

Демонстрация ЛР:  
Гапанюк. Ю.Е.

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

**"Классы"**

ИСПОЛНИТЕЛЬ:

студент группы **ИУ5-54**

**Наседкин И.А.**

## Задание ЛР

### Вход:

username или vk\_id пользователь : Содержит сущности

### 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 #
```

## Текст программы Main.py

```
import Der

user = Der.DerivID ("http://api.vk.com/method","users.get")
user.params="user_ids="+input('Введите username пользователя: ')
resp_user=user.execute()
id=user.find_id(resp_user.text)
print("id: ",id)
friends=Der.DerivFriends ("http://api.vk.com/method","friends.get")
friends.params="user_id="+id+"&fields=bdate"
resp_friends=friends.execute()
friends.build_gist(resp_friends)
```

## Base.py

```
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

    # 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
    )

```

## Der.py

```
import
```

```
Base
```

```

from datetime import datetime
import requests

class DerivID (Base.BaseClient):
    params = None

    def __init__(self, u, m):
        self.BASE_URL = u
        self.method = m

    def generate_url(self, method):
        return '{0}/{1}'.format(self.BASE_URL, method)

    def _get_data(self, method, http_method):
        response = requests.get(self.generate_url(self.method), self.params)
        return self.response_handler(response)

def find_id(self, aa):
    id_mas = aa.get("response")
    id = id_mas[0].get("uid")
    return str(id)

class DerivFriends(Base.BaseClient):
    params = None

    def __init__(self, u, m):
        self.BASE_URL = u
        self.method = m

    def generate_url(self, method):
        return '{0}/{1}'.format(self.BASE_URL, method)

    def _get_data(self, method, http_method):
        response = requests.get(self.generate_url(self.method), self.params)
        return self.response_handler(response)

    def build_gist(self, ss):
        a = ss.json()
        dict_age = {0: 0}
        for bdate in a.get("response"):

```

```

date_str = str(bdate.get("bdate"))
if (not (bdate.get("bdate")) == None) and (len(date_str) > 7):
    dt = datetime.strptime(date_str, "%d.%m.%Y")
    delta_date = str(datetime.now() - dt)
    i = int(delta_date.split()[0])
    val = i // 365
    if dict_age.get(val) == None:
        dict_age[val] = '#'
    else:
        dict_age[val] = str(dict_age.get(val)) + '#'
else:
    dict_age[0] = int(dict_age.get(0)) + 1
self.print_gist(dict_age)
def print_gist(self, dict_age):
    print()
    sum = 0
    for key in sorted(dict_age):
        if key == 0:
            sum += int(dict_age[0])
            print("Нет указан возраст: ", dict_age[0])
        else:
            sum += int(len(str((dict_age[key]))))
            print("%3d %s" % (key, dict_age[key]))
    print("Друзья: ", sum)

```

## Результаты работы программы

```

Run Main
Нет возраста: 80
18 **
20 *****
21 *****
22 ****
23 ****
24 *
27 **
31 *
34 *
36 *
38 *
51 *
89 *
92 *
96 *
110 *
Всего друзей: 131
Process finished with exit code 0

```