

SI 506 DYU 6

The primary take away from this week's lec prep was about how to use the REST API to gather information.

- On running `requests.get()`, a function defined in the Requests module, we get a response object
- This response object has a lot of attributes (`.text`, `.url`, `.headers`, etc)
- We can come to know about the status of a url by using the `.status_code` attribute of the response object
- Status Codes and their respective Meanings:
(200, 404, 301) = (Success, Error, not found, Page has moved to another location) (*hah, sneaked in a tuple assignment there)
- Once the website receives a request from the RESTAPI, it sends back data in JSON formatted string
- The `requests.get` function also takes an optional parameter called *params*. *Params* has to be a dictionary and the key-value pairs are appended to the base URL from where data is being fetched.
- To open JSON formatted data as a dictionary in Python, we used the `loads()` function
- The `dumps` function takes a dictionary and returns a string in JSON format
- When requesting data from websites, we often encounter errors
- Using try-except, we can ensure that we can get data and get notified if there is an error
- By putting the error name in the except part, and the data fetching part in try, we can successfully get data till no error is encountered. When an error is encountered, the statements in except part are executed
- By combining try-except and `requests.get`, we can fetch data from websites. Using `json.loads()`, we can convert the data into a dictionary and process it.
- After processing, the data can be dumped into a `.csv` format file where `.csv` stands for Comma Separated Values.