JÖNKÖPING UNIVERSITY

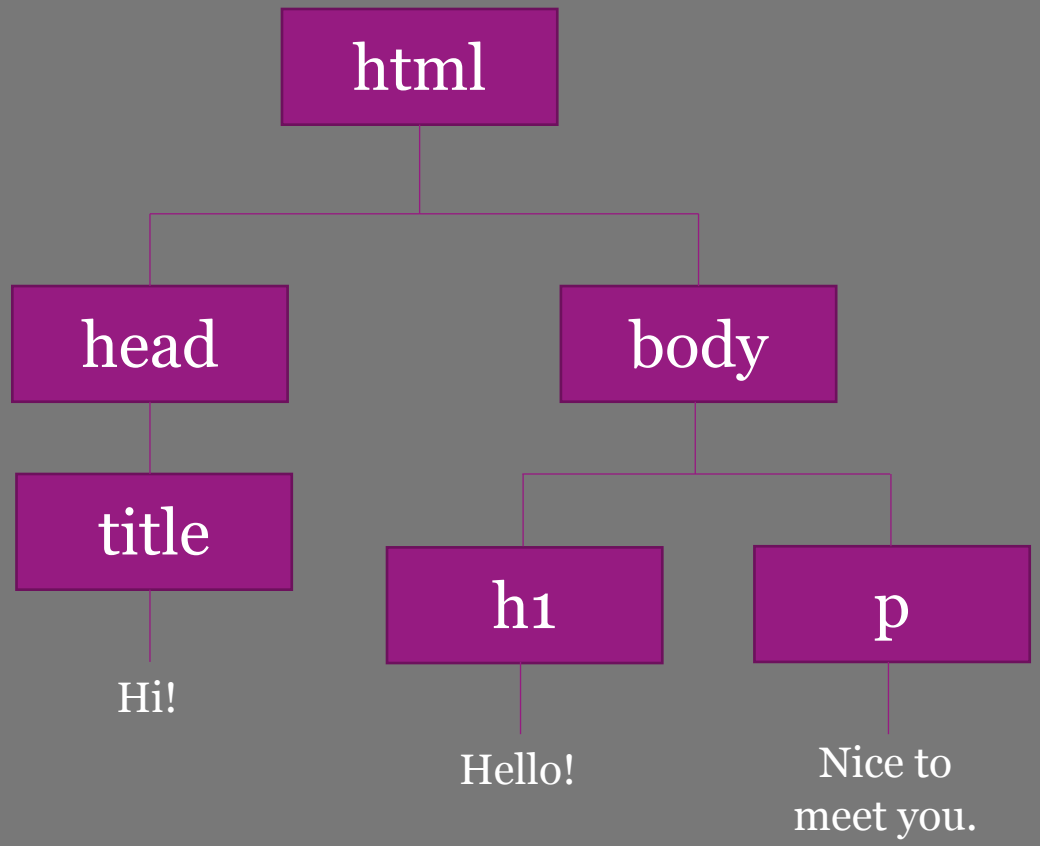
School of Engineering

DOCUMENT OBJECT MODEL

Peter Larsson-Green
Jönköping University
Spring 2019

THE DOCUMENT OBJECT MODEL

```
<!DOCTYPE html>
<html>
  <head>
    <title>Hi!</title>
  </head>
  <body>
    <h1>Hello!</h1>
    <p>Nice to meet you.</p>
  </body>
</html>
```



