

# JAVASCRIPT DEVELOPMENT

Sasha Vodnik, Instructor

### **HELLO!**

1. Pull changes from the svodnik/JS-SF-7 repo to your computer

(there is no starter code folder for tonight)

#### **JAVASCRIPT DEVELOPMENT**

## IN-CLASS LAB: FEEDR

### **LEARNING OBJECTIVES**

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

- 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.

### **AGENDA**

- Project 2 overview and questions
- Project 2 lab time

### **WEEKLY OVERVIEW**

WEEK 7

Asynchronous JS & callbacks / Advanced APIs

**WEEK 8** 

Project 2 Lab / this & Module pattern

WEEK 9

CRUD & Firebase / Deploying your app

## HOMEWORK REVIEW

8

## **Checkin and questions**

- The most significant thing I learned about using APIs is \_\_\_\_\_\_.
- My biggest outstanding question about using APIs is \_\_\_\_\_\_.

#### **EXERCISE** — PROJECT PLANNING



#### TYPE OF EXERCISE

▶ Individual, then groups of 2-3

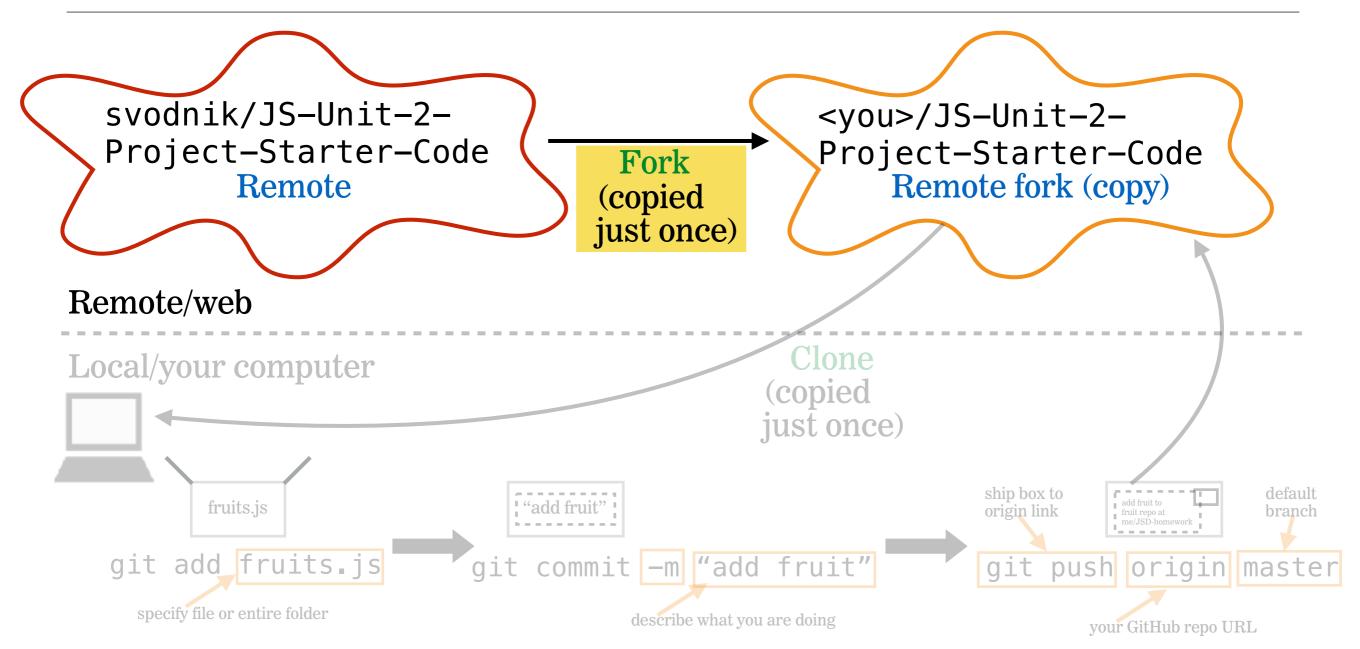
#### **TIMING**

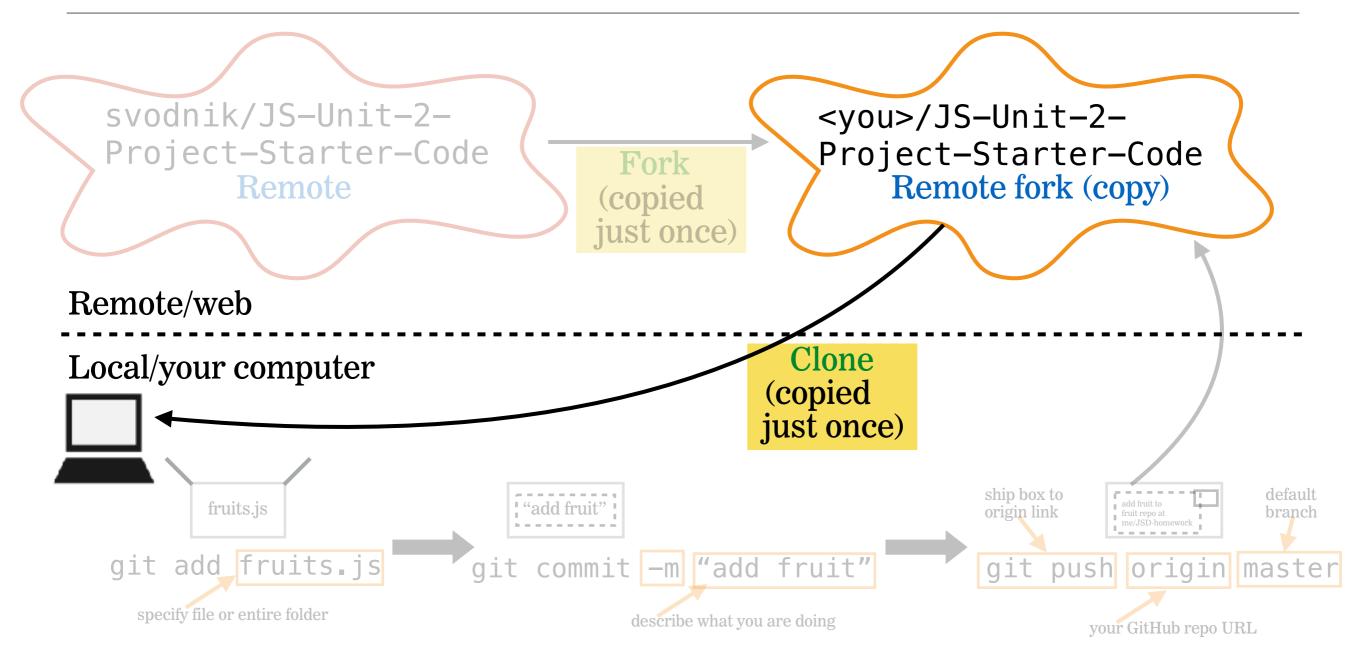
5 min

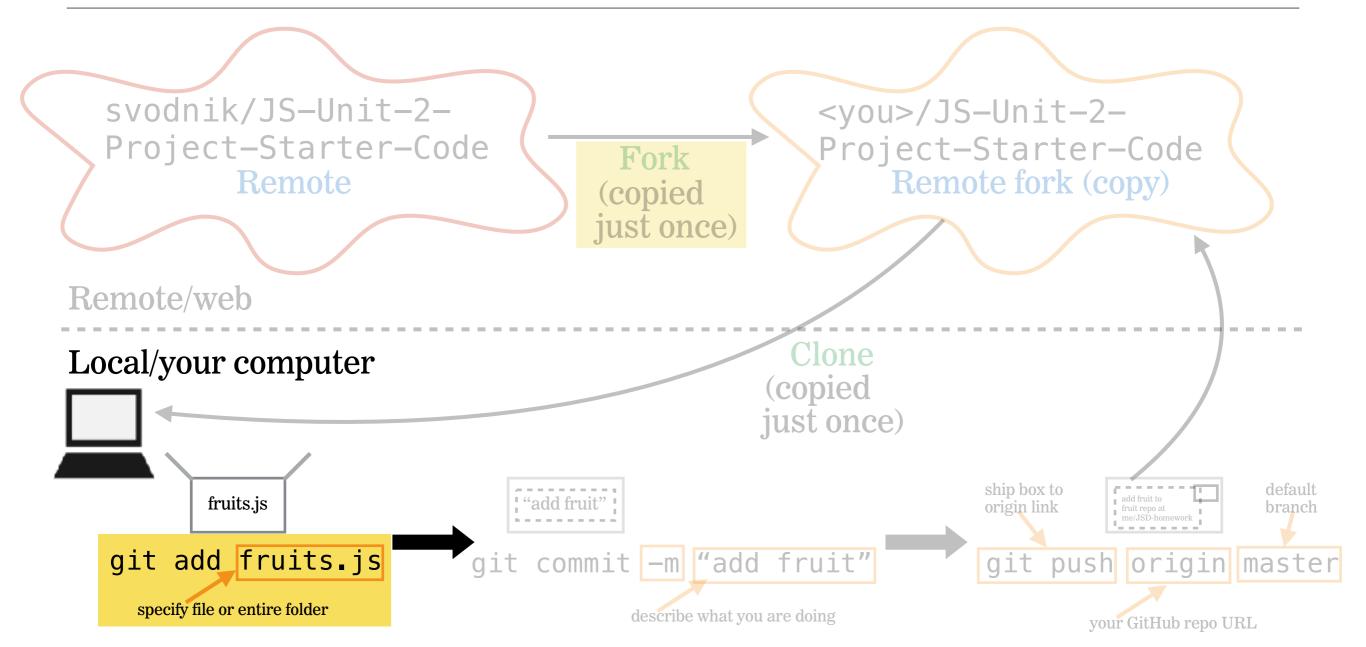
- 1. Think about how you approach a task with a lot of parts and steps. Jot a list of ideas.
- 2. Discuss your ideas with a partner or group, writing down any suggestions from your group members that seem helpful.
- 3. When everyone has had a chance to discuss, you'll have a chance to share your ideas with the rest of the class.

