GENERAL ASSEMBLY

JAVASCRIPT DEVELOPMENT

Sasha Vodnik, Instructor

HELLO!

1. Pull changes from the `svodnik/JS-SF-15-resources` repo to your computer
2. Open the `11-advanced-apis` folder in your editor
3. If you haven't already,
   - download Postman from https://getpostman.com
   - sign up for a Flickr account at https://flickr.com (or sign in with an existing Yahoo account)

LEARNING OBJECTIVES

At the end of this class, you will be able to

- Request data from a web service.
- Implement the geolocation API to request a location.
- Use Postman to construct and test an API request.
- Process a third-party API response and share location data on your website.
- Search documentation needed to make and customize third-party API requests.

AGENDA

- Configure Flickr account
- Implement geolocation
- Set up Postman
- Create and send API call
- Handle API response

WEEKLY OVERVIEW

WEEK 6	Asynchronous JS & callbacks / Advanced APIs
WEEK 7	Project 2 lab / Prototypal inheritance
WEEK 8	Closures & `this` / CRUD & Firebase

EXIT TICKET QUESTIONS

1. Will we talk about resolving/rejecting promises? Or is that out of scope for this class?
2. Are there any websites or resources that I could use to practice more of the concepts used in class?
3. While creating functions that rely on fetch data, is there any worry about needing to create too many single use functions?

ADVANCED APIS

BUILDING OUR APP

1. Get user's location
2. Create request to Flickr with user's location info
3. Parse API response and add returned images to view

BUILDING OUR APP

Our app

1

- Get user's location

2

- Create request containing user's location info

flickr.com server

- Parse API response
- Add returned images to view

3

ENDPOINTS

Basics

Authentication

- GET oauth/authenticate
- GET oauth/authorize
- POST oauth/access_token
- POST oauth/invalidate_token
- POST oauth/request_token
- POST oauth2/invalidate_token
- POST oauth2/token

Accounts and users

Create and manage lists

- GET lists/list
- GET lists/members
- GET lists/members/show
- GET lists/memberships
- GET lists/ownerships

EXERCISE

OBJECTIVE

- Search documentation needed to make and customize third-party API requests.

TIMING

4 min

1. Read the documentation for at least 2 endpoints ("API methods") from the list at https://www.flickr.com/services/api/
2. Identify an endpoint that will let us find photos based on a user's location.

Get User's Location

Call the Flickr endpoint

Handle the Response

EXERCISE

OBJECTIVE

- Process a third-party API response and share location data on your website.

TIMING

15 min

1. Create a handleResponseSuccess callback function to do the following:
   - Iterate through your response data, creating an `img` element each time with the given image URL from the API.
   - Add the class `image` to the `img` element
   - Append the new `img` element to the element with the class `images`, which already exists in the HTML.

Customize Search Results

EXERCISE

OBJECTIVE

- Search documentation needed to make and customize third-party API requests.

TIMING

until 9:20

Search the API documentation as necessary to modify your API request to do the following:

- Return 30 photos instead of the default 100
- Sort results by relevance

Bonus 1: Return URLs for larger images (Hint: Check out the extras argument at https://www.flickr.com/services/api/flickr.photos.search.html and look at the Size Suffixes section at https://www.flickr.com/services/api/misc.urls.html).

Bonus 2: Instead of landscapes, return photos from a different category (see tags at https://www.flickr.com/photos/tags/)

Bonus 3: Rewrite your app to use the Flickr SDK (https://github.com/flickr/flickr-sdk)

Exit Tickets!

(Class #11)

LEARNING OBJECTIVES - REVIEW

- Request data from a web service.
- Implement the geolocation API to request a location.
- Use Postman to construct and test an API request.
- Process a third-party API response and share location data on your website.
- Search documentation needed to make and customize third-party API requests.

NEXT CLASS PREVIEW

In-class lab: Feedr

- Familiarize yourself with the API documentation for news sources.
- Fork and clone your starter code.
- Strategize ways to hide the loader and replace the content of the `#main` container with that of the API.
- Integrate string and variable values using template literals

Q&A