

# APIs and the Requests Library: Takeaways



by Dataquest Labs, Inc. - All rights reserved © 2024

## Syntax

- Importing the requests library:

```
import requests
```

- Calling a GET method on a URL endpoint:

```
import requests
requests.get(url)
```

- Calling a POST method on a URL endpoint:

```
import requests
requests.post(url)
```

- Converting an HTTP response to a dictionary:

```
import requests
response = requests.get(url)
response_dict = response.json()
```

- Converting JSON data to a string:

```
import requests
import json
response = requests.get(url)
response_dict = response.json()
response_str = json.dumps(response_dict)
```

- Accessing the status code to check for errors:

```
import requests
response = requests.get(url)
response.status_code
```

## Concepts

- An **API**, or **Application Programming Interface**, is a set of definitions and rules that allows different software components to communicate and interact.
- The **HTTP request** is a message sent over the internet to a server requesting a specific action to be performed or certain information to be returned.
- The **HTTP response** includes the requested information, a confirmation message, or an error if something goes wrong.
- The body of an HTTP POST request contains all the data to be passed to the server.
- Headers specify additional information the server may need to fulfill the request, like authorization credentials.
- The `json` library module can be used to convert a dictionary to a string when we need to POST string data to the endpoint call.

# Resources

- [HTTP Request Methods](#)
- [HTTP Request Headers](#)
- [Practice with more APIs](#)

Takeaways by Dataquest Labs, Inc. - All rights reserved © 2024