

# Getting Child Elements of a Node in JavaScript

**Summary:** in this tutorial, you will learn how to get the first child element, the last child element, and all children of a specified element.

Suppose that you have the following HTML fragment:

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>JS Get Child Elements</title>
</head>
<body>
  <ul id="menu">
    <li class="first">Home</li>
    <li>Products</li>
    <li class="current">Customer Support</li>
    <li>Careers</li>
    <li>Investors</li>
    <li>News</li>
    <li class="last">About Us</li>
  </ul>
</body>
</html>
```

## Get the first child element

To get the first child element of a specified element, you use the `firstChild` property of the element:

```
let firstChild = parentElement.firstChild;
```

If the `parentElement` does not have any child element, the `firstChild` returns `null` .

The `firstChild` property returns a child node which can be any node type such as an element node, a text node, or a comment node.

The following script shows the first child of the `#menu` element:

```
let content = document.getElementById('menu');
let firstChild = content.firstChild.nodeName;
console.log(firstChild);
```

Output:

```
#text
```

The Console window shows `#text` because a text node is inserted to maintain the whitespace between the opening `<ul>` and `<li>` tags. This whitespace creates a `#text` node.

Note that any whitespace such as a single space, multiple spaces, returns, and tabs will create a `#text` node. To remove the `#text` node, you can remove the whitespaces as follows:

```
<article id="content"><h2>Heading</h2><p>First paragraph</p></article>
```

Or to get the first child with the Element node only, you can use the `firstElementChild` property:

```
let firstElementChild = parentElement.firstElementChild;
```

The following code returns the first list item which is the first child element of the menu:

```
let content = document.getElementById('menu');
console.log(content.firstElementChild);
```

Output:

```
<li class="first">Home</li>
```

In this example:

- First, select the `#menu` element by using the `getElementById()` method.
- Second, get the first child element by using the `firstElementChild` property.

## Get the last child element

To get the last child element of a node, you use the `lastChild` property:

```
let lastChild = parentElement.lastChild;
```

In case the `parentElement` does not have any child element, the `lastChild` returns `null`.

The `lastChild` property returns the last element node, text node, or comment node. If you want to select only the last child element with the element node type, you use the `lastElementChild` property:

```
let lastChild = parentElement.lastElementChild;
```

The following code returns the list item which is the last child element of the menu:

```
let menu = document.getElementById('menu');  
console.log(menu.lastElementChild);
```

Output:

```
<li class="last">About Us</li>
```

## Get all child elements

To get a live `NodeList` of child elements of a specified element, you use the `childNodes` property:

```
let children = parentElement.childNodes;
```

The `childNodes` property returns all child elements with any node type. To get the child element with only the element node type, you use the `children` property:

```
let children = parentElement.children;
```

The following example selects all child elements of the element with the Id `main` :

```
let menu = document.getElementById('menu');  
let children = menu.children;  
console.log(children);
```

Output:

```
▼ HTMLCollection(7) [li.first, li, li.current, li, li, li, li.last] ⓘ  
  length: 7  
  ▶ 0: li.first  
  ▶ 1: li  
  ▶ 2: li.current  
  ▶ 3: li  
  ▶ 4: li  
  ▶ 5: li  
  ▶ 6: li.last  
  ▶ __proto__: HTMLCollection
```

## Summary

- The `firstChild` and `lastChild` return the first and last child of a node, which can be any node type including text node, comment node, and element node.
- The `firstElementChild` and `lastElementChild` return the first and last child Element node.
- The `childNodes` returns a live `NodeList` of all child nodes of any node type of a specified node. The `children` return all child `Element` nodes of a specified node.

## Quiz