Forward School

Program Code: J620-002-4:2020

Program Name: FRONT-END SOFTWARE DEVELOPMENT

Title : Regular Expressions - Part 2

Name: Chong Mun Chen

IC Number: 960327-07-5097

Date : 3/7/2023

Introduction : Learning to Web Scrape with matching Regex.

Conclusion : Getting more familiar with the concept of Web Scraping and with Regex used.

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)
```

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

['cccsel@rmp.gov.my', 'pertanyaan@icu.gov.my', 'jpn.selangor@moe.gov.my', 'cprc_sel@moh.gov.my', 'hqweb@jupem.gov.my', 'p.selangor@jpj.gov.my', 'pinsel@imi.gov.my', 'pppselangor@jpn.gov.my', 'pensel@inform.gov.my', 'jbsselangor@gmail.com', 'kamarudin@hasil.gov.my', 'sel@sprm.gov.my', 'prn_selangor@moha.gov.my', 'webmaster@jpph.gov.my', 'penduduk@lppkn.gov.my', 'ppn_selangor@dof.gov.my', 'ppnselangor@perpaduan.gov.my', 'kemas_selangor2@kemas.gov.my', 'jbaudit@audit.gov.my', 'pneg_sgr@anm.gov.my', 'pjlwt@marine.gov.my', 'pro@jakoa.gov.my', 'wpkl@prison.gov.my', 'zikril@kpdnhep.gov.my', 'korporat@bomba.gov.my', 'webmaster@spr.gov.my', 'selangor@wildlife.gov.my', 'webmaster@mkn.gov.my', 'jmgselwp@jmg.gov.my', 'jhekssgor@mohr.gov.my', 'pmssubang@met.gov.my', 'ccc@customs.gov.my', 'jtknselangor@mohr.gov.my', 'jppmsel@mohr.gov.my', 'info@jkkn.gov.my', 'jpselangor@dosm.gov.my', 'pengarah_selangor@adk.gov.my', 'jkkpsl@mohr.gov.my', 'ppwnselangor@jpw.gov.my', 'pro@civildefence.gov.my']
```

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', '03-5518650
0', '03-55102133', '03-51237333', '03-51237209', '03-55144000', '03-55132
613', '03-55669555', '03-55432202', '03-55190653', '03-55107255', '03-551
17355', '03-55136755', '03-55192196', '03-55121411', '03-55192326', '03-5
5192231', '03-55215200', '03-55103500', '03-55256500', '03-55256514', '03
-55256515', '03-55103436', '03-55103436', '03-55184603', '03-55197825',
'03-55107397', '03-55110915', '03-55100575', '03-55190169', '03-5519069
0', '03-55132655', '03-55184617', '03-55184618', '03-55195114', '03-55195
319', '03-55102376', '03-55193044', '03-55193175', '03-55199533', '03-551
47400', '03-55147404', '03-55147405', '03-55205200', '03-55105049', '03-3
1695100', '03-31695190', '03-55190375', '03-55111063', '03-87328299', '03
-55144300', '03-55195255', '03-78464444', '03-78469892', '03-55194273',
'03-55400717', '03-55193915', '03-55101830', '03-55218790', '03-5521879
4', '03-55218791', '03-55447828', '03-55109705', '03-55101833', '03-55101
918', '03-55193233', '03-55193551', '03-55193457', '03-55199059', '03-784
63114', '03-78464982', '03-31693888', '03-31693600', '03-56328800', '03-5
6361605', '03-56361625', '03-56361573', '03-56361534', '03-56501600', '03
-56361534', '03-55102664', '03-55102791', '03-55102839', '03-55150200',
'03-55180408', '03-79568512', '03-79576396', '03-56236400', '03-5638915
9', '03-55118891', '03-55118706', '03-33411031', '03-33410506', '03-33410
443', '03-33411894', '03-55447000']
```

```
D:\Anaconda3\envs\python-dscourse\lib\site-packages\urllib3\connectionpoo
l.py:981: InsecureRequestWarning: Unverified HTTPS request is being made
to host 'www.selangor.gov.my'. Adding certificate verification is strongl
y advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.h
tml#ssl-warnings (https://urllib3.readthedocs.io/en/latest/advanced-usag
e.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)

