**Training Day 24 Report**

24 July 2025

**JSON and Arrays in JavaScript**

* **1. Introduction**
* JSON (JavaScript Object Notation) is a **lightweight data format** used to store and exchange data between server and client.  
  Arrays are **JavaScript objects** used to store **ordered lists of values**.
* Often, when working with APIs or local data, we need to **convert JSON to arrays** or **arrays to JSON** for processing.
* **2. JSON to Array**
* **a) What is JSON?**
* JSON is a **string representation** of objects or arrays.  
  It looks like JavaScript objects but is **always a string**.
* **Example JSON array (string):**
* let jsonData = '["Apple", "Banana", "Mango"]';
* **b) Converting JSON to Array**
* To convert a JSON string into a JavaScript array, we use **JSON.parse()**.
* **Syntax:**
* let array = JSON.parse(jsonString);
* **Example:**
* let jsonData = '["Apple", "Banana", "Mango"]';
* let fruits = JSON.parse(jsonData);
* console.log(fruits); // ["Apple", "Banana", "Mango"]
* console.log(fruits[0]); // Apple
* **Explanation:**
* JSON.parse() takes a JSON string and converts it to a **JavaScript object or array**.
* After parsing, we can access array elements using **indices**.
* **c) JSON Object to Array of Objects**
* Sometimes, JSON contains **objects**:
* let jsonData = '[{"name":"Shubhdeep","age":20},{"name":"Riya","age":19}]';
* let students = JSON.parse(jsonData);
* console.log(students[0].name); // Shubhdeep
* console.log(students[1].age); // 19
* Each object in JSON becomes an **object in the array** after parsing.
* **3. Array to JSON**
* **a) Converting Array to JSON**
* To convert a JavaScript array into a JSON string, we use **JSON.stringify()**.
* **Syntax:**
* let jsonString = JSON.stringify(array);
* **Example:**
* let fruits = ["Apple", "Banana", "Mango"];
* let jsonData = JSON.stringify(fruits);
* console.log(jsonData); // '["Apple","Banana","Mango"]'
* **Explanation:**
* JSON.stringify() converts a **JavaScript array or object** into a **JSON string**.
* JSON strings are often used to **send data to servers** or store in **local storage**.
* **b) Array of Objects to JSON**
* You can also convert an array of objects into JSON:
* let students = [
* { name: "Shubhdeep", age: 20 },
* { name: "Riya", age: 19 }
* ];
* let jsonData = JSON.stringify(students);
* console.log(jsonData);
* // '[{"name":"Shubhdeep","age":20},{"name":"Riya","age":19}]'
* Each object is converted into JSON format.
* This string can be transmitted to a **server** or saved in **local storage**.
* **4. JSON ↔ Array Example Together**
* **Example:**
* // JSON string
* let jsonData = '[{"name":"Aman","score":80},{"name":"Riya","score":90}]';
* // JSON to Array
* let students = JSON.parse(jsonData);
* console.log(students[0].name); // Aman
* // Modify array
* students.push({ name: "Karan", score: 85 });
* // Array to JSON
* let newJsonData = JSON.stringify(students);
* console.log(newJsonData);
* // '[{"name":"Aman","score":80},{"name":"Riya","score":90},{"name":"Karan","score":85}]'
* **Explanation:**
* **JSON.parse()** converts the JSON string to a JavaScript array of objects.
* You can **modify the array** (add/remove elements).
* **JSON.stringify()** converts the modified array back to a JSON string.
* **5. Using JSON with Local Storage**
* Often, arrays are stored in **browser local storage** using JSON.
* **Example:**
* let fruits = ["Apple", "Banana", "Mango"];
* // Store array as JSON string
* localStorage.setItem("fruits", JSON.stringify(fruits));
* // Retrieve array from JSON string
* let storedFruits = JSON.parse(localStorage.getItem("fruits"));
* console.log(storedFruits); // ["Apple", "Banana", "Mango"]
* **6. Key Points**
* **JSON.parse()** → Converts JSON string → JavaScript array/object
* **JSON.stringify()** → Converts JavaScript array/object → JSON string
* Useful for **APIs, server communication, and local storage**.
* JSON is **always a string**, arrays and objects are **actual JavaScript types**.
* **7. Conclusion**
* Converting between **JSON and arrays** is a common task in web development.
* JSON.parse() and JSON.stringify() make it simple and efficient.
* Understanding these conversions is essential for **data handling, storage, and server communication**.