**For the above JSON, iterate over all for loops(for, for in, for of, forEach).**

**Use a for Loop.**

The for statement creates a loop that consists of three optional expressions, enclosed in parentheses and separated by semicolons, followed by a statement (usually a [block statement](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/block)) to be executed in the loop.

let str = '';

for (let i = 0; i < 9; i++) {

str = str + i;

}

console.log(str);

// Expected output: "012345678"

**Use a for...in Loop**

A [for…in](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/for%E2%80%A6in) loop iterates over all enumerable properties of an object:

const res = JSON.parse(xhr.responseText);  
  
for (const key in res){  
 if(obj.hasOwnProperty(key)){  
 console.log(`${key} : ${res[key]}`)  
 }  
}

**Use a for...of Loop**

The for...of statement executes a loop that operates on a sequence of values sourced from an [iterable object](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Iteration_protocols#the_iterable_protocol). Iterable objects include instances of built-ins such as [Array](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array), [String](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/String), [TypedArray](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/TypedArray), [Map](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Map), [Set](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Set), [NodeList](https://developer.mozilla.org/en-US/docs/Web/API/NodeList) (and other DOM collections), as well as the [arguments](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Functions/arguments) object, [generators](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Generator) produced by [generator functions](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/function*), and user-defined iterables.

const array1 = ['a', 'b', 'c'];

for (const element of array1) {

console.log(element);

}

**Using in ForEach() Loop**

**This method is used for looping through an array element. Here’s an example of this.**

var names = ["jerry", "tom", "pluto", "micky", "mini"];  
names.forEach(function1);  
function function1(currentValue, index) {  
 console.log("Index in array is: "+index + " :: Value is: "+currentValue);  
}