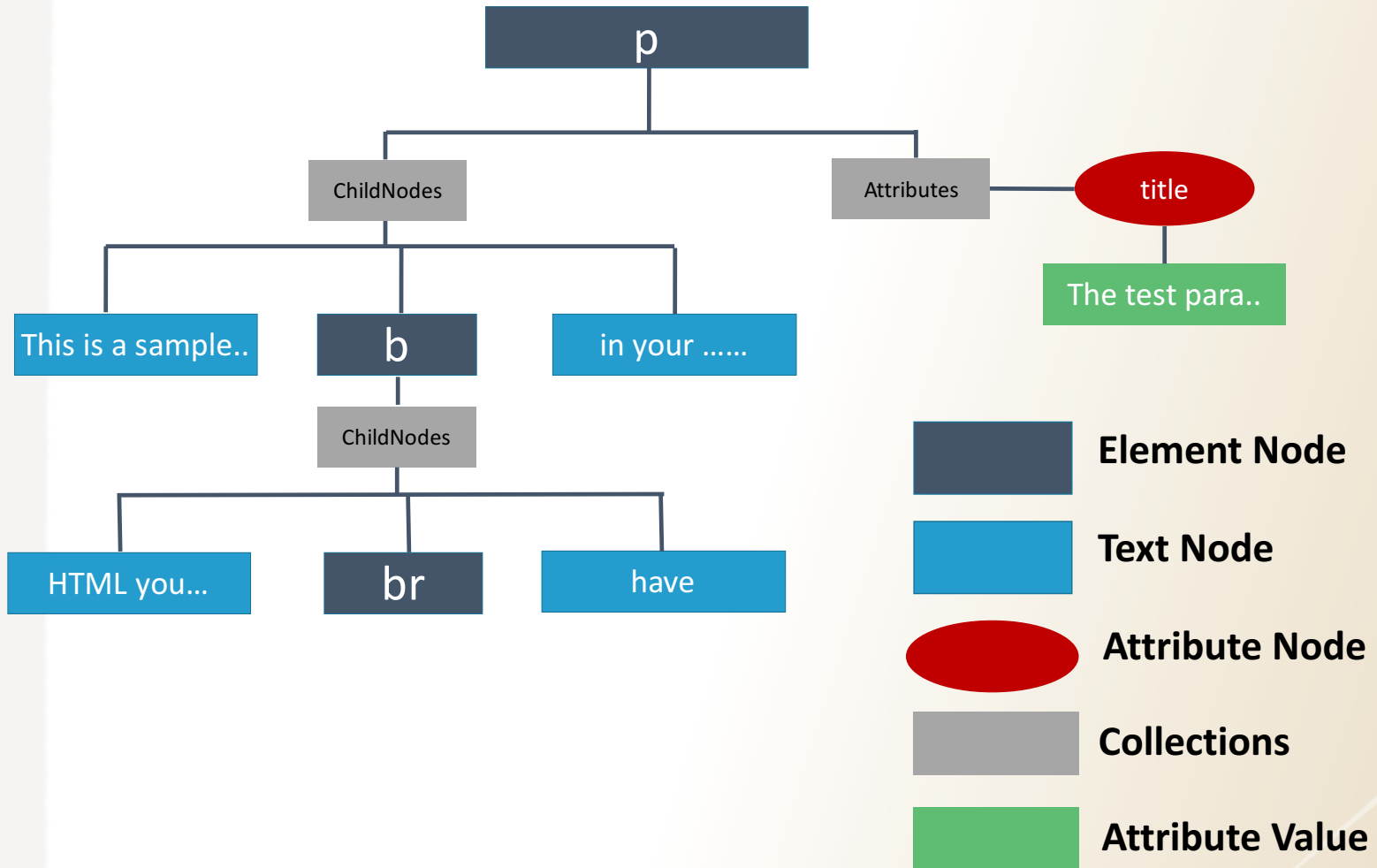


>> JavaScript: DOM Manipulation

(c) Dr. Mohammed Misbhaudhin



Document Object Model



DOM Traversal

- **Selecting a Node from the DOM**
 - Basic Methods
 - getElementById()
 - getElementsByTagName()
 - getElementsByClassName()
 - CSS Selector-based Methods
 - querySelector()
 - querySelectorAll()

WHAT do we do after selecting a node?

