

CD Lab Practical 8

Name: Gaurav Kedia

Roll No: 39

Batch: E-2

TASK A Consider the text file with following URL & perform the following operation using Regular Expression url = <https://www.gutenberg.org/files/2638/2638-0.txt>

```
import re
import requests
the_idiot_url = 'https://www.gutenberg.org/files/2638/2638-0.txt'

def get_book(url):
    # Sends a http request to get the text from project Gutenberg
    raw = requests.get(url).text
    # Discards the metadata from the beginning of the book
    start = re.search(r"\*\*\* START OF THIS PROJECT GUTENBERG EBOOK .* \*\*\*", raw).end()
    # Discards the metadata from the end of the book
    stop = re.search(r"II", raw).start()
    # Keeps the relevant text
    text = raw[start:stop]
    return text

def preprocess(sentence):
    return re.sub('[^A-Za-z0-9.]+' , '' , sentence)

book = get_book(the_idiot_url)
processed_book = preprocess(book)
print(processed_book)

# Q. 1 Find the number of the pronoun "the" in the corpus. Hint: Use the len() function.
len(re.findall(r'the', processed_book))

302

#Try to convert every single stand-alone instance of 'i' to 'I' in the corpus. Make sure not to change the 'i'
# occurring within a word:
processed_book = re.sub(r'\si\s', " I ", processed_book)
print(processed_book)

# Find the number of times anyone was quoted (") in the corpus.
len(re.findall(r'\"', book))

96

#What are the words connected by '--' in the corpus?
re.findall(r'[a-zA-Z0-9]*--[a-zA-Z0-9]*', book)

['ironical--it', 'malicious--smile', 'fur--or', 'astrachan--overcoat', 'it--the', 'Italy--was', 'malady--a',

#Find the numbers available in the text.
```

Phone Number Verification:

```
# Problem Statement - The need to easily verify phone numbers in any relevant scenario.
# Consider the following Phone numbers:
# 444-122-1234
# 123-122-78999
# 111-123-23
# 67-7890-2019
import re
phn = "412-555-1212"
if re.search("w{3}-w{3}-w{4}", phn):
    print("Valid phone number")
```

▼ E-mail Verification:

```
# Problem statement - To verify the validity of an E-mail address in any scenario.
# Consider the following examples of email addresses:
# Anirudh@gmail.com
# Anirudh @ com
# AC .com
# 123 @.com
# All E-mail addresses should include:
# 1 to 20 lowercase and/or uppercase letters, numbers, plus . _ % +
# An @ symbol
# 2 to 20 lowercase and uppercase letters, numbers and plus
# A period symbol
# 2 to 3 lowercase and uppercase letters
import re
email = "ac@aol.com md@.com @seo.com dc@.com"
print("Email Matches: ", len(re.findall("[w._%+-]{1,20}@[w.-]{2,20}.[A-Za-z]{2,3}", email)))

Email Matches: 0
```

▼ Password Verification:

```
# Write a Python program to check the validity of a password using Regular expression.
# Validation Rules:
# At least 1 letter between [a-z A-Z].
# At least 1 number between [0-9].
# At least 1 character from [&#@].
# Minimum length 6 characters.
import re
p= input("Input your password")
x = True
while x:
    if (len(p)<6):
        break
    elif not re.search("[a-z]",p):
        break
    elif not re.search("[0-9]",p):
        break
    elif not re.search("[A-Z]",p):
        break
    elif not re.search("[$#@]",p):
        break
    elif re.search("\s",p):
        break
    else:
        print("Valid Password")
        x=False
        break
if x:
    print("Not a Valid Password")

Valid Password
```

▼ Task C

```
# Problem Statement - Scrapping all of the phone numbers from a website for a requirement by making use of
# Python Regular Expressions & save it in CSV/ list
# Website URL: http://www.summet.com/dmsi/html/codesamples/addresses.html
import requests
from bs4 import BeautifulSoup
import re
r=requests.get('https://www.summet.com/dmsi/html/codesamples/addresses.html')
c=r.content
soup = BeautifulSoup(c, 'html.parser')
main_content = soup.findAll('li')

content=str(main_content)
res=re.findall(r'[\+\\(\\)?[1-9][0-9] .\\-\\(\\)]{8,}[0-9]',content)
res
```



```
'(121) 347-0086',  
'(304) 506-6314',  
'(425) 288-2332',  
'(145) 987-4962',  
'(187) 582-9707',  
'(750) 558-3965',  
'(492) 467-3131',  
'(774) 914-2510',  
'(888) 106-8550',  
'(539) 567-3573',  
'(693) 337-2849',  
'(545) 604-9386',  
'(221) 156-5026',  
'(414) 876-0865',  
'(932) 726-8645',  
'(726) 710-9826',  
'(622) 594-1662',  
'(948) 600-8503',  
'(605) 900-7508',  
'(716) 977-5775',  
'(368) 239-8275',  
'(725) 342-0650',  
'(711) 993-5187',  
'(882) 399-5084',  
'(287) 755-9948',  
'(659) 551-3389',  
'(275) 730-6868',  
'(725) 757-4047',  
'(314) 882-1496',  
'(639) 360-7590',  
'(168) 222-1592',  
'(896) 303-1164',  
'(203) 982-6130',  
'(906) 217-1470',  
'(614) 514-1269',  
'(763) 409-5446',  
'(836) 292-5324',  
'(926) 709-3295',  
'(963) 356-9268',  
'(736) 522-8584',  
'(410) 483-0352',  
'(252) 204-1434',  
'(874) 886-4174',  
'(581) 379-7573',  
'(983) 632-8597',  
'(295) 983-3476',  
'(873) 392-8802',  
'(360) 669-3923',  
'(840) 987-9449',  
'(422) 517-6053',  
'(126) 940-2753',  
'(427) 930-5255',  
'(689) 721-5145',  
'(676) 334-2174',  
'(437) 994-5270',  
'(564) 908-6970',  
'(577) 333-6244',  
'(655) 840-6139']
```