Welcome to the CoGrammar APIs

The session will start shortly...

Questions? Drop them in the chat. We'll have dedicated moderators answering questions.

Software Engineering Session Housekeeping

- The use of disrespectful language is prohibited in the questions, this is a supportive, learning environment for all - please engage accordingly. **(Fundamental British Values: Mutual Respect and Tolerance)**
- No question is daft or silly - **ask them!**
- There are **Q&A sessions** midway and at the end of the session, should you wish to ask any follow-up questions. Moderators are going to be answering questions as the session progresses as well.
- If you have any questions outside of this lecture, or that are not answered during this lecture, please do submit these for upcoming Academic Sessions. You can submit these questions here: **Questions**

Software Engineering Session Housekeeping cont.

- For all **non-academic questions**, please submit a query: **www.hyperiondev.com/support**
- Report a **safeguarding** incident: **www.hyperiondev.com/safeguardreporting**
- We would love your **feedback** on lectures: **Feedback on Lectures**

Skills Bootcamp
8-Week Progression Overview

Fulfil 4 Criteria to Graduation

✅ Criterion 1: Initial Requirements

- ***Timeframe:*** First 2 Weeks
- ***Guided Learning Hours (GLH):*** Minimum of 15 hours
- ***Task Completion:*** First four tasks

✅ Criterion 2: Mid-Course Progress

- ***Guided Learning Hours (GLH): 60***
- ***Task Completion:*** 13 tasks

CoGrammar

Skills Bootcamp Progression Overview

✅ Criterion 3: Course Progress

- ***Completion:*** All mandatory tasks, including Build Your Brand and resubmissions by study period end
- ***Interview Invitation:*** Within 4 weeks post-course
- ***Guided Learning Hours:*** Minimum of 112 hours by support end date (10.5 hours average, each week)

✅ Criterion 4: Demonstrating Employability

- ***Final Job or Apprenticeship Outcome:*** Document within 12 weeks post-graduation
- ***Relevance:*** Progression to employment or related opportunity

CoGrammar


SKILLS FOR LIFE
SKILLS BOOTCAMPS
Department for Education
CoGrammar
APIs
June 2024

Learning Objectives

- Define what an **API** is and explain its role in software development.
- Identify common **API** types (public, private, partner, internal) based on access restrictions.
- Grasp **REST API** Fundamentals, including HTTP methods (**GET**, **POST**, **PUT**, **DELETE**) and their usage in accessing resources.
- API Authentication and Authorization with common authentication methods (**API keys**, **OAuth**, **JWT**)

Learning Objectives

- Identify the structure of a **REST API endpoint** and interpret **API responses**.
- Develop a **client application** that **consumes** an existing **API** using Python's **requests**
- Develop and Deploy a Simple **RESTful API** Using **Flask** capable of **GET**ting and **POST**ing data

Poll

1. Imagine you're building a food delivery app. What connector (API) would helpful for this app?

    a. Weather API
    b. Payment Gateway API
    c. Social Media Login API
    d. Alien Language Translator API

Poll

2. You're building a fitness tracker app. Which connector (API) would likely provide the most relevant data?

a. Social Media Feed API
b. Weather API
c. Step Counter Sensor API
d. Movie Database API

Poll

3. You're browsing a travel website and see a section displaying live currency exchange rates. How might the website be using a connector (API)in this scenario?

    a. The website stores all the exchange rates internally and updates them manually,
    b. The website uses an API to access real-time exchange data from a financial service,
    c. The website calculates the exchange rates based on a complex algorithm,
    d. The website displays exchange rates based on your location

Introduction

CoGrammar

Intuition

In today's digital ecosystem, **APIs** (Application Programming Interfaces) are the glue that **connects** software systems. They enable seamless **communication** and **collaboration**, allowing different applications to interact, access functionalities, or **retrieve** data from other services, promoting **interoperability** and **efficiency**.