THE DOCUMENT OBJECT MODEL

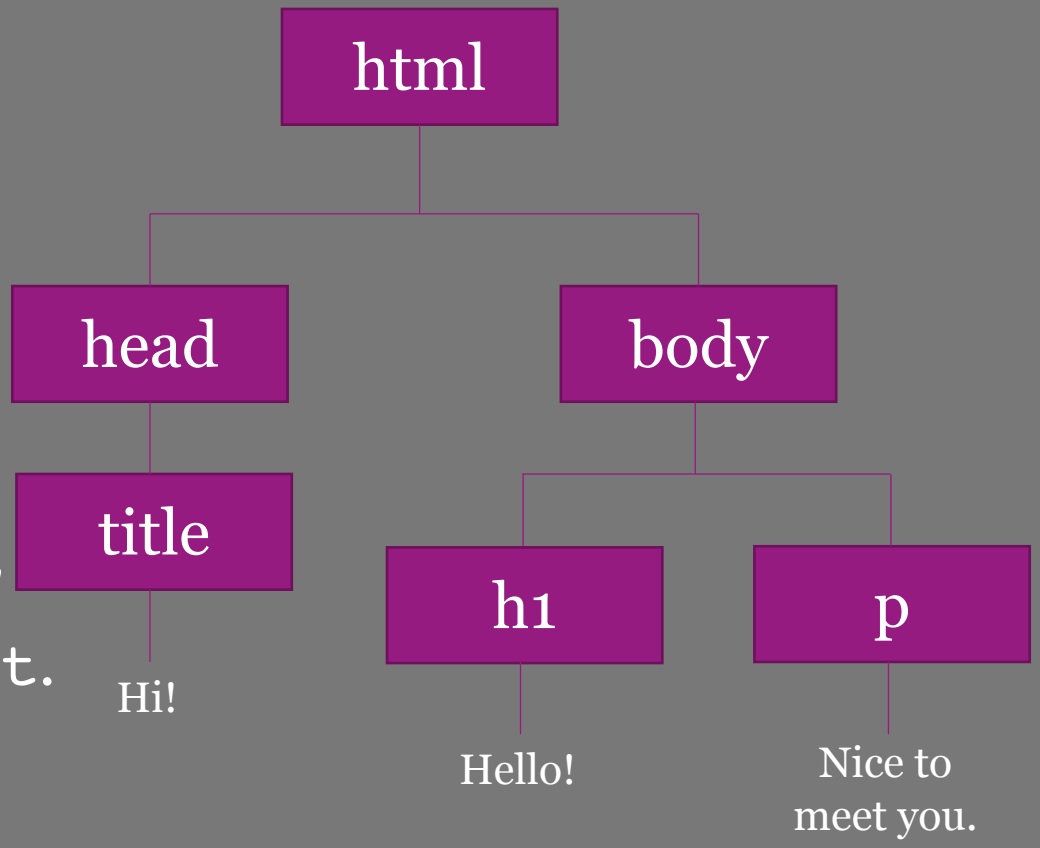
```
<!DOCTYPE html>
<html>
  <head>
    <title>Hi!</title>
  </head>
  <body>
    <h1>Hello!</h1>
    <p>Nice to meet you.</p>
  </body>
</html>
```

```
html
 |
 +-head
 |   |
 |   +-title - Hi!
 |
 +-body
     |
     +-h1 - Hello!
     |
     +-p - Nice to meet you.
```

THE DOCUMENT OBJECT MODEL

Web browsers give us access to the tree through the `document` variable.

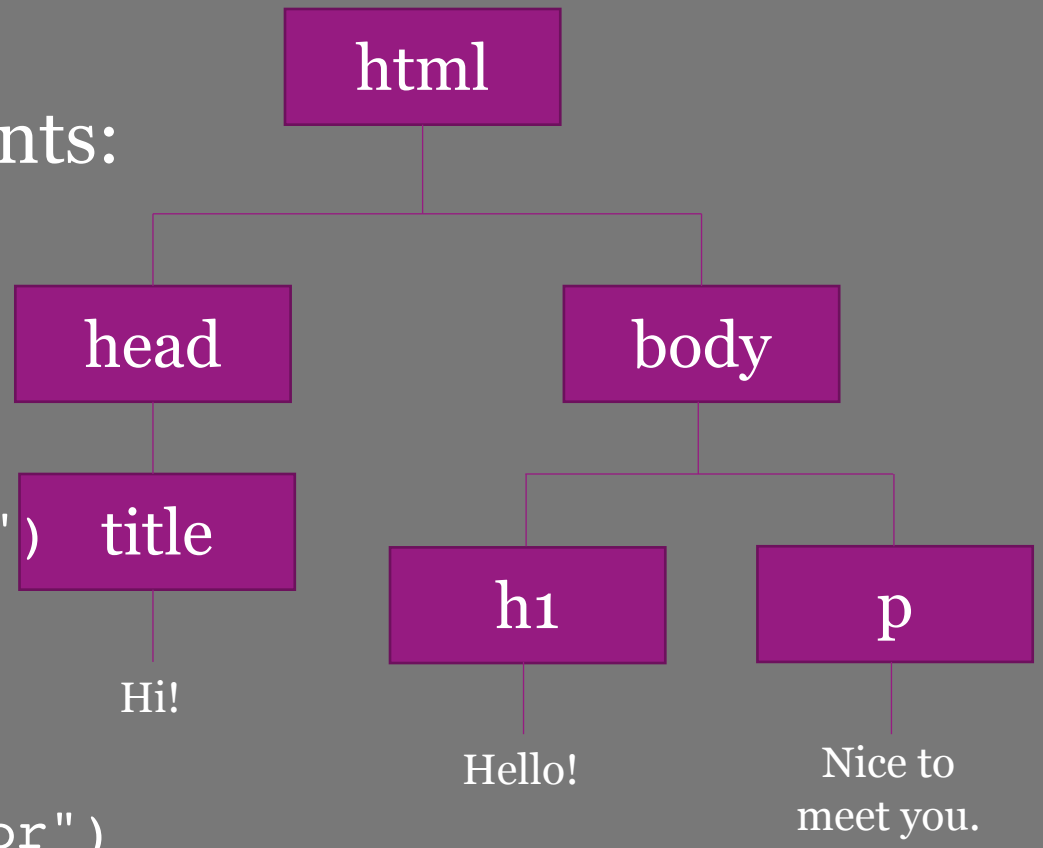
- `document.head` refers to the object representing the `<head>` element.
- `document.body` refers to the object representing the `<body>` element.
- `anHTMLElement.children` is an "array" containing the children of `anHTMLElement`.
- `anHTMLElement.parentNode` refers to the object representing the parent element of `anHTMLElement`.



