

## Ex. No: 10

**Aim:** *Demonstrate web scraping using python*

### Theory:

Web Scraping refers to extracting large amounts of data from the web. This is important for a data scientist who has to analyze large amounts of data.

Python provides a very handy module called requests to retrieve data from any website.

The requests.get() function takes in a URL as its parameter and returns the HTML response as its output.

The way it works is summarized in the following steps:

- It packages the Get request to retrieve data from webpage
- Sends the request to the server
- Receives the HTML response and stores in a response object

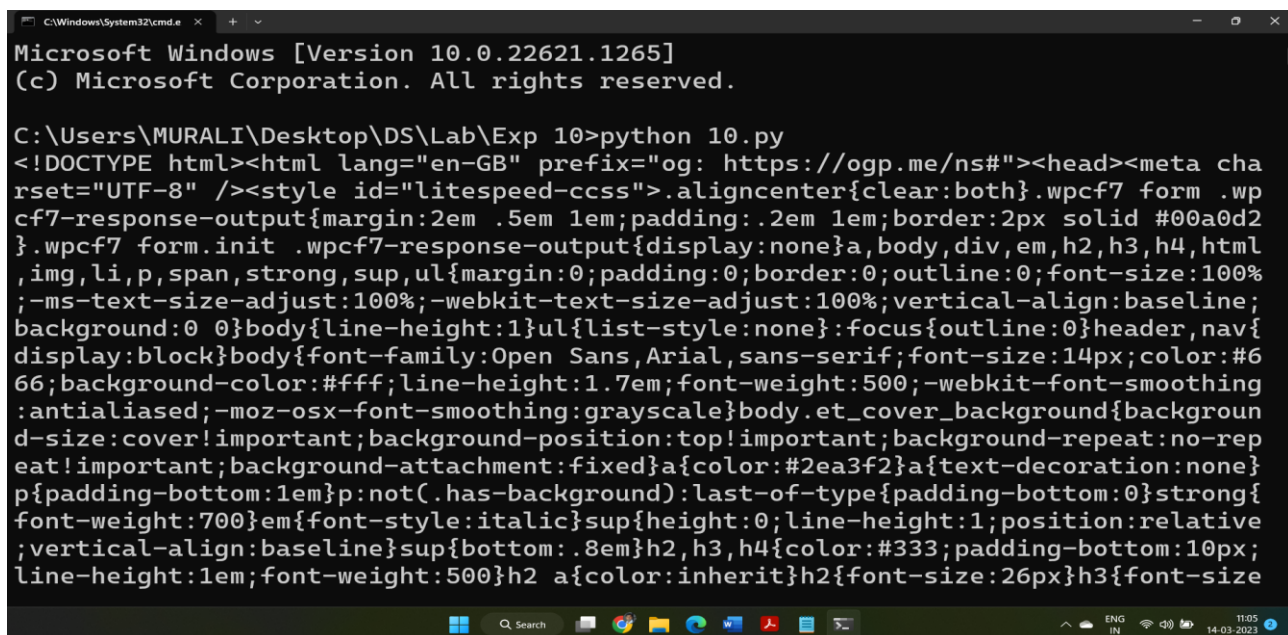
For this example, I want to show you a bit about our college – CMR Technical Campus. So, I will retrieve data from the "https://cmrtc.ac.in"

### Example 1:

```
import requests

url = "https://cmrtc.ac.in"
# response object
resp = requests.get(url)
# using text attribute of the response object, return the HTML of webpage as string
text = resp.text
print(text)
```

### Output:



```
Microsoft Windows [Version 10.0.22621.1265]
(c) Microsoft Corporation. All rights reserved.

C:\Users\MURALI\Desktop\DS\Lab\Exp 10>python 10.py
<!DOCTYPE html><html lang="en-GB" prefix="og: https://ogp.me/ns#"><head><meta cha
rset="UTF-8" /><style id="litespeed-ccss">.aligncenter{clear:both}.wpcf7 form .wp
cf7-response-output{margin:2em .5em 1em;padding:.2em 1em;border:2px solid #00a0d2
}.wpcf7 form.init .wpcf7-response-output{display:none}a,body,div,em,h2,h3,h4,html
,img,li,p,span,strong,sup,ul{margin:0;padding:0;border:0;outline:0;font-size:100%
;-ms-text-size-adjust:100%;-webkit-text-size-adjust:100%;vertical-align:baseline;
background:0 0}body{line-height:1}ul{list-style:none}:focus{outline:0}header,nav{
display:block}body{font-family:Open Sans,Arial,sans-serif;font-size:14px;color:#6
66;background-color:#fff;line-height:1.7em;font-weight:500;-webkit-font-smoothing
:antialiased;-moz-osx-font-smoothing:grayscale}body.et_cover_background{backgroun
d-size:cover!important;background-position:top!important;background-repeat:no-rep
eat!important;background-attachment:fixed}a{color:#2ea3f2}a{text-decoration:none}
p{padding-bottom:1em}p:not(.has-background):last-of-type{padding-bottom:0}strong{
font-weight:700}em{font-style:italic}sup{height:0;line-height:1;position:relative
;vertical-align:baseline}sup{bottom:.8em}h2,h3,h4{color:#333;padding-bottom:10px;
line-height:1em;font-weight:500}h2 a{color:inherit}h2{font-size:26px}h3{font-size
```

The tree-like structure of the HTML content retrieved by our request is not very comprehensible. To improve this readability, Python has another wonderful library called BeautifulSoup.

