Name: - L Prathyusha

PART 1 - Scraping of job website (Internshala and Indeed)

Website: -

1. https://internshala.com/internships/analytics-internship (https://internshala.com/internships/analytics-internship)
2. https://in.indeed.com/jobs?q=&l=india (https://in.indeed.com/jobs?q=&l=india)

In [1]:

```
from selenium import webdriver
from bs4 import BeautifulSoup
import time
import pandas as pd
import requests
```

In [2]:

```
url = "https://internshala.com/internships"
no_of_jobs = 9114
```

In [3]:

```
res = requests.get("https://internshala.com/internships/analytics-internship")

print(res.text)
print(res.status_code)
```

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml" xmlns:og="http://ogp.me/ns#" xm
lns:fb="https://www.facebook.com/2008/fbml">
    <head>
        <meta http-equiv="X-UA-Compatible" content="IE=9" />
        <meta charset="UTF-8" />
        <meta name="viewport" content="width=device-width, initial-scale=
1.0 user-scalable=0" />
        <meta property="fb:app_id" content="272234782795210" />
        <meta property="og:type" content="article" />
                    <meta property="og:image:width" content="1200" />
            <meta property="og:image:height" content="630" />
                    <meta name="twitter:site" content="@Internshala" />
                    <meta name="twitter:card" content="summary_large_imag
e" />

        <meta name="twitter:creator" content="@internshala" />
                <meta name="theme-color" content="#ffffff" />
        <meta name="msapplication-navbutton-color" content="#ffffff"/>
```

In [18]:

```
#body and head
import requests
from bs4 import BeautifulSoup

# Make a request
page = requests.get("https://internshala.com/internships/analytics-internship")
soup = BeautifulSoup(page.content, 'html.parser')

# Extract title of page
page_title = soup.title.text

# Extract body of page
page_body = soup.body

# Extract head of page
page_head = soup.head

# print the result
print(page_body, page_head)
```

```
<body>
<!-- Google Tag Manager (noscript) -->
<noscript><iframe height="0" src="https://www.googletagmanager.com/ns.htm
l?id=GTM-5NSPHG" style="display:none;visibility:hidden" width="0"></iframe
></noscript>
<!-- End Google Tag Manager (noscript) --> <div id="wrapper">
<!-- ribbon -->
<div class="notification" id="error">
</div>
<div class="notification notification_general" id="general_notification" s
tyle="display: none">
</div>
<!-- modal - error message -->
<div class="modal fade in internshala-modal" id="error_modal" style="displ
ay: none">
<div class="modal-dialog">
<div class="modal-content">
<div class="modal-body">
<div class="image_container error_image">
```

In [19]:

```
page_title = soup.title.text
page_title
```

Out[19]:

```
'Analytics Internship | Internshala'
```

In [21]:

```
def extract_job(soup):
    jobs = []
    for div in soup.find_all(name="div", attrs={"class":"row"}):
        for a in div.find_all(name="a", attrs={"analyst":"jobTitle"}):
            jobs.append(a["title"])
    return(jobs)
extract_job(soup)
```

Out[21]:

```
[]
```

In [22]:

```
#This code will extract the data and the infomation from the portal regarding the location

def extract_location(soup):
    locations = []
    spans = soup.findAll('span', attrs={'class': 'location'})
    for span in spans:
        locations.append(span.text)
    return(locations)
extract_location(soup)
```

Out[22]:

```
[]
```

Different website - "INDEED"

Here in this website the link is provided on the basis of the jobs that are being provided in INDIA and also the scraping is done on the basis of the provision of the jobs on the location and specifically analytics related work.

Website: - https://in.indeed.com/jobs?q=&l=india (https://in.indeed.com/jobs?q=&l=india)

In [23]:

```
URL = "https://in.indeed.com/jobs?q=&l=india"
#conducting a request of the stated URL above:
page = requests.get(URL)
#specifying a desired format of “page” using the html parser - this allows python to read t
soup = BeautifulSoup(page.text, "html.parser")
#printing soup in a more structured tree format that makes for easier reading
print(soup.prettify())
```

```
<!DOCTYPE html>
<html dir="ltr" lang="en">
 <head>
  <meta content="text/html;charset=utf-8" http-equiv="content-type"/>
  <script id="polyfill-script-bundle">
   /* Disable minification (remove `.min` from URL path) for more info */

