

JAVASCRIPT DEVELOPMENT

Sasha Vodnik, Instructor

HELLO!

1. Pull changes from the `svodnik/JS-SF-13-resources` repo to your computer
2. Open the `16-deploying` folder in your code editor

JAVASCRIPT DEVELOPMENT

DEPLOYING YOUR APP

LEARNING OBJECTIVES

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

- › Understand what hosting is.
- › Identify a program's needs in terms of host providers.
- › Ensure backward compatibility by using Babel to transpile code.
- › Optimize code before deployment
- › Deploy to a web host.

AGENDA

- Transpile with Babel
- Lint with ESLint
- Minify with Uglify-JS
- Add a polyfill
- Deploy with Firebase

DEPLOYING YOUR APP

WEEKLY OVERVIEW

WEEK 10

Deploying your app / Final project lab

WEEK 11

React / Final Project Presentations

EXIT TICKET QUESTIONS

1. why can't we set up proxy servers in class so that our apps can be used by the world? There are so few of us, can't we find time for it? Maybe after presenting final projects in class?

Final Project Checkin

Due Thursday, January 31

DEPLOYING YOUR APP

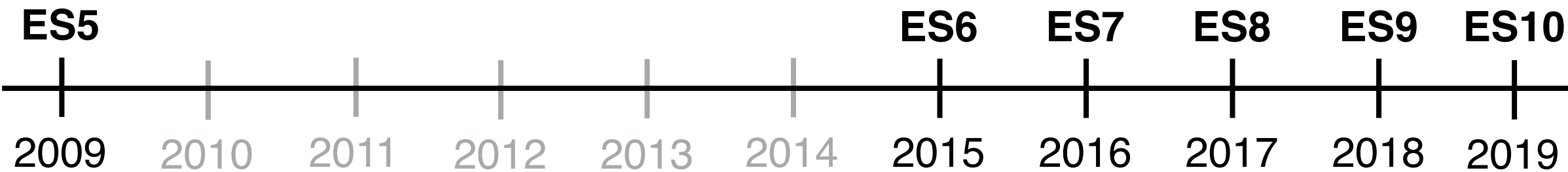
FINALIZING YOUR CODE

DEPLOYING YOUR APP

TRANSPILING

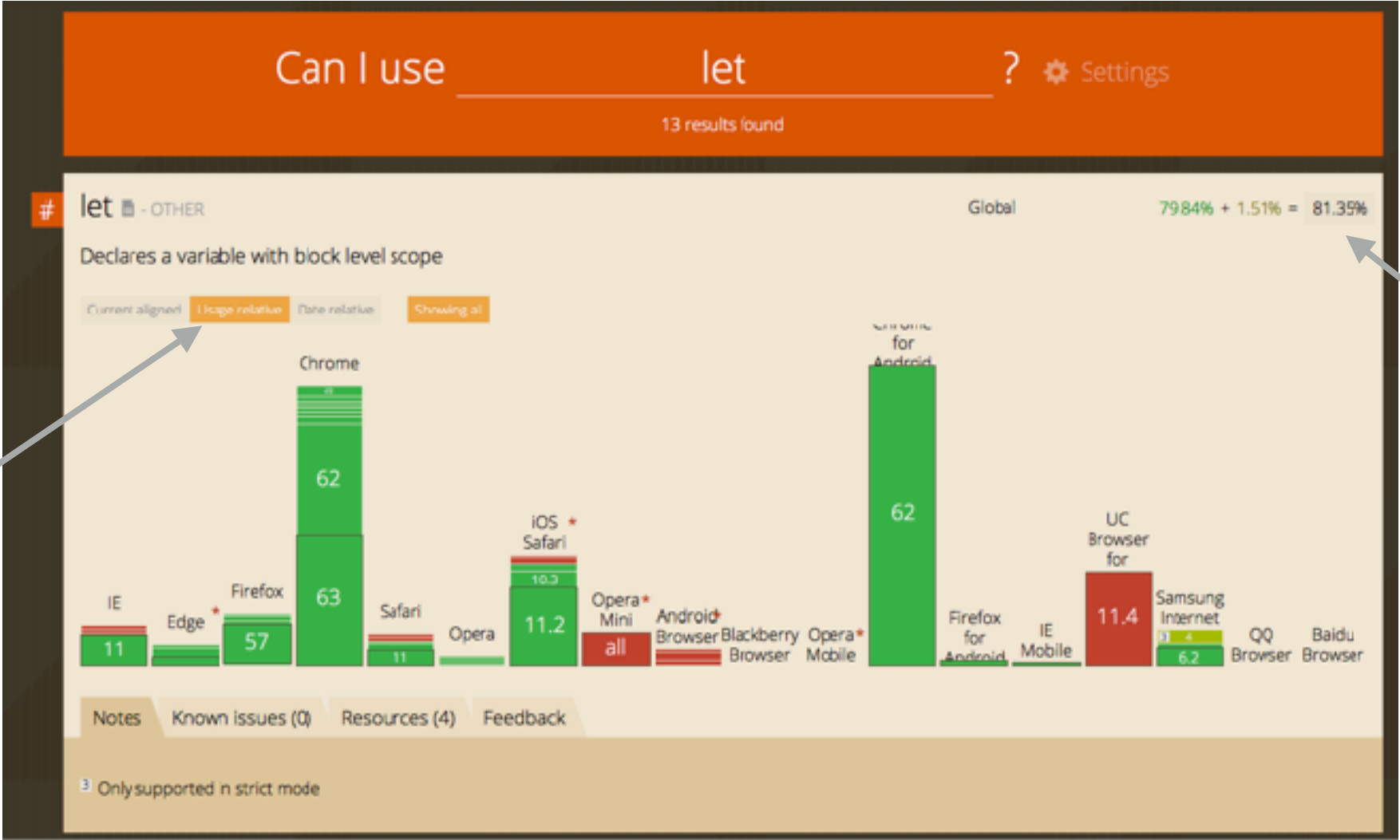
virtually all browsers
in use support ES5

