**Московский государственный технический университет им. Н.Э. Баумана.**

### Факультет «Информатика и системы управления»

### Кафедра ИУ5. Курс «Разработка интернет приложений»

### Отчет по лабораторной работе №2

**«Python-классы»**

|  |  |
| --- | --- |
| Выполнила: | Проверил: |
| студентка группы ИУ5-52 | доцент каф. ИУ5 |
| Бекбусинова Б. Б. | Гапанюк Ю. Е. |
| Подпись и дата: | Подпись и дата: |

### Москва, 2017 г.

#### Оглавление

[Исходный код 3](#_bookmark0)

[base\_client.py 3](#_bookmark1)

[lab\_classes.py 4](#_bookmark2)

[app.py 6](#_bookmark3)

[Результаты работы 7](#_bookmark4)

# Исходный код

## *base\_client.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, http\_method=self.http\_method

)

## *lab\_classes.py*

from base\_client import \* import requests

import json import datetime

class VK\_User(BaseClient): def init (self, username):

self.name = username

BASE\_URL = 'https://api.vk.com/method/' method = 'users.get'

http\_method = 'GET'

# Отправка запроса к VK API

def \_get\_data(self, method, http\_method): url=self.generate\_url(method)

response = requests.get(url=url+'?user\_ids='+self.name) return self.response\_handler(response)

def response\_handler(self, response): js = json.loads(response.text)

try:

return js['response'][0]['uid'] except:

raise Exception('There\'s no such user')

class VK\_Friends(BaseClient): def init (self, user\_id):

self.id = user\_id

BASE\_URL = 'https://api.vk.com/method/' method = 'friends.get'

http\_method = 'GET'

# Отправка запроса к VK API

def \_get\_data(self, method, http\_method): url=self.generate\_url(method)

response = requests.get(url=url+'?user\_id='+str(self.id)+'&fields=bdate') return self.response\_handler(response)

def response\_handler(self, response): js = json.loads(response.text)

try:

dates = []

for friend in js['response']: try:

dates.append(datetime.datetime.strptime(friend['bdate'],"%d.%m.%Y").date()) except:

pass

ages = [int((datetime.date.today()-date).days/365.2425) for date in dates] return ages

except KeyError:

return KeyError(js)

## *app.py*

from lab\_classes import \* from matplotlib import pylab

#user\_id = VK\_User('56337756').execute() name = input('Type username or id: ') print()

user\_id = None try:

user\_id = VK\_User(name).execute() except Exception as e:

print(e) quit(1)

ages= None try:

ages = VK\_Friends(user\_id).execute() except Exception as e:

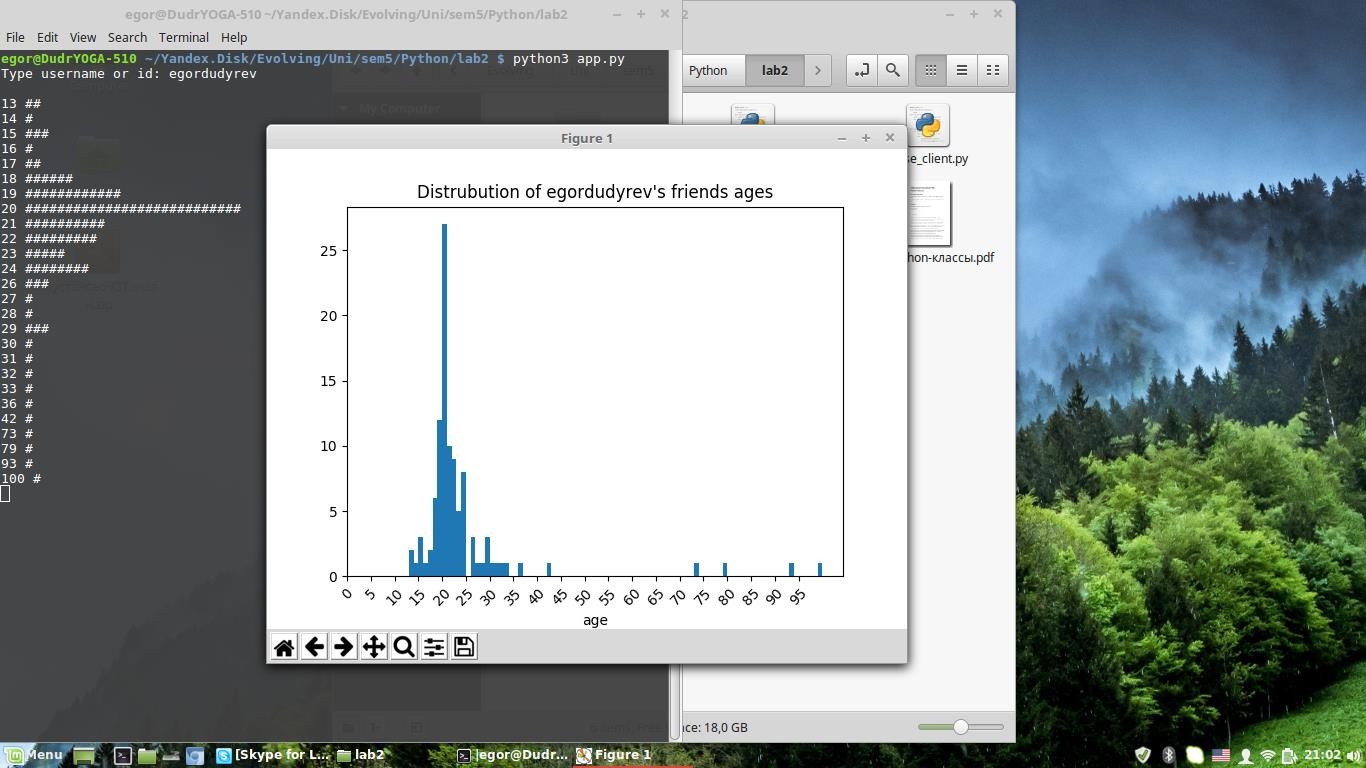
print(e) quit(1)

hist = [(x,ages.count(x)) for x in set(ages)] hist.sort(key=lambda x: x[0])

for age, number in hist: print(age,'#'\*number)

pylab.hist(ages,bins=((max(ages)-min(ages)))) pylab.xlabel('age') pylab.xticks(range(0,max(ages),5),rotation=45) pylab.title('Distrubution of {}\'s friends ages'.format(name)) pylab.show()

# Результаты работы

****