(function(self, undefined) {function ArrayCreate(r){if(1/r==-Infinity&&(r=
0),r>Math.pow(2,32)-1)throw new RangeError("Invalid array length");var n=
[];return n.length=r,n}function Call(t,l){var n=arguments.length>2?argumen
ts[2]:[];if(!1===IsCallable(t))throw new TypeError(Object.prototype.toStri
ng.call(t)+"is not a function.");return t.apply(l,n)}function Get(n,t){ret
urn n[t]}function HasOwnProperty(r,t){return Object.prototype.hasOwnProper
ty.call(r,t)}function HasProperty(n,r){return r in n}function IsArray(r){r
eturn"[object Array]"===Object.prototype.toString.call(r)}function IsCalla
ble(n){return"function"==typeof n}function RequireObjectCoercible(e){if(nu
ll===e||e===undefined)throw TypeError();return e}function SameValueNonNumb
er(e,n){return e===n}function ToBoolean(o){return Boolean(o)}function ToIn
teger(n){var i=Number(n);return isNaN(i)?0:1/i===Infinity||1/i==-Infinity|
```

In [11]:

```python
jobtitle = soup.find_all("a", attrs={"class":"jobtitle"})
content1 = str(jobtitle)
print(content1)

#for a in div.find_all(name="a", attrs={"data-tn-element":"jobTitle"}):
```

```
[<a class="jobtitle turnstileLink" data-tn-element="jobTitle" href="/pagead/
clk?mo=r&amp;ad=-6NYlbfkN0CgqSKHH_6evJONJcxU4VJC6ze_GQQ4HsXw0S_tB7OmpM8fG6ib
ESvTheEy80pQWlvl162XowVOsKm6vegNzcXijgY7MGuolmka80ruQFQgHy7aBusujx_JX-G9K0zl
ZH7tB_QYc4p1FHyn46xYT1di_LqvigBIeL9ATbiQb-9b_TcyhcjiOuzpmyJDk3Yq1oOrSlqeRqut
QCIBWzikKuI334LwCRyYhD3ZXtHVL4TkY95m2kkdR_z8ue4k6BZ_uI9YQBX73_7ccQdtircNglUR
-9GJUepd8MP1C65AQ9GKuAVhcu3LjIXJRrFLl5fAUCgLilRlM7HFMcJdLGJDT1JciMfdw9fydnmz
9Gi4ei8kc2d5OBAoAs0rtOAf1Rbla0xqDOS6zRqkbKZDaCHN60pHK5cn7XP5Y6HTPr0ZXKeqYtcb
VvfV1xdnhVWGpnymu9iVHkuWYN4mKA==&amp;p=0&amp;fvj=1&amp;vjs=3" id="sja0" oncl
ick="setRefineByCookie([]); sjoc('sja0', 1); convCtr('SJ'); rclk(this,jobmap
[0],true,1);" onmousedown="sjomd('sja0'); clk('sja0'); rclk(this,jobmap[0],
1);" rel="noopener nofollow" target="_blank" title="Delivery Boy / Delivery
Executive / Freshers / Biker / Driver / Rider">
Delivery Boy / Delivery Executive / Freshers / Biker / Drive...</a>, <a clas
s="jobtitle turnstileLink" data-tn-element="jobTitle" href="/pagead/clk?mo=r
&amp;ad=-6NYlbfkN0Aww0xA2HCyFUeK1iayRdkgn8MWtRH_nCnQA-eXRcpWqRgPaOosS2u4IUpB
tS1btmZ4tHXE5-M8p4mEn5yiEcJSh_ccqs2-MpDAYLy0NzqNhkt0DdomfNw6CyLanJjkexgwkRn7
ot9Ctl5SPZQ_6s1_LOUTm0Sk8As2xvvRuaXqkPbbB2bVBc1yPnnHOl5yz1Ah1errSnBIY-gwOH8a
JDFZ5NnYdqXwPZ8ZgLQnidwXDq8sxR6VJvCwPCBvURHBhAoTOT3wiTEMgf-t8cW7Dco-woASj2FU
i99QOOW1J2y9JKe9JZn4QjHvgTNWjOW28bjbEpQBrGEm-bUUIRuNO4oLmYNrPA7s1epuBwvPrDqW
xEHNBORLIFIZdCpXDEdRLQ2R9gUI53tWqSKedTuX6_tjZqOPqqYqV78kYa48X8UKRt_AiOlpif4B
jdCuvAnUvSxmMLlUx1zMAiTbJ7OoNxkaHKjSFywriYcv8qEQZ8pZ-2LvHIzi7hDR&amp;p=1&am
p;fvj=1&amp;vjs=3" id="sja1" onclick="setRefineByCookie([]); sjoc('sja1',
1); convCtr('SJ'); rclk(this,jobmap[1],true,1);" onmousedown="sjomd('sja1');
clk('sja1'); rclk(this,jobmap[1],1);" rel="noopener nofollow" target="_blan
k" title="Teacher / Trainer (All subject)">
Teacher / Trainer (All subject)</a>, <a class="jobtitle turnstileLink" data-
tn-element="jobTitle" href="/pagead/clk?mo=r&amp;ad=-6NYlbfkN0BaCcj4EYzU2-jj