In [127]:
```
# Quiz 1
import requests
import re

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

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


emailRegex = r'([\d\w\.]+\[at\][\d\w\.\-]+\.\w+)'
phoneRegex = r'(\+[0-9]{3,}\-[0-9 ]{7,})'

emails = re.findall(emailRegex, data.text)
phones = re.findall(phoneRegex, data.text)

print(emails)
print(phones)
```

```
['zuki[at]jpm.gov.my', 'zuki[at]jpm.gov.my', 'dr.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', 'nor.izuan[at]jpm.gov.my', 'mohd.abduh[at]jp
m.gov.my', 'hasif[at]jpm.gov.my', 'hanisah[at]jpm.gov.my', 'faizah.bastam
am[at]jpm.gov.my']
['+603-8872 7321', '+603-8872 7329', '+603-8872 7327', '+603-8872 7223',
'+603-8872 7200', '+603-8872 7216', '+603-8872 7224', '+603-8872 7218',
'+603-8872 7211', '+603-8872 7321', '+603-8872 7212']
```

Quiz 2

Extract the 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)

In [85]:
```
# Quiz 2
import requests
import re

url='https://www.jpn.gov.my/my/hubungi-kami/direktori-kakitangan/bahagian-k
data = requests.get(url, verify=False)
phoneRegex = r'([0-9]{2,}\-[0-9 ]{7,})'
phones = re.findall(phoneRegex, data.text)
print(phones)
```

```
['03-8880 7023', '03-8880 8185', '03-8880 8179', '03-8880 8185', '03-8880
8143', '03-8880 8185', '03-8880 7042', '03-8880 8185', '03-8880 8117', '0
3-8880 8185', '03-8000 8000', '03-8880 8288']
```