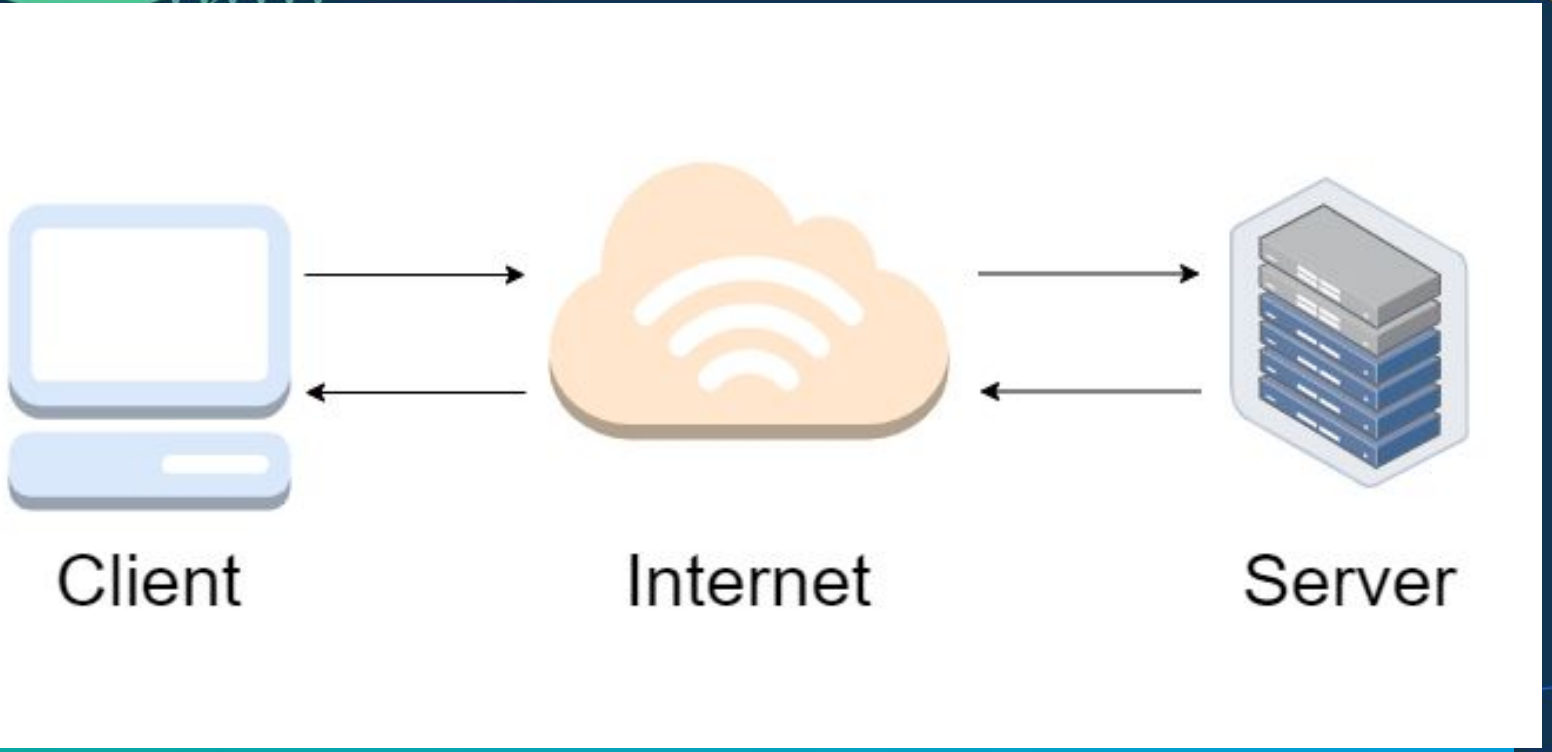
By integrating **third-party services** like **payment** gateways, enabling mobile apps to **fetch** data, or providing **weather** data and **social media feeds**, APIs empower developers to build **powerful**, connected **applications faster**. Understanding the **basics** of APIs unlocks new opportunities for innovation, efficiency, and growth, driving collaboration across various platforms and industries.