only modern browsers
support ES6+



caniuse.com

“Usage relative”
option shows
proportional graph



Estimated percent of
global browser traffic
that can parse this
feature

Transpiling involves rewriting code that uses ES6+ features to produce the same result using ES5 code

ES6

```
const taxRate = 0.0875;  
let items = [];  
  
let addToCart = () => {  
  // do something  
}
```

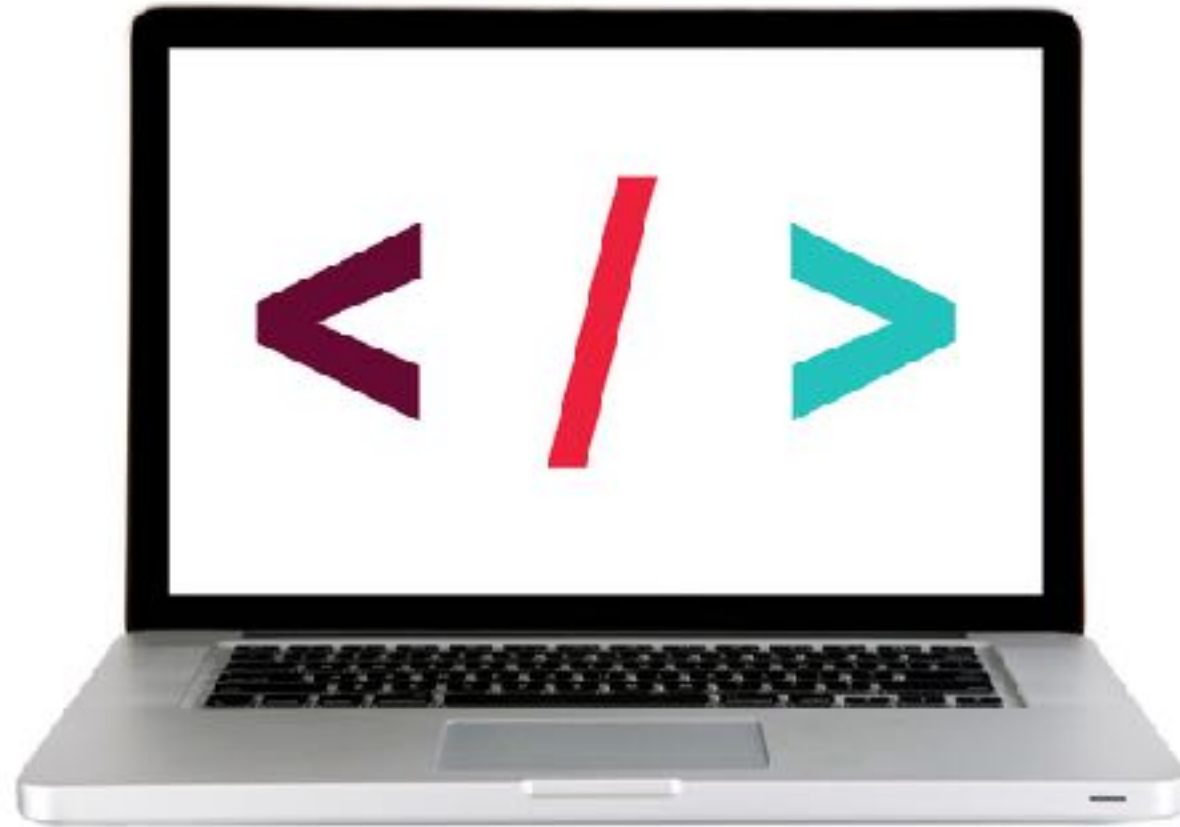
transpiling



ES5

```
var taxRate = 0.0875;  
var items = [];  
  
function addToCart() {  
  // do something  
}
```

LET'S TAKE A CLOSER LOOK



EXERCISE — TRANSPILE CODE USING BABEL



EXERCISE

KEY OBJECTIVE

- Ensure backward compatibility by using Babel to transpile code.

