Review of forEach and async

Week 1 Day 4 - June 9, 2022 - Doug Case

Week 1 Day 4 Goals

- Review forEach and for loops
- Review sync and await
- Leave lots of time for week 1 Flixster Project!

CODEPATH*ORG

Loops in JS - 4 ways to accomplish

```javascript
let names = ["Stephane", "Amber", "Valerie", "Nnaemeka"];

for (let i = 0; i < names.length; i++) {
  console.log(names[i]);
}

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

// the forEach() method executes a provided function once for each array element
names.forEach(function(name) {
  console.log(name);
});

names.forEach(function(name, idx) {  // optional second parm (index)
  console.log(name, idx);
});
```

Loops in JS - results

Stephane
Amber
Valerie
Nnaemeka

Stephane
Amber
Valerie
Nnaemeka

Stephane
Amber
Valerie
Nnaemeka

Stephane 0
Amber 1
Valerie 2
Nnaemeka 3

Lab3 displayResults for loop, forEach(): thx Miles/Dylan!

```
let resultsArea = document.querySelector("#my-results-area");

let myApiData = {
// <whole object is in here … >
};
console.log(myApiData);
displayResults(myApiData);   // two different ways to accomplish the same thing
displayResults2(myApiData);  // same effect as displayResult

function displayResults(apiData) {           // old-fashioned for loop
  for (let i = 0; i < apiData.data.length; i++) {
    resultsArea.innerHtml += `
      <img src=${apiData.data[i].images.original.url}>
    `
  }
}


function displayResults2(apiData) {           // forEach loop
  apiData.data.forEach(function(element){
    resultsArea.innerHTML += '<img src="' + element.images.original.url + '">';
  });
}
```

Review example of async function with await

```javascript
let pokemonForm = document.querySelector("form");
let pokemonArea = document.querySelector("#pokemon-area");

// (1) control the form behavior upon submit
pokemonForm.addEventListener("submit", getPokemon);

async function getPokemon(evt) {
    // this prevents the page from re-loading
    evt.preventDefault();
    let apiUrl = "https://pokeapi.co/api/v2/pokemon/" + evt.target.pokemon.value.toLowerCase();
    // (2) upon form submit, call the Pokemon API
    let response = await fetch(apiUrl);
    let responseData = await response.json();
    // (3) put some results of the pokemon API call onto page
    generateHTML(responseData);
}

function generateHTML(pokeData) {
    pokemonArea.innerHTML = `
        <h1>${pokeData.species.name}</h1>
        <p>Height: ${pokeData.height}</p>
        <p>Weight: ${pokeData.weight}</p>
        <img src="${pokeData.sprites.front_default}" alt="Pokemon image" />
    `;
}
```

JSON

- Called **JSON**: JavaScript Object Notation
- JSON is syntactically correct JS object - perfect to convert to JS
- There are packages that easily convert JSON to Java, C, Python, etc
- https://jsonlint.com   will format JSON and tell you if it is valid format

Happy Coding - short break first!