

APIs

FinTech
Lesson 5.1



Class Objectives

By the end of today's class, you will be able to:



Describe the client-server model.



Read documentation and identify endpoints from a given API.



Perform a GET request and view the JSON response using the Postman client.



Interpret the JSON structure from a GET request-response.



Use the requests library to request JSON data from an API within Python.

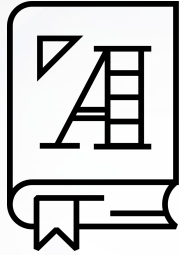


Parse a JSON response and print a selected field using Python.



Query an API using URL parameters with the requests library.

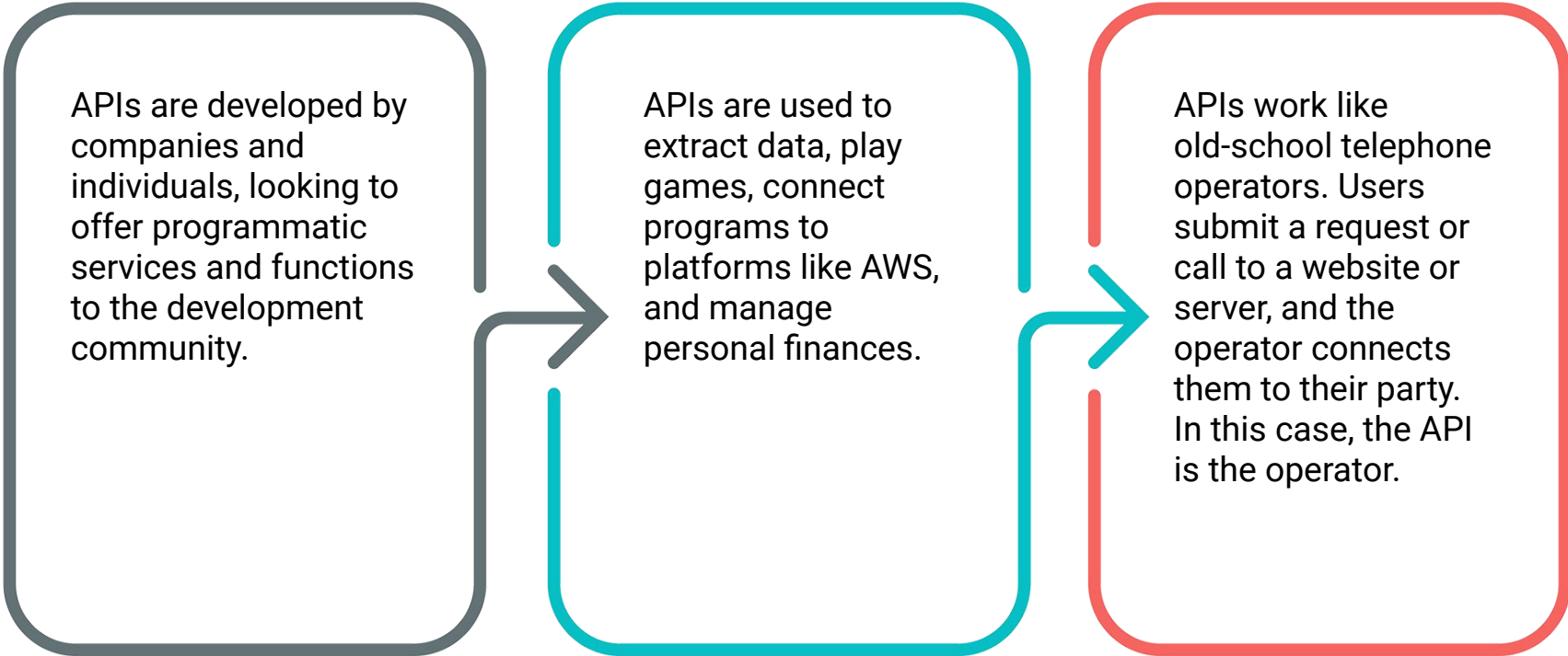
Disclaimer: The response and content of live data cannot be censored or predicted.



Application Programming Interface (API)

is a set of functions packaged together that lets developers communicate with a server and integrate third-party software and technology into new applications.

