



JavaScript cloneNode() Method

Summary: in this tutorial, you will learn how to use the JavaScript `cloneNode()` method to clone an element.

Introduction to cloneNode() method

The `cloneNode()` is a method of the `Node` interface that allows you to clone an element:

```
let clonedNode = originalNode.cloneNode(deep);
```

Parameters

deep

The `cloneNode()` method accepts an optional parameter `deep` :

- If the `deep` is `true` , then the original node and all of its descendants are cloned.
- If the `deep` is `false` , only the original node will be cloned. All the node's descendants will *not* be cloned.

The `deep` parameter defaults to `false` if you omit it.

originalNode

The `originalNode` is the element to be cloned.

Return value

The `cloneNode()` returns a copy of the `originalNode` .

Usage notes

Besides the DOM structure, the `cloneNode()` copies all attributes and inline listeners of the original node. However, it doesn't copy the event listeners added via `addEventListener()` or assignment to an element's properties such as `originalNode.onclick = eventHandler()` .

If you clone a node with an `id` attribute and place the cloned node in the same document, the `id` will be duplicated. In this case, you need to change the `id` before adding it to the DOM tree.

JavaScript cloneNode() method example

The following example uses the `cloneNode()` method to copy the `` element and place it in the same document:

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>JavaScript cloneNode() Demo</title>
</head>
<body>
  <ul id="menu">
    <li>Home</li>
    <li>Services</li>
    <li>About</li>
    <li>Contact</li>
  </ul>
  <script>
    let menu = document.querySelector('#menu');
    let clonedMenu = menu.cloneNode(true);
    clonedMenu.id = 'menu-mobile';
    document.body.appendChild(clonedMenu);
  </script>
</body>
</html>
```

How it works.

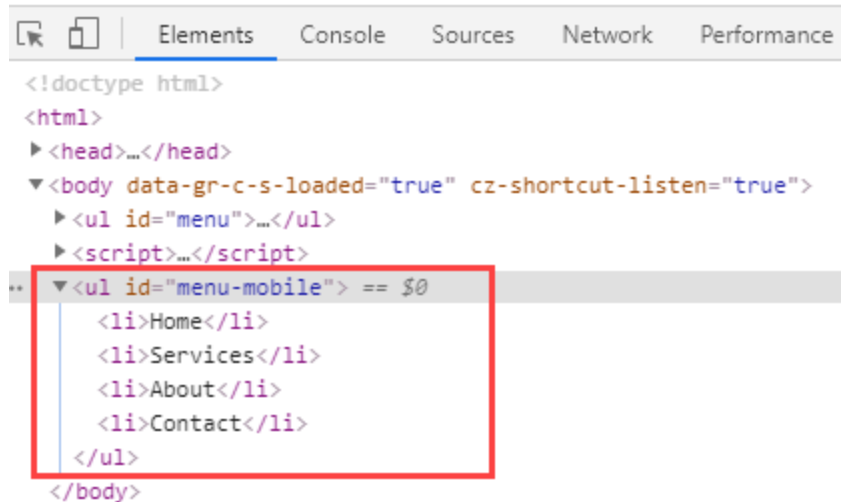
- First, select the `` with the id menu by using the `querySelector()` method.

- Second, create a deep clone of the `` element using the `cloneNode()` method.
- Third, change the `id` of the cloned element to avoid duplicates.
- Finally, append the cloned element to the child nodes of the `document.body` using the `appendChild()` method.

Output:

- Home
- Services
- About
- Contact

- Home
- Services
- About
- Contact



```
<!doctype html>
<html>
  <head>...</head>
  <body data-gr-c-s-loaded="true" cz-shortcut-listen="true">
    <ul id="menu">...</ul>
    <script>...</script>
    <ul id="menu-mobile"> == $0
      <li>Home</li>
      <li>Services</li>
      <li>About</li>
      <li>Contact</li>
    </ul>
  </body>
```

Summary

- Use the `node.cloneNode()` method to clone the node.
- Pass `true` into the `cloneNode()` method to create a deep clone of a DOM element.

Quiz