APIs

- **API:** Application Programming Interface – Interface that provides a set of functions to a user where the underlying mechanism of that function is hidden.

- **Web API:** It is an API that travels through the internet. Usually called client-server architecture.

Web APIs



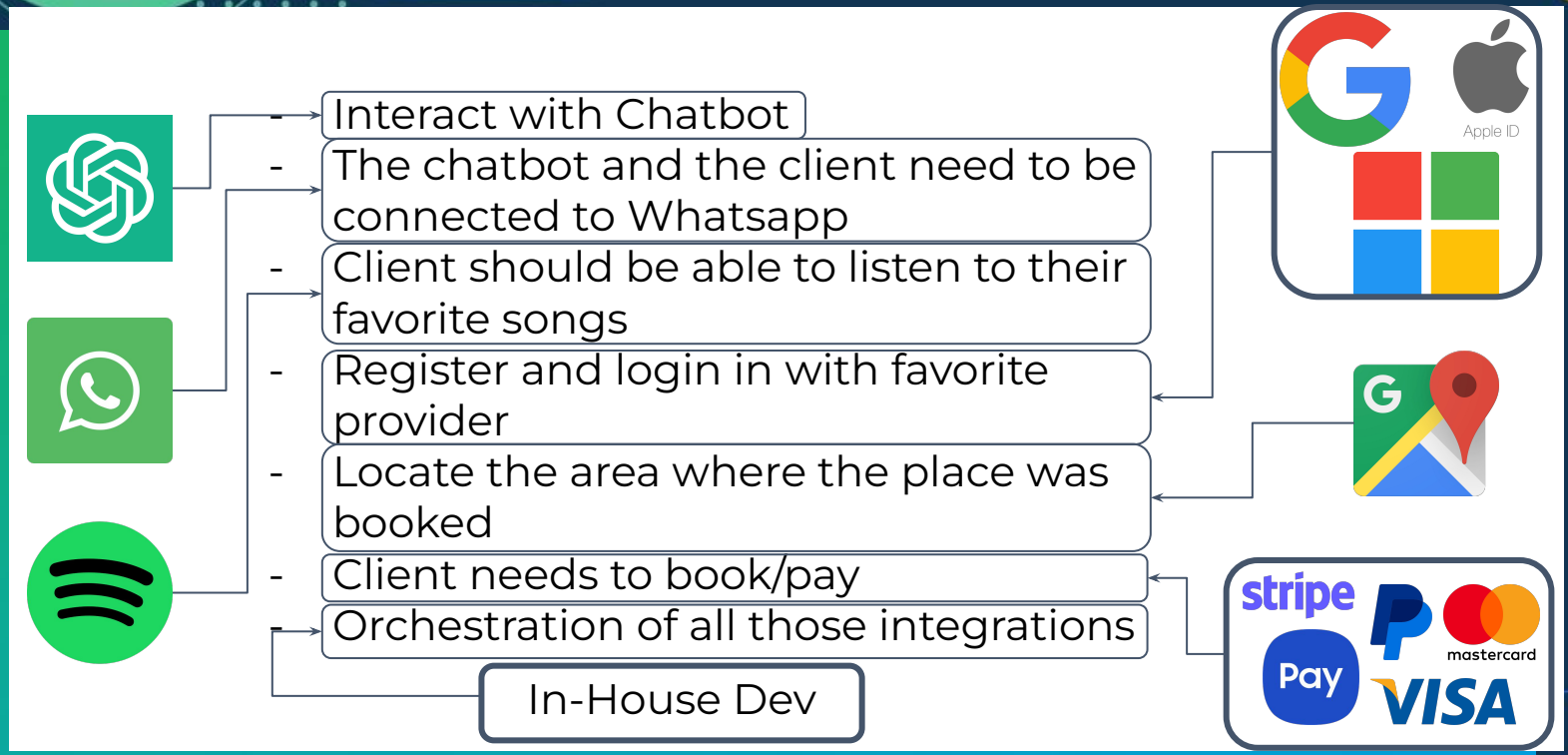
API Terminologies

Integration: API integration refers to the process of **connecting** two or more **applications** or **systems** by using **APIs** (Application Programming Interfaces) to **exchange data** and **perform actions.**

Call: The process of **sending** a **request** to your API after setting up the right **endpoints**

Monetization: API monetization is a process by which a business can create **revenue** from its **APIs**.