Application Programming Interfaces (APIs)



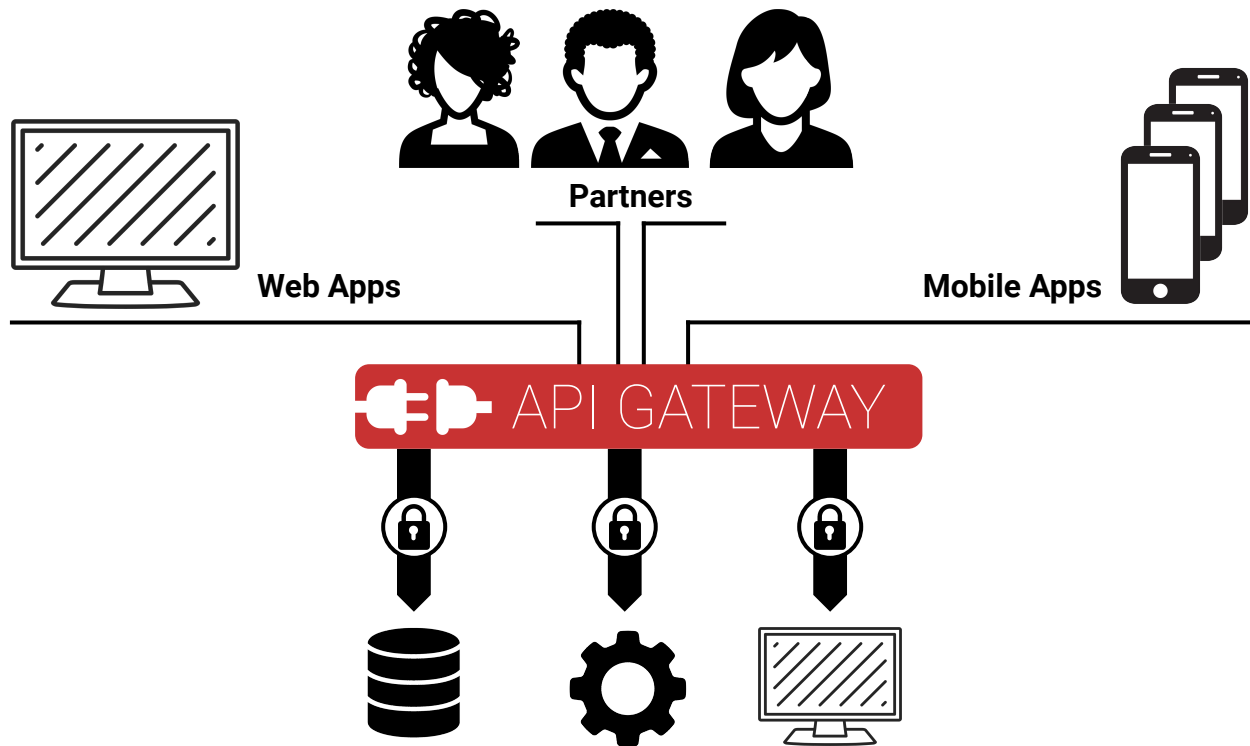
APIs are developed by companies and individuals, looking to offer programmatic services and functions to the development community.

APIs are used to extract data, play games, connect programs to platforms like AWS, and manage personal finances.

APIs work like old-school telephone operators. Users submit a request or call to a website or server, and the operator connects them to their party. In this case, the API is the operator.

