



HTML Document Object Model

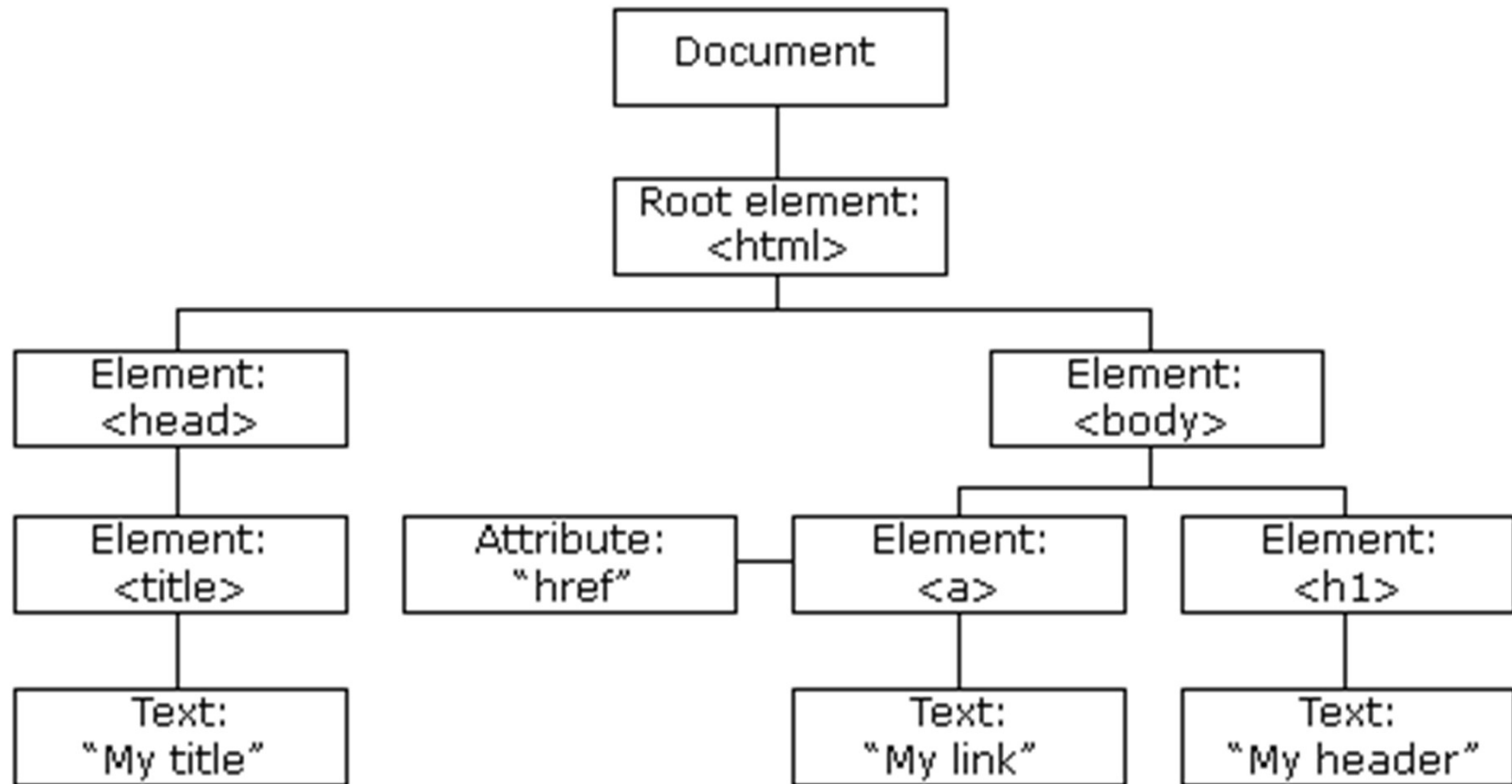
DOM

DOM

- When a web page is loaded, the browser creates a Document Object Model of the page.
- The **HTML DOM** model is constructed as a tree of **Objects**
- The HTML DOM is a standard for how to get, change, add, or delete HTML elements.



