Московский государственный технический университет им. Н.Э.Баумана

|  |  |  |
| --- | --- | --- |
| Защищено:  Гапанюк Ю. Е.  "\_\_"\_\_\_\_\_\_\_\_\_\_\_\_\_201**6** г. |  | Демонстрация ЛР:  Гапанюк Ю. Е.  "\_\_"\_\_\_\_\_\_\_\_\_\_\_\_\_201**6** г. |

**Отчет по лабораторной работе № 3 по курсу**

**Разработка интернет-приложений**

5

(количество листов)

|  |  |
| --- | --- |
| ИСПОЛНИТЕЛЬ: |  |
| студентка группы **ИУ5-51** | \_\_\_\_\_\_\_\_\_\_\_ |
|  | (подпись) |
| **Мишенёва Е.Ю.** | "\_\_"\_\_\_\_\_\_\_\_\_\_\_201**6** г. |

Москва, МГТУ - 201**6**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**СОДЕРЖАНИЕ**

[1. Цель лабораторной работы 3](file:///E:\Универ\4_семестр\Системное_программирование\Отчеты\Отчет_лаб_1\Отчет_лаб_1.doc#_Toc441916406)

[2. Задача лабораторной работы](file:///E:\Универ\4_семестр\Системное_программирование\Отчеты\Отчет_лаб_1\Отчет_лаб_1.doc#_Toc441916407) 3

[3. Файлы проекта 3](file:///E:\Универ\4_семестр\Системное_программирование\Отчеты\Отчет_лаб_1\Отчет_лаб_1.doc#_Toc441916408)

[4. Результат 5](file:///E:\Универ\4_семестр\Системное_программирование\Отчеты\Отчет_лаб_1\Отчет_лаб_1.doc#_Toc441916409)

# Цель лабораторной работы

# Знакомство с модулями и ООП в Python, а также работа с сетью. В лабораторной работе необходимо создать набор классов для реализации работы с VK API VK API.

# Задание лабораторной работы

***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 #

1. **Файлы проекта**

base.py

import requests

class BaseClient:

BASE\_URL = "https://api.vk.com/method/"

method = None

http\_method = None

def get\_params(self):

pass

def get\_json(self):

pass

def get\_headers(self):

pass

def generate\_url(self, method):

return '{0}{1}'.format(self.BASE\_URL, method)

def \_get\_data(self, method, http\_method):

response = None

response = requests.get(self.generate\_url(method), params=self.get\_params())

return self.response\_handler(response)

def response\_handler(self, response):

return response

def execute(self):

return self.\_get\_data(

self.method,

http\_method=self.http\_method

)

methods.py

from base import BaseClient

class UserGet(BaseClient):

BASE\_URL = "https://api.vk.com/method/"

method = "users.get"

def \_\_init\_\_(self, username):

self.username = username

def get\_params(self):

return{

"user\_ids": self.username

}

def response\_handler(self, response):

b = response.json()

return b['response'][0]['uid']

class FriendsGet(BaseClient):

BASE\_URL = "https://api.vk.com/method/"

method = "friends.get"

def \_\_init\_\_(self, username):

self.username = username

def get\_params(self):

return{

"user\_id": self.username,

"fields": "bdate"

}

def response\_handler(self, response):

data = response.json()

res = {}

for i in data['response']:

try:

if i["bdate"].split('.')[2] in res.keys():

res[i["bdate"].split('.')[2]] += 1

else:

res[i["bdate"].split('.')[2]] = 1

except:

pass

return res

main.py

from base import BaseClient

from methods import FriendsGet

from methods import UserGet

r = UserGet('178981827')

b = r.execute()

print(b)

f = FriendsGet(b);

a = f.execute()

print(a)

list\_date = sorted(list(a.keys()), reverse=True)

for i in list\_date:

print(2016 - int(i), end=": ")

print("#" \* int(a[i]), end="")

print()

# Результат

