Problem Statement: Recipe Management System

You are tasked with creating a simple Recipe Management System for a food blog. The system will manage a list of recipes, each recipe containing a title, a list of ingredients, and instructions. The goal is to implement a variety of JavaScript concepts to manipulate and interact with the recipe data.

Requirements:

1. **Recipe Data Structure:**
    - Each recipe should be an object with the following properties:
        - `title` (string): The name of the recipe.
        - `ingredients` (array of strings): List of ingredients.
        - `instructions` (string): A description of how to prepare the dish.
2. **Functions to Manage Recipes:**
    - **Add Recipe:** Create a function `addRecipe` that takes a recipe object and adds it to an array of recipes.
    - **Remove Recipe:** Create a function `removeRecipe` that removes a recipe from the array based on the recipe title.
    - **Find Recipe by Ingredient:** Create a function `findRecipeByIngredient` that takes an ingredient and returns an array of recipes that include that ingredient.
3. **Array Manipulations:**
    - Use `forEach` to display the title of each recipe in the console.
    - Use `map` to create an array of recipe titles.
    - Use `filter` to find recipes that contain a specific ingredient.
    - Use `reduce` to count the total number of recipes.
4. **Object Manipulations:**
    - Use dot notation and bracket notation to access and modify recipe properties.
    - Use object destructuring to extract `title` and `ingredients` from recipes.
5. **Additional Features:**
    - **Recipe Destructuring:** Implement a function `describeRecipe` that takes a recipe object and logs a description in the format: "Title: [title], Ingredients: [ingredients], Instructions: [instructions]".
    - **Recipe Update:** Create a function `updateRecipe` that updates the ingredients of a recipe based on the title.
    - **Recipe Statistics:** Implement a function `getStatistics` that provides the average number of ingredients per recipe.
6. **Optional Advanced Features:**
    - Implement `sort` to display recipes sorted by title.
    - Implement `find` to get the first recipe that contains a specified ingredient.

- Use `fill` to initialize a list of recipes with placeholder values.
- Use `splice` to remove a specific recipe based on its index.

🍽 Example Usage:

```
// 🌟 Add Recipes
addRecipe({
    title: "🍲 Veggie Stir-Fry",
    ingredients: ["broccoli", "carrots", "bell peppers", "soy sauce", "garlic"],
    instructions: "🍴 Stir-fry vegetables with soy sauce and garlic until
tender."
});

addRecipe({
    title: "🥑 Avocado Toast",
    ingredients: ["avocado", "whole grain bread", "lemon juice", "salt",
"pepper"],
    instructions: "🔪 Mash avocado with lemon juice, salt, and pepper. Spread on
toasted bread."
});

// 📜 Display Recipe Titles
console.log('📜 Recipe Titles:');
recipes.forEach(recipe => console.log(`- ${recipe.title}`));

// 🔍 Find Recipes with "avocado"
const avocadoRecipes = findRecipeByIngredient("avocado");
console.log('🔍 Recipes with avocado:');
avocadoRecipes.forEach(recipe => console.log(`- ${recipe.title}`));

// 🔄 Update Avocado Toast Recipe
updateRecipe("🥑 Avocado Toast", ["avocado", "whole grain bread", "lemon juice",
"salt", "pepper", "chili flakes"]);
console.log('🔄 Avocado Toast updated with new ingredients!');

// 🔍 Describe a Specific Recipe
console.log('🔍 Recipe Description:');
describeRecipe(recipes[0]);

// 📊 Get and Display Recipe Statistics
const stats = getStatistics();
console.log('📊 Recipe Statistics:');
console.log(`- Total Recipes: ${stats.totalRecipes}`);
console.log(`- Average Ingredients per Recipe: ${stats.averageIngredients}`);

// 🔠 Sort Recipes by Title
const sortedRecipes = [...recipes].sort((a, b) => a.title.localeCompare(b.title));
console.log('🔠 Sorted Recipes by Title:');
sortedRecipes.forEach(recipe => console.log(`- ${recipe.title}`));

// 🎯 Find the First Recipe with "carrots"
```

```
const carrotRecipe = recipes.find(recipe =>
recipe.ingredients.includes("carrots"));
console.log('🥕 First Recipe with Carrots:');
console.log(carrotRecipe ? carrotRecipe.title : 'No recipe found.');

// 🛠️ Fill Placeholder Recipes
const placeholderRecipes = new Array(3).fill({
    title: "🐵 Placeholder Recipe",
    ingredients: ["ingredient1", "ingredient2"],
    instructions: "📜 Placeholder instructions."
});
console.log('🛠️ Placeholder Recipes:');
console.log(placeholderRecipes);

// 🍽️ Remove a Recipe by Index
recipes.splice(1, 1); // Removes the second recipe (index 1)
console.log('🍽️ Recipes after Splice:');
recipes.forEach(recipe => console.log(`- ${recipe.title}`));
```

Breakdown of Example Usage:

1. **Adding Recipes:** 🍲 (Veggie Stir-Fry), 🥑 (Avocado Toast)
2. **Displaying Titles:** 📜
3. **Finding Ingredients:** 🔍
4. **Updating Recipes:** 🔄
5. **Describing Recipes:** 🔍
6. **Statistics:** 📊
7. **Sorting Recipes:** 🔠
8. **Finding Ingredients in Recipes:** 🥕
9. **Placeholder Recipes:** 🛠️
10. **Removing Recipes:** 🍽️
