1. How do web APIs work? Why are they useful?

Web APIs (Application Programming Interfaces) are a set of rules and protocols that allow different software applications to communicate and interact with each other over the internet. It is an application programing for either web server or web browser.

*1. **Request and Response Model**: Web APIs work on a client-server model, where one application acts as the client that makes requests to the server, which hosts the API. The client sends an HTTP request to the server, specifying the desired action and data. The server processes the request and sends back an HTTP response, containing the requested data or the result of the action.*

*2. **Data Exchange Formats**: APIs often use common data exchange formats such as JSON (JavaScript Object Notation) or XML (eXtensible Markup Language) to represent the data being sent and received.*

*3. **RESTful APIs**: Representational State Transfer (REST) is style for designing web APIs. RESTful APIs use standard HTTP methods like GET (retrieve data), POST (create data), PUT/PATCH (update data), and DELETE (remove data) to perform operations on resources. The URL identifies the resource, and the HTTP method specifies the action to be performed on that resource.*

*4. **Statelessness**: Web APIs are usually stateless, meaning each request from a client to a server must contain all the information necessary to understand and process the request. The server does not retain any knowledge of the client's previous requests.*

*5. **Cross-Platform Compatibility**: Web APIs allow different applications to communicate and work together regardless of the programming languages, platforms, or devices they are running on. This cross-platform compatibility is one of the main reasons for their popularity.*

Now, why are web APIs useful?

*1. **Integration and Interoperability**: Web APIs enable different software systems, services, and platforms to work together seamlessly. Businesses can integrate their applications with external services to enhance functionality, offer new features, and streamline processes.*

*2. **Data Access and Sharing**: APIs provide a standardized way to access data from various sources, like social media platforms, weather services, financial institutions.*

*3. **Third-Party Development**: Companies can open up their APIs to third-party developers, allowing them to build applications or services that use the company's data or functionality. This encourages innovation and extends the reach of the platform or service.*

*4. **Modularity and Scalability**: APIs promote a modular approach to software development, where different components can be developed independently. This modular design enhances scalability, as new features can be added or updated without affecting the entire system.*

*5. **Mobile App Development**: APIs play a crucial role in mobile app development, allowing mobile applications to communicate with backend servers and access data and services in real-time.*

Web APIs allows developers to create complex functionality so much more easy. The construct of it, helps developers to easily communicate with the server and other developers over the internet.
https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Client-side_web_APIs/Introduction

2.

Look up 3 different public web APIs online and describe 3 endpoints from each.

In your description include the following:

(a) The URI (or endpoint)

(b) All of the HTTP verb methods that can be sent to that endpoint.

(c) A description of what happens when each verb method is sent.

*1. **GitHub API**:*

Source: https://developer.github.com/v3/

(a) Endpoint: https://api.github.com/users/{username}

(b) HTTP Verb Methods: GET

(c) Description: Sending a GET request to this endpoint with a specific GitHub username retrieves information about that user, such as their public profile, repositories, followers, and other relevant details.

(a) Endpoint: https://api.github.com/repos/{owner}/{repo}

(b) HTTP Verb Methods: GET

(c) Description: A GET request to this endpoint with the owner's username and repository name fetches details about a specific GitHub repository, including information about the repository, its branches, issues, contributors, and more.

(a) Endpoint: https://api.github.com/search/repositories

(b) HTTP Verb Methods: GET

(c) Description: By sending a GET request to this endpoint with specific search parameters, developers can search for GitHub repositories based on various criteria, such as keywords, stars, forks, and more.

*2. **Chuck Norris Jokes API**:*

Source: https://api.chucknorris.io/

(a) Endpoint: https://api.chucknorris.io/jokes/random

(b) HTTP Verb Methods: GET

(c) Description: A GET request to this endpoint retrieves a random Chuck Norris joke. The API response includes a humorous statement or fact about Chuck Norris.

(a) Endpoint: https://api.chucknorris.io/jokes/categories

(b) HTTP Verb Methods: GET

(c) Description: This endpoint provides a list of all available joke categories related to Chuck Norris. Sending a GET request to this endpoint fetches the categories, allowing developers to filter jokes based on specific topics.

(a) Endpoint: https://api.chucknorris.io/jokes/search

(b) HTTP Verb Methods: GET

(c) Description: By sending a GET request to this endpoint with a specific keyword, users can search for Chuck Norris jokes containing that keyword. The API responds with a list of jokes matching the search query.

*3. **The Cat API**:*

Source: https://thecatapi.com/

(a) Endpoint: https://api.thecatapi.com/v1/images/search

(b) HTTP Verb Methods: GET

(c) Description: When a GET request is sent to this endpoint, the API returns a random image or a list of random images of cats. This endpoint is used to fetch cat images for entertainment purposes, for example, to display random cat images in an application.

(a) Endpoint: https://api.thecatapi.com/v1/breeds

(b) HTTP Verb Methods: GET

(c) Description: By sending a GET request to this endpoint, the API provides a list of cat breeds along with their characteristics and details. Developers can use this endpoint to get information about various cat breeds programmatically.

(a) Endpoint: https://api.thecatapi.com/v1/favourites

(b) HTTP Verb Methods: GET, POST, DELETE

(c) Description: This endpoint allows users to interact with their favorite cat images.

A GET request returns a list of the user's favorite cat images. A POST request is used to add an image to the user's favorites. And a DELETE request removes a specific favorite image from the list.

(b) HTTP Verb Methods: GET

(c) Description: A GET request to this endpoint retrieves a list of all Pokémon abilities available in the API. The response includes information about each ability, such as its name, description, and effects.

(a) Endpoint: https://pokeapi.co/api/v2/type

(b) HTTP Verb Methods: GET

(c) Description: By sending a GET request to this endpoint, the API returns a list of all Pokémon types, such as Grass, Fire, Water, etc. The response provides details about each type, including its strengths, weaknesses, and relations with other types.