JSON:

What is JSON

* JSON stands for JavaScript Object Notation.
* JSON is lightweight data-interchange format.
* JSON is easy to read and write than XML.
* JSON is language independent.
* JSON supports array, object, string, number and values.

Example:

1. {"employees":[
2. {"name":"Sonoo", "email":"sonoojaiswal1987@gmail.com"},
3. {"name":"Rahul", "email":"rahul32@gmail.com"},
4. {"name":"John", "email":"john32bob@gmail.com"}
5. ]}

# **JSON Example**

JSON example can be created by object and array. Each object can have different data such as text, number, boolean etc. Let's see different JSON examples using object and array.

### **JSON Object Example**

A JSON object contains data in the form of key/value pair. The keys are strings and the values are the JSON types. Keys and values are separated by colon. Each entry (key/value pair) is separated by comma.

The **{** (curly brace) represents the JSON object.

1. {
2. "employee": {
3. "name":       "sonoo",
4. "salary":      56000,
5. "married":    **true**
6. }
7. }

Let's see the example of JSON array having objects.

1. [
2. {"name":"Ram", "email":"Ram@gmail.com"},
3. {"name":"Bob", "email":"bob32@gmail.com"}
4. ]

JSON Example 1

1. {"employees":[
2. {"name":"Shyam", "email":"shyamjaiswal@gmail.com"},
3. {"name":"Bob", "email":"bob32@gmail.com"},
4. {"name":"Jai", "email":"jai87@gmail.com"}
5. ]}

The XML representation of above JSON example is given below.

1. **<employees>**
2. **<employee>**
3. **<name>**Shyam**</name>**
4. **<email>**shyamjaiswal@gmail.com**</email>**
5. **</employee>**
6. **<employee>**
7. **<name>**Bob**</name>**
8. **<email>**bob32@gmail.com**</email>**
9. **</employee>**
10. **<employee>**
11. **<name>**Jai**</name>**
12. **<email>**jai87@gmail.com**</email>**
13. **</employee>**
14. **</employees>**

# **JSON Object**

JSON object holds key/value pair. Each key is represented as a string in JSON and value can be of any type. The keys and values are separated by colon. Each key/value pair is separated by comma.

The curly brace **{** represents JSON object.

Let's see an example of JSON object.

1. {
2. "employee": {
3. "name":       "sonoo",
4. "salary":      56000,
5. "married":    **true**
6. }
7. }

In the above example, employee is an object in which "name", "salary" and "married" are the key. In this example, there are string, number and boolean value for the keys.

Competitive questions on Structures in Hindi

Keep Watching

## JSON Object with Strings

The string value must be enclosed within double quote.

1. {
2. "name":       "sonoo",
3. "email":      "sonoojaiswal1987@gmail.com"
4. }

## JSON Object with Numbers

JSON supports numbers in double precision floating-point format. The number can be digits (0-9), fractions (.33, .532 etc) and exponents (e, e+, e-,E, E+, E-).

1. {
2. "integer": 34,
3. "fraction": .2145,
4. "exponent": 6.61789e+0
5. }

## JSON Object with Booleans

JSON also supports boolean values true or false.

1. {
2. "first": **true**,
3. "second": **false**
4. }

## JSON Nested Object Example

A JSON object can have another object also. Let's see a simple example of JSON object having another object.

1. {
2. "firstName": "Sonoo",
3. "lastName": "Jaiswal",
4. "age": 27,
5. "address" : {
6. "streetAddress": "Plot-6, Mohan Nagar",
7. "city": "Ghaziabad",
8. "state": "UP",
9. "postalCode": "201007"
10. }
11. }

JSON Array of Objects

Let's see a simple JSON array example having 4 objects.

1. {"employees":[
2. {"name":"Ram", "email":"ram@gmail.com", "age":23},
3. {"name":"Shyam", "email":"shyam23@gmail.com", "age":28},
4. {"name":"John", "email":"john@gmail.com", "age":33},
5. {"name":"Bob", "email":"bob32@gmail.com", "age":41}
6. ]}