Web Front-End Development
Week 6: Server APIs

Many of today's web applications are part of a large ecosystem where applications are built on top of other applications, integrate with other applications or use data from other applications.

In web applications APIs are often used to integrate with other web applications and access their data. Web pages rarely include static data, so we're going to look at how you can use APIs to access data from other sites and incorporate into your web site. APIs usually deliver data in the JSON format. We accessed JSON data from our own site, now let's look at how we can use another site's API to access their data.

Many large web sites including Google, Facebook, Twitter, Yelp, Instagram, etc. provide APIs to let developers access their data.
You have to look at their APIs and documentation to understand what data is accessible and how you can access it.

Authentication
APIs use different authentication techniques to authenticate a client.

- Basic authentication
  - Username/password
  - Using the same username and password to access the API and manage the account is not ideal
  - Sometimes you want the client to have different permissions than the account owner
- API Key Authentication
  - requires the API to be accessed with a unique key which is a string of characters that uniquely identifies you in your script
  - distinct from the account owner's login password so they do not grant access to any account information, and are not used for authorization.
  - Whatever limits are placed on the key affect every client equally
  - Using an API key does not require user action or consent.
  - Each site can implement API key authorization differently, no standard
- Open Authorization (OAuth)
  - Automates the exchange of the access code
  - Ability to set different limits for different clients
  - becoming the most widely used authentication scheme on the web
  - this must be supported by the server running your web site. The creative server does NOT support OAuth so we can't use APIs using this method.

Youtube
Google has many APIs available depending on the data you're looking for.
https://developers.google.com/apis-explorer/#p/
Let's look at the YouTube Data API
Home
https://developers.google.com/youtube/v3/

A good API is well documented
Reference
https://developers.google.com/youtube/v3/docs
Guides
https://developers.google.com/youtube/v3/getting-started
Google Developer's Console https://console.developers.google.com/iam-admin/projects?pli=1
Create Project
Name: YouTube example
Organization: colorado.edu
Location: colorado.edu
In the API library manager look under YouTube APIs and click on YouTube Data API v3
Enable
Now we need credentials.
Create Credentials
Select the API, Web browser, and public data
This will give you an API key.
You can now use this API key to access the YouTube APIs.

Setting up API keys
https://support.google.com/googleapi/answer/6158862?hl=en&ref_topic=7013279

Best practices for securely using API keys
https://support.google.com/googleapi/answer/6310037

- Do not embed API keys directly in code, instead store them in environment variables or in files outside of your application's source tree.
- Do not store API keys in files inside your application's source tree to help ensure your keys do not end up in your source code control system (such as GitHub).
- Restrict your API keys to be used by only the IP addresses, referrer URLs, and mobile apps that need them.
- Restrict your API keys to be usable only for certain APIs
- Delete unneeded API keys
- Regenerate your API keys periodically. Old keys will continue to work for 24 hours after you generate replacement keys.

Example:
https://repl.it/@aileenjp/Youtube-API-JSON-Fetch

- Search API reference https://developers.google.com/youtube/v3/docs/search/list
  - part: snippet
  - q: cats
- Look at the JSON sent back as the response. Same as https://www.googleapis.com/youtube/v3/search?key=AIzaSyBz7S_VPKC4Y_Z9CfrRGEGMQgiul1v3dc8&part=snippet&maxResults=10&q=cats
- Use the URL interface to create a URL object https://developer.mozilla.org/en-US/docs/Web/API/URL
- Use the URLSearchParams interface set the search parameters https://developer.mozilla.org/en-US/docs/Web/API/URLSearchParams

- Use the Fetch API to make the request
- By requesting the part “snippet” we’re returned an object that includes an items property which is an array of objects. Each object has a snippet property which is an object with keys such as “title” and a “thumbnails” object with a default object that has a url property whose value is a url to a thumbnail image.
- Each item also includes an “id” property that’s an object with a “videoId” key whose value we can use to link to each video in an iframe in our page.

Lab: Create a web page that loads and displays JSON data from an API.

API Directory http://www.programmableweb.com/apis/directory
Public APIs (shows Auth method) https://github.com/public-apis/public-apis