Quiz 3

Extract the Phone Numbers from this page using RegEx

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

In [9]:

```
# Quiz 3
import requests
import re

url='https://www.ptptn.gov.my/hubungi-ptptn'
data = requests.get(url, verify=False)
phoneRegex = r'([0-9\-\-]{2,}\-[0-9 ]{7,})'
phones = re.findall(phoneRegex, data.text)
print(phones)
```

```
['03-21931177', '019-3273814', '03-21931197', '017-4230544', '017-871606
9', '011-51813747', '019-3649405', '03-21931179', '017-8706413', '019-701
4450', '012-2384653', '012-2793861', '03-21931184', '017-3040187', '010-9
309668', '013-2593416', '012-2526162', '017-2385664', '018-3535811', '014
-5034527', '019-2635071', '013-2225981', '011-21326228', '019-2552681',
'019-3057771', '017-2937556', '017-7197560', '019-3986284', '019-505847
5', '013-6973323', '011-12327988', '019-3380363', '011-10952566', '010-44
84213', '012-6892851', '019-9289373', '0111-5996537', '014-6726355', '010
-4513823', '014-5045556', '013-3775478', '018-3666658', '019-2343492', '0
11-11567607', '016-9592884', '017-6350035', '012-4979300', '012-7894113',
'013-3924689', '011-33518922', '012-2793949', '017-2666271', '018-355168
6', '012-5731831', '012-2793946', '017-6744296', '019-2340117', '010-8844
898', '012-6369827', '019-2633071', '011-17911236', '012-2299576', '011-1
4201982', '012-3354671', '011-40224241', '012-2794367', '013-3962810', '0
17-2090581', '019-2551430', '012-9248412', '019-3030490', '013-5773275',
'011-14201829', '017-7273220', '017-2856818', '017-3546273', '013-950294
3', '011-62016403', '012-9548597', '012-2794397', '019-9913229', '013-255
0204', '011-14201832', '012-2493001', '013-2526566', '012-2794408', '018-
2308575', '018-3921577', '017-6301826', '012-2793467', '011-23718675', '0
17-7763076', '011-12834055', '013-5272728', '014-3271970', '013-2127206',
'019-2629071', '017-3294318', '014-2621703', '010-6554540', '011-1420205
6', '011-33178276', '011-1140 5767', '012-2794010', '014-6131607', '018-9
464985', '012-2794076', '011-1420 1914', '011-11181518', '011-27765184',
'012-5083642', '019-4066625', '014-5125304', '019-3180868', '012-279402
6', '019-4364504', '011-28932548', '012-2728060', '013-3990639', '0115-62
56325', '03-55231630', '019-3641299', '014-3200162', '017-6087784', '019-
3606110', '017-4150598', '017-8940494', '019-2753062', '012-2794456', '01
7-5210890', '019-8152634', '017-3274760', '012-2341568', '014-5493183',
'011-73449057', '019-3647199', '018-3757794', '012-2896425', '011-1420201
7', '018-2022457', '013-2374103', '013-4367671', '019-3271933', '011-1191
3055', '017-8121519', '019-4074903', '012-2759594', '017-6967633', '014-5
134935', '010-2786390', '019-2663071', '011-27282150', '011-21178900', '0
19-3278773', '019-4169525', '011-33091677', '011-25489744', '011-1420206
3', '017-9329956', '014-5020244', '011-23637492', '019-2939630', '013-296
5706', '014-3215269', '012-2794527', '017-3083081', '012-2995300', '018-6
654083', '012-9201327', '013-3043963', '011-37030877', '019-2142117', '01
7-6572252', '019-2060564', '011-11294543', '019-2141550', '013-2112182',
'019-4489566', '012-8658500', '011-14202084', '017-3511091', '018-966486
7', '019-5162281', '019-3433989', '014-5397033', '011-31715313', '017-296
7951', '017-9284240', '017-6775383', '010-5065876', '019-3391553', '017-9
635331', '018-3839070', '012-3043067', '019-4603873', '017-6300331', '06-
7606902', '019-3272599', '019-2152151', '011-21106861', '017-2437561', '0
14-7156933', '019-3368340', '013-6673642', '011-12294739', '018-2089929',
'012-3043271', '016-3521929', '013-5944876', '019-3983117', '017-241199
3', '017-3059182', '016-4293707', '011-12379899', '016-7946561', '011-377
11539', '019-7703928', '011-35065884', '012-5087614', '013-9066654', '010
-2879195', '019-2318094', '012-2506407', '018-2543217', '016-8887261', '0
12-9529017', '012-3042957', '013-5890449', '011-28013600', '019-2197761',
'014-5159084', '010-3245995', '019-2176614', '017-6275237', '014-332716
1', '06-2336210', '012-3049321', '013-6042113', '011-31929187', '017-6624
480', '011-12849088', '019-7560920', '017-5734105', '012-2507695', '017-3
051442', '013-6203805', '012-2507312', '016-3355074', '012-4455414', '019
-218 9132', '014-2737780', '010-6634059', '019-3369278', '017-6074825',
'017-6429284', '012-2507214', '011-23625105', '016-6558745', '012-231497
2', '019-4902007', '017-2928226', '013-3302537', '012-6897100', '017-2932
046', '012-9712429', '011-31749310', '016-2034001', '019-6703868', '013-6
549577', '07-237 1088', '011-26421223', '011-26384286', '013-7909403', '0
```

```
19-2087748', '011-28767644', '018-7631025', '019-3370402', '018-7745771',
'016-7069004', '011-21349402', '012-2343960', '017-3902797', '012-826158
6', '010-5050244', '019-3374605', '016-5151824', '017-7362256', '016-5388
130', '019-2662536', '017-7420745', '019-6346796', '018-4647844', '012-26
5 7306', '011-74120538', '014-5125738', '012-6892360', '017-6073772', '01
7-7098093', '018-2468121', '012-2480836', '014-6292360', '017-6690170',
'012-2343241', '013-7877321', '012-7785859', '019-2186911', '017-744521
3', '019-6047348', '019-2659071', '013-7401101', '016-7041430', '012-2341
926', '014-3122195', '017-7329319', '017-7595273', '019-2090731', '013-75
08971', '019-7000115', '019-2178911', '019-7961126', '011-10707204', '010
-7136162', '05-2498406', '013-3783018', '019-5765279', '010-5618958', '01
1-17471051', '019-2446388', '013-5315871', '010-4675394', '019-6453091',