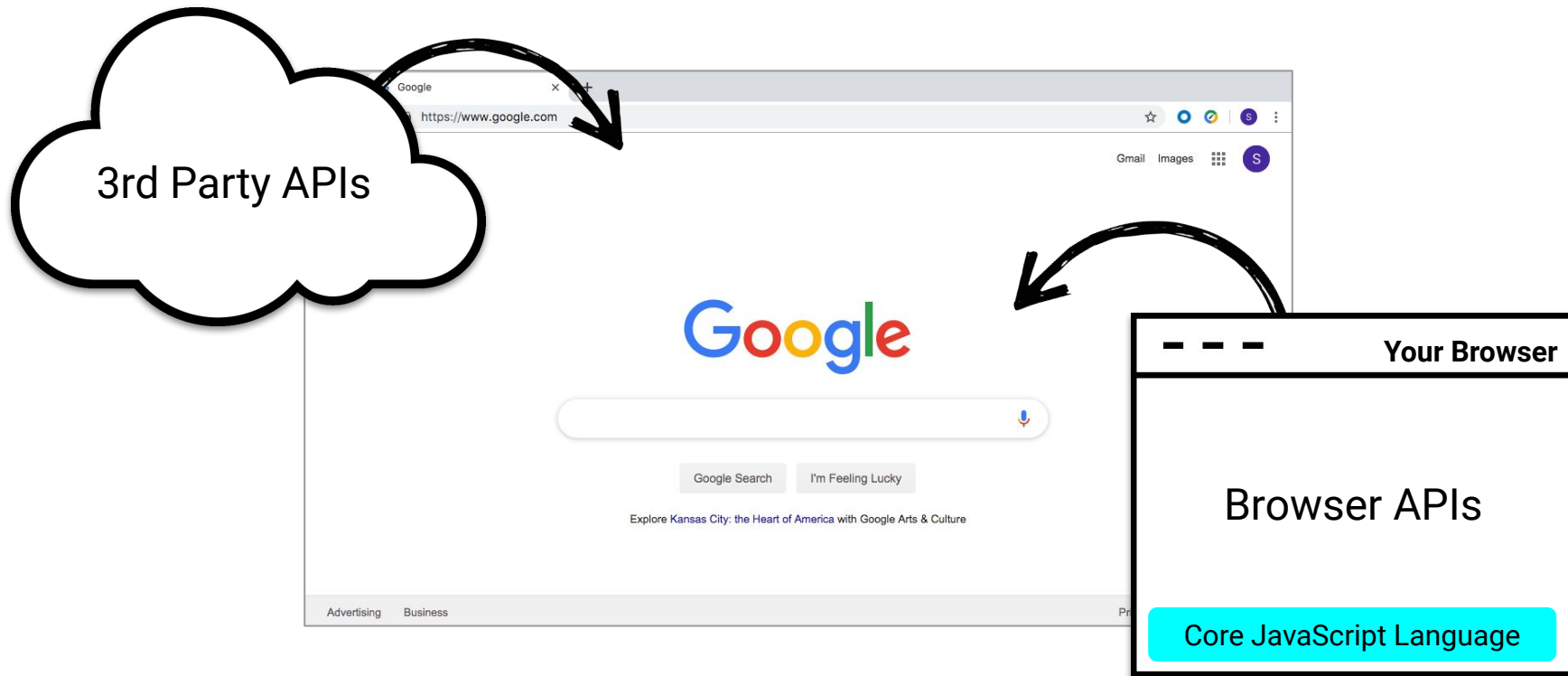
API Recap

In software development, APIs are often the bridge between different components.



API Recap

In software development, APIs are often the bridge between different components.

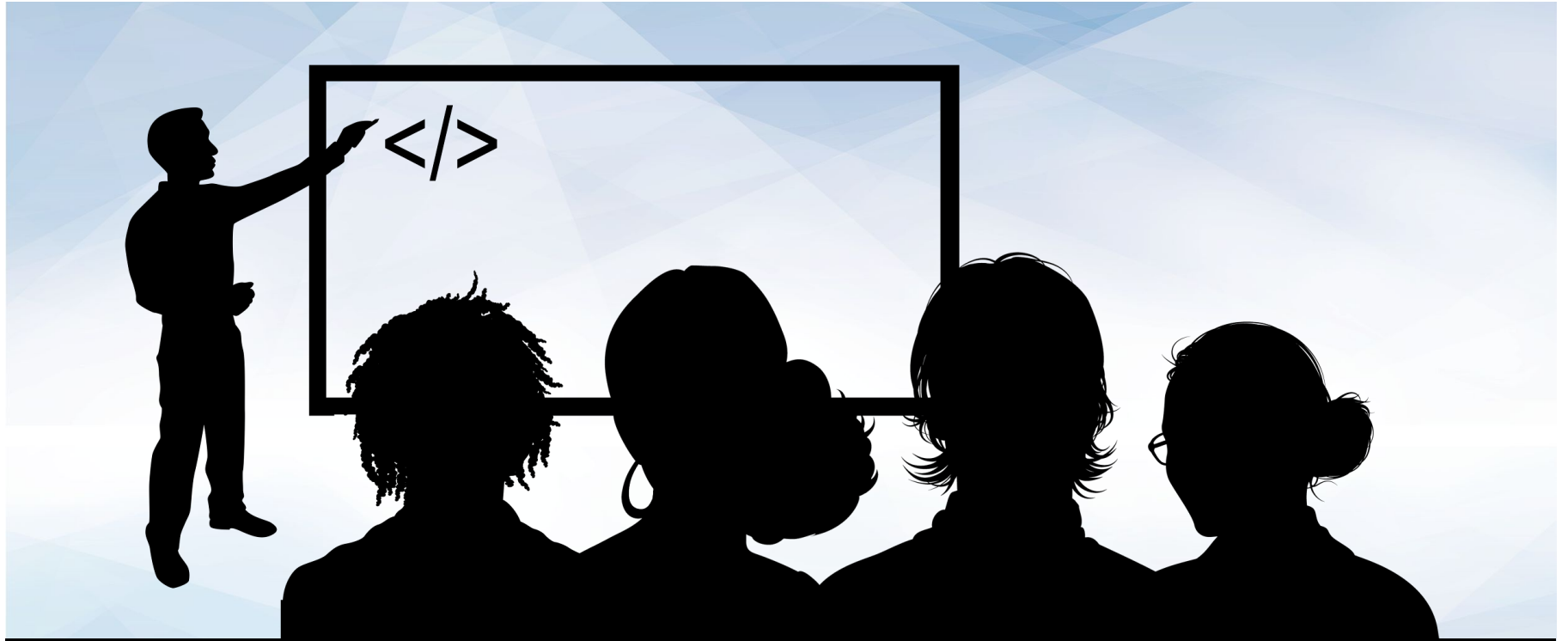


Quandl

A marketplace of financial data, Quandl collects and consolidates data from various sources, then makes it available to users. Quandl is ideal for extracting financial data to calculate ROI, risk-to-reward ratio, etc.

