JAVASCRIPT DEVELOPMENT

Sasha Vodnik, Instructor

HELLO!

1. Pull changes from the `svodnik/JS-SF-13-resources` repo to your computer

2. Open the `11-advanced-apis > starter-code` folder in your code 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 7	(holiday) / Advanced APIs
WEEK 8	Project 2 Lab / Prototypal inheritance
WEEK 9	Closures & the Module Pattern / CRUD & Firebase

HOMEWORK REVIEW

HOMEWORK — GROUP DISCUSSION

EXERCISE

TYPE OF EXERCISE

▸ Pairs

TIMING

6 min

1. Share your solutions for the homework.

2. Share 1 thing you found challenging. If you worked it out, share how; if not, brainstorm with your group how you might approach it.

ADVANCED APIS

BUILDING OUR APP

1. Configure our systems for development and testing
2. Get user's location
3. Create request to Flickr with user's location info
4. Parse API response and add returned images to view

BUILDING OUR APP

Our app

flickr.com server

3

2

‣ Get user's location

‣ Create request containing user's location info

‣ Parse API response
‣ Add returned images to view

4

ENDPOINTS

‣ Examples from openweathermap.org

By geographic coordinates

API call:

api.openweathermap.org/data/2.5/weather?lat={lat}&lon={lon}

Parameters:

lat, lon coordinates of the location of your interest

By city name

API call:

api.openweathermap.org/data/2.5/weather?q={city name}

api.openweathermap.org/data/2.5/weather?q={city name},{country code}

By ZIP code

Description:

Please note if country is not specified then the search works for USA as a default.

API call:

api.openweathermap.org/data/2.5/weather?zip={zip code},{country code}

EXERCISE

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.

EXERCISE

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 popular tags at https://www.flickr.com/photos/tags/)

EXERCISE

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.

‣ Look up other news sources that might be useful for the project.

Q&A