DOM (2)



JavaScript and DOM

- JavaScript can:

- change all the HTML elements in the page
- change all the HTML attributes in the page
- change all the CSS styles in the page
- remove existing HTML elements and attributes
- add new HTML elements and attributes
- react to all existing HTML events in the page
- create new HTML events in the page



getElementById

- The most common way to access an HTML element is to use the id of the element.
- In the example above, the `getElementById` method used `id="demo"` to find the element.



innerHTML

- The easiest way to get the **content** of an element is by using the `innerHTML` property
- The `innerHTML` property is useful for getting or replacing the content of HTML elements.
- The `innerHTML` property can be used to get or change any HTML element, including `<html>` and `<body>`.



Example

```
<html>
```

```
<body>
```

```
<p id="demo"></p>
```

```
<script>
```

```
document.getElementById("demo").innerHTML = "Hello World!";
```

```
</script>
```

```
</body>
```

```
</html>
```



Finding HTML Elements

Method	Description
<code>document.getElementById(<i>id</i>)</code>	Find an element by element id
<code>document.getElementsByTagName(<i>name</i>)</code>	Find elements by html tag name (h1, h2, p...)
<code>document.getElementsByClassName(<i>name</i>)</code>	Find elements by class name



Changing HTML Elements

Property	Description
<i>element.innerHTML = new html content</i>	Change the inner HTML of an element
<i>element.attribute = new value</i>	Change the attribute value of an HTML element
<i>element.setAttribute(attribute, value)</i>	Change the attribute value of an HTML element
<i>element.style.property = new style</i>	Change the style of an HTML element

Adding and Deleting Elements

Method	Description
<code>document.createElement(<i>element</i>)</code>	Create an HTML element
<code>document.removeChild(<i>element</i>)</code>	Remove an HTML element
<code>document.appendChild(<i>element</i>)</code>	Add an HTML element
<code>document.replaceChild(<i>new</i>, <i>old</i>)</code>	Replace an HTML element
<code>document.write(<i>text</i>)</code>	Write into the HTML output stream



Adding Events Handlers

Method	Description
<code>document.getElementById(<i>id</i>).onclick = function(){<i>code</i>}</code>	Adding event handler code to an onclick event

See next lecture



Finding HTML Objects

Property	Description
document.anchors	Returns all <a> elements that have a name attribute
document.applets	Deprecated
document.baseURI	Returns the absolute base URI of the document
document.body	Returns the <body> element
document.cookie	Returns the document's cookie
document.doctype	Returns the document's doctype
document.documentElement	Returns the <html> element
document.documentMode	Deprecated
document.documentURI	Returns the URI of the document
document.domain	Returns the domain name of the document server
document.domConfig	Obsolete.
document.embeds	Returns all <embed> elements
document.URL	Returns the complete URL of the document



Finding HTML Objects (2)

Property	Description
document.forms	Returns all <form> elements
document.head	Returns the <head> element
document.images	Returns all elements
document.implementation	Returns the DOM implementation
document.inputEncoding	Returns the document's encoding (character set)
document.lastModified	Returns the date and time the document was updated
document.links	Returns all <area> and <a> elements that have a href attribute
document.readyState	Returns the (loading) status of the document
document.referrer	Returns the URI of the referrer (the linking document)
document.scripts	Returns all <script> elements
document.strictErrorChecking	Returns if error checking is enforced
document.title	Returns the <title> element
document.URL	Returns the complete URL of the document



Finding HTML Element by Id

```
const element =  
document.getElementById("intro");
```



Finding HTML Elements by Tag Name

```
const elements =  
document.getElementsByTagName("p");
```



Find elements inside an element

```
const x =  
document.getElementById("main");
```

```
const y =  
x.getElementsByTagName("p");
```



Finding HTML Elements by Class Name

```
const x =  
document.getElementsByClassName("intro");
```



Finding HTML Elements by CSS Selectors

```
const x =  
document.querySelectorAll("p.intro");
```



Finding HTML Elements by HTML Object Collections

- This example finds the form element with id="frm1", in the forms collection, and displays all elements' values:

```
const x = document.forms["frm1"];
let text = "";
for (let i = 0; i < x.length; i++) {
    text += x.elements[i].value + "<br>";
}
document.getElementById("demo").innerHTML =
text;
```

