

Tạo mới thêm xóa phần tử Dom - JavaScript

Tạo ra một nút phần tử HTML trong DOM, chèn phần tử HTML, xóa phần tử và thay thế phần tử trong DOM HTML

Tạo phần tử

Bạn có thể dùng JavaScript để tạo mới phần tử HTML sau đó chèn vào trang web. Đây là các cách giúp bạn tạo mới một phần tử HTML

`document.createElement(tag_name)` tạo ra phần tử có thể `tag_name` như `a`, `p`, `div` ...

`element.cloneNode()` Tạo ra một phần tử bằng cách nhân bản phần tử chỉ ra (`element`)

`document.createTextNode(text)` tạo phần tử từ văn bản `text` HTML

```
var node = document.createTextNode("Tạo ra một phần tử");

var linknode = document.createElement("a");
linknode.href="https://xuanthulab.net/";
linknode.innerText="xuanthulab.net";
```

Ví dụ trên sẽ tạo ra một nút text, nhưng nó chưa hiện thị cho đến khi bạn gắn phần tử đó vào HTML document để nó là phần tử con của một phần tử nào đó, có một số cách để gắn phần tử tạo ra từ JavaScript vào DOM HTML

`element.appendChild(newNode)` Thêm phần tử `newNode` vào phần tử `element` nó trở thành phần tử con sau cùng của `element`

`element.insertBefore(newNode, node2)` Chèn phần tử `newNode` nằm phía trước `node2`

`element.replaceChild(newNode, oldNode)` Thay thế phần tử `oldNode` bằng phần tử `newNode`

Ví dụ sau tạo ra một phần tử đoạn văn `p` sau đó chèn nó vào phần tử `div` đang có sẵn trong DOM HTML

```
<div id ="demo">nội dung ví dụ</div>

<button onclick="add_child()">KẾT QUẢ</button>
<script>
  function add_child() {
    //tạo phần tử p
    var p = document.createElement("p");

    //tạo phần tử text
    var node = document.createTextNode("Some new text");

    //gắn node vào p
    p.appendChild(node);

    //Thay đổi một số thuộc tính của p
    p.appendChild(node);
    p.style.backgroundColor = 'red';
    p.style.padding = "10px";
    p.style.color = "white";
```

```
var div = document.getElementById("demo");
//gắn p vào div
div.appendChild(p);
}
</script>
```

nội dung ví dụ

KẾT QUẢ

Loại bỏ phần tử

Để loại bỏ phần tử HTML, bạn chọn phần tử cha rồi sử dụng phương thức `removeChild(node)`

```
<div id="demo">
  <p id="p1">This is a paragraph.</p>
  <p id="p2">This is another paragraph.</p>
</div>

<script>
  var parent = document.getElementById("demo");
  var child = document.getElementById("p1");
  parent.removeChild(child);
</script>
```

Ví dụ trên sẽ xóa bỏ phần tử đoạn văn thứ nhất

Bạn có thể sử dụng thủ thuật lấy thuộc tính `parentNode` để bỏ qua bước tìm phần tử cha trong DOM: `child.parentNode.removeChild(child);`

```
<div id="demo">
  <p id="p1">This is a paragraph.</p>
  <p id="p2">This is another paragraph.</p>
</div>

<script>
  var child = document.getElementById("p1");
  child.parentNode.removeChild(child);
</script>
```

Thay thế phần tử

Để thay thế một phần tử bằng một phần tử khác dùng cú pháp `element.replaceChild(newNode, oldNode)`. Trong đó `element` là nút cha

```
<div id="demo">
  <p id="p1">This is a paragraph.</p>
  <p id="p2">This is another paragraph.</p>
</div>

<script>
  var p = document.createElement("p");
  p.innerText = 'Đây là đoạn văn mới tạo ra';
```

```
var parent = document.getElementById("demo");  
var child = document.getElementById("p1");  
parent.replaceChild(p, child);  
</script>
```