TIMING

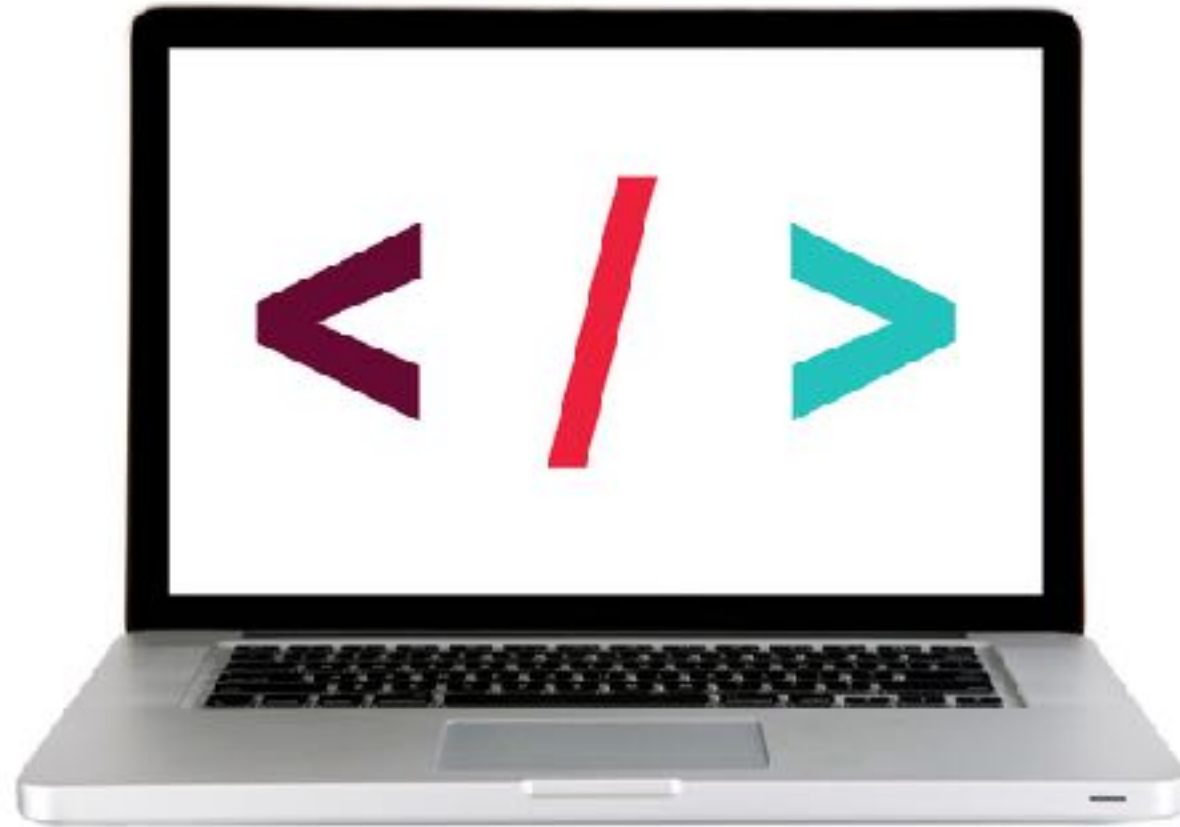
5 min

1. Configure Babel for the Firebase app you created in the previous class. (If your code isn't quite working, use the code in the `starter-code > 3-transpiling-exercise` folder as a starting point.)
2. Run Babel to create an ES5-compatible version of your code.
3. Open the converted file in your editor and verify the code was transpiled.
4. Open `index.html` and change the source for the script element to the JavaScript file created by Babel.
5. Test your app in the browser and make sure it still works as it did previously.

DEPLOYING YOUR APP

LINTING

LET'S TAKE A CLOSER LOOK



EXERCISE — LINT CODE USING ESLINT



EXERCISE

KEY OBJECTIVE

- Optimize code for deployment.