Users access Quandl's data via their API, which supports multiple programming languages, including Python.



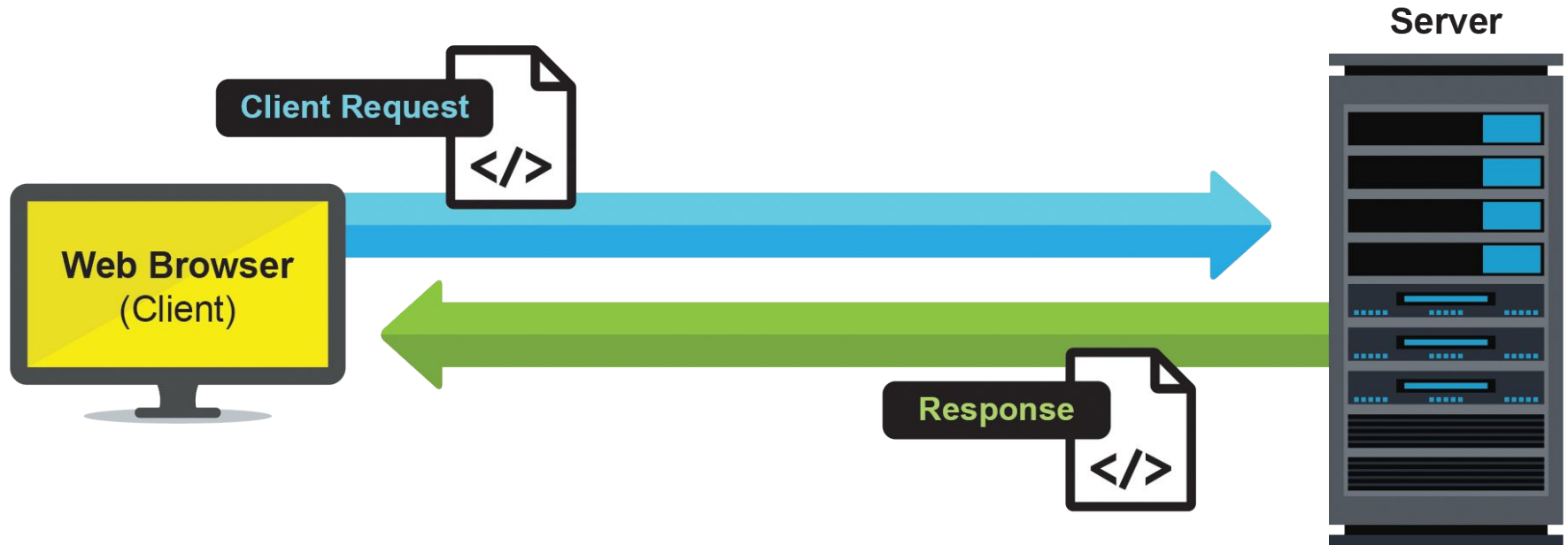


Instructor Demonstration
Review Homework

Client-Server Model

Client-Server Model

The client-server model is a structure that outlines the relationship and flow of communication between two components: a client and a server.



The Client-Server Model

A client is any tool or application used to connect to, or communicate with, a server. Example **clients** include:



Web browsers

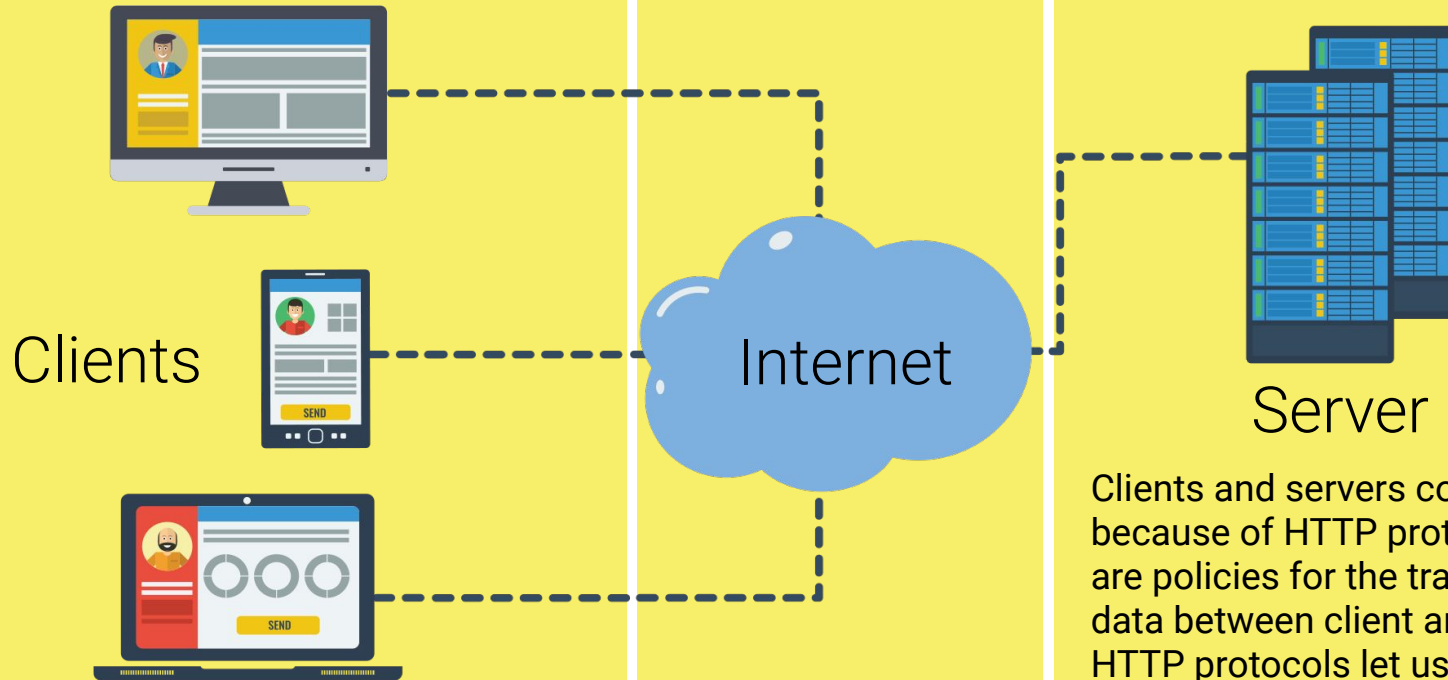


Mobile devices



Command-line interfaces

The Client-Server Model



Clients submit requests to servers, and then wait for a response from the server.

Clients and servers communicate because of HTTP protocols, which are policies for the transmission of data between client and server. HTTP protocols let users execute server functions, and clients and servers to exchange data.



Activity: Eavesdropping on the Server

In this activity, you will surf the web with the browser's developer console open, visit websites like Facebook and Yahoo, and complete a Google search. You will read the standard output from the console to get a better idea of what data is sent between client and server as you navigate sites.

(Instructions sent via Slack.)

Suggested Time:
10 Minutes





Time's Up! Let's Review.

Postman



Postman is a service that provides users with a UI to submit and store API calls and requests.



POSTMAN



API requests must be submitted in some type of development environment. Postman offers an API specific development environment that is free to users.



Postman is a great tool to use when onboarding onto a new API. All that is needed to execute an API with Postman is the request URL.



Because Postman is a development environment, users can save API requests, configure environments, and even create mock servers.



Activity: I Spy an API

In this activity, you will go through a list of FinTech APIs and test out their functionality using Postman. This will give you a better understanding of what Postman is, how it should be used, and will expose you to one of the most common tools used in the FinTech industry.

(Instructions sent via Slack.)

Suggested Time:
15 Minutes





Activity: Parlez-vous le JSON?

In this activity, you will choose a sub-selection of the JSON output to decipher, and then explain the sub-selection to a peer.

(Instructions sent via Slack.)

Suggested Time:
5 Minutes



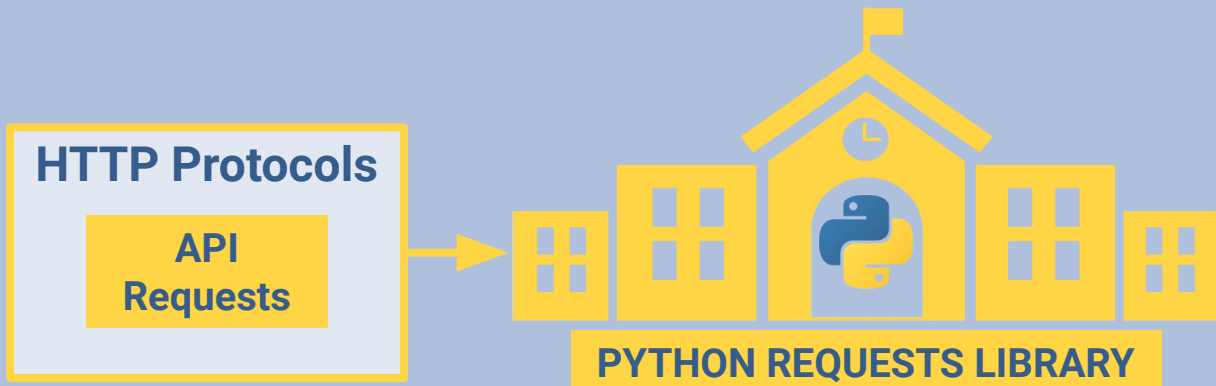


Time's Up! Let's Review.