'012-2691508', '013-3063630', '011-24047600', '016-5576394', '018-387832
1', '019-2184577', '019-4882387', '012-5008794', '012-3042795', '017-5570
201', '012-5300468', '016-4144210', '019-2097796', '011-51488755', '018-2
114021', '013-2458519', '017-4660206', '014-9053518', '019-3392782', '010
-2887060', '013-4122870', '014-9293769', '012-2692083', '017-5393432', '0
12-5845868', '019-3373798', '013-5913165', '011-11318943', '012-2692781',
'017-4619776', '019-4007050', '05-801 2398', '019-3390474', '017-524212
9', '017-5127992', '019-3391021', '017-5332445', '012-5455188', '019-3375
542', '016-4446894', '016-6209297', '013-5958537', '04-3838427', '019-327
4906', '013-4898068', '017-9145698', '013-4556084', '018-9573293', '012-2
693086', '012-4725195', '019-4985659', '012- 4690670', '012-2341270', '01
7-5650481', '013-6420149', '019-3533972', '013-5217785', '012-5513771',
'017-4576881', '019-3377840', '011-35134245', '017-4158058', '019-492300
5', '012-2341802', '017-4603602', '016-2579064', '04-226 2430', '019-3375
875', '019-4203014', '019-7671587', '011-14709963', '019-3966343', '019-6
630506', '017-4347770', '019-3600963', '016-4887879', '013-4619057', '013
-7994647', '012-2693280', '014-9036329', '011-40737367', '019-4202366',
'012-2341785', '017-4907775', '011-36306991', '012-5088615', '017-307749
2', '012-5796584', '019-3271328', '019-3378799', '013-2151951', '013-5206
013', '017-4224656', '019-5688895', '010-8465640', '019-3379862', '018-27
94765', '013-6666890', '017-4100639', '017-4670650', '017-8702195', '012-
5135527', '019-4576460', '011-54251470', '012-5085670', '013-5909380', '0
19-5712046', '017-8703507', '013-6347119', '013-5815540', '019-3973151',
'017-5439233', '019-5994749', '012-2694051', '014-3050102', '017-430737
4', '013-2265312', '012-8937595', '012-5988313', '010-3724786', '012-5082
165', '019-4503855', '011-37846617', '013-2709287', '018-2890757', '011-1
2350288', '017-4432457', '019-4887837', '013-4048214', '04-9708181', '019
-2674071', '010-3042264', '011-88880690', '012-2794067', '017-5654438',
'019-3415147', '012-3727957', '014-8744548', '013-9471845', '09-5173209',
'019-3397085', '010-8082073', '013-9389511', '013-2228942', '019-339438
7', '012-9615618', '018-2525336', '017-8709074', '016-9038387', '014-5181
880', '019-2662529', '019-9060095', '017-9216196', '013-9165528', '019-21
84145', '017-2837540', '019-9235290', '016-2474001', '014-5183135', '013-
3320057', '019-2553442', '013-9380180', '019-7581693', '018-9019456', '01
2-2480857', '019-2580237', '011-19909159', '017-3214070', '019-3398215',
'014-8372328', '017-9425627', '019-5623768', '019-2125748', '012-950161
0', '014-7949272', '09-7416776', '011-14201934', '012-9108843', '013-9229
258', '019-9715656', '012-2481087', '013-4662045', '014-8464787', '012-23
40851', '014-5066737', '017-8510694', '012-2344372', '011-25519411', '017
-6363285', '012-9867839', '019-2699145', '017-9410786', '013-9993512', '0
11-16121236', '019-3565811', '019-2684614', '013-9978230', '019-3544238',
'018-9031697', '09-960 2800', '019-9600558', '012-9540785', '019-909181
9', '012-2341029', '011-37533717', '019-9078053', '09-6233353', '012-6892
980', '019-9902873', '013-9287796', '019-2455749', '019-3412532', '011-31
212003', '018-3876499', '019-3409456', '013-9972779', '017-9867195', '014
```

```
-8408965', '017-9896476', '013-5173174', '019-3237827', '014-5088815', '0
11-6363 3000', '011-12171847', '019-9795146', '011-20668682', '017-923133
2', '013-365 2998', '017-8700563', '011-23749448', '013-5339445', '019-29
08725', '012-2007384', '017-7665185', '011-11163836', '013-2665624', '019
-2094825', '010-4087796', '011-25779475', '012-5090402', '012-8727741',
'016-8515192', '012-2793491', '010-9730581', '013-5604177', '019-220164
9', '014-6846474', '019-8866619', '019-7919375', '017-2714525', '012-5094
027', '016-8918539', '011-10069030', '019-2139161', '016-9649984', '016-4
304001', '012-8716654', '013-8177830', '014-6825188', '011-31735788', '01
3-8321999', '012-6893526', '011-29993920', '016-3045700', '014-8822786',
'016-5722843', '014-8822452', '017-8053856', '019-3657095', '016-583758
0', '011-16037479', '010-2252494', '016-8031403', '011-31449144', '012-25
11715', '013-5423922', '011-33329492', '019-2156170', '012-8014020', '014
-3536832', '016-5828485', '019-8736861', '013-9649197', '012-8150545', '0
12-5088658', '0111-7143319', '010-2402494', '011-36016122', '016-815115
2', '011-31725631', '012-8447305', '014-8587005', '019-3431140', '016-585
1095', '010-3860487', '019-2154960', '017-6917002', '011-31236318']

C:\Users\ACER\anaconda3\envs\python-dscourse\lib\site-packages\urllib3\co
nnectionpool.py:1056: InsecureRequestWarning: Unverified HTTPS request is
being made to host 'www.ptptn.gov.my'. Adding certificate verification is
strongly advised. See: https://urllib3.readthedocs.io/en/1.26.x/advanced-
usage.html#ssl-warnings (https://urllib3.readthedocs.io/en/1.26.x/advance
d-usage.html#ssl-warnings)
  warnings.warn(
C:\Users\ACER\anaconda3\envs\python-dscourse\lib\site-packages\urllib3\co
nnectionpool.py:1056: InsecureRequestWarning: Unverified HTTPS request is
being made to host 'www.ptptn.gov.my'. Adding certificate verification is
strongly advised. See: https://urllib3.readthedocs.io/en/1.26.x/advanced-
usage.html#ssl-warnings (https://urllib3.readthedocs.io/en/1.26.x/advance
d-usage.html#ssl-warnings)
  warnings.warn(
```