API Integration



HTTP and Status Codes

HTTP : **H**yper**T**ext **T**ransfer **P**rotocol

https://www.google.com/search/q=python

200 **400** **500**

HTTP Payload Format

Payload

```
<?xml version="1.0" encoding="UTF-8"?>
<root>
    <name>Tim</name>
    <age>23</age>
    <salary>50000</salary>
</root>
```

```
{
    "name: "Tim",
    "age: "23",
    "salary: "50000"
}
```

REST

Representational State Transfer

Endpoint

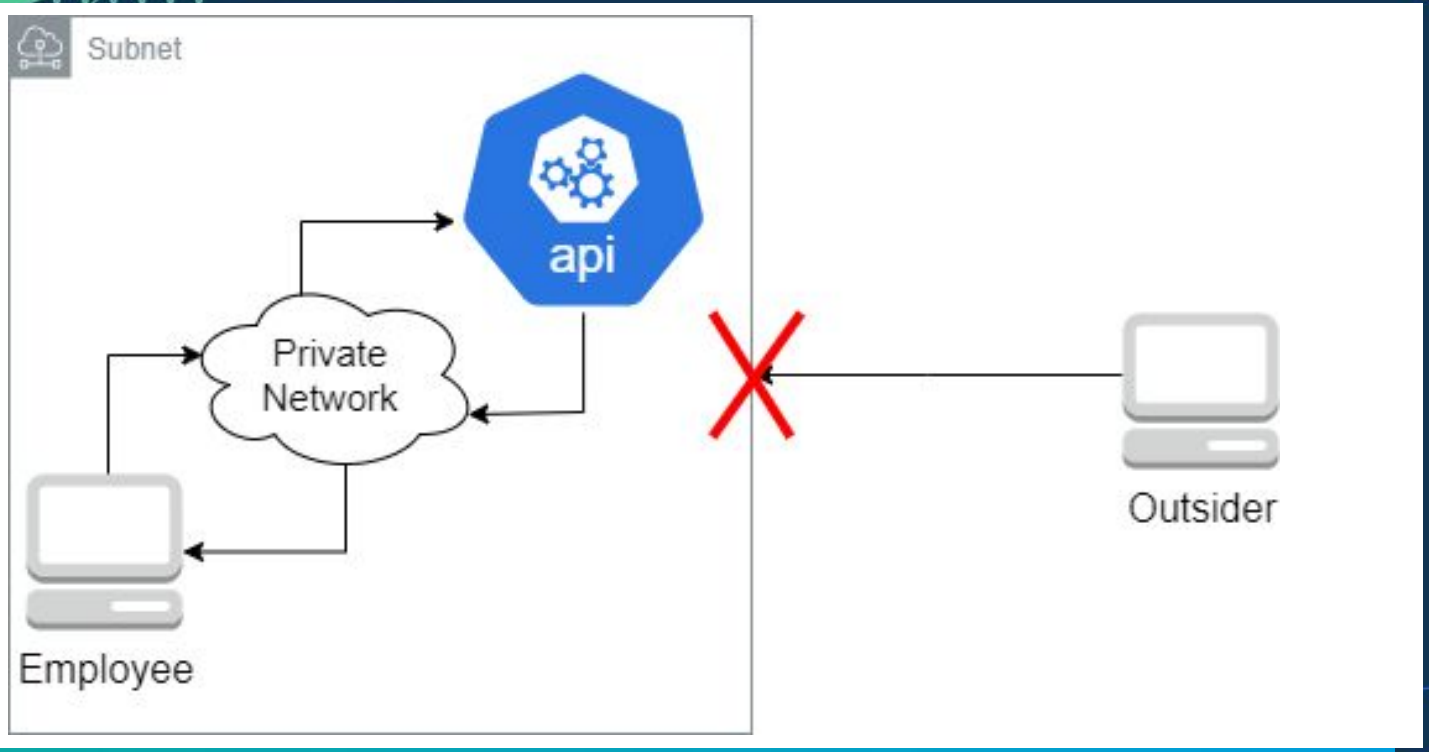
Method

Header

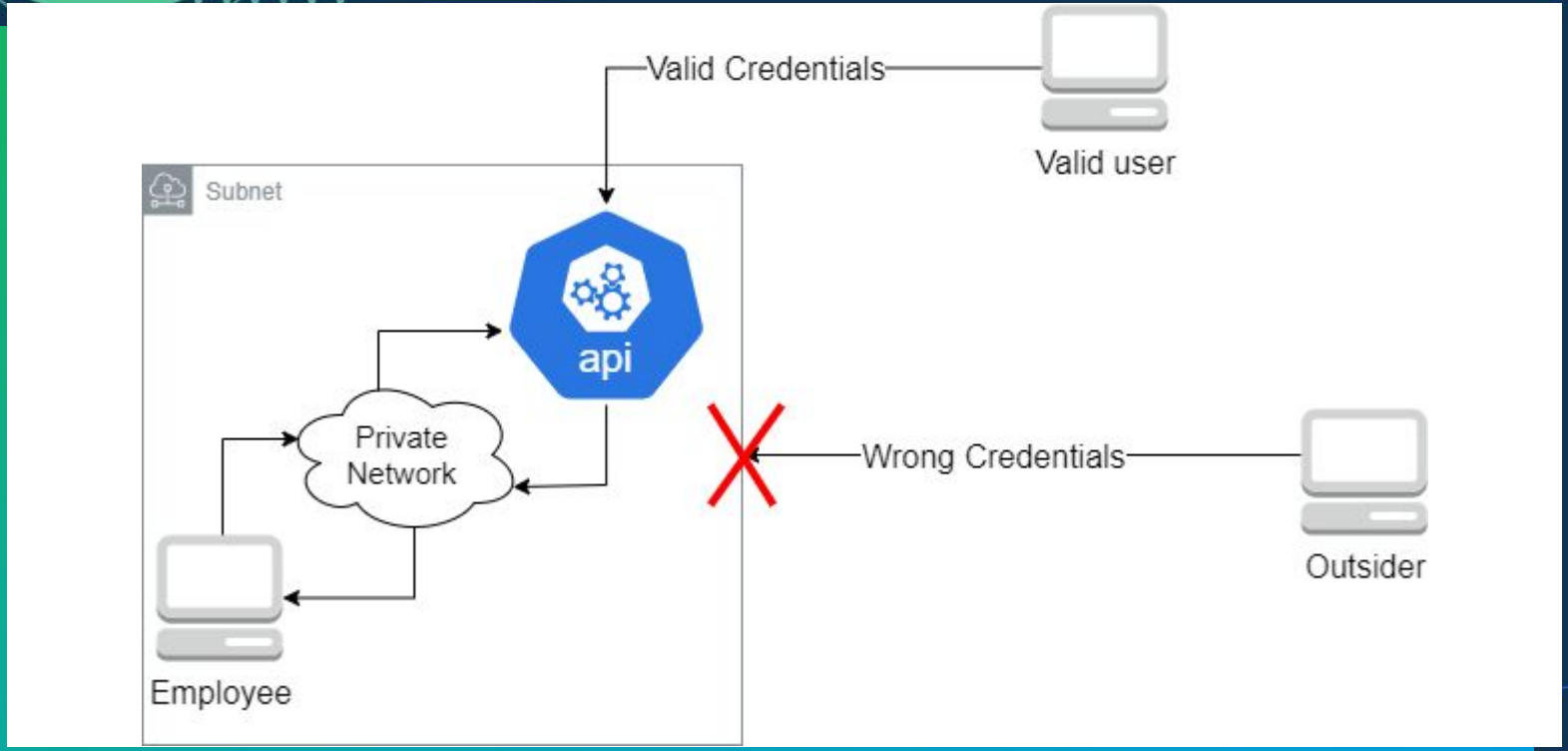
Body

