JavaScript Iterables

Back to JS page

Table of Contents

- JavaScript Iterables
  - Iterating Over a String
    - Example 1
  - Iterating Over an Array
    - Example 2
  - Iterating Over a Set
    - Example 3
  - Iterating Over a Map
    - Example 4
  - Home Made Iterable
    - Example 5
  - Document
  - Reference

An Iterable is an Iterable Object

An iterable is an object that can be iterated over with `for...of`. Iterables include: strings, arrays, Sets, and Maps.

Iterating Over a String

Strings are iterable, so you can loop over each character:

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Iterables</h2>
<p>From w3schools.com, Experiment by Teeratus_R</p>

<h4>Iterating Over a String</h4>
<p id="demo"></p>

<script>
const word = "Hello";
let text = "";

for (let char of word) {
  text += char + " ";
}

document.getElementById("demo").innerHTML =
  "String: " + word + "<br>" +
  "Characters: " + text;
</script>

</body>
</html>
```

Example 1

Result View Example

Iterating Over an Array

Arrays are iterable, so you can loop over each element:

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Iterables</h2>
<p>From w3schools.com, Experiment by Teeratus_R</p>

<h4>Iterating Over an Array</h4>
<p id="demo"></p>

<script>
const fruits = ["Apple", "Banana", "Orange"];
let text = "";

for (let fruit of fruits) {
  text += fruit + "<br>";
}

document.getElementById("demo").innerHTML = text;
</script>

</body>
</html>
```

Example 2

Result View Example

Iterating Over a Set

Sets are iterable, so you can loop over each value:

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Iterables</h2>
<p>From w3schools.com, Experiment by Teeratus_R</p>

<h4>Iterating Over a Set</h4>
<p id="demo"></p>

<script>
const letters = new Set(["a", "b", "c"]);
let text = "";

for (let letter of letters) {
  text += letter + " ";
}

document.getElementById("demo").innerHTML =
  "Set: a, b, c<br>" +
  "Iterated: " + text;
</script>

</body>
</html>
```

Example 3

Result View Example

Iterating Over a Map

Maps are iterable, so you can loop over each entry:

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Iterables</h2>
<p>From w3schools.com, Experiment by Teeratus_R</p>

<h4>Iterating Over a Map</h4>
<p id="demo"></p>

<script>
const fruits = new Map([
  ["apple", 500],
  ["banana", 300]
]);
let text = "";

for (let entry of fruits) {
  text += entry[0] + " = " + entry[1] + "<br>";
}

document.getElementById("demo").innerHTML = text;
</script>

</body>
</html>
```

Example 4

Result View Example

Home Made Iterable

You can create your own iterable by implementing the `Symbol.iterator` method:

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Iterables</h2>
<p>From w3schools.com, Experiment by Teeratus_R</p>

<h4>Home Made Iterable</h4>
<p id="demo"></p>

<script>
// Create a home made iterable
function myIterable() {
  let i = 0;
  return {
    [Symbol.iterator]: function() {
      return {
        next: function() {
          i++;
          return i <= 5 ? {value: i, done: false} : {done: true};
        }
      };
    }
  };
}

let text = "";
for (let num of myIterable()) {
  text += num + " ";
}

document.getElementById("demo").innerHTML =
  "Home made iterable: " + text + "<br><br>" +
  "Custom iterable using Symbol.iterator and next()";
</script>

</body>
</html>
```

Example 5

Result View Example

Document

Document in project

You can Download PDF file.

Reference

- W3Schools JavaScript Iterables