Forward
School

Program Code: J620-002-4:2020

Program Name: FRONT-END SOFTWARE DEVELOPMENT

Title : Regular Expressions - Part 2

Name: Phua Yan Han

IC Number: 050824070059

Date : 3/7/23

Introduction : learning more regular expressions

Conclusion : learn more basic regex function

P12 - Regular Expressions - Part 2

Extract Email using Regex

In [1]:

```
import requests
import re

url='https://www.selangor.gov.my/index.php/pages/view/339'

# get the data
data = requests.get(url, verify=False)


emails = re.findall(r'([\d\w\.]+@[\d\w\.\-]+\.\w+)', data.text)

print(emails)
```

```
['cccsel@rmp.gov.my', 'pertanyaan@icu.gov.my', 'jpn.selangor@moe.gov.my', 'cp
rc_sel@moh.gov.my', 'hqweb@jupem.gov.my', 'p.selangor@jpj.gov.my', 'pinsel@im
i.gov.my', 'adminshahalam@jpn.gov.my', 'pensel@inform.gov.my', 'jbsselangor@g
mail.com', 'norisham@hasil.gov.my', 'sel@sprm.gov.my', 'prn_selangor@moha.go
v.my', 'penduduk@lppkn.gov.my', 'noraisyah@dof.gov.my', 'ppnselangor@perpadua
n.gov.my', 'kemas_selangor2@kemas.gov.my', 'zirawati@audit.gov.my', 'pneg_sgr
@anm.gov.my', 'p_wisma@marine.gov.my', 'haslinda@marine.gov.my', 'pro@jakoa.g
ov.my', 'm_anis@jakoa.gov.my', 'wpkl@prison.gov.my', 'm.zuhairi@kpdnhep.gov.m
y', 'korporat@bomba.gov.my', 'norazam.khamis@bomba.gov.my', 'webmaster@spr.go
v.my', 'selangor@wildlife.gov.my', 'webmaster@mkn.gov.my', 'assri_ramli@cgso.
gov.my', 'jmgselwp@jmg.gov.my', 'jhekssgor@mohr.gov.my', 'pmssubang@met.gov.m
y', 'ccc@customs.gov.my', 'zuraini.othman@customs.gov.my', 'jtknselangor@moh
r.gov.my', 'jppmsel@mohr.gov.my', 'info@jkkn.gov.my', 'jpselangor@dosm.gov.m
y', 'tini@dosm.giv.my', 'pengarah_selangor@adk.gov.my', 'jkkpsl@mohr.gov.my',
'ppwnselangor@jpw.gov.my', 'pro@civildefence.gov.my']

C:\Users\Asus\anaconda3\envs\python-dscourse\lib\site-packages\urllib3\connec
tionpool.py:1056: InsecureRequestWarning: Unverified HTTPS request is being m
ade to host 'www.selangor.gov.my'. Adding certificate verification is strongl
y advised. See: https://urllib3.readthedocs.io/en/1.26.x/advanced-usage.html#
ssl-warnings (https://urllib3.readthedocs.io/en/1.26.x/advanced-usage.html#ss
l-warnings)
  warnings.warn(
```

Extract Phone Number using Regex

In [2]:

```
import requests
import re

url='https://www.selangor.gov.my/index.php/pages/view/339'

# get the data
data = requests.get(url, verify=False)

phones = re.findall(r'[0-9]{2}\-[0-9]{8}', data.text)

print(phones)
```

```
['03-55145004', '03-55195175', '03-55213600', '03-55213700', '03-55186500',
'03-55102133', '03-51237333', '03-51237209', '03-55144000', '03-55132613', '0
3-55669555', '03-55432202', '03-55190653', '03-55107255', '03-55117355', '03-
55136755', '03-55192196', '03-55121411', '03-55192326', '03-55192231', '03-55
215200', '03-55103500', '03-55256500', '03-55256514', '03-55256515', '03-5510
3436', '03-55103436', '03-55184603', '03-55197825', '03-55107397', '03-551109
15', '03-55100575', '03-55190169', '03-55190690', '03-55132655', '03-5518461
7', '03-55184618', '03-55195114', '03-55195319', '03-55102376', '03-5519304
4', '03-55193175', '03-55199533', '03-55147400', '03-55147404', '03-5514740
5', '03-55205200', '03-55105049', '03-31695100', '03-31695190', '03-5519037
5', '03-55111063', '03-87328299', '03-55144300', '03-55195255', '03-7846444
4', '03-78469892', '03-55194273', '03-55400717', '03-55193915', '03-5510183
0', '03-55218790', '03-55218794', '03-55218791', '03-55447828', '03-5510970
5', '03-55101833', '03-55101918', '03-55193233', '03-55193551', '03-5519345
7', '03-55199059', '03-78463114', '03-78464982', '03-31693888', '03-3169360
0', '03-56328800', '03-56361605', '03-56361625', '03-56361573', '03-5636153
4', '03-56501600', '03-56361534', '03-55102664', '03-55102791', '03-5510283
9', '03-55150200', '03-55180408', '03-79568512', '03-79576396', '03-5623640
0', '03-56389159', '03-55118891', '03-55118706', '03-33411031', '03-3341050
6', '03-33410443', '03-33411894', '03-55447000']
```

```
D:\Anaconda3\envs\python-dscourse\lib\site-packages\urllib3\connectionpool.p
y:981: InsecureRequestWarning: Unverified HTTPS request is being made to host
'www.selangor.gov.my'. Adding certificate verification is strongly advised. S
ee: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings
(https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings)
  warnings.warn(
```