Python Requests

Python Requests

Python's requests library is similar to Postman and can be used to make API calls, which lets developers submit API requests using HTTP protocols.



The requests library lets developers use Python like glue, connecting Python code with multiple third-party APIs. This allows developers to create programs that are a blend of multiple technologies!

Python Requests

- The requests library has its own functions, such as **GET** and **POST**. These can be used to execute API calls programmatically with Python.

- The difference between Python's request library and Postman is that **developers can use Python objects to make APIs interact with one another.**
- Allows developers to:
 - pass the output of one API as input to another API
 - utilize conditionals
 - leverage loops

Python Requests

Each type of request serves a different purpose.

GET

GET requests are used to extract/acquire data from a server

POST

POST requests are used to push new or updated data to the server

PUT

PUT requests are used to overwrite content on the server



Instructor Demonstration

Python Requests



Activity: Ice Breakers On Request

In this activity, you will be given a list of request URLs to execute using the Python requests library. You will also be able to put your JSON knowledge to use by interpreting JSON output.

(Instructions sent via Slack.)

Suggested Time:
20 Minutes





Activity: Engagement

In this activity, you will reveal an interesting fact or joke you discovered while working with APIs.

Suggested Time:
15 Minutes





Time's Up! Let's Review.





Countdown timer

15:00

(with alarm)

URL Parameters

URL Parameters

URL parameters serve as a means of configuring and changing API functionality.

Parameters can be specified in one of two ways. Parameters can follow / forward slashes, or be specified by parameter name and then parameter value.

Parameter provided after /

```
http://numbersapi.com/42
```

Parameter provided using
parameter name and value

```
http://numbersapi.com/random?min=10?json
```


URL Parameters

URL



```
graph TD; A[URL] --> B[http://www.site.com/page.html?parameter1=[@field:fieldname1]&parameter2=[@field:fieldname2]]
```

`http://www.site.com/page.html?parameter1=[@field:fieldname1]¶meter2=[@field:fieldname2]`

URL Parameters

URL

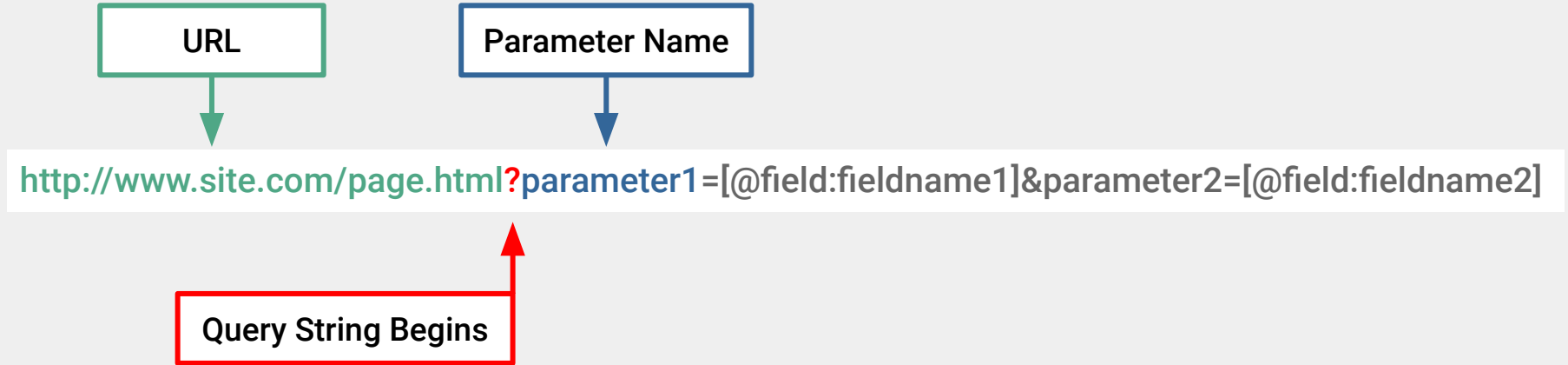


`http://www.site.com/page.html?parameter1=[@field:fieldname1]¶meter2=[@field:fieldname2]`