- Update the node
 - Update Content
 - Update Attribute Values
 - Update Style
- Create a new node
- Delete a node

Practice Setup

- Please open browser
- Go this Link
<https://codepen.io/drmisbha/pen/YBRQbp>
– (The link is on the Class JS Website)
- Write the code in the **JS Window**

UPDATE A DOM NODE

Update Content

INNERHTML

Change the HTML content of an element

DOM Element Property

```
node.innerHTML = "Paragraph changed!";
```

Selected Node
Variable

New Content

Assume that the node is selected in the variable node

```
<!DOCTYPE html>
<html>
  <head>
    <title> Test Page </title>
  </head>
  <body>
    <div>
      <p>First Test Paragraph</p>
      <p>Second Test Paragraph</p>
    </div>
    <a href="www.kfu.edu.sa">KFU</a>
  </body>
</html>
```

New First Paragraph
Second Test Paragraph
[KFU](#)

JS Window

```
<script>
  var p = document.querySelectorAll("p");
  p[0].innerHTML = "New First Paragraph";
</script>
```


TEXTCONTENT

Get or Set the text content of an element

DOM Element Property

```
node.textContent = "Paragraph changed!";
```

Selected Node
Variable

New Content

Assume that the node is selected in the variable node

```
<!DOCTYPE html>
<html>
  <head>
    <title> Test Page </title>
  </head>
  <body>
    <div>
      <p>First Test Paragraph</p>
      <p>Second Test Paragraph</p>
    </div>
    <a href="www.kfu.edu.sa">KFU</a>
  </body>
</html>
```

First Test Paragraph
New Second Paragraph
[KFU](#)

JS Window

```
<script>
  var p = document.querySelectorAll("p");
  p[1].textContent = "New Second Paragraph";
</script>
```

innerHTML vs textContent

- The content inside **innerHTML** can contain HTML tags and it will be parsed.
- **textContent** will consider the content as plain text only
- Since **innerHTML** parses content as HTML, it takes longer
- Since **textContent** uses straight text, it does not parse HTML and is faster.

```
<!DOCTYPE html>
<html>
  <head>
    <title> Test Page </title>
  </head>
  <body>
    <div>
      <p>First Test Paragraph</p>
      <p>Second Test Paragraph</p>
    </div>
    <a href="www.kfu.edu.sa">KFU</a>
  </body>
</html>
```

New First Para
New Second Para
KFU

JS Window

```
<script>
  var p = document.querySelectorAll("p");
  p[0].innerHTML = "<em>New First Para</em>";
  p[1].textContent = "<em>New Second Para</em>";
</script>
```

UPDATE A DOM NODE

Update Attribute Values

GETATTRIBUTE()

Get the value of an attribute in the element

DOM Element Method

```
node.getAttribute("href");
```

Selected Node Variable Attribute Name

Assume that the node is selected in the variable node

```
<!DOCTYPE html>
<html>
  <head>
    <title> Test Page </title>
  </head>
  <body>
    <div>
      <p>First Test Paragraph</p>
      <p>Second Test Paragraph</p>
    </div>
    <a href="www.kfu.edu.sa">KFU</a>
  </body>
</html>
```

Console

"www.kfu.edu.sa"

JS Window

```
<script>
  var a = document.querySelector("a");
  var attr = a.getAttribute("href");
  console.log(attr);
</script>
```

SETATTRIBUTE()

