## What is JSON? Explain its purpose.

JSON (JavaScript Object Notation) is a lightweight, text-based, human-readable data interchange format that is used to store and transmit data objects consisting of attribute-value pairs and arrays. It originated from JavaScript object syntax but is language-independent and supported by many programming languages for data exchange between systems, especially between web clients and servers.

Its primary purpose is to provide a simple and efficient way to represent structured data that can be easily read and written by humans and machines. JSON is commonly used in web APIs to send and receive data, replacing heavier formats like XML because it is more compact and faster to parse. It allows data to be transmitted as plain text, making it ideal for communication across different platforms and languages.

## Which data types are supported in JSON? List all.

- a string

- a number - int, float...

- an object (JSON object) {}

- an array []

- a boolean

- \_null\_

## What is the correct JSON format? Give an example.

The correct JSON format follows these key rules:

* Data is represented as name/value pairs, where the name (key) is always a string enclosed in double quotes.
* A colon (:) separates each key and its value.
* Multiple key/value pairs within an object are separated by commas.
* Curly braces {} enclose objects.
* Square brackets [] enclose arrays.
* Values can be strings (in double quotes), numbers (integer or floating point), booleans (true or false), null, objects, or arrays.
* Strings must be in double quotes, not single quotes.
* No trailing commas or comments are allowed.

A simple example of a valid JSON object:

{

"name": "John",

"age": 30,

"isMarried": false,

"children": ["Anna", "Jake"],

"address": {

"street": "123 Main St",

"city": "New York"

},

"pets": null

}

This example shows:

## What is the difference between a JSON object and a JSON array? Give examples.

The difference between a JSON object and a JSON array lies primarily in their structure and usage:

JSON Object:

It is an unordered collection of key-value pairs.

Keys are always strings enclosed in double quotes.

Values can be any valid JSON data type: string, number, boolean, null, object, or array.

It is enclosed in curly braces {}.

It is used to represent a single entity or complex data with named properties.

**Example of a JSON object:**

json

{

"name": "Alice",

"age": 25,

"isStudent": false,

"address": {

"city": "New York",

"zipcode": "10001"

}

}

JSON Array:

It is an ordered list of values.

Values can be any valid JSON data type, including objects.

It is enclosed in square brackets [].

It is commonly used to represent collections or lists of similar items.

Order of elements in an array matters.

**Example of a JSON array:**

json

[

"apple",

"banana",

"cherry"

]

**Or an array of objects:**

json

[

{"name": "Alice", "age": 25},

{"name": "Bob", "age": 30},

{"name": "Charlie", "age": 35}

]

## How do we represent null in JSON? Give an example.

In JSON, the null value is represented simply as the literal null without quotes. This indicates the absence of a value or a null value for a key.

Example of representing null in JSON:

{

"name": "John",

"middleName": null,

"age": 30

}

## Does JSON allow comments? Explain why.

JSON does not allow comments as part of its official specification. The reason for this is to keep JSON a pure data-interchange format, ensuring simplicity, consistent parsing, and interoperability across different systems. Including comments could introduce non-standard content that may break parsers or cause compatibility issues.

Douglas Crockford, the creator of JSON, deliberately removed support for comments to prevent misuse of JSON as a configuration language laden with parsing directives. Instead, JSON is designed to only contain data structures without extraneous elements like comments

## What is the default character encoding for JSON?

The default character encoding for JSON is **UTF-8**. According to the JSON specification (RFC 4627 and later RFC 8259), JSON text must be encoded in Unicode with UTF-8 being the default and most commonly used encoding for interoperability. JSON can also be encoded in UTF-16 or UTF-32, but UTF-8 is standard and preferred because it is 8-bit compatible and widely supported across systems and platforms

## Convert this data into JSON format:

Name: Pramod

Age: 25

Skills: Java, API

{

“Name” : “Pramod”,

“Age” : 25,

“Skills” : [ “Java”,”API”]

}