BeautifulSoup is a Python library for parsing the tree-like structure of HTML and extracting data from the HTML document.

To make it work, we need to pass the text response from the request object to BeautifulSoup() which creates its own object – “soup” in this case. Calling prettify() on BeautifulSoup object parses the tree-like structure of the HTML document:

### Example 2:

```
import requests
from bs4 import BeautifulSoup

url = "https://cmrtc.ac.in"

# Package the request, send the request and catch the response: r
r = requests.get(url)

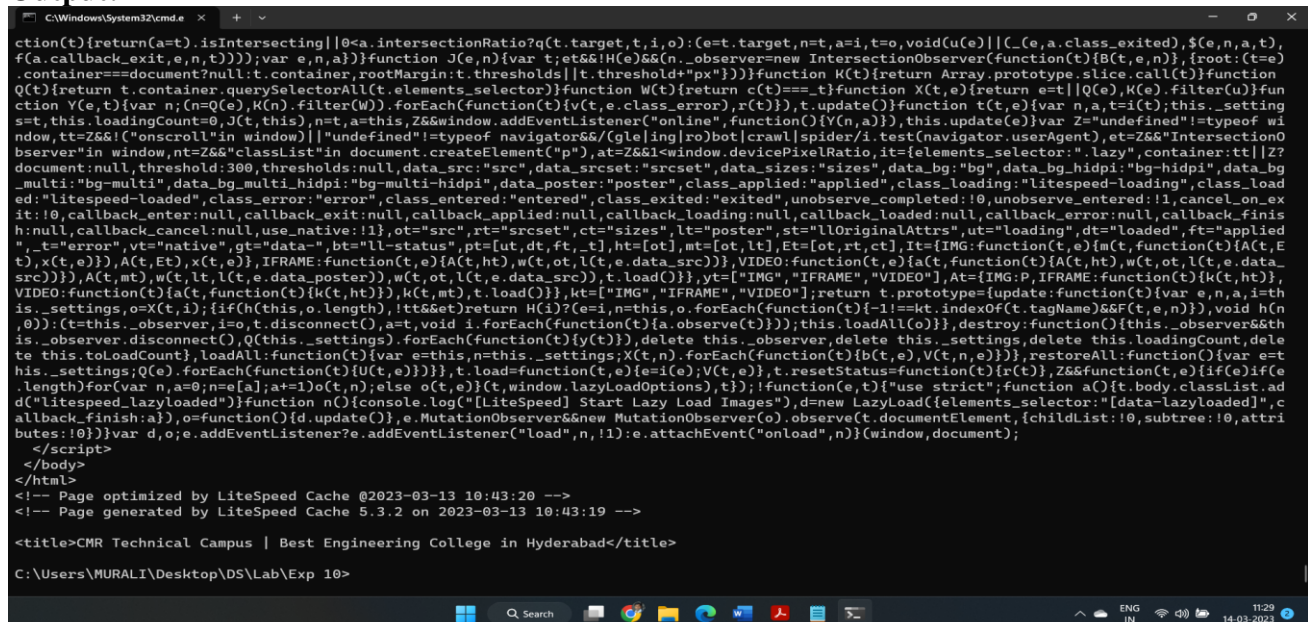
# Extracts the response as html: html_doc
html_doc = r.text

# Create a BeautifulSoup object from the HTML: soup
soup = BeautifulSoup(html_doc)

# Print the response
print(soup.prettify())

# Getting the title tag
print(soup.title)
```