Quiz

Quiz 1

Extract the Emails and Phone numbers from this page using RegEx

https://www.ksn.gov.my/mengenai-pejabat/direktori-pejabat (https://www.ksn.gov.my/mengenai-pejabat/direktori-pejabat)

Quiz 2

Extract the Emails and Phone numbers from this page using RegEx

https://www.jpn.gov.my/my/hubungi-kami/direktori-kakitangan/bahagian-kewarganegaraan (https://www.jpn.gov.my/my/hubungi-kami/direktori-kakitangan/bahagian-kewarganegaraan)

Quiz 3

Extract the Emails and Phone Numbers from this page using RegEx

https://www.ptptn.gov.my/hubungi-ptptn (https://www.ptptn.gov.my/hubungi-ptptn)

In [22]:

```
# Method 1

import requests
import re

url='https://www.ksn.gov.my/mengenai-pejabat/direktori-pejabat'
# get the data
data = requests.get(url)


emails = re.findall(r'([\w]+\[at][\w]{3}\.[\w]{3}\.\w+)', data.text)

print(emails)
phones = re.findall(r'\+[0-9]{3}\-[0-9]{4} [0-9]{4}', data.text)

print(phones)
```

```
['zuki[at]jpm.gov.my', 'zuki[at]jpm.gov.my', 'haniff[at]jpm.gov.my', 'hasnah
[at]jpm.gov.my', 'akram[at]jpm.gov.my', 'zulamri[at]jpm.gov.my', 'ahmadnizam
[at]jpm.gov.my', 'izuan[at]jpm.gov.my', 'abduh[at]jpm.gov.my', 'hasif[at]jpm.
gov.my', 'hanisah[at]jpm.gov.my', 'bastamam[at]jpm.gov.my']
['+603-8872 7321', '+603-8872 7329', '+603-8872 7327', '+603-8872 7223', '+60
3-8872 7200', '+603-8872 7216', '+603-8872 7224', '+603-8872 7218', '+603-887
2 7211', '+603-8872 7321', '+603-8872 7212']
```

In [44]:

```python
# Method 2

import requests
import re

url='https://www.jpn.gov.my/my/hubungi-kami/direktori-kakitangan/bahagian-kewar
data = requests.get(url, verify=False)
phones = re.findall(r'[0-9]{2}\-[0-9 ]{9}', data.text)

print(phones)
# get the data
```

```
C:\Users\Asus\anaconda3\envs\python-dscourse\lib\site-packages\urllib3\connec
tionpool.py:1056: InsecureRequestWarning: Unverified HTTPS request is being m
ade to host 'www.jpn.gov.my'. Adding certificate verification is strongly adv
ised. See: https://urllib3.readthedocs.io/en/1.26.x/advanced-usage.html#ssl-w
arnings (https://urllib3.readthedocs.io/en/1.26.x/advanced-usage.html#ssl-war
nings)
  warnings.warn(
```

```
['03-8880 7023', '03-8880 8185', '03-8880 8179', '03-8880 8185', '03-8880 814
3', '03-8880 8185', '03-8880 7042', '03-8880 8185', '03-8880 8117', '03-8880
8185', '03-8000 8000', '03-8880 8288']
```