zXm_zx0aYdP8FnDawaQA2ggejfXH3O18tOG6jOGedBwD2RZBINQuzyuaVQknBrKvQO-TreeswNOV
oHBlZuz_RAgUmw9T_cQxiyESNDPDNp1xSzVS4JTgAy6Y1SGnIRBx3P5lzqBEiWZvvVA78BeUrMfJ
urh_peEsBPafHkxI79Nf8JrmO7siwawhk0zDW2lkMs_4Z9lTtv2EjGf0HIMppBubr1D3SlE1Grk8
2_7JJIFN23QN_EaU-nKUV_JbsmSQKP3iuu6RZRuIgYTXqK8kBMvHUfzDaDpoIUD7gAqcmQx6Mawi
VtdzgeWGfQCQVBu1yCs1J3Dr_NRlUlhMdPGsvpPnANcHLFYykY3NlmXT_GwoxhOQTEWZlsZOnwrR
ofKYZt1vD9e9Uzcdft2PtiRdSTZ8O0UuhCeRxTHrStDzsv2R2pHKsC82tvBBpH6DEA==&amp;p=2
&amp;fvj=1&amp;vjs=3" id="sja2" onclick="setRefineByCookie([]); sjoc('sja2',
1); convCtr('SJ'); rclk(this,jobmap[2],true,1);" onmousedown="sjomd('sja2');
clk('sja2'); rclk(this,jobmap[2],1);" rel="noopener nofollow" target="_blan
k" title="Online Teachers - English (Female Only)">
Online Teachers - English (Female Only)</a>, <a class="jobtitle turnstileLin
k" data-tn-element="jobTitle" href="/company/FirstHandle-Pvt-Ltd/jobs/Data-E
ntry-Operator-68b215e5edb1d9d6?fccid=ce12bb4a988b7783&amp;vjs=3" id="jl_68b2
15e5edb1d9d6" onclick="setRefineByCookie([]); return rclk(this,jobmap[3],tru
e,1);" onmousedown="return rclk(this,jobmap[3],1);" rel="noopener nofollow"
target="_blank" title="Data Entry Operator">
Data Entry Operator</a>, <a class="jobtitle turnstileLink" data-tn-element
="jobTitle" href="/pagead/clk?mo=r&amp;ad=-6NYlbfkN0DFi1nmQQWK2fa3N4W3y7EUOE
ocZkWPqKP_f_xZ7ne8RSGQbz1xcvPKO_15W0ph925uDEj0HZxoBNLkAJX44TXtpypCi23ur-apCW
ko7gN2o2jxpFMemzFpIFUKClII1a1Gg7M-wL7mcQkiWtFq82Li3PetmNeyhkcE26wEaV2cVhYdAI
oklNd6P61515oxZwInc2pb9VCeRCQr9VYZegklSeDI4iBtEjRemvXPyVPBjUOzjm2CIWPis4AjEX
xQgqGsIB0DGk4V-wfcghG6np93e6bV9sfcqFhrJiRtZxdoSUx-6LgNwRfs70QFFneI6kI9lCh2ul
3brakaWPFkX0OznoWcB157KF4GT5U5B2P-8Evm0EBTfwo-DJOFlbDw2USng0dsm4bwfRRPFY1YCZ
bLQEyc6Nk6Tv0-uFwJQ0CjeQimRHG2lfIODxE_18b28R1iwOZbvnVTe4z0YnKL5pINXixHUlaP_I
jt5iFiuvptyI5t1fq6xSDEy_16nFQYroc8237qJEplFknJBUuYf2pv5C1XCQX0P6w=&amp;p=4&a
mp;fvj=0&amp;vjs=3" id="sja4" onclick="setRefineByCookie([]); sjoc('sja4',
0); convCtr('SJ'); rclk(this,jobmap[4],true,0);" onmousedown="sjomd('sja4');
clk('sja4'); rclk(this,jobmap[4],0);" rel="noopener nofollow" target="_blan
k" title="Network Administrator">
Network Administrator</a>, <a class="jobtitle turnstileLink" data-tn-element
="jobTitle" href="/rc/clk?jk=520485306f9dd599&amp;fccid=74e36d81c894b4b7&am
p;vjs=3" id="jl_520485306f9dd599" onclick="setRefineByCookie([]); return rcl
k(this,jobmap[5],true,0);" onmousedown="return rclk(this,jobmap[5],0);" rel
="noopener nofollow" target="_blank" title="Call Center Supervisor I">
Call Center Supervisor I</a>, <a class="jobtitle turnstileLink" data-tn-elem
ent="jobTitle" href="/rc/clk?jk=3c6ed17e1d216076&amp;fccid=47cca1a92b73a5b2&
amp;vjs=3" id="jl_3c6ed17e1d216076" onclick="setRefineByCookie([]); return r
clk(this,jobmap[6],true,0);" onmousedown="return rclk(this,jobmap[6],0);" re
l="noopener nofollow" target="_blank" title="Junior Systems Admin">
Junior Systems Admin</a>, <a class="jobtitle turnstileLink" data-tn-element
="jobTitle" href="/company/dezine-di-lusso/jobs/Computer-Typist-77f7cc4473ae
1c32?fccid=798a47d74a71d375&amp;vjs=3" id="jl_77f7cc4473ae1c32" onclick="set
