CO324 Lab 2: HTTP Clients

# **Objective**

Use command line tools to make HTTP requests to various APIs. At the end of the lab you should be able to

* Use the Python urllib module to make HTTP client requests.
* Understand the difference between various HTTP request methods (verbs.)
* Be able to interpret HTTP responses and status codes.
* Awareness of common HTTP headers.

# **Preparation**

Install Python version 3.7 or newer on your computer. You may use whichever IDE or editor that you prefer. You will need a functioning Internet connection to do the exercises.

# **Instructions**

* Use only the standard built-in modules in Python to complete these exercises.
* Put the solutions to coding exercises in files named ex1a.py, ex2b.py, etc.
* Solutions to short answer questions should be in a single plain text file named answers.txt. State your E-number at the top of answers.txt.
* Zip all your files and submit them (please don’t use other formats.)

Submissions that do not follow instructions will incur a 10% penalty.

# **References**

* <https://developer.mozilla.org/en-US/docs/Web/HTTP/Overview>
* <https://pymotw.com/3/urllib.request/index.html>
* <https://www.urlencoder.io/python/>

# **Exercise**

1. We perform an HTTP GET request in Python as follows.

|  |
| --- |
| from urllib import request  response = request.urlopen("<http://eng.pdn.ac.lk>")  body = response.read()  response.close() |

1. What is the response code you received?
2. What is the web server and OS used to host this site? Hint: look at the headers stored in response.headers.items().
3. What is the size of the response body?
4. Read the response body into a variable named ‘body’. What is the Python type of the ‘body’ variable?
5. Explain why ‘body’ has that particular type with reference to the structure of HTTP responses.
6. What happens if you request the URL “<http://eng.pdn.ac.lk/unknown>" ? What about “[http://unknown.pdn.ac.lk](http://eng.pdn.ac.lk)"?
7. Request the following URL and store the response body in a variable ‘body’.<https://ta.wikipedia.org/wiki/%E0%AE%9A%E0%AE%BF%E0%AE%99%E0%AF%8D%E0%AE%95%E0%AE%B3%E0%AE%AE%E0%AF%8D>. Try printing the body data.
8. What difference do you see if you call the method .decode(“utf-8”) on the body data received in (g) before printing it?
9. Let’s explore DDG search. To submit a search query we do:

|  |
| --- |
| from urllib import request  with request.urlopen(“https://www.duckduckgo.com/?q=University+of+Peradeniya”) as query:  headers = query.headers.items()  body = query.read() |

1. Append the string “&format=json&pretty=1” to the search query. What difference do you see in the response?
2. Use request.urlopen to search for the phrase “Rocco's basilisk”. Hint: try it in your browser first and look at the request URL carefully.
3. What is URL encoding/decoding? Why is it necessary?
4. How would you do a DDG search in Python for your name written in Tamil or Sinhala? Use this for Unicode input<https://www.lexilogos.com/keyboard/>.
5. Complete this function.

|  |
| --- |
| def ddg\_query(search\_term: str, nr\_results: int) -> List[str]  '''Extracts the top ‘nr\_result’ number of URLs from the response.  Returns extracted URLs as a list.''' |

For example ddg\_query(“University of Peradeniya”, 1) should return something like [“<http://www.pdn.ac.lk/>”]

1. Complete this function.

|  |
| --- |
| def spider\_metadata(urls: List[str]) -> List[List]  '''Returns the HTTP headers for each of the URLs in the list.''' |

Hint: refer back to your answer to exercise 1(b).

1. How can you avoid downloading the response bodies? Hint: Use the appropriate HTTP method (<https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods>)