In [55]:
```
# Method 3

import requests
import re

url='https://www.ptptn.gov.my/hubungi-ptptn'
data = requests.get(url)
phones = re.findall(r'[0-9]{2}\-[0-9 ]{9}|[0-9]{2}\-[0-9 ]{8}|[0-9]{3}\-[0-9 ]{

print(phones)
# get the data
```

```
['03-21931177', '03-21931197', '011-51813747', '03-21931179', '03-21931184',
'011-21326228', '011-12327988', '011-10952566', '011-11567607', '011-3351892
2', '011-17911236', '011-14201982', '011-40224241', '011-14201829', '011-6201
6403', '011-14201832', '011-23718675', '011-12834055', '011-14202056', '011-3
3178276', '011-1140 576', '011-1420 191', '011-11181518', '011-27765184', '01
1-28932548', '03-55231630', '011-73449057', '011-14202017', '011-11913055',
'011-27282150', '011-21178900', '011-33091677', '011-25489744', '011-1420206
3', '011-23637492', '011-37030877', '011-11294543', '011-14202084', '011-3171
5313', '011-21106861', '011-12294739', '011-12379899', '011-37711539', '011-3
5065884', '011-28013600', '011-31929187', '011-12849088', '019-218 9132', '01
1-23625105', '011-31749310', '07-237 1088', '011-26421223', '011-26384286',
'011-28767644', '011-21349402', '012-265 7306', '011-74120538', '011-1070720
4', '011-17471051', '011-24047600', '011-51488755', '011-11318943', '05-801 2
398', '012- 4690670', '011-35134245', '04-226 2430', '011-14709963', '011-407
37367', '011-36306991', '011-54251470', '011-37846617', '011-12350288', '011-
88880690', '011-19909159', '011-14201934', '011-25519411', '011-16121236', '0
9-960 2800', '011-37533717', '011-31212003', '011-6363 300', '011-12171847',
'011-20668682', '013-365 2998', '011-23749448', '011-11163836', '011-2577947
5', '011-10069030', '011-31735788', '011-29993920', '011-16037479', '011-3144
9144', '011-33329492', '011-36016122', '011-31725631', '011-31236318']
```

In [39]:
```
# Answer to Exercise 2

import requests
import re

user_agent = 'user-agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit

headers={'User-Agent':user_agent}

url='https://www.jpn.gov.my/my/hubungi-kami/direktori-kakitangan/bahagian-kewar

# get the data
# disable SSL cert verification
```

In [9]:
```
data.text
```

Out[9]:
```
'<!DOCTYPE html>\n<html lang="ms-my" dir="ltr" vocab="http://schema.org/">
\n    <head>\n        <meta http-equiv="X-UA-Compatible" content="IE=edge">
\n        <meta name="viewport" content="width=device-width, initial-scale=
1">\n        <link rel="shortcut icon" href="/images/logojpn40px.png">\n
<link rel="apple-touch-icon" href="/images/logojpn40px.png">\n        <meta
charset="utf-8" />\n\t<base href="https://www.jpn.gov.my/my/hubungi-kami/di
rektori-kakitangan/bahagian-kewarganegaraan" />\n\t<meta name="description"
content="Portal Rasmi JPN" />\n\t<meta name="generator" content="XXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX" />\n\t<title>Portal JPN - Bahagian Kewargan
egaraan</title>\n\t<link href="/my/hubungi-kami/direktori-kakitangan/bahagi
an-kewarganegaraan?format=feed&amp;type=rss" rel="alternate" type="applicat
ion/rss+xml" title="RSS 2.0" />\n\t<link href="/my/hubungi-kami/direktori-k
akitangan/bahagian-kewarganegaraan?format=feed&amp;type=atom" rel="alternat
e" type="application/atom+xml" title="Atom 1.0" />\n\t<link href="https://w
ww.jpn.gov.my/my/hubungi-kami/direktori-kakitangan/bahagian-kewarganegaraa
n" rel="alternate" hreflang="ms-MY" />\n\t<link href="https://www.jpn.gov.m
y/en/contact-us/direktori-kakitangan/citizenship-division" rel="alternate"
hreflang="en-GB" />\n\t<link href="/plugins/content/pdf_embed/assets/css/st
yle.css" rel="stylesheet" />\n\t<link href="https://www.jpn.gov.my/modules/
```