RefineByCookie([]); return rclk(this,jobmap[7],true,1);" onmousedown="return
rclk(this,jobmap[7],1);" rel="noopener nofollow" target="_blank" title="Comp
uter Typist">
Computer Typist</a>, <a class="jobtitle turnstileLink" data-tn-element="jobT
itle" href="/rc/clk?jk=6c5a17f7d99789b0&amp;fccid=fe2d21eef233e94a&amp;vjs=
3" id="jl_6c5a17f7d99789b0" onclick="setRefineByCookie([]); return rclk(thi
s,jobmap[8],true,0);" onmousedown="return rclk(this,jobmap[8],0);" rel="noop
ener nofollow" target="_blank" title="Office Administrator">
Office Administrator</a>, <a class="jobtitle turnstileLink" data-tn-element
="jobTitle" href="/rc/clk?jk=e2e9604dc57e67ea&amp;fccid=c67533a523c83a4c&am
p;vjs=3" id="jl_e2e9604dc57e67ea" onclick="setRefineByCookie([]); return rcl
k(this,jobmap[9],true,0);" onmousedown="return rclk(this,jobmap[9],0);" rel
="noopener nofollow" target="_blank" title="Process Executive - Customer Ser
vice Representative">
Process Executive - Customer Service Representative</a>, <a class="jobtitle
turnstileLink" data-tn-element="jobTitle" href="/company/Pitney-Bowes-Softwa
re-India-Pvt-Ltd/jobs/Amazon-Voice-Non-ac1d46821b16c7a3?fccid=a2dbe6797d69dc
eb&amp;vjs=3" id="jl_ac1d46821b16c7a3" onclick="setRefineByCookie([]); retur
n rclk(this,jobmap[10],true,1);" onmousedown="return rclk(this,jobmap[10],
1);" rel="noopener nofollow" target="_blank" title="AMAZON VOICE AND NON - V
OICE">
AMAZON VOICE AND NON - VOICE</a>, <a class="jobtitle turnstileLink" data-tn-
element="jobTitle" href="/rc/clk?jk=4653b0d90955f69a&amp;fccid=c67533a523c83
a4c&amp;vjs=3" id="jl_4653b0d90955f69a" onclick="setRefineByCookie([]); retu
rn rclk(this,jobmap[11],true,0);" onmousedown="return rclk(this,jobmap[11],
0);" rel="noopener nofollow" target="_blank" title="Member - Learning &amp;
Developmentt">
Member - Learning &amp; Developmentt</a>, <a class="jobtitle turnstileLink"
data-tn-element="jobTitle" href="/rc/clk?jk=b28b7621293dcd37&amp;fccid=a5b44
99d9e91a5c6&amp;vjs=3" id="jl_b28b7621293dcd37" onclick="setRefineByCookie
([]); return rclk(this,jobmap[12],true,0);" onmousedown="return rclk(this,jo
bmap[12],0);" rel="noopener nofollow" target="_blank" title="Systems Adminis
trator, Customer Conversations Platform">
Systems Administrator, Customer Conversations Platform</a>, <a class="jobtit
le turnstileLink" data-tn-element="jobTitle" href="/company/Miri-Infotech-Pv
t-Ltd/jobs/Amazon-Voice-Non-321860b65fa547b0?fccid=011bd6912128c9de&amp;vjs=
3" id="jl_321860b65fa547b0" onclick="setRefineByCookie([]); return rclk(thi
s,jobmap[13],true,1);" onmousedown="return rclk(this,jobmap[13],1);" rel="no
opener nofollow" target="_blank" title="AMAZON VOICE AND NON - VOICE">
AMAZON VOICE AND NON - VOICE</a>, <a class="jobtitle turnstileLink" data-tn-
element="jobTitle" href="/rc/clk?jk=ebb5aafa259d4c9e&amp;fccid=c9b9e477b3c84
b4d&amp;vjs=3" id="jl_ebb5aafa259d4c9e" onclick="setRefineByCookie([]); retu
rn rclk(this,jobmap[14],true,0);" onmousedown="return rclk(this,jobmap[14],
0);" rel="noopener nofollow" target="_blank" title="Intern">
Intern</a>]
```

In [12]:

```
def extract_job(soup):
    jobs = []
    for div in soup.find_all(name="div", attrs={"class":"row"}):
        for a in div.find_all(name="a", attrs={"data-tn-element":"jobTitle"}):
            jobs.append(a["title"])
    return(jobs)
extract_job(soup)
```

Out[12]:

```
['Delivery Boy / Delivery Executive / Freshers / Biker / Driver / Rider',
 'Teacher / Trainer (All subject)',
 'Online Teachers - English (Female Only)',
 'Data Entry Operator',
 'Network Administrator',
 'Call Center Supervisor I',
 'Junior Systems Admin',
 'Computer Typist',
 'Office Administrator',
 'Process Executive - Customer Service Representative',
 'AMAZON VOICE AND NON - VOICE',
 'Member - Learning & Developmentt',
 'Systems Administrator, Customer Conversations Platform',
 'AMAZON VOICE AND NON - VOICE',
 'Intern']
```

In [13]:

```
company = soup.find_all(name="span", attrs={"class":"company"})
company_name=str(company)
company_name
```

Out[13]:

```
'[<span class="company">\n<a class="turnstileLink" data-tn-element="companyN
ame" href="/cmp/Ride.swiggy" onmousedown="this.href = appendParamsOnce(this.
href, \'from=SERP&amp;campaignid=serp-linkcompanyname&amp;fromjk=1c11ad91878
61aa5&amp;jcid=db5415bd052ef74f\')" rel="noopener" target="_blank">\nRide.Sw
iggy</a></span>, <span class="company">\n<a class="turnstileLink" data-tn-el
ement="companyName" href="/cmp/Home-Guru" onmousedown="this.href = appendPar
amsOnce(this.href, \'from=SERP&amp;campaignid=serp-linkcompanyname&amp;fromj
k=5eb05cbd3b2b3d34&amp;jcid=827cb4e557add265\')" rel="noopener" target="_bla
nk">\nHome Guru</a></span>, <span class="company">\nUdayy</span>, <span clas
s="company">\nFirstHandle Pvt Ltd</span>, <span class="company">\n<a class
="turnstileLink" data-tn-element="companyName" href="/cmp/Infosys" onmousedo
wn="this.href = appendParamsOnce(this.href, \'from=SERP&amp;campaignid=serp-
linkcompanyname&amp;fromjk=21f107e712350ae9&amp;jcid=a7275015a5759d99\')" re
l="noopener" target="_blank">\nInfosys Limited</a></span>, <span class="comp
any">\n<a class="turnstileLink" data-tn-element="companyName" href="/cmp/Fis
-9" onmousedown="this.href = appendParamsOnce(this.href, \'from=SERP&amp;cam
paignid=serp-linkcompanyname&amp;fromjk=520485306f9dd599&amp;jcid=728c395148
e5a2db\')" rel="noopener" target="_blank">\nFIS Global</a></span>, <span cla
ss="company">\n<a class="turnstileLink" data-tn-element="companyName" href
="/cmp/Itc-Infotech" onmousedown="this.href = appendParamsOnce(this.href,
\'from=SERP&amp;campaignid=serp-linkcompanyname&amp;fromjk=3c6ed17e1d216076&
