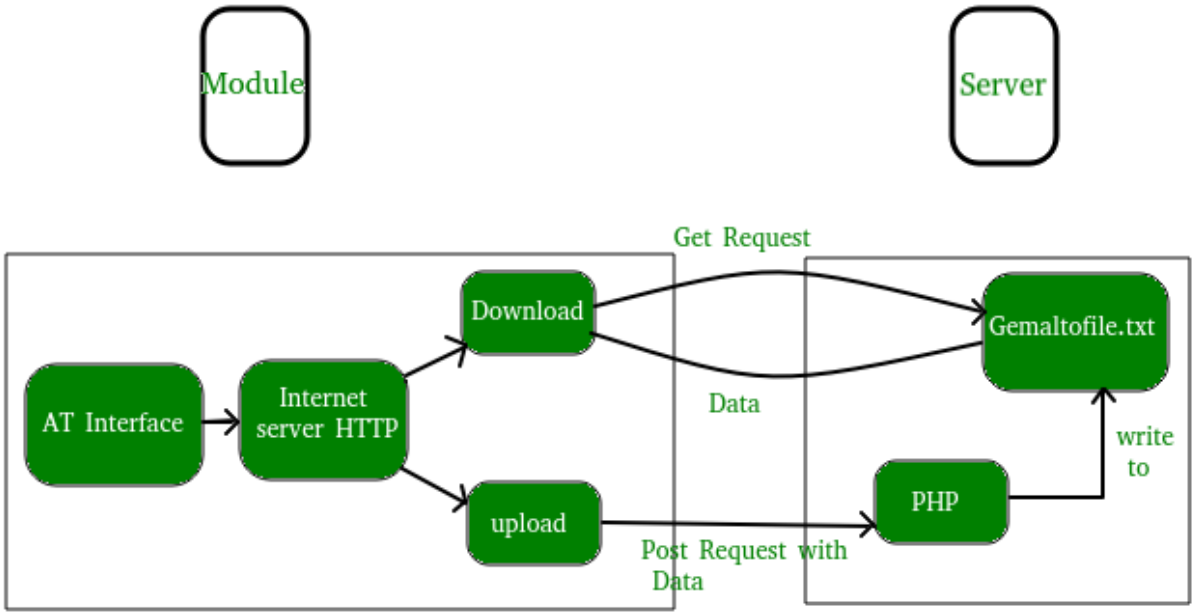
Read The Content of a URL Using HTTP Request Library in Python



source:(https://www.geeksforgeeks.org/get-post-requests-using-python/ (https://www.geeksforgeeks.org/get-post-requests-using-python/))

1- Libraries:

1- requests Library: for making HTTP requests in Python.
2- re Library: regular expression library need it to print only the contect without html tags.

2- HTTP request methods:

- get( ): HTTP has 2 methods GET and POST each on determine which action you want to perform in this code I used GET method to request the data (content) from the URL.
- content: gives you access to the raw bytes of the URL.
- text: convert raw bytes that show in content into a string using a character encoding such as UTF-8.
- status_code: informs you of the status of the request.
- sub( ): replace the substrings that match with the search pattern with a string of user's choice.
- headers: to give a useful information such as a content type.

In [27]:
```
# importing the requests and regular expression libraries
import requests as req
import re

resp = req.get("https://ai.googleblog.com/2020/03/visual-transfer-learni

if resp:
    print('Response OK')
else:
    print('Response Failed')

print(resp.status_code)
```

```
Response OK
200
```

In [29]:
```
print(resp.content)
```

```
b'<!DOCTYPE html>\n<html class=\'v2 detail-page\' dir=\'ltr\' itemscop
e=\'\' itemtype=\'http://schema.org/Blog\' lang=\'en\' xmlns=\'http://
www.w3.org/1999/xhtml\' xmlns:b=\'http://www.google.com/2005/gml/b\' x
mlns:data=\'http://www.google.com/2005/gml/data\' xmlns:expr=\'http://
www.google.com/2005/gml/expr\'>\n<head>\n<link href=\'https://www.blog
ger.com/static/v1/widgets/1243919952-css_bundle_v2.css\' rel=\'stylesh
eet\' type=\'text/css\'/>\n<title>\nGoogle AI Blog: Visual Transfer Le
arning for Robotic Manipulation\n</title>\n<meta content=\'width=devic
e-width, height=device-height, minimum-scale=1.0, initial-scale=1.0, u
ser-scalable=0\' name=\'viewport\'/>\n<meta content=\'IE=Edge\' http-e
quiv=\'X-UA-Compatible\'/>\n<meta content=\'article\' property=\'og:ty
pe\'/>\n<meta content=\'Visual Transfer Learning for Robotic Manipulat
ion\' property=\'og:title\'/>\n<meta content=\'http://2.bp.blogspot.co
m/-qRz-hnwUdY4/WulXSQ6Rv4I/AAAAAAAATvQ/shk7KsphA0c3E3nUMsDVASqYaH0PhLP
NwCK4BGAYYCw/s1600/GoogleAI_logo_horizontal_color_rgb.png\' property=\
'og:image\'/>\n<meta content=\'en_US\' property=\'og:locale\'/>\n<meta
content=\'http://ai.googleblog.com/2020/03/visual-transfer-learning-fo
r-robotic.html\' property=\'og:url\'/>\n<meta content=\'Google AI Blog
\' property=\'og:site_name\'/>\n<!-- Twitter Card properties -->\n<met
```

