Московский государственный технический университет им. Н.Э. Баумана Факультет «Информатика и системы управления»

Кафедра «Автоматизированные системы обработки информации и управления»



**Отчет по Лабораторной работе 3**

**По курсу**

**“Разработка интернет-приложений”**

Выполнил:

Колосков А. Н.

Студент группы ИУ5-51

Москва 2017

**Цели работы:** ознакомиться с модулями и ООП в Python, освоить работу с сетью, работу с VK API.

**Задание:**

Вход: username или vk\_id пользователя

Выход: Гистограмма распределения возрастов друзей пользователя, поступившего на вход

Пример:

Вход: reigning

Выход:

19 #

20 ##

21 ##

22 ####################

23 ################

24 ####

25 #

28 #

29 #

30 #

37 #

38 ##

45 #

**Код программы:**

Модуль Base\_class.py

class BaseClient:  
 # URL vk api  
  
 BASE\_URL = None  
  
 # метод vk api  
  
 method = None  
  
 # GET, POST, ...  
  
 http\_method = None  
  
 # Получение GET параметров запроса  
  
 def get\_params(self):  
 return None  
  
 # Получение данных POST запроса  
  
 def get\_json(self):  
 return None  
  
 # Получение HTTP заголовков  
  
 def get\_headers(self):  
 return None  
  
 # Склейка url  
  
 def generate\_url(self, method):  
 return '{0}{1}'.format(self.BASE\_URL, method)  
  
 # Отправка запроса к VK API  
  
 def \_get\_data(self, method, http\_method):  
 response = None  
  
 # *todo выполнить запрос* return self.response\_handler(response)  
  
 # Обработка ответа от VK API  
  
 def response\_handler(self, response):  
 return response  
  
 # Запуск клиента  
  
 def execute(self):  
 return self.\_get\_data(  
  
 self.method,  
  
 self.http\_method  
  
 )

Модуль Heir\_classes.py:

import requests  
import Base\_class  
import datetime

import matplotlib.pyplot as plt

class GetVkUserInfo(Base\_class.BaseClient):  
 BASE\_URL = 'https://api.vk.com/method/'  
 method = 'users.get'  
 user\_id\_or\_username = str  
  
 def get\_params(self, user\_id\_or\_username):  
 return {'user\_ids': user\_id\_or\_username}  
  
 def \_get\_data(self, method, http\_method):  
 r = requests.get(self.generate\_url(self.method), params=self.get\_params(self.user\_id\_or\_username))  
 # print(r.url)  
 return self.response\_handler(r)  
  
 # Обработка ответа от VK API  
  
 def response\_handler(self, response):  
 print(response.json()['response'][0]['first\_name'])  
 print(response.json()['response'][0]['last\_name'])  
 return response.json()['response'][0]['uid']  
  
class GetVkFriendsInfo(Base\_class.BaseClient):  
 BASE\_URL = 'https://api.vk.com/method/'  
 method = 'friends.get'  
 user\_id = str  
  
 def get\_params(self, user\_id):  
 return {'user\_id': user\_id, 'fields': 'bdate', }  
  
 def \_get\_data(self, method, http\_method):  
 r = requests.get(self.generate\_url(self.method), params=self.get\_params(self.user\_id))  
 return self.response\_handler(r)  
  
 # Обработка ответа от VK API  
  
 def response\_handler(self, response):  
 mas\_ages = []  
 count\_of\_friends = len(response.json()['response'])  
 print(count\_of\_friends, ' friends')  
 for i in range(count\_of\_friends):  
 try:  
 bd = response.json()['response'][i]['bdate']  
 if len(bd) >= 8:  
 ageindays = datetime.datetime.today()-datetime.datetime.strptime(bd, '%d.%m.%Y')  
 mas\_ages.append(int(ageindays.days/365))  
 except:  
 pass  
 # print(mas\_ages)  
 setofages = list(set(mas\_ages))  
 setofages.sort()  
 for el in setofages:  
 count = 0  
 for i in mas\_ages:  
 if i == el:  
 count += 1  
 print(el, ' ', '|'\*count)  
 n, bins, patches = plt.hist(mas\_ages, 50, facecolor='g')  
 plt.xlabel('Возраст')  
 plt.ylabel('Количество')  
 plt.title('Гистограмма распределения')  
 plt.grid(True)  
 plt.show()  
 return response.json()

Модуль Main.py:

import Heir\_classes  
  
  
a = Heir\_classes.GetVkUserInfo()  
a.user\_id\_or\_username = 'carpediemmm'  
b = Heir\_classes.GetVkFriendsInfo()  
b.user\_id = a.execute()  
b.execute()

**Результаты выполнения:**

 