amp;jcid=47cca1a92b73a5b2\')" rel="noopener" target="_blank">\nITC Infotech
</a></span>, <span class="company">\nRanga documentation centre</span>, <spa
n class="company">\n<a class="turnstileLink" data-tn-element="companyName" h
ref="/cmp/Amazon.com" onmousedown="this.href = appendParamsOnce(this.href,
\'from=SERP&amp;campaignid=serp-linkcompanyname&amp;fromjk=6c5a17f7d99789b0&
amp;jcid=308e6a517ad137d7\')" rel="noopener" target="_blank">\nAISPL - Telan
gana</a></span>, <span class="company">\n<a class="turnstileLink" data-tn-el
ement="companyName" href="/cmp/Infosys" onmousedown="this.href = appendParam
sOnce(this.href, \'from=SERP&amp;campaignid=serp-linkcompanyname&amp;fromjk=
e2e9604dc57e67ea&amp;jcid=d4392e8051f7dc43\')" rel="noopener" target="_blan
k">\nInfosys BPM Limited</a></span>, <span class="company">\nPitney Bowes So
ftware India Pvt Ltd</span>, <span class="company">\n<a class="turnstileLin
k" data-tn-element="companyName" href="/cmp/Infosys" onmousedown="this.href
= appendParamsOnce(this.href, \'from=SERP&amp;campaignid=serp-linkcompanynam
e&amp;fromjk=4653b0d90955f69a&amp;jcid=d4392e8051f7dc43\')" rel="noopener" t
arget="_blank">\nInfosys BPM Limited</a></span>, <span class="company">\n<a
class="turnstileLink" data-tn-element="companyName" href="/cmp/Google" onmou
sedown="this.href = appendParamsOnce(this.href, \'from=SERP&amp;campaignid=s
erp-linkcompanyname&amp;fromjk=b28b7621293dcd37&amp;jcid=a5b4499d9e91a5c6
\')" rel="noopener" target="_blank">\nGoogle</a></span>, <span class="compan
y">\nMiri Infotech Pvt Ltd</span>, <span class="company">\n<a class="turnsti
leLink" data-tn-element="companyName" href="/cmp/Cbre" onmousedown="this.hre
f = appendParamsOnce(this.href, \'from=SERP&amp;campaignid=serp-linkcompanyn
ame&amp;fromjk=ebb5aafa259d4c9e&amp;jcid=c9b9e477b3c84b4d\')" rel="noopener"
target="_blank">\nCBRE</a></span>]'
```

In [14]:

```
def extract_com(soup):
    company = []
    spans = soup.findAll("span", attrs={"class": "company"})
    for span in spans:
        company.append(span.text.strip())
    return(company)
extract_com(soup)
```

Out[14]:

```
['Ride.Swiggy',
 'Home Guru',
 'Udayy',
 'FirstHandle Pvt Ltd',
 'Infosys Limited',
 'FIS Global',
 'ITC Infotech',
 'Ranga documentation centre',
 'AISPL - Telangana',
 'Infosys BPM Limited',
 'Pitney Bowes Software India Pvt Ltd',
 'Infosys BPM Limited',
 'Google',
 'Miri Infotech Pvt Ltd',
 'CBRE']
```

In [15]:

```
location = soup.find_all(name="span", attrs={"class":"location"})
location_name=str(location)
location
```

Out[15]:

```
[<span class="location accessible-contrast-color-location">Secunderabad, Tel
angana</span>,
 <span class="location accessible-contrast-color-location">Gurgaon, Haryana
</span>,
 <span class="location accessible-contrast-color-location">Hyderabad, Telang
ana</span>,
 <span class="location accessible-contrast-color-location">Gachibowli, Hyder
abad, Telangana</span>,
 <span class="location accessible-contrast-color-location">Hyderabad, Telang
ana</span>,
 <span class="location accessible-contrast-color-location">Gurgaon, Haryana
</span>,
 <span class="location accessible-contrast-color-location">Hyderabad, Telang
ana</span>,
 <span class="location accessible-contrast-color-location">Pune, Maharashtra
</span>,
 <span class="location accessible-contrast-color-location">Hyderabad, Telang
ana</span>,
 <span class="location accessible-contrast-color-location">Nainital, Uttarak
hand</span>,
 <span class="location accessible-contrast-color-location">Hyderabad, Telang
ana</span>]
```

In [16]:

```
#This code will extract the data and the infomation from the portal regarding the location

def extract_location(soup):
    locations = []
    spans = soup.findAll('span', attrs={'class': 'location'})
    for span in spans:
        locations.append(span.text)
    return(locations)
extract_location(soup)
```

Out[16]:

```
['Secunderabad, Telangana',
 'Gurgaon, Haryana',
 'Hyderabad, Telangana',
 'Gachibowli, Hyderabad, Telangana',
 'Hyderabad, Telangana',
 'Gurgaon, Haryana',
 'Hyderabad, Telangana',
 'Pune, Maharashtra',
 'Hyderabad, Telangana',
 'Nainital, Uttarakhand',
 'Hyderabad, Telangana']
```

In [21]:

```
import requests

URL = 'https://www.monster.com/jobs/search/?q=Software-Developer&where=Australia'
page = requests.get(URL)
print(page.text)
print(page.status_code)
```

```
<!doctype html>
<html>
  <head>
    <style data-styled="true" data-styled-version="5.2.1">.bMDofZ{display:
inline-block;-webkit-text-decoration:none;text-decoration:none;}/*!sc*/
.bMDofZ,.bMDofZ:visited{padding:0px 16px;border:2px solid #6e46ae;border-r
adius:4px;margin:0px 16px;background-color:transparent;font-family:Proxima
Nova;font-weight:700;font-size:14px;color:#6e46ae;line-height:32px;}/*!sc
*/
.bMDofZ:focus{padding:0px 16px;border:2px solid #402965;border-radius:4px;
margin:0px 16px;background-color:#402965;font-family:Proxima Nova;font-wei
ght:700;font-size:14px;color:#ffffff;line-height:32px;}/*!sc*/
.bMDofZ:hover{cursor:pointer;padding:0px 16px;border:2px solid #57378a;bor
der-radius:4px;margin:0px 16px;background-color:#57378a;font-family:Proxim
a Nova;font-weight:700;font-size:14px;color:#ffffff;line-height:32px;}/*!s
c*/
data-styled.g2[id="sc-gsTCUz"]{content:"bMDofZ,"}/*!sc*/
html,body,body div,span,object,iframe,h1,h2,h3,h4,h5,h6,p,blockquote,pre,a
bbr,address,cite,code,del,dfn,em,img,ins,kbd,q,samp,small,strong,sub,sup,v
```

In [23]:

```
for job_elem in job_elems:
    # Each job_elem is a new BeautifulSoup object.
    # You can use the same methods on it as you did before.
    title_elem = job_elem.find('h2', class_='title')
    company_elem = job_elem.find('div', class_='company')
    location_elem = job_elem.find('div', class_='location')
    print(title_elem)
    print(company_elem)
    print(location_elem)
    print()
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-23-eb5491849f84> in <module>
----> 1 for job_elem in job_elems:
      2     # Each job_elem is a new BeautifulSoup object.
      3     # You can use the same methods on it as you did before.
      4     title_elem = job_elem.find('h2', class_='title')
      5     company_elem = job_elem.find('div', class_='company')

NameError: name 'job_elems' is not defined
```

In [15]:

```
d1 = {'title':title,'company':company,'location':location}
df = pd.DataFrame.from_dict(d1)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-15-bff9bf15b81a> in <module>
----> 1 d1 = {'title':title,'company':company,'location':location}
      2 df = pd.DataFrame.from_dict(d1)

NameError: name 'location' is not defined
```

In [ ]:

```

```