Отчет по лабораторной работе №3

По курсу «Разработка интернет-приложений»

«Python­классы»

|  |  |  |
| --- | --- | --- |
| Выполнил: |  | Преподаватель: |
| Марков Артем, ИУ5-51 |  | Гапанюк Ю.Е. |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
|  |  |  |

2016 г.

1. Задание лабораторной работы.

Вход:

username или vk\_id пользователя

Выход:

Гистограмма распределения возрастов друзей пользователя, поступившего на вход

Пример:

Вход:

Godlik96

Выход:

14: #

18: ##

19: #####

20: ###########################

21: #######

22: #

25: #

26: ##

27: #

32: #

44: #

49: #

96: #

2) Листинг

**test.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) *#(self.BASE\_URL=https://api.vk.com/method/)* print()  
 **def** \_get\_data(self, method, http\_method):  
 *#r=requests.get(self.generate\_url(method), params=self.get\_params())  
 #print(r)* response = requests.get(self.generate\_url(method), params=self.get\_params())  
 *# print(requests.get(self.generate\_url(method), params=self.get\_params()))  
 #print(self.response\_handler(response))* **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** test **import** BaseClient  
**class** MyBaseClient(BaseClient):  
 BASE\_URL = **"https://api.vk.com/method/"** *# метод vk api* 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()  
 *#print(data)* res = {}  
 **for** i **in** data[**'response'**]:  
 **try**:  
 **if** i[**"bdate"**].split(**'.'**)[2] **in** res.keys():  
 res[i[**"bdate"**].split(**'.'**)[2]] += 1  
 *#print(i['first\_name'], i['last\_name'], i["bdate"])* **else**:  
 res[i[**"bdate"**].split(**'.'**)[2]] = 1  
 *#print('! ',i['first\_name'], i['last\_name'], i["bdate"])* **except**:  
 **pass  
 return** res

**main.py**

**import** numpy **as** np  
**import** matplotlib.mlab **as** mlab  
**import** matplotlib.pyplot **as** plt  
**from** test **import** BaseClient  
**from** methods **import** FriendsGet  
**from** methods **import** MyBaseClient  
  
r = MyBaseClient(**'Godlik96'**)  
b = r.execute()  
print(b,**'!'**)  
  
f = FriendsGet(b);  
a = f.execute()  
print(a)  
arr1=[]  
arr2=[]  
list\_date = sorted(list(a.keys()), reverse=**True**)  
**for** i **in** list\_date:  
 arr2.append(int(a[i]))  
 arr1.append(2016-int(i))  
 x=2016-int(i)  
 print(x, end=**": "**)  
 print(**"#"** \* int(a[i]), end=**""**)  
 print()  
  
  
plt.bar(arr1, arr2,color=**"Green"**)  
plt.xlim(min(arr1) - 2, max(arr1) + 3)  
plt.ylim(0, max(arr2) + 2)  
plt.xlabel(**'Age'**, fontsize=24, color=**'Green'**)  
plt.ylabel(**'Number of friends'**, fontsize=24, color=**'Green'**)  
plt.show()