https://dummy.restapiexample.com/

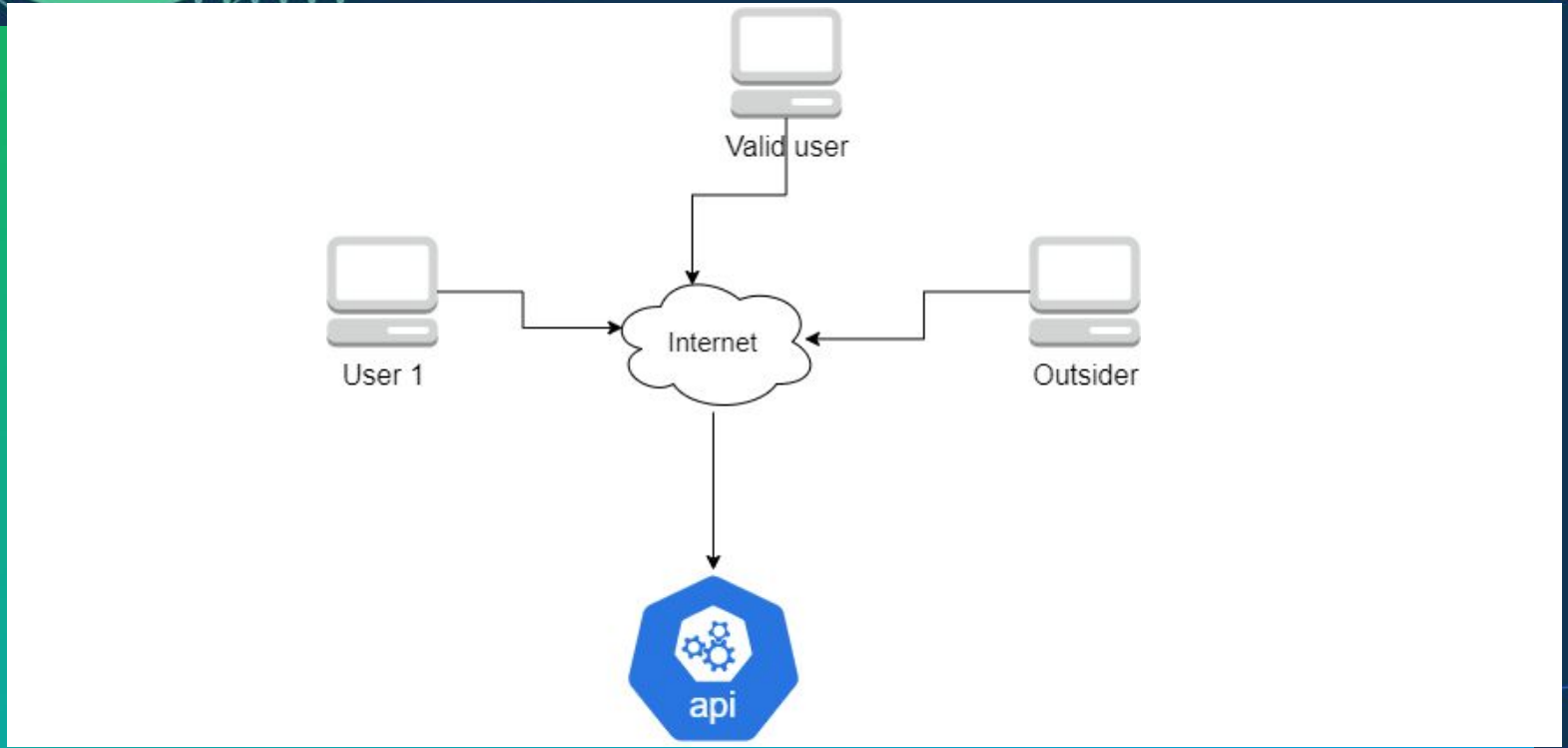
API Types: Private



API Types: Partner



API Types: Public



Backend API

Flask

VS

dj

Django

These differences will help you understand which one you should choose for your web application.

