

STEM Digital Academy

School of Science & Technology

Methods for Document and Node Interface

Web Technologies



Lecture by Dr Elahe Kani-Zabihi

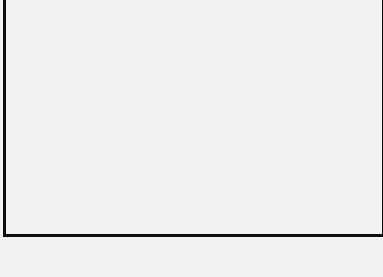
```
<!doctype html>
<html>
    <head>
     <title>
       Web Technologies
     </title>
     <style>
         color: ■blue;
     </style>
     <script>
     console.log("client-side");
     </script>
    </head>
    <body>
       >
           Welcome!
       </body>
</html>
```



What we will cover

Methods for document interface

Methods for node interface





console.log()

- Displaying messages
- Outputting values
- Debugging
- Inspecting objects





Methods for Document Interface 1

createElement (element)

```
const pElement = document.createElement ("p");
pElement.id = "newText";
pElement.textContent = "Just a text";

console.log(pElement.textContent);
//output in Console is: Just a text
```



Methods for Document Interface 2

createTextNode (value)

const addText = document.createTextNode("New text added
be new method");



appendChild (node)

```
const pElement = document.createElement ("p");
const addText = document.createTextNode("New text added
be new method");
pElement.appendChild (addText);
```



insertBefore (new, existing)

```
<main id="mainId">
  <h1></h1>
  <h1 id="head">
  </h1>
</main>
```

```
const parent = document.getElementById
("mainId");
const pElement = document.createElement
("p");
const existingChild = document.getElementById
("head");

parent.insertBefore (pElement, existingChild);
```



replaceChild (new, existing)

```
<main id="mainId">
  <h1></h1>
  <h1 id="head">
  </h1>
</main>
```

```
const parent = document.getElementById
("mainId");
const pElement = document.createElement
("p");
const existingChild =
document.getElementById ("head");
parent.replaceChild (pElement, existingChild);
```



removeChild (node)

```
<main id="mainId">
  <h1></h1>
  <h1 id="head">
  </h1>
</main>
```

```
const parent = document.getElementById
("mainId");
const existingChild =
document.getElementById ("head");
parent.removeChild (existingChild);
```



getRootNode ()

```
const nodChild = document.getElementById ("head");
const rooNode = nodChild.getRootNode();
console.log(rooNode);
//Output in Console: #document
```



Try It Yourself

In this activity you can try the new methods.

So,

- 1. Create a new HTML file with the <main> and <script> elements.
- 2. Use the code in slides 5 to 11 to practice the new methods for document and node interfaces.

