

## ES 6 Part 1 - Assessment

### Introduction

As a passionate **home chef** or **culinary enthusiast**, managing your recipes can be both exciting and overwhelming. Whether you're crafting signature dishes or experimenting with new cuisines, having an organized and accessible **recipe management system** makes cooking more enjoyable and efficient.

Let's build a **digital cookbook web page** – a platform where you can **store, view, edit, and manage** your favorite recipes all in one place. With the knowledge gained so far in the program, let's develop a clean interface with interactive features. This cookbook isn't just for you – you can develop this web page further where **the world can explore, appreciate, and try your creations**. Share your passion, inspire others, and let your recipes reach kitchens far and wide.

### Objectives of the Assessment:

- ✓ Create functions in JS to complete the tasks.
- ✓ Use the concepts of ES-6 studied this week where needed.
- ✓ Write clean and error-free backend JavaScript logic.
- ✓ Recognize the role and importance of JavaScript in web development.

### Problem Statement

The HTML and CSS files are given to you in the Source code. You will also see a JS file with a pre-defined Array "**recipes**", to hold all recipe objects.

Your tasks are as follows:

1. Add an event listener for the form to collect user input and call necessary functions to add the recipe, display the recipes, and reset the form.
2. Create a function **addRecipe**. This function takes one **object**(form data) as a parameter.
  - a. **Validate** the object for any null values.
  - b. If **none** of the object's properties **are empty**, add the object to the **recipes array**.
  - c. If **any property is empty**, **display an alert** instructing the user to fill in all input fields.
3. Create a function **displayRecipes**.
  - a. For each **recipe**, generate a card containing the recipe's **Name, Ingredients, and Instructions**, along with **Edit** and **Delete** buttons.
  - b. The **Ingredients** and **Instructions** are **comma-separated** at every step.
  - c. Using the **recipes array** and **template strings**, code to **dynamically display** all recipes in the "Recipes" section of the web page as shown below.

## Recipe Book

### Add a Recipe

Title:

Ingredients:

Instructions:

**Add Recipe**

### Recipes

**Spaghetti Bolognese**

**Ingredients:**

spaghetti
minced meat
tomato sauce
onion

**Instructions:**

1. Cook spaghetti
2. brown minced meat
3. add sauce
4. mix together

**Edit**
**Delete**

4. Create a function **editRecipe** with an **index** parameter.
- When the **edit button** is clicked, use the **index** to find the recipe and **display its details in the input form**.
  - Change the "Add Recipe" button to "**Save Recipe**".
  - Once "Save Recipe" is clicked, **update** the recipe's details in the **array** and **refresh** the displayed list on the web page. Switch the button back to "**Add Recipe**".

### Add a Recipe

Title:

Ingredients:

Instructions:

**Save Recipe**

### Recipes

**Ingredients:**

spaghetti
minced meat

tomato sauce
onion
garlic

**Instructions:**

1. Cook spaghetti
2. brown minced meat
3. add sauce
4. mix together

**Edit**
**Delete**

5. Create a function **deleteRecipe** with an **index** parameter.
  - a. Use this **index** to locate and **remove** the recipe from the **recipes array**.
  - b. Also, **delete** it from **the web page**.
6. Use the provided **CSS** or add additional styles to **styles.css** to ensure the webpage is clean, styled, user-friendly, and responsive.

!!!! Happy Coding !!!!