TIMING

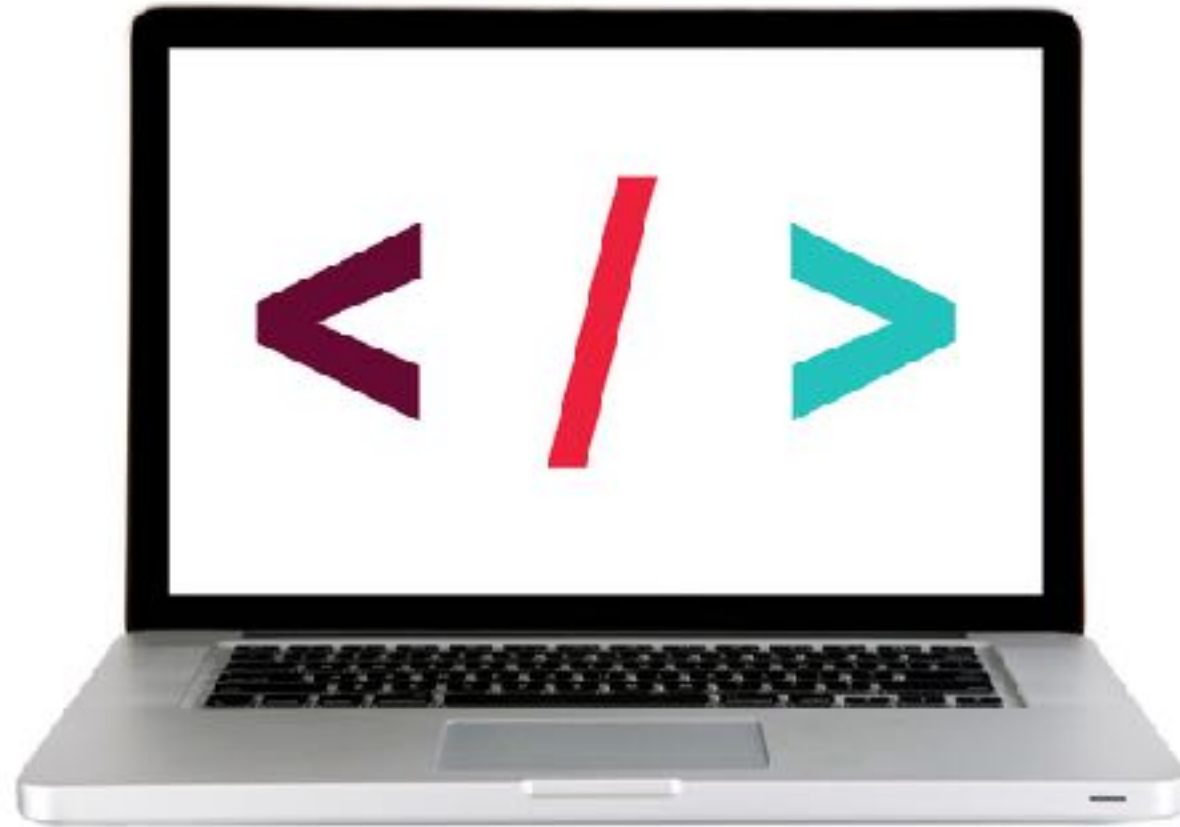
3 min

1. In your browser, open <https://eslint.org/demo>.
2. Copy the contents of `app.js` from your Firebase project, paste in the left pane of the ESLint interface, and verify that no errors are shown.
3. If errors are flagged, fix them in the web interface, then when the code is error-free, copy the code from the web interface (click in the code and press `command+A`), then replace the code in `app.js` with the copied code. Save your changes.
4. Test your app in the browser and make sure it still works as it did previously.

DEPLOYING YOUR APP

MINIFYING

LET'S TAKE A CLOSER LOOK



EXERCISE — MINIFY CODE



EXERCISE

KEY OBJECTIVE

- Optimize code for deployment.

TIMING

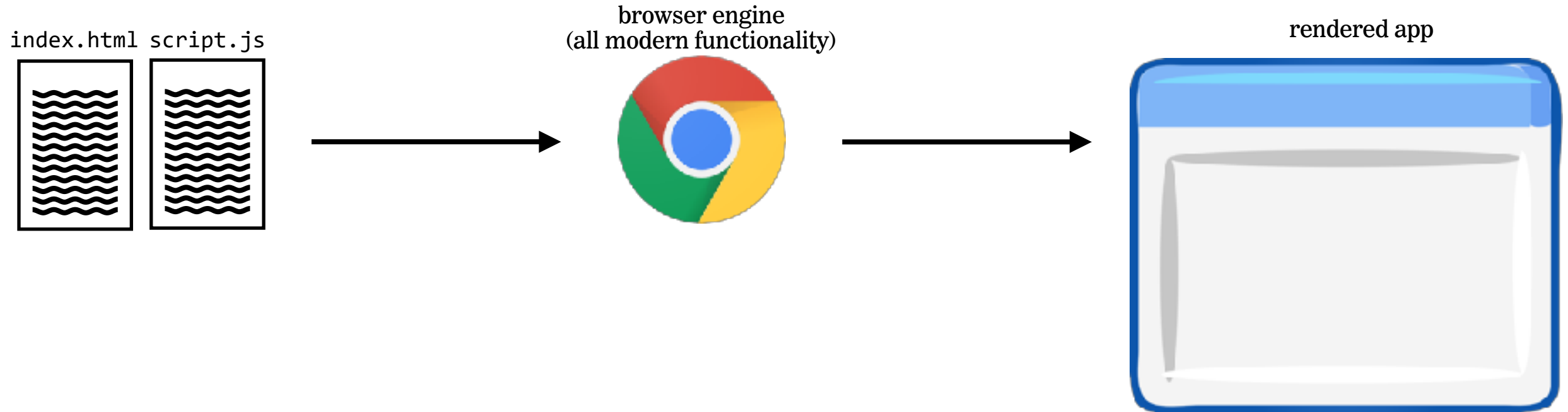
3 min

1. At the command line, navigate to the folder containing your Firebase project.
2. Use uglify to create a minified version of `app.js`, outputting to `app.min.js`.
3. Open `index.html` and change the source for the script element to `app.min.js`.
4. Test your app in the browser and make sure it still works as it did previously.

