

**Отчет по лабораторной работе №3
по курсу «Разработка Интернет-приложений»**

Вариант № <21>

ИСПОЛНИТЕЛЬ:
студент группы ИУ5-52
Ромичева Е.В.
09.09.2016 г.

Содержание

Описание задания лабораторной работы	3
Исходный код	3
Результат работы	5

Описание задания лабораторной работы

В этой ЛР Вы знакомитесь с модулями и ООП в Python, а также осваиваете работу с сетью. В лабораторной работе необходимо создать набор классов для реализации работ с VK API. Построить гистограмму распределения возрастов друзей пользователя ВКонтакте.

Исходный код

basecl.py

```
import requests
class BaseClient:
    BASE_URL = None

    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())
        # todo выполнить запрос

        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
        )
```

heircl.py

```
import requests
from basecl import BaseClient
```

```

from datetime import date
from _overlapped import NULL

def get_age(birthday):
    today = date.today()
    age = today.year - int(birthday[2],base=10)
    if today.month < int(birthday[1],base=10):
        age -= 1
    elif today.month == int(birthday[2],base=10) and today.day <
int(birthday[0], base=10):
        age -= 1
    return age

class GetUser (BaseClient):
    BASE_URL = 'https://api.vk.com/method/'
    method = 'users.get'

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

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

class GetFriends (BaseClient):
    BASE_URL = 'https://api.vk.com/method/'
    method = 'friends.get'

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

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

```

main.py

```

import sys
from collections import Counter
from basecl import BaseClient
import requests
from heirc1 import get_age, GetFriends, GetUser

import matplotlib.pyplot as plt
import numpy as np

# print ('Введите id')
# cust_id=input()

```

```

#a=GetUser(cust_id)

a=GetUser('reigning')
b=a.execute()
us1 = b.json()
#print(us1)
us_id=us1['response'][0]['uid']
print(us_id)
j=0
res=[]
friends_list = GetFriends(us_id)
bb=friends_list.execute()
friends=bb.json() #массив словарей
for i in range(len(friends['response'])):
    if ('bdate' in friends['response'][i] and
len(friends['response'][i]['bdate'].split('.')==3):

res.append(get_age(friends['response'][i]['bdate'].split('.')))

res1=Counter(res)
#print(res1)
data1=res1
ages = []
for i in list(res1):
    ages.append(res1[i])

locs = ages
plt.grid()
plt.minorticks_on()
plt.axis([0, 120, 0, 50])
plt.figure(num=1, figsize=(8, 6))
plt.xlabel('age', size=14)
plt.ylabel('count', size=14)
plt.bar(data1, locs, width=0.5)
plt.show()

```

Результат работы