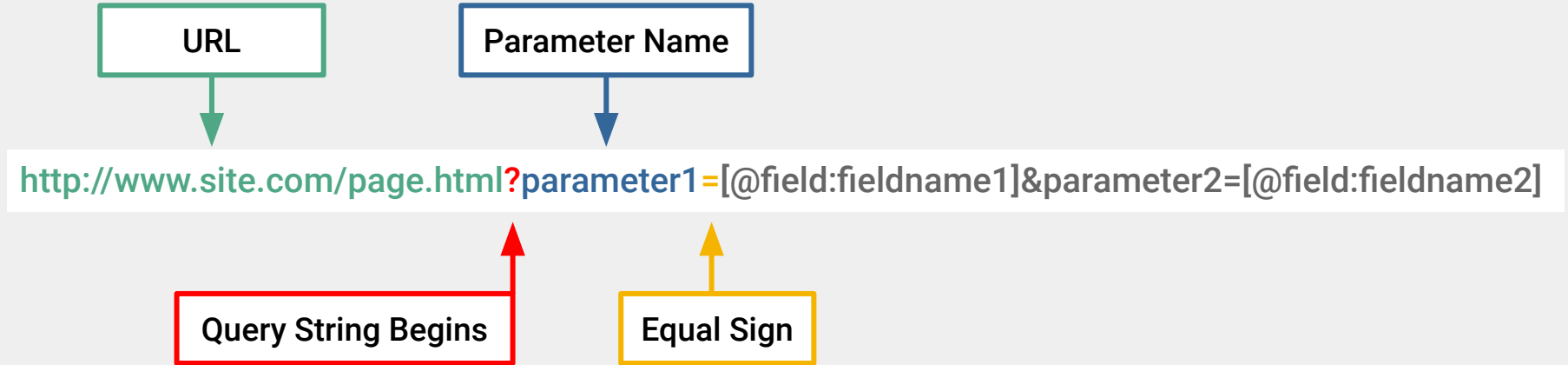
Query String Begins



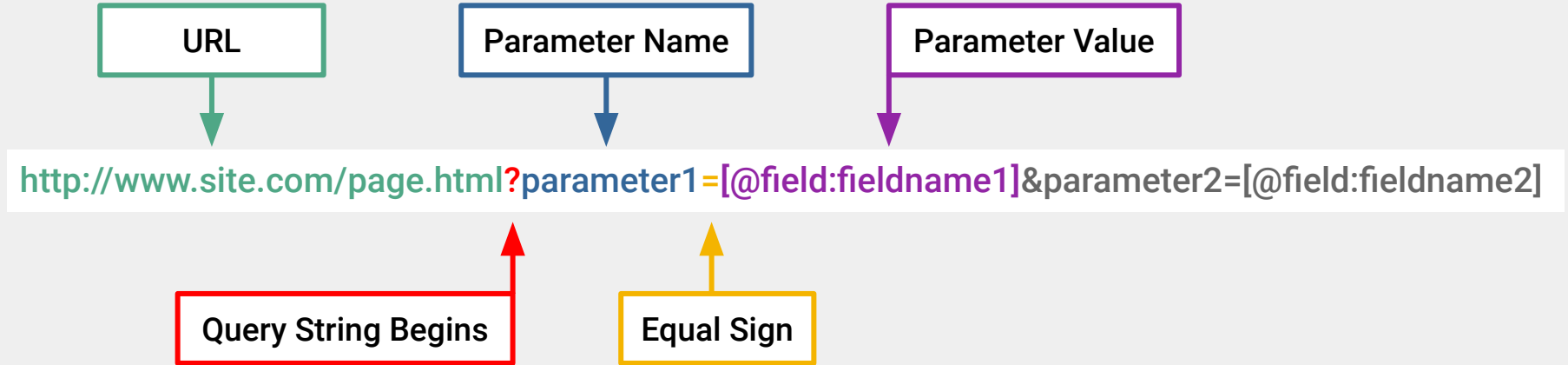
URL Parameters



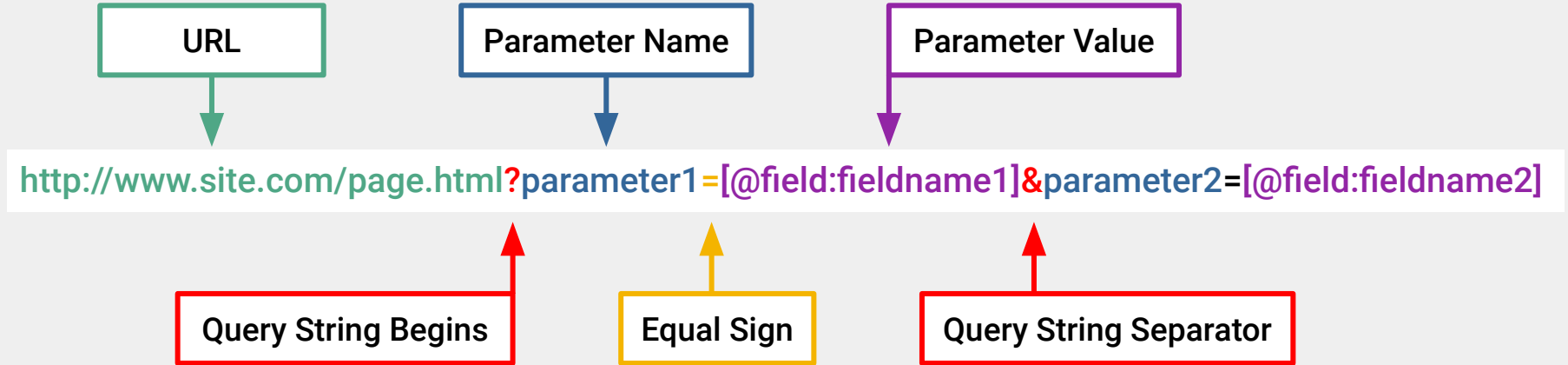
URL Parameters

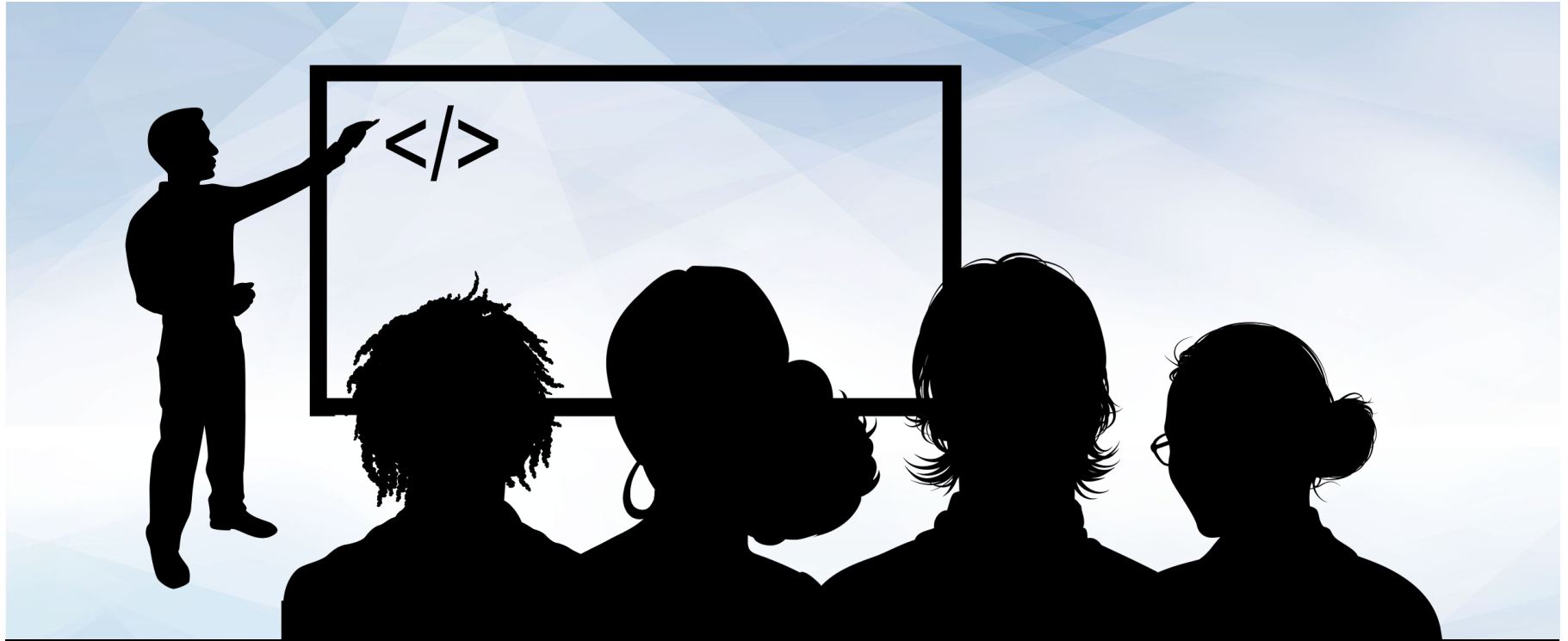


URL Parameters



URL Parameters





Instructor Demonstration

URL Parameters



Activity: House of Requests

In this activity, you will play a game of blackjack using the Deck of Cards API. You will test skills including the execution of GET requests using the Python requests library, extraction of JSON elements, and parameterization of API request URLs.

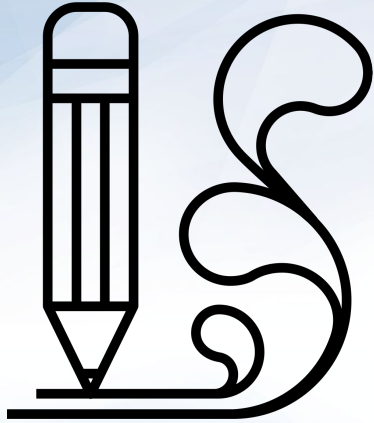
(Instructions sent via Slack.)

Suggested Time:
25 Minutes





Time's Up! Let's Review.



Homework: Sign Up for APIs

Sign up for these APIs before coming to the next class; you will use them in Lesson 5.2.

(Links sent via Slack.)

Due Date:
Next Class





Questions?

*The
End*