Set the value of an attribute in the element

DOM Element Method

```
node.setAttribute("href", "www.kfu.edu.sa/sites");
```

Selected Node
Variable

Attribute Name

New Attribute Value

Assume that the node is selected in the variable node



http://www.kfu.edu.sa/ccsit/is/web-based-systems

```
<!DOCTYPE html>
<html>
  <head>
    <title> Test Page </title>
  </head>
  <body>
    <div>
      <p>First Test Paragraph</p>
      <p>Second Test Paragraph</p>
    </div>
    <a href="www.kfu.edu.sa">KFU</a>
  </body>
</html>
```

Console

"https://www.kfu.edu.sa"

JS Window

```
<script>
  var a = document.querySelector("a");
  a.setAttribute("href", "https://www.kfu.edu.sa");
  console.log(a.getAttribute("href"));
</script>
```

UPDATE A DOM NODE

Update Style

STYLE

Set the CSS property for
a selected element

DOM Element Property

node.style.color = "red";

Selected Node
Variable

CSS Property

Value

Assume that the node is selected in the variable node

```
<!DOCTYPE html>
<html>
  <head>
    <title> Test Page </title>
  </head>
  <body>
    <div>
      <p>First Test Paragraph</p>
      <p>Second Test Paragraph</p>
    </div>
    <a href="www.kfu.edu.sa">KFU</a>
  </body>
</html>
```

First Test Paragraph
Second Test Paragraph
KFU

JS Window

```
<script>
  var p = document.querySelectorAll("p");
  p[0].style.color = "Green";
</script>
```

Properties with a “-”

- Some CSS Properties have a hyphen “-” in the name
 - Such as font-family, text-decoration, font-style and so on
- Using a “-” is not allowed in JS
- Hence, such properties are written using Camel Case

font-family => font**F**amily

text-decoration => text**D**ecoration

```
<!DOCTYPE html>
<html>
  <head>
    <title> Test Page </title>
  </head>
  <body>
    <div>
      <p>First Test Paragraph</p>
      <p>Second Test Paragraph</p>
    </div>
    <a href="www.kfu.edu.sa">KFU</a>
  </body>
</html>
```

First Test Paragraph
Second Test Paragraph
KFU

JS Window

```
<script>
  var p = document.querySelectorAll("p");
  p[0].style.textDecoration = "none";
</script>
```

CREATE A NEW DOM NODE



STEP 1

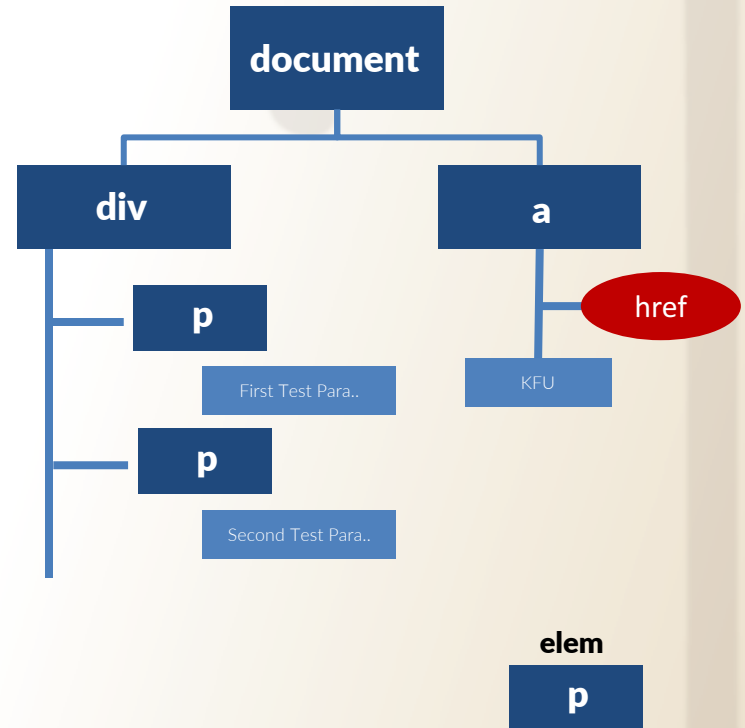
Create an Empty Node `createElement()`

```
var elem = document.createElement("P");
```