In [38]:

```
# Method 1

import requests
import re

url='https://www.ksn.gov.my/mengenai-pejabat/direktori-pejabat'

# get the data
```

In [40]:

```
# Method 2

import requests
import re

url='https://www.ksn.gov.my/mengenai-pejabat/direktori-pejabat'

# get the data
```

In [41]:
```
# Method 3

import requests
import re

url='https://www.ksn.gov.my/mengenai-pejabat/direktori-pejabat'

# get the data
```

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

import requests
import re

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

headers={'User-Agent':user_agent}

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

# 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.or
g/">\n    <head>\n        <meta http-equiv="X-UA-Compatible" content="I
E=edge">\n        <meta name="viewport" content="width=device-width, in
itial-scale=1">\n        <link rel="shortcut icon" href="/images/logojp
n40px.png">\n        <link rel="apple-touch-icon" href="/images/logojpn
40px.png">\n        <meta charset="utf-8" />\n\t<base href="https://ww
w.jpn.gov.my/my/hubungi-kami/direktori-kakitangan/bahagian-kewarganegar
aan" />\n\t<meta name="description" content="Portal Rasmi JPN" />\n\t<m
eta name="generator" content="XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
/>\n\t<title>Portal JPN - Bahagian Kewarganegaraan</title>\n\t<link hre
f="/my/hubungi-kami/direktori-kakitangan/bahagian-kewarganegaraan?forma
t=feed&amp;type=rss" rel="alternate" type="application/rss+xml" title
="RSS 2.0" />\n\t<link href="/my/hubungi-kami/direktori-kakitangan/baha
gian-kewarganegaraan?format=feed&amp;type=atom" rel="alternate" type="a
pplication/atom+xml" title="Atom 1.0" />\n\t<link href="https://www.jp
n.gov.my/my/hubungi-kami/direktori-kakitangan/bahagian-kewarganegaraan"
rel="alternate" hreflang="ms-MY" />\n\t<link href="https://www.jpn.gov.
my/en/contact-us/direktori-kakitangan/citizenship-division" rel="altern
ate" hreflang="en-GB" />\n\t<link href="/plugins/content/pdf_embed/asse
```

In [37]:
```
# 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 host 'www.ptptn.gov.my'. Adding certificate verification is strongly advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings (https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings)
  warnings.warn(

['ahmadshahril@ptptn.gov.my', 'sarah@ptptn.gov.my', 'umminadira@ptptn.gov.my', 'nabillah@ptptn.gov.my', 'noorshaheera@ptptn.gov.my', 'fadzlina@ptptn.gov.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', 'masliana@ptptn.gov.my', 'arif_h@ptptn.gov.my', 'kamarul_a@ptptn.gov.my', 'fazaitum@ptptn.gov.my', 'wan_norasyikin@ptptn.gov.my', 'karthinee@ptptn.gov.my', 'izzasuria@ptptn.gov.my', 's.nurul@ptptn.gov.my', 'nursyafira@ptptn.gov.my', 'hafsoh@ptptn.gov.my', 'nuraishah@ptptn.gov.my', 'norhidayati@ptptn.gov.my', 's.nurhaishah@ptptn.gov.my', 'roslan@ptptn.gov.my', 'haslida@ptptn.gov.my', 'md_ismail@ptptn.gov.my', 'amani@ptptn.gov.my', 'malissa@ptptn.gov.my', 'm.shafik@ptptn.gov.m
```

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)

```
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/)

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 consume
```

Out[18]:
```
[]
```

In [19]:
```
re.findall('\\btest\\b', 'test this is a test') # backslash is explicitly e
```

Out[19]:
```
['test', 'test']
```

In [14]:
```
re.findall(r'\btest\b', 'test this is a test') # often this syntax is easie
```

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|NE

print(matching)
```

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