In [37]:
```python
# Answer to Exercise 3

import requests
import re

url='https://www.ptptn.gov.my/hubungi-ptptn'

# get the data
```

```
D:\Anaconda3\envs\python-dscourse\lib\site-packages\urllib3\connectionpool.
py:981: InsecureRequestWarning: Unverified HTTPS request is being made to h
ost 'www.ptptn.gov.my'. Adding certificate verification is strongly advise
d. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-wa
rnings (https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-wa
rnings)
  warnings.warn(

['ahmadshahril@ptptn.gov.my', 'sarah@ptptn.gov.my', 'umminadira@ptptn.gov.m
y', 'nabillah@ptptn.gov.my', 'noorshaheera@ptptn.gov.my', 'fadzlina@ptptn.g
ov.my', 'nurulhusna@ptptn.gov.my', 'nurul_atiqah@ptptn.gov.my', 'nur_atikah
@ptptn.gov.my', 'nadhirah@ptptn.gov.my', 'noorehan@ptptn.gov.my', 'ismail.m
@ptptn.gov.my', 'norzaidah@ptptn.gov.my', 'farahanim_a@ptptn.gov.my', 'masl
iana@ptptn.gov.my', 'arif_h@ptptn.gov.my', 'kamarul_a@ptptn.gov.my', 'fazai
tum@ptptn.gov.my', 'wan_norasyikin@ptptn.gov.my', 'karthinee@ptptn.gov.my',
'izzasuria@ptptn.gov.my', 's.nurul@ptptn.gov.my', 'nursyafira@ptptn.gov.m
y', 'hafsoh@ptptn.gov.my', 'nuraishah@ptptn.gov.my', 'norhidayati@ptptn.go
v.my', 's.nurhaishah@ptptn.gov.my', 'roslan@ptptn.gov.my', 'haslida@ptptn.g
ov.my', 'md_ismail@ptptn.gov.my', 'amani@ptptn.gov.my', 'malissa@ptptn.gov.
my', 'm_shafik@ptptn.gov.my', 'fadzilah_m@ptptn.gov.my', 'norhasimah@ptptn.
```

In [ ]:
```
data.text
```

Need to use Selenium because of the Javascript on the page.

Discussion

Is regex good for scraping non regular texts from Web pages?

Eg. Look for all text between

- and

can work? yes but how usable?

Eg. Look for all names starting with Mr. and then extract the name. How?

We need a better way to extract the meta data and elements about the web page.

r'

[https://docs.python.org/3/library/re.html (https://docs.python.org/3/library/re.html)](https://docs.python.org/3/library/re.html)

```
Regular expressions use the backslash character ('\') to indicate spec
ial forms or to allow special characters to be used without invoking t
heir special meaning. This collides with Python’s usage of the same ch
aracter for the same purpose in string literals; for example, to match
a literal backslash, one might have to write '\\\\' as the pattern str
ing, because the regular expression must be \\, and each backslash mus
t be expressed as \\ inside a regular Python string literal. Also, ple
ase note that any invalid escape sequences in Python’s usage of the ba
ckslash in string literals now generate a DeprecationWarning and in th
e future this will become a SyntaxError. This behaviour will happen ev
en if it is a valid escape sequence for a regular expression.

The solution is to use Python’s raw string notation for regular expres
sion patterns; backslashes are not handled in any special way in a str
ing literal prefixed with 'r'. So r"\n" is a two-character string cont
aining '\' and 'n', while "\n" is a one-character string containing a
newline. Usually patterns will be expressed in Python code using this
raw string notation.
```

Note:

Alternative 3rd party regex implementation

[https://pypi.org/project/regex/ (https://pypi.org/project/regex/)](https://pypi.org/project/regex/)

Try it out own your own

In [15]:

```
# Example

teststring = 'this is \n a test'

print(teststring)
```

```
this is
 a test
```

In [16]:

```
# Example escape character \ is now a raw string not an escape char

teststring = r'this is \n a test'

print(teststring)
```

```
this is \n a test
```

In [18]:

```
# \b Word boundary, allow to perform "whole words only" search

import re

re.findall('\btest\b', 'test this is a test') # the backslash gets consumed by
```

Out[18]:

```
[]
```

In [19]:

```
re.findall('\\btest\\b', 'test this is a test') # backslash is explicitly escap
```

Out[19]:

```
['test', 'test']
```

In [14]:

```
re.findall(r'\btest\b', 'test this is a test') # often this syntax is easier
```

Out[14]:

```
['test', 'test']
```

In [21]:

```
# some example using regex to extract the URL

import re

x="""<!DOCTYPE html>

<html itemscope itemtype="http://schema.org/QAPage">

<head>
"""

matching = re.findall(r"[a-zA-Z0-9\-\.]+\.(?:com|org|net|mil|edu|COM|ORG|NET|MI

print(matching)
```

```
['schema.org/QAPage']
```