Diagram illustrating the components of the `createElement()` function call:

- `var elem`: Variable to store the new Element
- `document`: Object representing the document
- `createElement`: Function Name
- `"P"`: HTML Element Name


```
<!DOCTYPE html>
<html>
  <head>
    <title> Test Page </title>
  </head>
  <body>
    <div>
      <p>First Test Paragraph</p>
      <p>Second Test Paragraph</p>
    </div>
    <a href="www.kfu.edu.sa">KFU</a>
  </body>
</html>
```



JS Window

```
<script>
  var elem = document.createElement("P");
</script>
```

STEP 2

Add Content to the Node **textContent()**

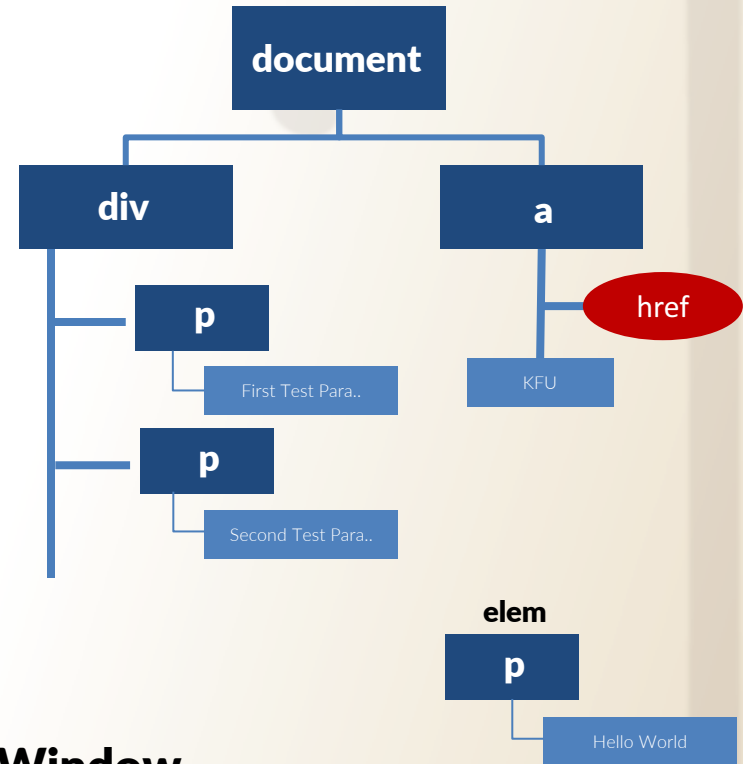
DOM Element Property

elem.textContent = "Hello World";

Element Node

Element Content

```
<!DOCTYPE html>
<html>
  <head>
    <title> Test Page </title>
  </head>
  <body>
    <div>
      <p>First Test Paragraph</p>
      <p>Second Test Paragraph</p>
    </div>
    <a href="www.kfu.edu.sa">KFU</a>
  </body>
</html>
```



JS Window

```
<script>
  var elem = document.createElement("P");
  elem.textContent = "Hello World";
</script>
```

