

JAVASCRIPT - ARRAY MAP METHOD

http://www.tutorialspoint.com/javascript/array_map.htm

Copyright © tutorialspoint.com

Description

JavaScript array **map** method creates a new array with the results of calling a provided function on every element in this array.

Syntax

Its syntax is as follows –

```
array.map(callback[, thisObject]);
```

Parameter Details

- **callback** – Function that produces an element of the new Array from an element of the current one.
- **thisObject** – Object to use as **this** when executing callback.

Return Value

Returns the created array.

Compatibility

This method is a JavaScript extension to the ECMA-262 standard; as such it may not be present in other implementations of the standard. To make it work, you need to add the following code at the top of your script.

```
if (!Array.prototype.map)
{
    Array.prototype.map = function(fun /*, thisp*/)
    {
        var len = this.length;

        if (typeof fun !== "function")
            throw new TypeError();

        var res = new Array(len);
        var thisp = arguments[1];

        for (var i = 0; i < len; i++)
        {
            if (i in this)
                res[i] = fun.call(thisp, this[i], i, this);
        }
        return res;
    };
}
```

Example

Try the following example.

```
<html>
<head>
    <title>JavaScript Array map Method</title>
</head>

<body>
```

```
<script type="text/javascript">
  if (!Array.prototype.map)
  {
    Array.prototype.map = function(fun /*, thisp*/)
    {
      var len = this.length;

      if (typeof fun != "function")
        throw new TypeError();

      var res = new Array(len);
      var thisp = arguments[1];

      for (var i = 0; i < len; i++)
      {
        if (i in this)
          res[i] = fun.call(thisp, this[i], i, this);
      }
      return res;
    };
  }

  var numbers = [1, 4, 9];
  var roots = numbers.map(Math.sqrt);
  document.write("roots is : " + roots );
</script>

</body>
</html>
```

Output

roots is : 1 2 3

Loading [Mathjax]/jax/output/HTML-CSS/fonts/TeX/fontdata.js