://api.vk.com/method/"
    # метод vk api
    method = None
    # GET, POST, ...
    http_method = None

    def __init__(self):
        # Инициализация
        self.success = True

    def get_params(self):
        return None

    def get_json(self):
        # Получение данных POST запроса
        return None

    def _get_data(self, method, http_method):
        # Отправка запроса к VK API
        try:
            response = requests.get(self.BASE_URL + self.method + "." +
self.http_method, params=self.get_params())
            print(self.BASE_URL + self.method + "." + self.http_method)

        except Exception:
            raise SystemExit("Нет ответа от VK")

        return self.response_handler(response)

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

    def is_success(self):
        # Проверка найден ли ID
        return self.success

    def execute(self):
        # Запуск клиента
        try:
            self.success = True
            return self._get_data(
                self.method,
                http_method=self.http_method
            )
        except Exception:
```

```
            #print('Ошибка')
            self.success = False
```

clientVK.py

```
import base_client
from datetime import datetime


class ClientGetID(base_client.BaseClient):
    # метод vk api
    method = "users"
    # GET, POST, ...
    http_method = "get"

    def __init__(self, username):
        # Инициализация
        super().__init__()
        self.json_data = None
        self.username = username

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

    def response_handler(self, response):
        # Получение ID пользователя
        self.json_data = response.json()
        print(self.json_data)
        # print("Json data = ", self.json data)
        return self.json_data["response"][0]["uid"]

    def get_json(self):
        # Получить json строку
        return self.json_data


def calculate_age(born, today):
    # Вычисление возраста
    return today.year - born.year - ((today.month, today.day) < (born.month,
born.day))


class ClientGetFriendsAges(base_client.BaseClient):
    # метод vk api
    method = "friends"
    # GET, POST, ...
    http_method = "get"

    def __init__(self, user_id):
        # Инициализация
        super().__init__()
        self.json_data = None
        self.user_id = user_id

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

    def response_handler(self, response):
        # Получение списка возрастов
```

```
        self.json_data = response.json()
        ages = list()
        today = datetime.utcnow()

        for friend in self.json_data["response"]:
            date_of_birth = friend.get("bdate")
            print (date_of_birth)

            try:
                date_of_birth = datetime.strptime(date_of_birth, "%d.%m.%Y")
            except (ValueError, TypeError):
                print("Ошибка формата даты")
                date_of_birth = datetime.today()
                # continue

            ages.append(calculate_age(date_of_birth, today))