STEP 3

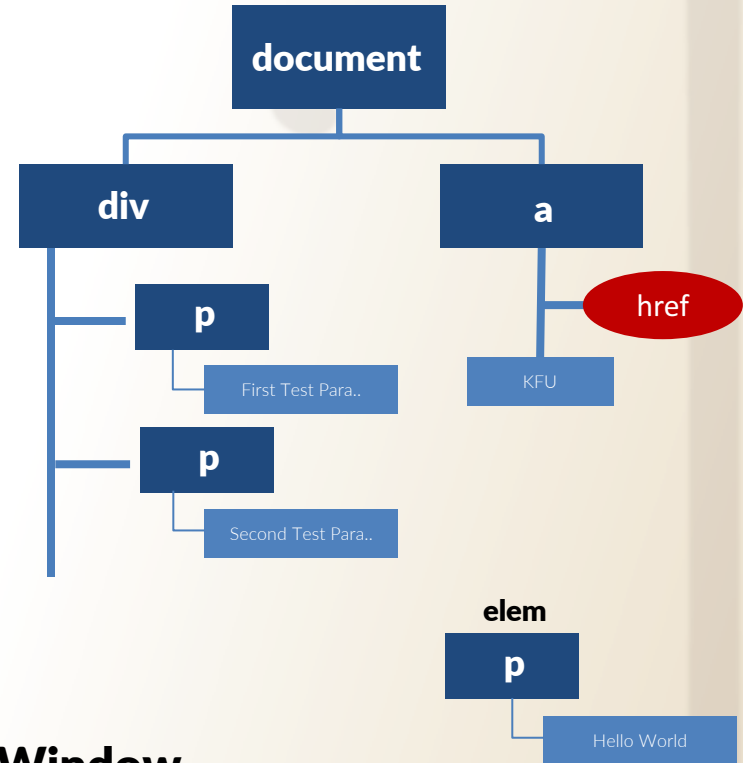
Get the Parent to Attach the Node to
querySelector()

```
var parent = document.querySelector("div");
```

Diagram illustrating the components of the code:

- Function Name**: `querySelector()`
- Variable to Store the Parent Element**: `parent`
- Element Name**: `"div"`

```
<!DOCTYPE html>
<html>
  <head>
    <title> Test Page </title>
  </head>
  <body>
    <div>
      <p>First Test Paragraph</p>
      <p>Second Test Paragraph</p>
    </div>
    <a href="www.kfu.edu.sa">KFU</a>
  </body>
</html>
```



JS Window

```
<script>
  var elem = document.createElement("P");
  elem.textContent = "Hello World";
  var parent = document.querySelector("div");
</script>
```

STEP 4

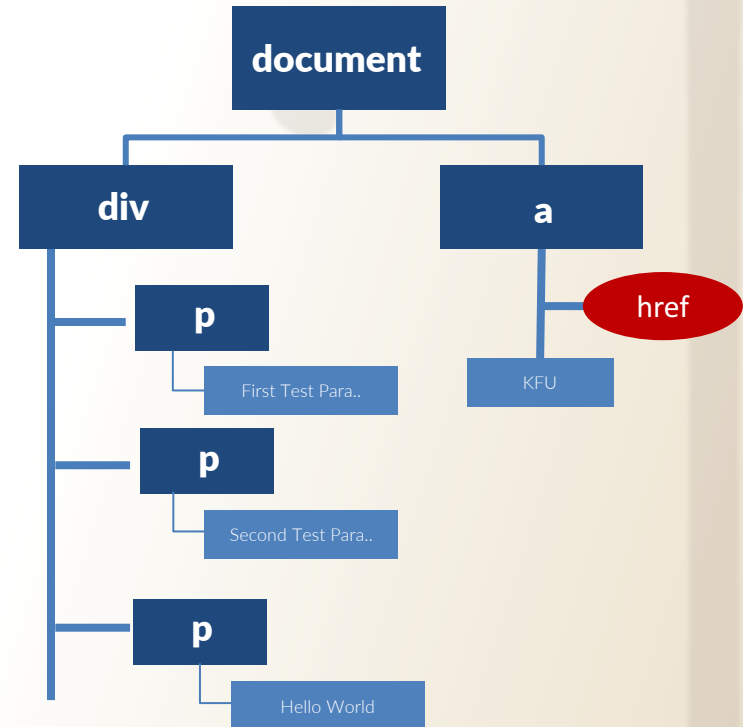
Attach the node to the DOM as a child of the parent
appendChild()

Function Name

```
parent.appendChild(elem);
```

Parent Node Node to Attach

```
<!DOCTYPE html>
<html>
  <head>
    <title> Test Page </title>
  </head>
  <body>
    <div>
      <p>First Test Paragraph</p>
      <p>Second Test Paragraph</p>
    </div>
    <a href="www.kfu.edu.sa">KFU</a>
  </body>
</html>
```



