JavaScript insertAfter

Summary: in this tutorial, you will learn how to insert a new node after an existing node as a child node of a parent node.

> This tutorial was written when DOM API did not support the `after()` method. Now, you can use the `after()` method to insert a node after an element.

JavaScript DOM provides the `insertBefore()` method that allows you to insert a new node after an existing node as a child node.

To insert a new node after an existing node as a child node, you can use the following approach:

- First, select the next sibling node of the existing node.
- Then, select the parent node of the existing node and call the `insertBefore()` method on the parent node to insert a new node before that immediate sibling node.

The following `insertAfter()` function illustrates the logic:

```
function insertAfter(newNode, existingNode) {
    existingNode.parentNode.insertBefore(newNode, existingNode.nextSibling);
}
```

Suppose that you have the following list of items:

```
<ul id="menu">
    <li>Home</li>
    <li>About</li>
    <li>Contact</li>
</ul>
```

The following inserts a new node after the last list item:

```javascript
let menu = document.getElementById('menu');
// create a new li node
let li = document.createElement('li');
li.textContent = 'Services';


// insert a new node after the last list item
insertAfter(li,  menu.lastElementChild);
```

How it works:

- First, select the `ul` element by its id (`menu`) using the `getElementById()` method.

- Second, create a new list item using the `createElement()` method.

- Third, use the `insertAfter` () method to insert a list item element after the last list item element.

Put it all together.

```html
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>JavaScript insertAfter() Demo</title>
</head>
<body>
    <ul id="menu">
        <li>Home</li>
        <li>About</li>
        <li>Contact</li>
    </ul>
    <script>
        function insertAfter(newNode, existingNode) {
            existingNode.parentNode.insertBefore(newNode, existingNode.nextSibling);
        }

        let menu = document.getElementById('menu');
```

```
        // create a new li node
        let li = document.createElement('li');
        li.textContent = 'Services';
        insertAfter(li,  menu.lastElementChild);
    </script>
</body>


</html>
```

The menu looks like the following after the insert:

```
<ul id="menu">
    <li>Home</li>
    <li>About</li>
    <li>Contact</li>
    <li>Services</li>
</ul>
```

Summary

- JavaScript DOM hasn't supported the `insertAfter()` method yet.

- Use the `insertBefore()` method and the `nextSibling` property to insert a new before an existing node as a child of a parent node.