Document Object Model (DOM)

The **DOM**

document

document object
root of html
owns other nodes

HTML

HEAD

BODY

H1

text

P

text

H2

text

H2

text

```
<html>
  <head>
  </head>
  <body>
    <h1>Heading 1</h1>
    <p>Paragraph</p>
    <h2>Heading 2</h2>
  </body>
</html>
```
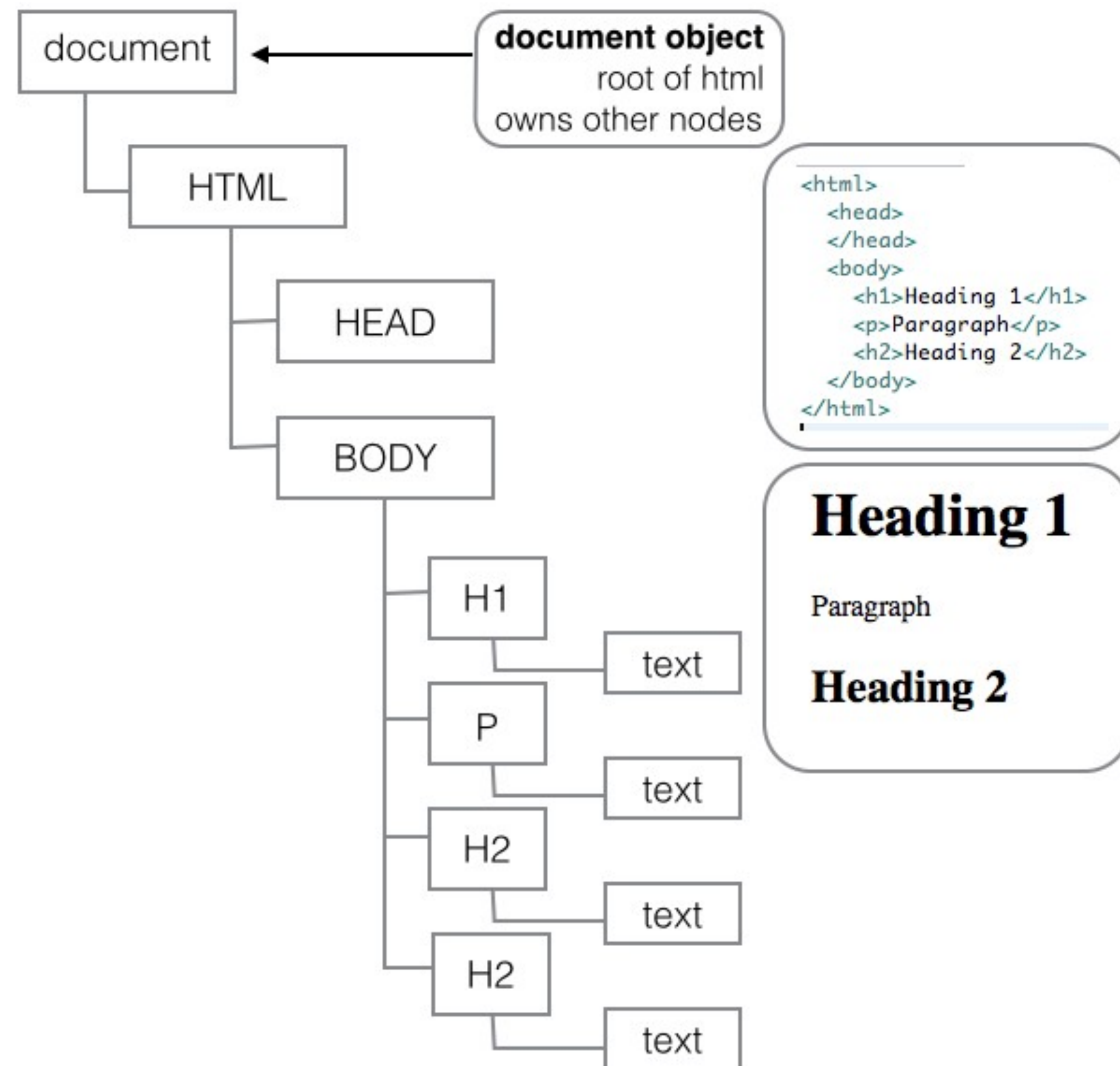
Heading 1

Paragraph

Heading 2

The **DOM**

- A World Wide Web Consortium (W3C) Standard
- Defines standard for accessing web documents
- Represents the displayed web page
- Each element represented in the DOM by its own object
  - Access and modify individual elements
  - Add and delete elements

```
<script>
document.getElementById('demo').innerHTML = 'Hello JavaScript!';
</script>
```

HTML DOM Document Object

HTML DOM Nodes

In the HTML DOM (Document Object Model), everything is a node:

- The document itself is a document node
- All HTML elements are element nodes
- All HTML attributes are attribute nodes
- Text inside HTML elements are text nodes
- Comments are comment nodes

DOM

The Web Browser

On opening HTML document in browser:
- It becomes a **document object**
  - The **document object** is root node of HTML document
  - **document object** provides properties and methods to access node objects from within JavaScript.

```
<script>
let x = document.getElementsByName('map');
alert(map.length);

</script>
```

DOM Access

Click the button to change this text.

