Notes on JSON

Hia Al Saleh

September 9th, 2024

Contents

1   Final Exercise

- **Objective**: Create a valid JSON document using the given information.

- **Guidance**:

  - Structure the document with appropriate key-value pairs.
  - Use meaningful names for the keys.
  - Verify the validity of the JSON after completion.

2   What is JSON

- **Definition**: JSON stands for *JavaScript Object Notation.*

- **Purpose**:

  - Text format used for exchanging data between platforms.
  - It is a subset of JavaScript but works independently of it.

- **Characteristics**:

  - Portable, works across platforms and systems.
  - Based on the JavaScript object literal syntax.
  - Simple and easy to read.
  - Files use the `.json` extension.
  - Shorter than XML, as it does not require closing tags, leading to smaller file sizes and faster transfer.

3   How is JSON Formatted

- **Syntax**:

  - JSON is marked like a JavaScript object with `{}` for objects and `[]` for arrays.
  - Key-value pairs are separated by colons (`:`), and multiple pairs are separated by commas (`,`).

- **Types of JSON Values**:

  - JSON values can include: Objects, Strings, Numbers, Booleans, Null, and Arrays.

- **Rules**:

  - Keys (names) must always be in double quotes.
  - Values only require quotes depending on the datatype (e.g., strings).
  - Escape characters like backslashes and quotes must be used appropriately.

- **Data Types**:

  - **Numbers**: Can be integers, decimals, negative numbers, or exponents, and do not require quotes.
  - **Boolean**: Values must be lowercase (`true` or `false`).
  - **Null**: Indicates no value (not the same as undefined in JavaScript).

- **Arrays**:

  - Always enclosed in square brackets (`[]`), containing valid JSON data types.

- **Nested Objects**:

  - JSON allows objects to be nested within other objects, using a tree-like structure.

- **Portability Considerations**:

  - Avoid special characters or spaces in key names for compatibility across platforms.