In [14]:
```
content = resp.text
print(content)
```

```
<title>
Google AI Blog: Visual Transfer Learning for Robotic Manipulation
</title>
<meta content='width=device-width, height=device-height, minimum-scale
=1.0, initial-scale=1.0, user-scalable=0' name='viewport'/>
<meta content='IE=Edge' http-equiv='X-UA-Compatible'/>
<meta content='article' property='og:type'/>
<meta content='Visual Transfer Learning for Robotic Manipulation' prop
erty='og:title'/>
<meta content='http://2.bp.blogspot.com/-qRz-hnwUdY4/WulXSQ6Rv4I/AAAAA
AAATvQ/shk7KsphA0c3E3nUMsDVASqYaH0PhLPNwCK4BGAYYCw/s1600/GoogleAI_logo
_horizontal_color_rgb.png' property='og:image'/>
<meta content='en_US' property='og:locale'/>
<meta content='http://ai.googleblog.com/2020/03/visual-transfer-learni
ng-for-robotic.html' property='og:url'/>
<meta content='Google AI Blog' property='og:site_name'/>
<!-- Twitter Card properties -->
<meta content='Google AI Blog' property='twitter:site'/>
<meta content='Visual Transfer Learning for Robotic Manipulation' prop
erty='twitter:title'/>
```

In [24]:
```
stripped = re.sub('<[^<]+?>', '', content)
print(stripped)
```

```
        ga('blogger.send', 'pageview');




      //<![CDATA[
      var axel = Math.random() + "";
      var a = axel * 10000000000000;
      document.write('');
      //]]>
```

In [20]:
```
print(resp.headers)
```

```
{'Strict-Transport-Security': 'max-age=86400; includeSubDomains', 'Con
tent-Type': 'text/html; charset=UTF-8', 'Expires': 'Sun, 05 Jul 2020 1
3:20:08 GMT', 'Date': 'Sun, 05 Jul 2020 13:20:08 GMT', 'Cache-Control'
: 'private, max-age=0', 'Last-Modified': 'Sat, 04 Jul 2020 10:35:29 GM
T', 'ETag': 'W/"f291a04262644ba1eb8ffcb7bce50f2eb1ade3e9f9c54004d3f85f
798548cdfd"', 'Content-Encoding': 'gzip', 'X-Content-Type-Options': 'n
osniff', 'X-XSS-Protection': '1; mode=block', 'Server': 'GSE', 'Alt-Sv
c': 'h3-29=":443"; ma=2592000,h3-27=":443"; ma=2592000,h3-25=":443"; m
a=2592000,h3-T050=":443"; ma=2592000,h3-Q050=":443"; ma=2592000,h3-Q04
6=":443"; ma=2592000,h3-Q043=":443"; ma=2592000,quic=":443"; ma=259200
0; v="46,43"', 'Transfer-Encoding': 'chunked'}
```

- **References:**

1- Python Requests tutorial: http://zetcode.com/python/requests/ (http://zetcode.com/python/requests/)
2- How To Get Started With the Requests Library in Python: https://www.digitalocean.com/community/tutorials/how-to-get-started-with-the-requests-library-in-python (https://www.digitalocean.com/community/tutorials/how-to-get-started-with-the-requests-library-in-python)
3- re — Regular expression operations: https://docs.python.org/2/library/re.html (https://docs.python.org/2/library/re.html)
4- https://www.geeksforgeeks.org/get-post-requests-using-python/ (https://www.geeksforgeeks.org/get-post-requests-using-python/)

In [ ]: