Readme.md

DOM

🌿 **Technical Exploration Route** ✨ Discovering how a webpage responds to what we do, challenge by challenge. This activity will be guided and divided into stations with clear instructions to develop at your own pace, including suggested breaks.

🎯 OBJECTIVE

Understand the DOM, how to change things using JavaScript, and how to make forms safer for users. We're not aiming for perfection—just to explore, test, and share what we discover.

🔔 IMPORTANT

- You don't have to complete all the challenges.
- You can ask for help anytime or work individually, in pairs, or in small groups—whatever makes you feel comfortable.
- Take breaks whenever you need.

🧭 **CHALLENGE 1: Explore the DOM (Document Object Model)** Choose the option you like best or feel most comfortable with. You don't need to do all three.

Option 1 – Visual (if you're focused and ready to concentrate)

- LINK TO EXPLANATORY VIDEO

Option 2 – Playful (if you prefer something visual or interactive)

- https://flukeout.github.io/

Option 3 – Reflective (if you prefer creativity) Complete this sentence:

> "If a webpage were a theater, the DOM would be…" You can create a visual diagram using tools we've worked with or write a dialogue with your peers.

🧭 **CHALLENGE 2: Let's play with changing things Goal:** Make a button greet you.

Project structure:

```
greeting/
├── index.html
└── script.js
```

1. Create an `index.html` file with this code:

html

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<title>Greeting with JavaScript</title>
</head>
<body>
<h1>Hello!</h1>
<button id="greetButton">Click here</button>
<p id="responseText"></p>
<script src="script.js"></script>
</body>
</html>
```

2. Create a `script.js` file with this code:

javascript

```
document.getElementById("greetButton").addEventListener("click", () => {
  document.getElementById("responseText").textContent = "Nice to meet you! 😊";
});
```

3. Right-click `index.html` and choose “Open with Live Server.” Your browser will open and show your working page.

🆘 **Don’t have Live Server?**

1. Go to the left sidebar in Visual Studio Code (cube icon → Extensions)
2. Search for “Live Server”
3. Click Install
4. Go back and right-click `index.html` → “Open with Live Server”

💡 Tip: You can enable Auto Save in VSCode so changes save automatically. Steps:

1. Open VSCode
2. Top menu: File → Auto Save (or “Guardar automáticamente”)
3. A ✔️ checkmark will appear if it’s enabled.

🧭 **CHALLENGE 3: Explore a form with basic JavaScript validations** 🎯 **Goal:** See how a form works with built-in JavaScript validation. Editable, modifiable, and safe to experiment with.

⚠️ Breaking things while coding isn’t bad—it’s a chance to learn, fix, and grow as a developer.

Project structure:

validated-form/
├── index.html
└── script.js

1. Create two files: `index.html` and `script.js`
2. Add this to `index.html`:

html
```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<title>Validated Form</title>
</head>
<body>
<h1>Contact Form</h1>
<form id="form">
  <label for="name">Name:</label>
  <input type="text" id="name" placeholder="Enter your name" required>
  <br><br>
  <label for="email">Email:</label>
  <input type="email" id="email" placeholder="example@email.com" required>
  <br><br>
  <button type="submit">Send</button>
</form>
<div id="errors" style="color: red;"></div>
<div id="successMessage" style="color: green;"></div>
<script src="script.js"></script>
</body>
</html>
```

3. Add this to `script.js`:

javascript
```
document.getElementById("form").addEventListener("submit", function(e) {
  e.preventDefault();
  const name = document.getElementById("name").value.trim();
  const email = document.getElementById("email").value.trim();
  const errors = document.getElementById("errors");
  const successMessage = document.getElementById("successMessage");
  errors.textContent = "";
```

```
  successMessage.textContent = "";
  let hasErrors = false;

  if (name === "") {
    errors.innerHTML += "🔴 Name is required.<br>";
    hasErrors = true;
  }
  if (email === "") {
    errors.innerHTML += "🔴 Email is required.<br>";
    hasErrors = true;
  } else if (!email.includes("@") || !email.includes(".")) {
    errors.innerHTML += "🔴 Email must be in a valid format (like example@email.com).<br>";
    hasErrors = true;
  }

  if (!hasErrors) {
    successMessage.textContent = "✅ Form submitted successfully. Thank you!";
  }
});
```

🔎 **Explore the form:**

1. Open it in your browser
2. Try deleting or entering incorrect data
3. Observe the error messages
4. Customize the error messages to be fun, serious, or personal e.g., “Hey! Your name’s missing!” or “😬 Required field.”
5. Delete a line of JS validation—what breaks?
6. Remove `required` and `type="email"` from HTML—what still works?
7. Try creating your own success message with emojis, colors, or a GIF!

🧭 **EXTREME CHALLENGE 4: Make your form think for itself** 🎯 **Goal:** Create a form that validates all fields using only JavaScript—no `required`, no `type="email"`.

Here, each validation is a mini logic function—and you’re the programmer!

Project structure:

```
js-form-validation/
├── index.html
└── script.js
```

1. Add this to `index.html`:

html

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<title>Smart Form</title>
</head>
<body>
<h1>Form with JavaScript Validations</h1>
<form id="form">
  <label for="name">Name:</label>
  <input type="text" id="name" placeholder="Enter your name">
  <br><br>
  <label for="email">Email:</label>
  <input type="text" id="email" placeholder="example@email.com">
  <br><br>
  <label for="age">Age:</label>
  <input type="text" id="age" placeholder="Your age">
  <br><br>
  <button type="submit">Send</button>
</form>
<div id="errors" style="color: red;"></div>
<div id="successMessage" style="color: green;"></div>
<script src="script.js"></script>
</body>
</html>
```

2. Add this to `script.js`:

javascript

```
document.getElementById("form").addEventListener("submit", function(e) {
  e.preventDefault();
  const name = document.getElementById("name").value.trim();
  const email = document.getElementById("email").value.trim();
  const age = document.getElementById("age").value.trim();
  const errors = document.getElementById("errors");
  const successMessage = document.getElementById("successMessage");
  errors.innerHTML = "";
  successMessage.textContent = "";
  let hasErrors = false;

  // TODO 1: Validate name has at least 3 letters
  if (name.length < 3) {
```

```
    errors.innerHTML += "🔴 Name must have at least 3 letters.<br>";
    hasErrors = true;
  }

  // TODO 2: Validate email contains @, . and ends in .com or .es
  if (!email.includes("@") || !email.includes(".") ||
      (!email.endsWith(".com") && !email.endsWith(".es"))) {
    errors.innerHTML += "🔴 Email must be valid (.com or .es).<br>";
    hasErrors = true;
  }

  // TODO 3: Validate age is a number ≥ 18
  const ageNumber = parseInt(age);
  if (isNaN(ageNumber) || ageNumber < 18) {
    errors.innerHTML += "🔴 Age must be 18 or older.<br>";
    hasErrors = true;
  }

  // TODO 4: Final success message
  if (!hasErrors) {
    successMessage.textContent = "✅ All good! Form submitted successfully.";
  }
});
```