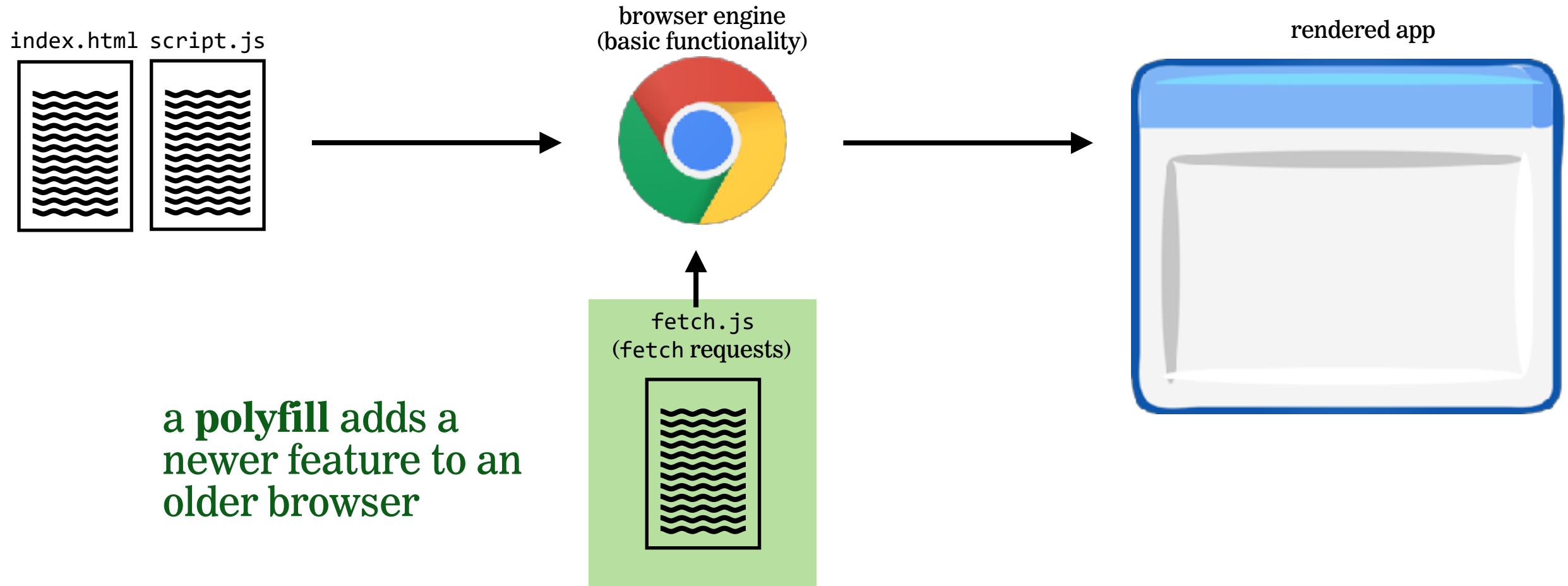
DEPLOYING YOUR APP

POLYFILLS

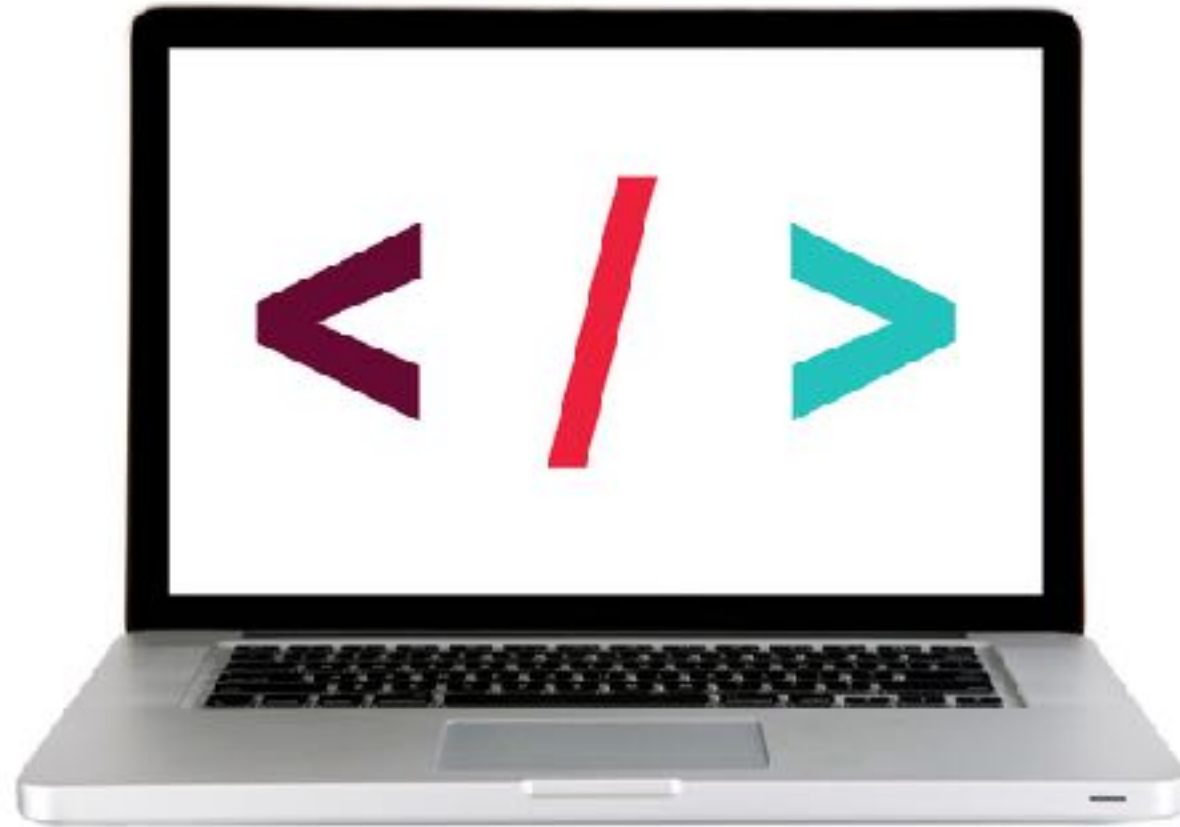
APP FUNCTIONALITY IN A MODERN BROWSER



APP FUNCTIONALITY IN AN OLDER BROWSER

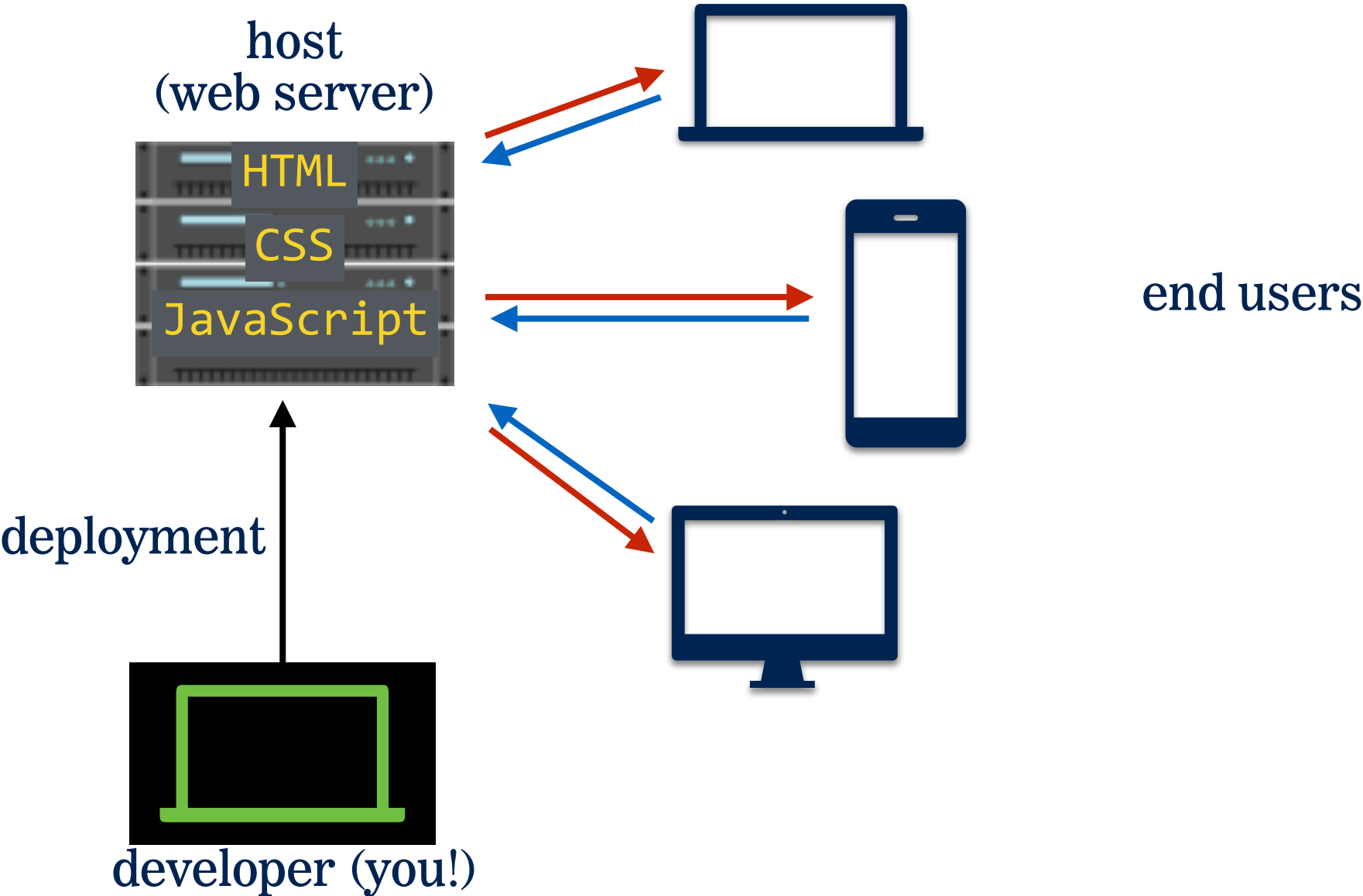


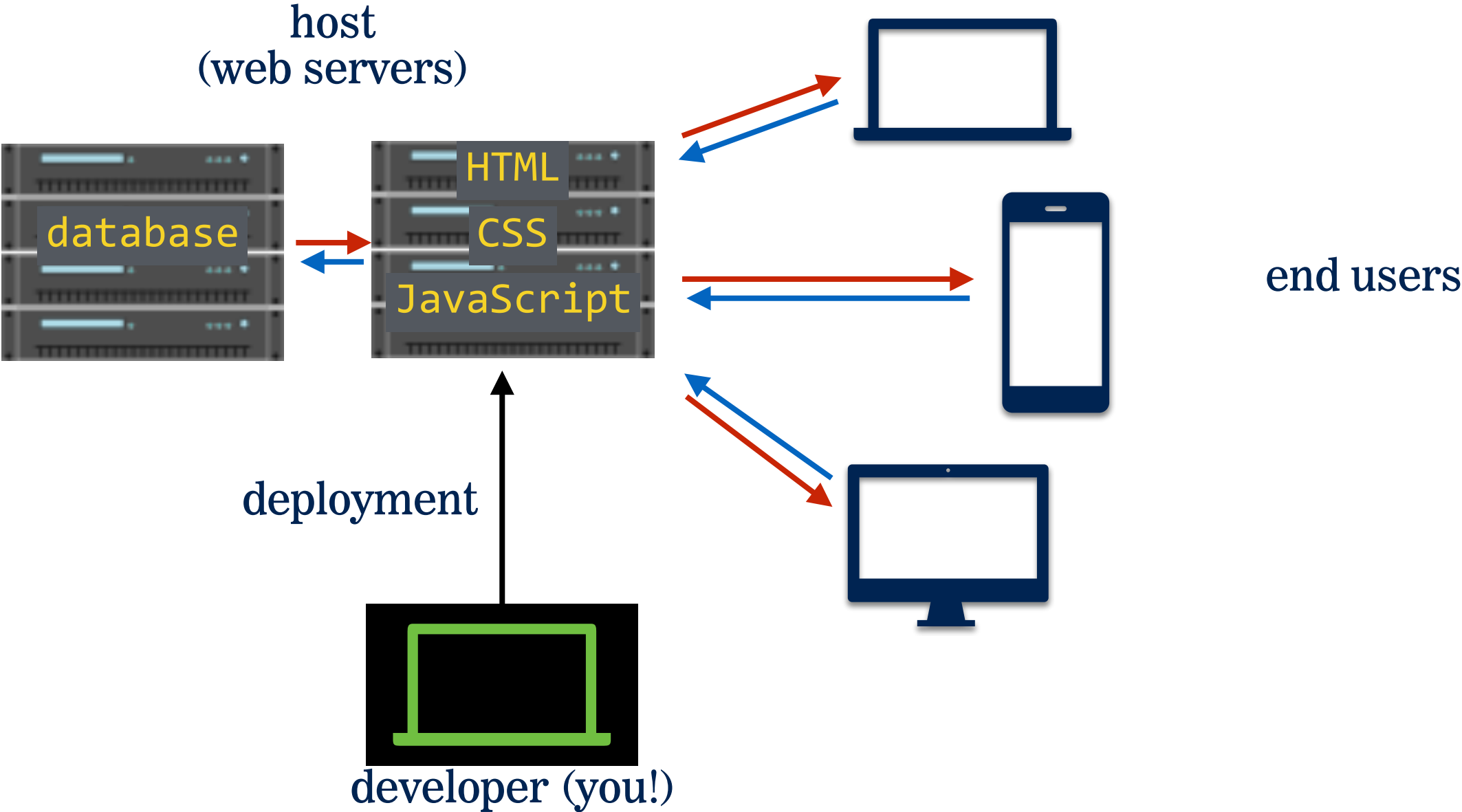
LET'S TAKE A CLOSER LOOK











DEPLOYING YOUR APP

DEPLOYMENT

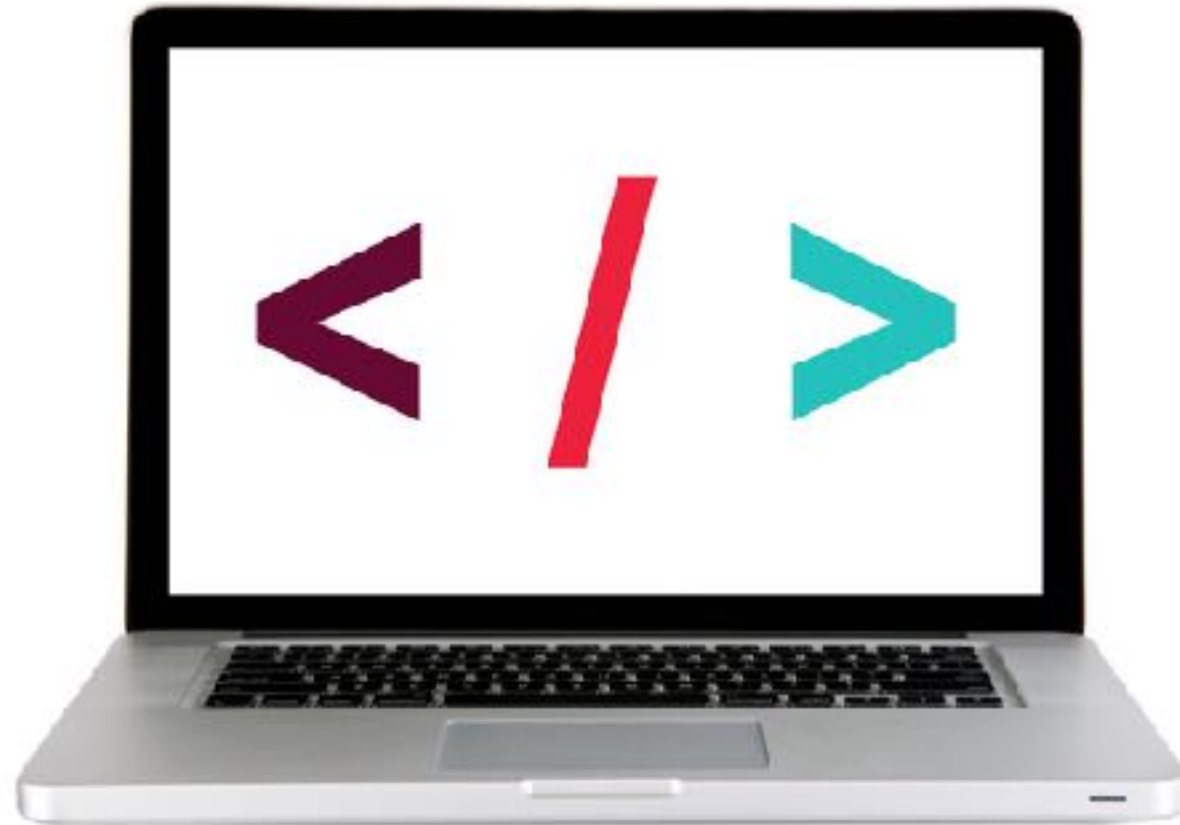




ALTERNATIVE “SERVERLESS” SERVICES

 Google Firebase Relevant Capabilities <ul style="list-style-type: none">AuthRealtime DatabaseMedia StorageCloud Functions Quick Overview <p>Google Firebase is very powerful while being very easy to use. For example, you can run cloud functions, but you don't even need to for most data storage and retrieval or auth. It might be expensive to scale on though.</p>	 Google Cloud Platform Relevant Capabilities <ul style="list-style-type: none">Everything Quick Overview <p>More of a major infrastructure provider in vein of Amazon Web Services than a toolkit for building out an app like Firebase is.