

Methods for Document and Node Interface

Web Technologies



Lecture by
Dr Elahe Kani-Zabihi

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


What we will cover

- **Methods for document interface**
- **Methods for node interface**

console.log ()

- Displaying messages
- Outputting values
- Debugging
- Inspecting objects

```
13 </form>
14 <script>
15     const dayRadio = document.
16     const nightRadio = document.
17
18     console.log(dayRadio.value);
19     console.log(nightRadio.value);
20     console.log(dayRadio.checked);
21     console.log(nightRadio.checked);
22
23 </script>
24 </body>
25 </html>
```

PROBLEMS OUTPUT **DEBUG CONSOLE** TERMINAL Filter (e.g. text, lexcl... Open radio.html

day
night
false
true

radio.html:18
radio.html:19
radio.html:20
radio.html:21

Console
Section

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");
```

Methods for Node Interface 1

- `appendChild (node)`

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

Methods for Node Interface 2

- 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);
```


Methods for Node Interface 3

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);
```

Methods for Node Interface 4

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);
```

Methods for Node Interface 5

- `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.