Try it

Hi ICTSkills

Try it

DOM Access
Using DOM method

```html
<!DOCTYPE html>
<html>
  <body>
    <p id="demo">Click the button to change this text.</p>
    <button onclick="domAccess()">Tryit</button>
    <script src="dom.js"></script>
  </body>
</html>
```

```javascript
//file: dom.js
function domAccess(){
  document.getElementById('demo').innerHTML = 'Hi ICTSkills';
}
```

DOM Access

Using jQuery

```html
<!DOCTYPE html>
<html>
   <body>
     <p id="demo">Click the button to change this text.</p>
      <button onclick="domAccess()">Tryit</button>
      <script src="jquery.js"></script>
    </body>
  </html>
```

```javascript
//file: jq.js
function jQueryAccess(){
 $('#demo').html('Hi ICTSkills');
}
```

HTML Nodes
Methods to retrieve nodes

- document.getElementById(id)
  - **id** unique on a page hence *getElementById*
- document.getElementsByName(name)
  - returns array of elements with **name** attribute = *name*
  - **name** need not be unique hence *getElementsByName*
- node.getElementsByTagName(tagName)
  - returns array of elements with **tagName** attribute = *tagName*

Get element by id
Simple demo document.getElementById(id)

Prints the height of image whose *id="img1"*

Native JavaScript

```javascript
//in html file
<img src="img/01.png" id="img1">
//in javascript file
let image = document.getElementById('img1');
alert('Image height is ' + image.height);
```

jQuery

```javascript
//in html file
<img src="img/01.png" id="img1">
//in javascript file
alert('Image height is ' + $('#img1').height());
```

Get elements by name

Simple demo `document.getElementsByName(name)`

Discovers images with attribute *name="imgs"*

Native JavaScript

```javascript
let images = document.getElementsByName('imgs');

for (let i = 0; i < images.length; i++) {
 alert('Image height is ' + images[i].height);
}
```

jQuery

```javascript
let $images = $('[name="imgs"]');
images.each(function (){
 alert('Image height is ' + $(this).height());
});
```

Get elements by tagName

Can be used on a sub-tree, not just entire document

Native JavaScript

```
let imgDiv = document.getElementById('ictskills-images'); let
images = imgDiv.getElementsByTagName('img');
for (let i = 0; i < images.length; i++) {
 alert('Image height is ' + images[i].height);
}
```

jQuery

```
//let images: only those contained in node <div id="ictskills-imgs">
//with attribute name="imgs", e.g.: <img src="img/01.png" name="imgs">
let $images = $('#ictskills-imgs [name=\'imgs\']');
images.each(function (){
 alert('Image height is ' + $(this).height());
});
```

Hide | Reveal Elements

Using Native JavaScript

HTML

```html
<p id="text">Watch me appear and disappear</p>
<button onclick="hide()">Hide</button>
<button onclick="reveal()">Reveal</button>
```

JavaScript

```javascript
function hide(){
 document.getElementById('text').style.visibility = 'hidden';
}

function reveal(){
 document.getElementById('text').style.visibility = 'visible';
}
```

Hide | Reveal Elements

Using jQuery

HTML

```html
<p id="text">Watch me appear and disappear</p>
<button onclick="hide()">Hide</button>
<button onclick="reveal()">Reveal</button>
```

jQuery

```javascript
function hide(){
    $('#text').hide();
}

function reveal(){
    $('#text').show();
}
```

Semantic UI Enable Dropdown Box using JQuery

HTML

```html
<div class="ui selection dropdown">
<input name="amountDonated" type="hidden">
<div class="default text">Amount</div>
<i class="dropdown icon"></i>
<div class="menu">
<div class="item" data-value="100">$100</div>
<div class="item" data-value="200">$200</div>
<div class="item" data-value="300">$300</div>
</div>
</div>
```

jQuery

```javascript
//Ensure you use latest version jQuery
//http://code.jquery.com/jquery-latest.min.js
<script>$('.ui.selection.dropdown').dropdown('enable');</script>
```

Summary

- Document Object Model (DOM)
  - HTML page underlying data structure.
  - Difficult development environment.
  - Better to use jQuery v native JavaScript?