</p>	 Amazon Web Services Relevant Capabilities <ul style="list-style-type: none">Everything Quick Overview <p>Lambda, API Gateway, S3, and Cognito (auth) are probably the most relevant things to front-end developers. AWSync is a hit like Firebase. There are frameworks that help you deploy to Lambda, like Craffie and Functional Fleet.</p>	 Microsoft Azure Relevant Capabilities <ul style="list-style-type: none">Everything Quick Overview <p>A major infrastructure provider with solutions for about just everything, and generally considered the cheapest. For working with cloud functions, there is an online editor, but it also allows Git-hub sync and integrates directly with VS Code. Data storage is through Cosmos DB.</p>	 StdLib Relevant Capabilities <ul style="list-style-type: none">Cloud Functions Quick Overview <p>StdLib is based on Function as a Service ("server-less") architecture, popularized by AWS Lambda. You can use StdLib to build modular, scalable APIs for yourself and other developers in minutes without having to manage servers, gateways, domains, write documentation, or build SDKs. They also offer Codeboxz, and online code editor for working with the APIs.</p>	 Webtask Relevant Capabilities <ul style="list-style-type: none">Cloud FunctionsBasic JSON data store Quick Overview <p>An in-browser editor for creating and testing cloud functions. Seems like the nicest experience for this particular job. It's kinda of an elaborate demonstration of Auth0 Extend, which is essentially a way to take Webtask and put it in your own app.</p>	 IBM Cloud Functions Relevant Capabilities <ul style="list-style-type: none">Cloud Functions Quick Overview <p>Based on Apache OpenWhisk.</p>	 Backendless Relevant Capabilities <ul style="list-style-type: none">Realtime DatabaseAuth Quick Overview <p>All-in-one kind of service similar to Firebase, including the realtime database. Has a PHP version you can host yourself if you're, ya know, into running servers.</p>
--	---	--	--	--	--	---	--

LET'S TAKE A CLOSER LOOK



EXERCISE — PUSH CHANGES TO FIREBASE



EXERCISE

KEY OBJECTIVE

- Deploy to a web host.

TIMING

5 min

1. Make a change to the HTML, CSS, and/or JavaScript for the project you deployed to Firebase.
2. Push your changes to Firebase and verify that your updated code is what you see in your browser at `appname.firebaseio.com`

Exit Tickets!

(Class #16)

LEARNING OBJECTIVES – REVIEW

- Understand what hosting is.
- Identify a program's needs in terms of host providers.
- Ensure backward compatibility by using Babel to transpile code.
- Optimize code before deployment
- Deploy to a web host.

NEXT CLASS PREVIEW

Final Project Lab

Q&A