### Output:



```
C:\Windows\System32\cmd.exe
ctio(t){return(a=t).isIntersecting||0<a.intersectionRatio?q(t.target,t,i,o):(e=t.target,n=t,a=i,t=o,void(u(e)||(_(e,a.class_exited),$(e,n,a,t),f(a,callback_exit,e,n,t)))));var e,n,a}}function J(e,n){var t;et&&H(e)&&(n._observer=new IntersectionObserver(function(t){B(t,e,n)},{root:(t=e).container===document?null:t.container,rootMargin:t.thresholds||t.threshold+"px"}))}function K(t){return Array.prototype.slice.call(t)}function Q(t){return t.container.querySelectorAll(t.elements_selector)}function W(t){return c(t)}function X(t,e){return e=t||Q(e).K(e).filter(u)}function Y(e,t){var n;(n=Q(e).K(n).filter(W)).forEach(function(t){v(t,e.class_error,r(t)),t.update()}function t(t,e){var n,a,t=i(t);this._setting s=t,this.loadingCount=0,J(t,this),n=t,a=this,Z&&window.addEventListener("online",function(){Y(n,a)}),this.update(e)}var Z="undefined"!=typeof wi ndow,tt=Z&&("onscroll"in window)||"undefined"!=typeof navigator&&(g|e|ing|ro)bot|crawl|spider/i.test(navigator.userAgent),et=Z&&"Intersection0 bserver"in window,nt=Z&&"classList"in document.createElement("p"),at=Z&&1<window.devicePixelRatio,it={elements_selector:".lazy",container:tt||Z? document:null,threshold:300,thresholds:null,data_src:"src",data_srcset:"srcset",data_sizes:"sizes",data_bg:"bg",data_bg_hidpi:"bg-hidpi",data_bg _multi:"bg-multi",data_bg_multi_hidpi:"bg-multi-hidpi",data_poster:"poster",class_applied:"applied",class_loading:"litespeed-loading",class_load ed:"litespeed-loaded",class_error:"error",class_entered:"entered",class_exited:"exited",unobserve_completed:10,unobserve_entered:11,cancel_on_ex it:10,callback_enter:null,callback_exit:null,callback_applied:null,callback_loading:null,callback_loaded:null,callback_error:null,callback_finis h:null,callback_cancel:null,use_native:11},ot="src",rt="srcset",ct="sizes",lt="poster",st=110,originalAttrs="ut="loading",dt="loaded",ft="applied ",_t="error",vt="native",gt="data-",bt="ll-status",pt=[ut,dt,ft,_t],ht=[ot],mt=[ot,lt],Et=[ot,rt,ct],It=[IMG:function(t,e){m(t,function(t){A(t,E t),x(t,e)}),A(t,Et),x(t,e)},IFRAME:function(t,e){A(t,ht),w(t,ot,lt(t,e.data_src))},VIDEO:function(t,e){A(t,function(t){A(t,ht),w(t,ot,lt(t,e.data _src))}),A(t,mt),w(t,lt,lt(t,e.data_poster)),w(t,ot,lt(t,e.data_src)),t.load()}],yt=["IMG","IFRAME","VIDEO"],At=[IMG:P.IFRAME,function(t){k(t,ht)}, VIDEO:function(t){a(t,function(t){k(t,ht)}),k(t,mt),t.load()}],kt=["IMG","IFRAME","VIDEO"];return t.prototype={update:function(t){var e,n,a,i,th is._settings,o=X(t,i);if(h(this,o.length),lt&&et)return H(i)?(e=i,n=this,o.forEach(function(t){f-1==kt.indexOf(t.tagName)&&f(t,e,n)},void(h(n ,0))):(t=this._observer,i=o,t.disconnect(),a=t,void i.forEach(function(t){fa.observe(t)});this.loadAll(o)},destroy:function(t){this._observer&&th is._observer.disconnect(),Q(this._settings).forEach(function(t){y(t)},delete this._observer,delete this._settings,delete this.loadingCount,dele te this.toLoadCount),loadAll:function(t){var e=this,n=this._settings;x(t,n).forEach(function(t){b(t,e),V(t,n,e)}),restoreAll:function(t){var e=t this._settings;(e).forEach(function(t){U(t,e)}),t.load=function(t,e){e=i(e),V(t,e),t.resetStatus=function(t){x(t)},Z&&function(t,e){if(e){e .length}for(var n,a=0;n=a[a],a+=1)o(t,n);else o(t,e)}(t,window.lazyLoadOptions,t)};function(e,t){function a(){t.body.classList.ad d("litespeed_lazyloaded")}function n(){console.log("[LiteSpeed] Start Lazy Load Images"),d=new LazyLoad({elements_selector:"[data-lazyloaded]",c allback_finish:a}),o=function(){d.update(),e.MutationObserver&&new MutationObserver(o).observe(t.documentElement,{childList:!0,subtree:!0,attri butes:!0})}}var d,o,e.addEventListene?e.addEventListener("load",n,!1):e.attachEvent("onload",n)}(window,document);
</script>
</body>
</html>
<!-- Page optimized by LiteSpeed Cache @2023-03-13 10:43:20 -->
<!-- Page generated by LiteSpeed Cache 5.3.2 on 2023-03-13 10:43:19 -->

<title>CMR Technical Campus | Best Engineering College in Hyderabad</title>

C:\Users\MURALI\Desktop\DS\Lab\Exp 10>
```