DOM Challenge 4

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;
  }
```
Project setup
Create a folder called js-form-validation. Inside it, make two files:

index.html — this is your webpage

script.js — this is your JavaScript brain

📄 index.html — The form layout
This is the basic form with three fields: name, email, and age.

html
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Smart Form</title>
</head>
<body>
```

```
  <h1>Form with JavaScript Validations</h1>

  <!-- The form users will fill out -->
  <form id="form">
    <!-- Name input -->
    <label for="name">Name:</label>
    <input type="text" id="name" placeholder="Enter your name">
    <br><br>

    <!-- Email input -->
    <label for="email">Email:</label>
    <input type="text" id="email" placeholder="example@email.com">
    <br><br>

    <!-- Age input -->
    <label for="age">Age:</label>
    <input type="text" id="age" placeholder="Your age">
    <br><br>

    <!-- Submit button -->
    <button type="submit">Send</button>
  </form>

  <!-- Where error messages will show -->
  <div id="errors" style="color: red;"></div>

  <!-- Where success message will show -->
  <div id="successMessage" style="color: green;"></div>

  <!-- Connects the JavaScript file -->
  <script src="script.js"></script>
</body>
</html>
```

📜 script.js — Making the form smart

This JavaScript code checks each field and shows messages if something's wrong.

javascript

```
// When the form is submitted, run this function
document.getElementById("form").addEventListener("submit", function(e) {
  // Stop the form from refreshing the page
  e.preventDefault();

  // Get the values the user typed in
  const name = document.getElementById("name").value.trim();
```

```
  const email = document.getElementById("email").value.trim();
  const age = document.getElementById("age").value.trim();

  // Get the message areas
  const errors = document.getElementById("errors");
  const successMessage = document.getElementById("successMessage");

  // Clear old messages
  errors.innerHTML = "";
  successMessage.textContent = "";

  // Keep track of problems
  let hasErrors = false;

  // ✅ Check 1: Name must be at least 3 letters
  if (name.length < 3) {
    errors.innerHTML += "🔴 Name must have at least 3 letters.<br>";
    hasErrors = true;
  }

  // ✅ Check 2: Email must include @ and . and end 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;
  }

  // ✅ Check 3: Age must be a number and at least 18
  const ageNumber = parseInt(age); // Turn age into a number
  if (isNaN(ageNumber) || ageNumber < 18) {
    errors.innerHTML += "🔴 Age must be 18 or older.<br>";
    hasErrors = true;
  }

  // ✅ If no errors, show success message
  if (!hasErrors) {
    successMessage.textContent = "✅ All good! Form submitted successfully.";
  }
});
```

🧪 What to try (like a curious beginner)

Leave fields empty → See the red error messages.

Type a short name like “Jo” → You’ll get a name warning.

Type a weird email like hello@banana → It’ll say the email isn’t valid.

Type age as abc or 17 → It’ll say age must be 18 or older.

Type everything correctly → You’ll see the green success message!

🎨 Bonus: Make it fun!
Change the success message to something personal:

javascript
successMessage.innerHTML = "🎉 Great job, Ana! You're officially a form master!";
Or add a GIF:

javascript
successMessage.innerHTML = "✅ All good!
<img
src='https://media.giphy.com/media/3o7aD2saalBwwftBIY/giphy.gif' width='100'>";
```
  // 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.";
  }
});
```
