The `requests` Package

> Prerequisite: Computer Networks and APIs

The `requests` package provides an easy way for Python programs to issue HTTP requests, whether scraping the contents of a web page, or exchanging data with an API.

Reference:

- Source Code
- Docs

Installation

First install the package, if necessary:

```
pip install requests
```

Usage

Issuing HTTP Requests

Issue a "GET" request (perhaps the most common):

```
import requests

request_url = "https://raw.githubusercontent.com/prof-rossetti/intro-to-python/master/data/products/1.json"
response = requests.get(request_url)
print(response.status_code)
print(response.text)
```

In addition to "GET" requests, you can also issue other types of requests like "POST", "PUT", and "DELETE", sending data to the server as necessary:

```
# where request_url is a URL that accepts POST requests
# ... and my_data is a dictionary of data to POST
response = requests.post(request_url, json=my_data)

# where request_url is a URL that accepts PUT requests
# ... and my_data is a dictionary of data to PUT
response = requests.put(request_url, json=my_data)
```

```
# where request_url is a URL that accepts DELETE requests
response = requests.delete(request_url)
```

Example POST request:

```
import requests

request_url = "https://www.ssa.gov/cgi-bin/babyname.cgi"
params = {"name": "Jane", "sex": "F"}
response = requests.post(request_url, params)
print(response.status_code)
print(response.text)
```

Parsing HTTP Responses

If the response contains JSON, you can use the `json` module to parse it:

```
response = requests.get(some_url)
response_data = json.loads(response.text)
print(type(response_data)) #> <class 'list'> or <class 'dict'>
```

If the response contains data in CSV format, you can use the familiar CSV-processing mechanisms like the the `csv` module or the `pandas` package, with some possible modifications for parsing a CSV-formatted string instead of a CSV file.

If the response contains data in HTML or XML format, you can use the `BeautifulSoup` package to parse it.