FEWD WEEK 8 • CLASS 13:

Javascript Objects

https://slides.com/jennifermeade/fewd-8-13/live

QUICK REVIEW

ARRAY SYNTAX

```
const fruits = ["🍏", "🍊", "🍋", "🍇", "🍓", "🍑"];
                 0    1    2    3    4    5

console.log( fruits[4] ); // returns 🍓
console.log( fruits[0] ); // returns 🍏
```

- Array elements are **indexed** meaning they are assigned a number starting with 0.
- Individual elements are accessed with the name of the variable followed by the number of the element inside square brackets.

FOR-OF LOOP SYNTAX

```
const names = ['Markus','David','Kelly','Ann','Dina'];

for (let name of names) {
  console.log(name);
}
```

1. Use the **for** keyword followed by parentheses
2. Create a variable with **let** to store the current value of the array element.
3. Use the **of** keyword followed by an array to loop over.
4. The code inside the curly braces get run on each loop.

TEMPLATE LITERALS

```
let projectname = $(this).text();
const template = `
  <div>
    <h2>${projectName}</h2>
    <p class="summary">Here's the project summary</p>
  </div>
`
$('.pop-up').append(template);
```

- Backticks can be used to surround strings instead of single or double quotes.
- Can span multiple lines.
- Insert variables or expressions with: `${  }`

CUSTOMER RECEIPT WARM UP

//codepen.io/jme11/embed/NEzvOY/?height=265&theme-id=default&default-tab=html,result&embed-version=2&editable=true

https://codepen.io/jme11/pen/NEzvOY

LAB SOLUTION


```
let total = 0;

for (let purchase of purchases) {
 let item = purchase[0];
 let price = purchase[1];
 total += price;
 let template = `
   <div>
     ${item}
     <span class="price">$${price.toFixed(2)}</span>
   </div>`
 $('#receipt').append(template);
}

$('#receipt').append(`
  <div class="total">
    Total
    <span class="price">$${total.toFixed(2)}</span>
  </div>`);
```

OBJECTIVES

- Understand the structure and use of objects in Javascript
- Understand how to create and use object methods

OBJECTS IN JAVASCRIPT

WHAT ARE OBJECTS

Ummm... you know, they're objects. 🙂

```
const table = {
  legs: 4,
  color: "white",
  style: "contemporary",
  materials: ["metal", "formica", "particleboard"]
};
```

OBJECT SYNTAX

```
const person = {
  firstName: "Jane", /* property: value */
  lastName: "Smith",
  age: 30,
  eyes: "blue",
  hair: "brown" /* no comma after the last property */
};
```

- Objects are surrounded by curly braces.
- Objects store data in **key/value pairs**.
- Individual pieces of data (known as **object properties**) are separated by commas (no comma after the last key/value pair).

GETTING AT STUFF IN AN OBJECT

```
const pet = {
  name: "Cosmo",
  age: 13,
  cute: true,
  color: ["black", "white"]
};
pet.age; /* dot notation output 13 */
pet["age"]; /* square bracket notation output 13 */
pet.color[0]; /* What is this one? */
```

- Whereas arrays have elements you access with an index, objects have **properties you access by name**.
- Object can use either dot notation or square bracket notation.

TRY SOME OTHERS

```
const projects = [
  {name: "City Skylines", date: "02/05/13", url: "../img/skylines.jpg"},
  {name: "Maker's Mission", date: "8/12/15", url: "../img/mm.png"},
  {name: "Book Cover", date: "12/22/15", url: "../img/cover1.jpg"},
  {name: "Web and Mobile App UX", date: "07/23/18", url: "../img/ux-proj.png"}
];

projects[0].name /* outputs "City Skylines" */
projects[3]['url'] /* outputs "../img/ux-proj.png" */
```

- "Maker's Mission" == projects[1].name
- Book Cover project date == projects[2].date
- projects[1].url == "../img/mm.png"
- "../img/skylines.jpg" == projects[0].url

ADDING & MODIFYING PROPERTIES

```
const pet = {
  name: "Cosmo",
  age: 12,
  cute: true,
  color: ["black", "white"]
};

pet.birthday = "September 12, 2005"; /* Add new property */
pet.breed = "Border Collie"; /* Add new property */

pet.age = 13; /* Update the property value */
```

OBJECT METHODS

OBJECT METHODS

Functions added to objects are called **methods**. The keyword **this** allows access other properties inside the same object.

```
let attendee = {
  firstName: "Jane",
  lastName: "Smith",
  company: "Apple",
  fullName: function() { /* object method */
    return this.firstName + " " + this.lastName;
  }
}

attendee.fullName() /* outputs "Jane Smith" */
```

PASSING DATA TO METHODS

Data can also be **passed into our method from outside** our object too.

```
let attendee = {
  firstName: "Jane",
  lastName: "Smith",
  greeting: function(salutation) {
    return salutation + " " + this.firstName;
  }
}

attendee.greeting("Hello!") /* outputs: Hello! Jane */
```

DOES THIS LOOK FAMILIAR?

JQUERY METHODS

You already know how to use object methods.
You've been doing it for weeks now.

`$( ... )` is a method that creates an object — specifically, a jQuery object. It takes the string value you pass to it as a selector and finds all of the matches in the document and puts them in an object. That object has properties and methods that we have been accessing with dot notation, like `.html()`, `.text()`, etc.!

PANELISTS LAB

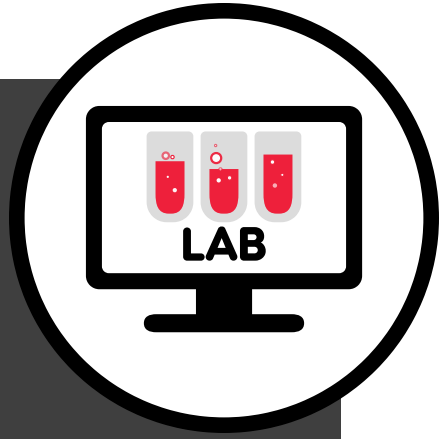
//codepen.io/jme11/embed/deXwNq/?height=265&theme-id=default&default-tab=html,result&embed-version=2&editable=true

https://codepen.io/jme11/pen/deXwNq

PANELISTS SOLUTION

```
for (let panelist of panelists) {
  const template = `
    <section>
      <div>
        <img src="${panelist.image}">
      </div>
      <p class="name">${panelist.name}</p>
      <p class="company">${panelist.company}</p>
    </section>`

  $('body').append(template);
}
```

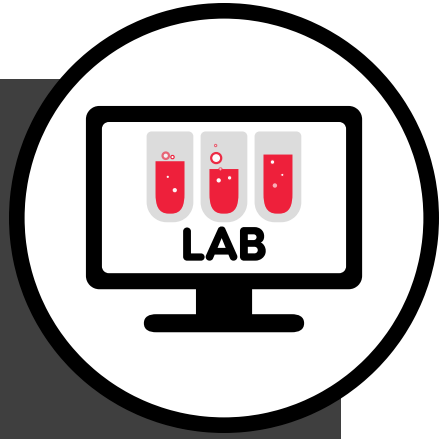


PANELISTS BONUS SOLUTION

```
let html = '';

for (let panelist of panelists) {
  html += `
    <section>
      <div>
        <img src="${panelist.image}">
      </div>
      <p class="name">${panelist.name}</p>
      <p class="company">${panelist.company}</p>
    </section>`
}

$('body').append(html);
```



CARD GALLERY

//codepen.io/jme11/embed/ZoQqrg/?height=265&theme-id=default&default-tab=html,result&embed-version=2&editable=true

https://codepen.io/jme11/pen/ZoQqrg

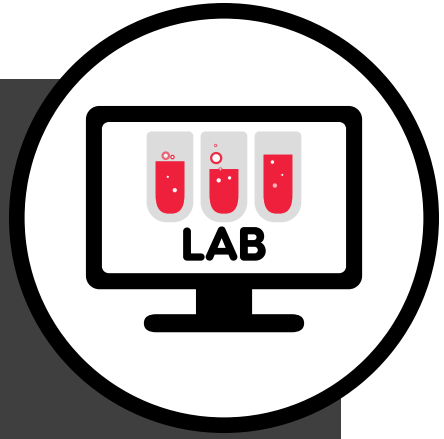
CARD GALLERY SOLUTION

```
let html = "";

for(let story of stories) {

  const template = `
    <div class="card">
      <img src="${story.image}" alt="${story.title}">
      <div class="content">
        <h3>featured</h3>
        <p>${story.title}</p>
      </div>
    </div>`;

  html += template;
}
```



JAVASCRIPT BASICS

JAVASCRIPT BASICS YOU KNOW

You've now had an introduction to all the basic topics essential to programming in Javascript.

Variables

Operators

If Statements

For Loops

Strings, Numbers, Booleans

Arrays

Objects

Functions

this

WHAT'S LEFT

Tons... We'll try and tackle some of these in the next classes:

AJAX, Async & APIs

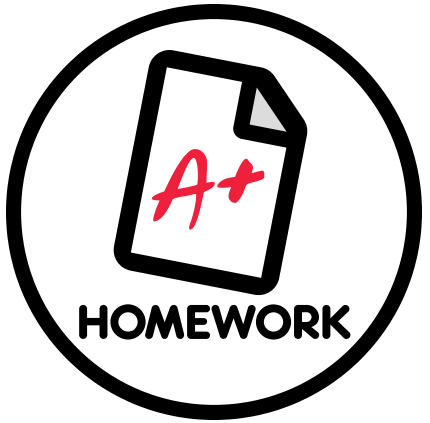
Local Data Persistance

SASS & Webpack

Working with Forms

jQuery Plugins & JS Libraries

HOMEWORK



- Submit your HTML/CSS via Slack
- There are two weeks to finish the Javascript for your final projects
- Make an appointment with me to review your project and course progress
  - Tues/Thurs: Day or Evening by Skype
  - Fri: 9:00 AM - 6:00 PM on campus
  - Sat: 10:00 AM - 2:00 PM on campus
  - Sat/Sun: Day or Evening by Skype

EXIT SURVEY

https://goo.gl/EB4XFw

GO BUILD AWESOME THINGS!