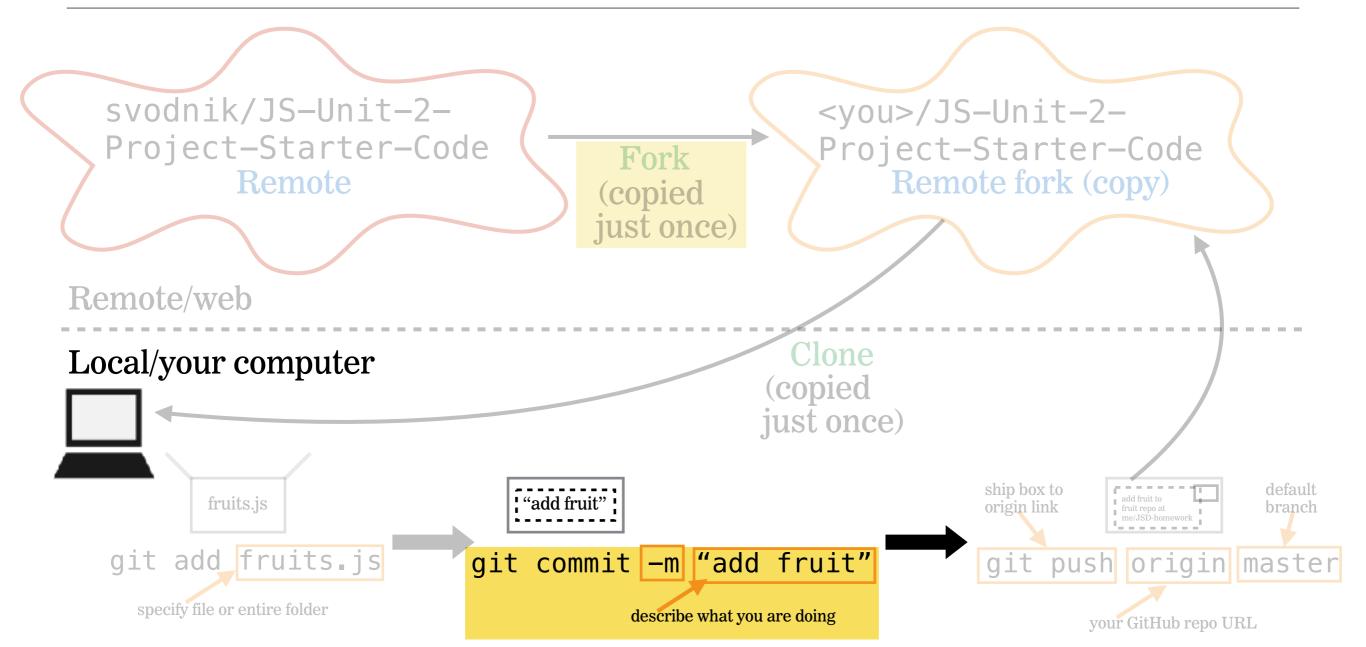
## **Project 2: Feedr**

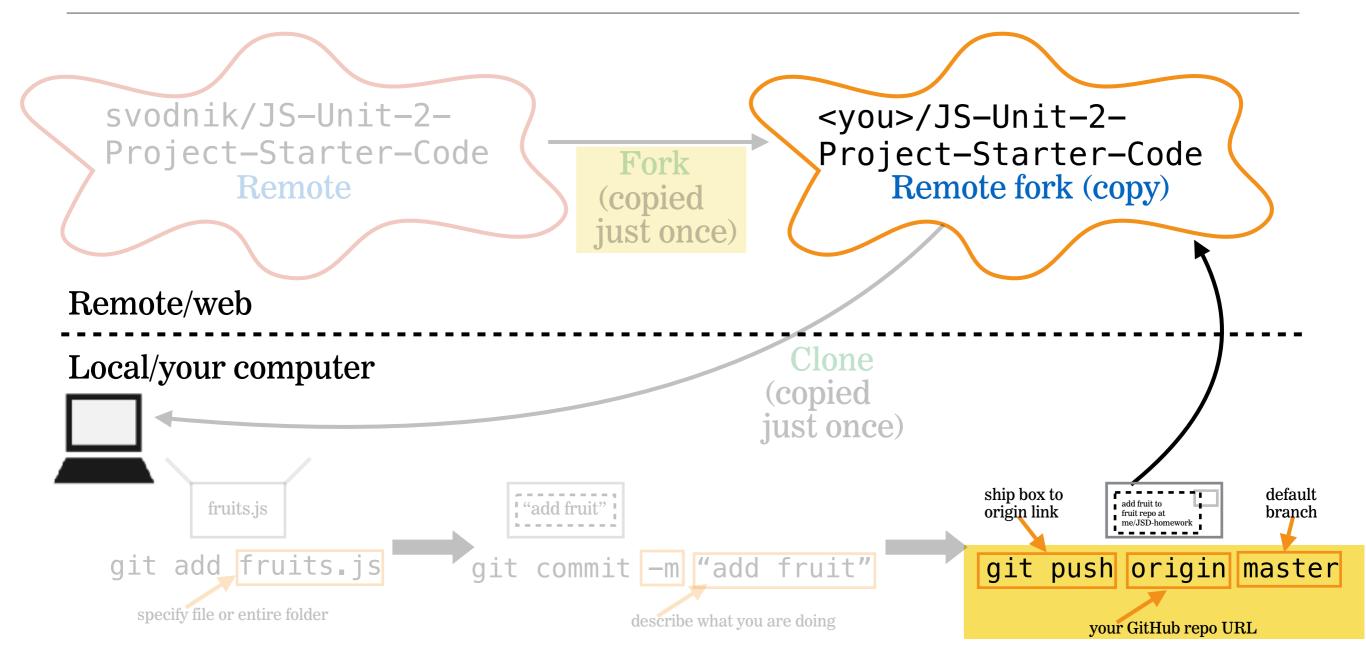
- GitHub repo to fork: https://github.com/svodnik/JS-Unit-2-Project-Starter-Code
- Project overview & instructions: https://github.com/svodnik/JS-Unit-2-Project-Starter-Code/blob/master/ README.md











### **LEARNING OBJECTIVES - REVIEW**

- 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.

## NEXT CLASS PREVIEW Prototypal Inheritance

- Explain the difference between literal and constructed objects.
- Write a constructor for a JavaScript object.
- Explain prototypal inheritance and its purpose.
- Recognize the difference between prototypal and classical inheritance.
- Create and extend prototypes.

## Exit Tickets!

## QSA