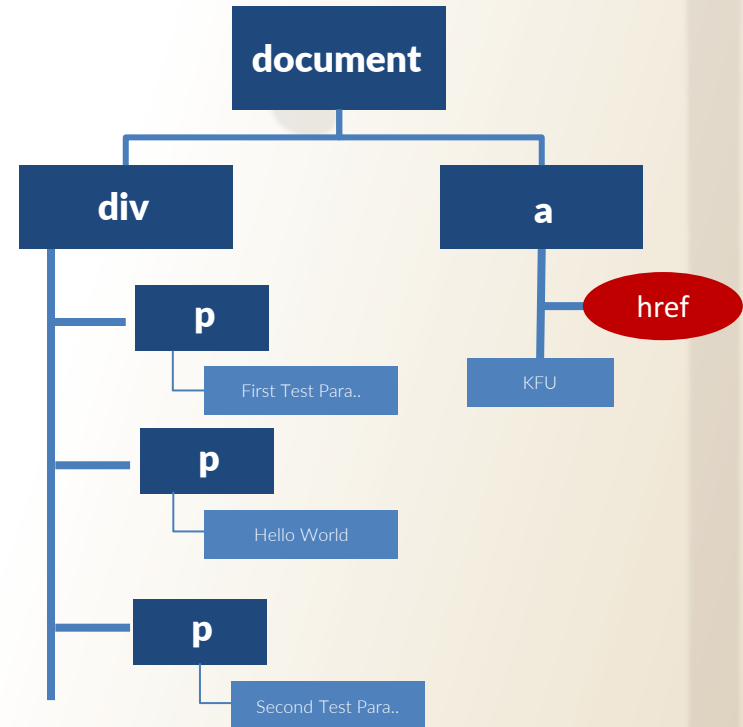
JS Window

```
<script>
  var elem = document.createElement("P");
  elem.textContent = "Hello World";
  var parent = document.querySelector("div");
  parent.appendChild(elem);
</script>
```

appendChild()

- The **appendChild()** function attaches the element to the end of all the children of the selected parent
- Cannot be used if we want to attach in the middle.
- To do that, we need a sibling as a reference and use the “**insertBefore**” method.


```
<!DOCTYPE html>
<html>
  <head>
    <title> Test Page </title>
  </head>
  <body>
    <div>
      <p>First Test Paragraph</p>
      <p>Second Test Paragraph</p>
    </div>
    <a href="www.kfu.edu.sa">KFU</a>
  </body>
</html>
```



JS Window

```
<script>
  var elem = document.createElement("P");
  elem.textContent = "Hello World";
  var parent = document.querySelector("div");
  parent.insertBefore(elem, parent.children[1]);
</script>
```

insertBefore()

Insert the new element before the selected child element

Function Name

```
parent.insertBefore(elem, parent.children[1]);
```

Parent Node

Node to Attach

Element before to attach to

REMOVE A DOM NODE

removeChild()

Removes the identified child element

Function Name

```
parent.removeChild(parent.children[0]);
```

Parent Node

Element before to attach to

```
<!DOCTYPE html>
<html>
  <head>
    <title> Test Page </title>
  </head>
  <body>
    <div>
      <p>First Test Paragraph</p>
      <p>Second Test Paragraph</p>
    </div>
    <a href="www.kfu.edu.sa">KFU</a>
  </body>
</html>
```

Second Test Paragraph

KFU

JS Window

```
<script>
  var parent = document.querySelector("div");
  parent.removeChild(parent.children[0]);
</script>
```

Summary

- How to change HTML content (Update) in the DOM?
 - innerHTML
 - textContent
- How to create and add a new element to the DOM?
 - createElement()
 - appendChild()
 - insertBefore()
- How to remove elements from the DOM?
 - removeChild()
- How to set and get attributes for HTML Elements in the DOM?
 - setAttribute() and getAttribute()
- How to manipulate CSS Styles from JavaScript?
 - Use style property
 - Use setAttribute() function