Framework	WSGI Framework	Full-stack Framework
Database	SQLAlchemy	Built-in ORM
Flexibility	Complete flexibility	Feature-packed
Design	Minimalistic Design	Batteries Included
Forms	Integrates with WTForms	Uses ModelForms
Template Engine	Jinja2 templating engine	Django Template language (DTL)
Learning curve	Simple to learn	Might take time to learn
Community support	Low community support	Big developer community
Applications	Netflix, Uber, Lyft, Red Hat	Pinterest, Instagram, Accenture

cynoteck.com

CoGrammar

Borrowed from:

https://cynoteck.com/blog-post/flask-vs-django/

Let’s get coding!

CoGrammar

Agenda

1. Introduce Postman for making API requests.
2. Guide learners through a step-by-step process of making basic GET and POST requests to a public API (e.g., Chat GPT 4 API , Rapid API, Free APIs , Chat GPT 3)
3. Setup Flask and build a Simple API

Agenda

4. Create Endpoints: Show how to create a few basic endpoints (GET and POST).
5. Run the API: Demonstrate how to run the Flask app and test the endpoints using Postman or curl.
6. Consume the API Using Requests Library

Final Assessment

CoGrammar

Poll

1. Which data format is **commonly** used for API responses?

   a. XML
   b. CSV
   c. JSON

Poll

2. REST APIs leverage HTTP methods for communication. What does the HTTP status code "**404 Not Found**" typically indicate in an API response?

   a. Successful request processing
   b. User authentication failure
   c. The requested resource does not exist
   d. Internal server error

CoGrammar

Poll

3. APIs can be used to retrieve data in various formats. What are some advantages of using JSON as a data format for API communication?

a. JSON is more secure than other formats
b. JSON is human-readable and easy to understand
c. JSON is the only format supported by all APIs
d. JSON requires less processing power than other formats

Lesson Conclusion and Recap

CoGrammar

Lesson Conclusion and Recap

- APIs: The Glue of the Digital World: APIs (Application Programming Interfaces) act as bridges between software systems, enabling seamless communication, data exchange, promoting interoperability and integration.
- RESTful Foundations: REST (Representational State Transfer) is a common architectural style for APIs, using HTTP methods (GET, POST, PUT, DELETE) to interact with resources.
- Consuming APIs: With Python's requests library, we can send HTTP requests to interact with APIs, retrieving data with GET requests and sending data with POST requests.
- Creating APIs with Flask: Using Flask, we can easily set up a web server and define endpoints to handle various HTTP requests, making it simple to create and test APIs.

Homework or Follow-up Activities

CoGrammar

Homework or Follow-up Activities

1. **Mid-Hard**
    a. Let's assume that you are the CEO of a startup. Your main idea is to collect data from estate agent companies around London, say company A, B and C for a start.
    b. Your solution will happen in 2 folds:
        i. You provide a webpage where customers can log on and be recommended the best property to buy for their budget. On the front end, the web page.
        ii. You will need to make money.
            1. That would happen through premium services, like showing more than one house if the fee is paid. This is still the frontend,
            2. Now, banks need your solution. Create an API endpoint for the same service, billing them per request.

Homework or Follow-up Activities

Instructions:

- The assumption here for exercise 2:
  - You will create your population in terms of the people who will register through the Django form or a similar form using Flask. Get all the required and validated data, name, age, home address ...
  - The houses will also be fictitious. Create enough variation in prices and location such that it is random enough
  - You will use the Django REST framework or Flask for the rest endpoints. Please use the provided code, used during the practical.

References

- Flask Documentation
- Postman
- Requests Library Documentation
- REST API Tutorial
- What is a Web API?
- https://pypi.org/project/requests/

- https://platform.openai.com/docs/api-reference/audio/createSpeech

Thank you for attending

SKILLS FOR LIFE
SKILLS BOOTCAMPS

Department for Education

CoGrammar