        return ages

    def get_json(self):
        return self.json_data()
```

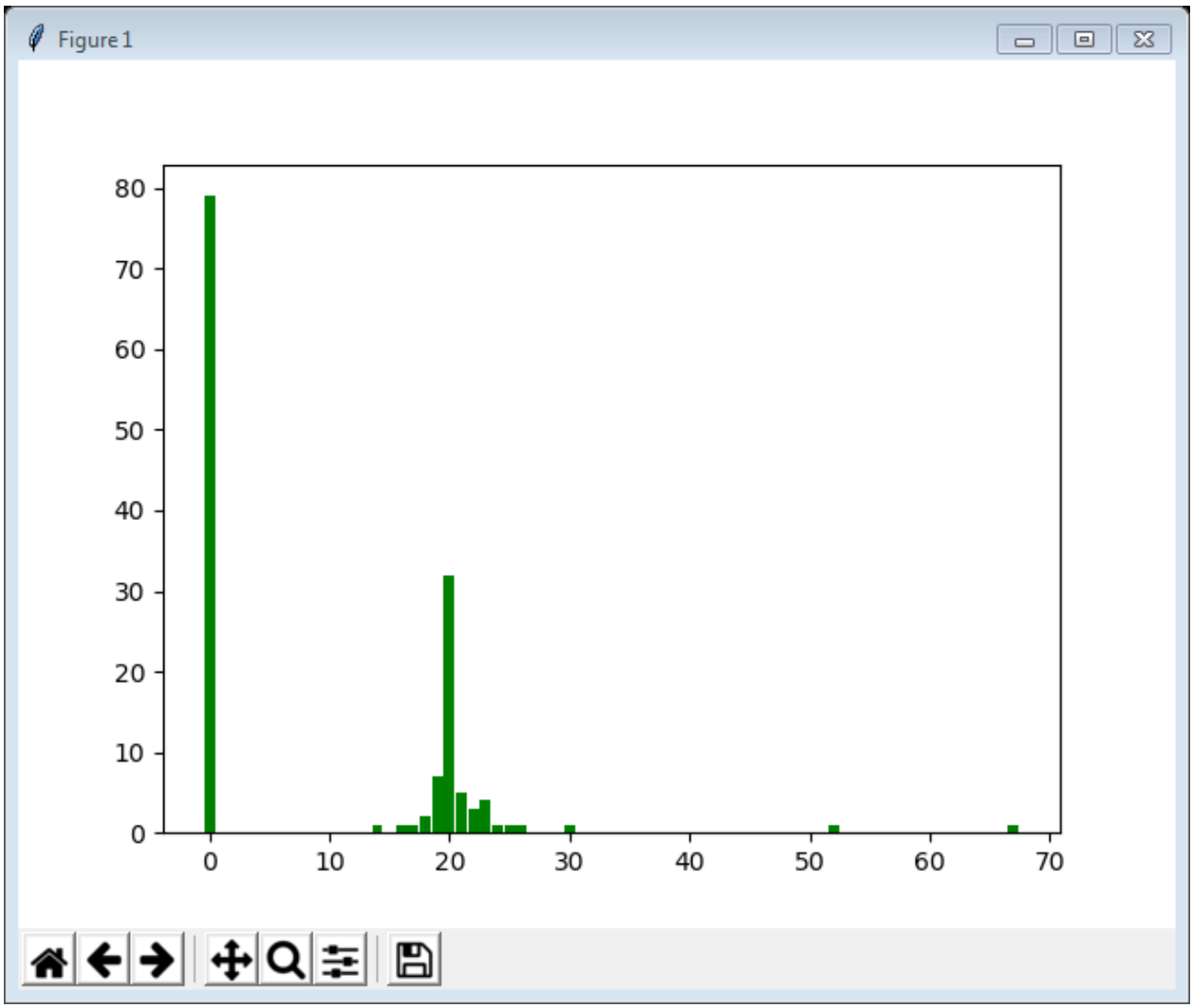
main.py

```
from clientVK import *
from histogram import Hist
import matplotlib.pyplot as plt


def main():
    username = input('Введите логин VK: ')

    client_get_id = ClientGetID(username)
    user_id = client_get_id.execute()

    if client_get_id.is_success():
        print("ID: ", user_id)
    else:
        print('Не существует пользователя с таким ID.')
        return 0

    # find age list
    friends_ages_list = ClientGetFriendsAges(user_id).execute()
    if not friends_ages_list:
        print('Друзей не найдено')
        return 0
    else:
        print("Ages: ", friends_ages_list)
        # write gist
        username_friend_gist = Hist(friends_ages_list)
        username_friend_gist.printGist()

    # show gist
    title = "Гистограмма"
    title_x = "Возраст"
    title_y = "Количество друзей"
    username_friend_gist.showBar(title, title_x, title_y)


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

histogram.py

```
import matplotlib.pyplot as plt


class Hist:
    # данные гистаграммы
    _age_dictionary = dict()

    def __init__(self, age_list):
        self._ages_list = sorted(age_list)
        for value in self._ages_list:
            self._age_dictionary.update(
                {value: self._age_dictionary.get(value, 0) + 1})

    def get_data(self):
        return self._ages_list

    def printGist(self):
        for age, count in self._age_dictionary.items():  # dict.items
возвращает пары
            print(str(age).ljust(4) + ":" + "#" * count)

    def showBar(self, title1, title_x, title_y):
        plt.bar(list(self._age_dictionary.keys()),
                self._age_dictionary.values(), color='g', width=0.9,
linewidth=20)
        plt.show()
```

```
0   :#################################################################################
14  :#
16  :#
17  :#
18  :##
19  :#######
20  :################################
21  :#####
22  :###
23  :####
24  :#
25  :#
26  :#
30  :#
52  :#
67  :#
```



Ключевой особенностью программы является обработка пользователей с некорректной датой рождения на страницы в ВК (отсутствует год рождения или дата отсутствует полностью). Такие пользователи получают «нулевой» год рождения, как показано на гистограмме