THE DOCUMENT OBJECT MODEL

Different ways to obtain references to objects representing the HTML elements:

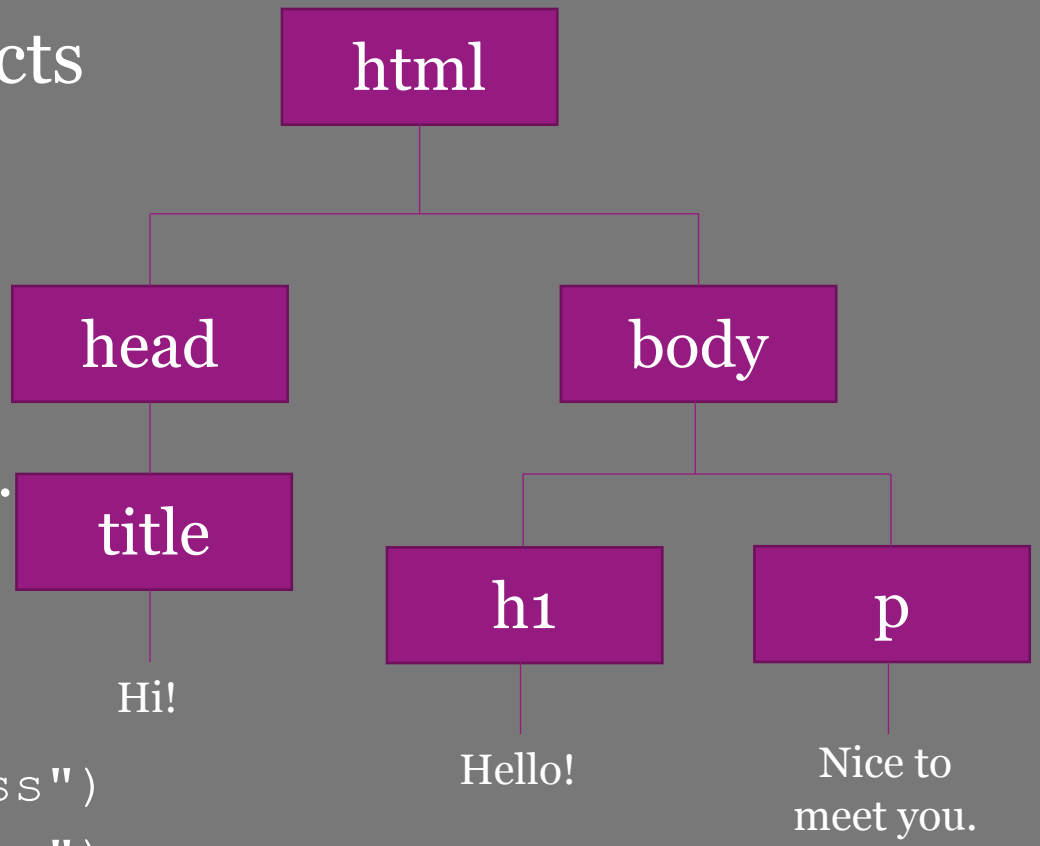
- `document.getElementById("theId")`
  - `id="theId"`
- `document.getElementsByTagName("name")`
  - `<name ...>...</name>`
- `document.getElementsByClassName("name")`
  - `class="name"`
- `document.querySelector("aCSSSelctor")`
  - First one matching `aCSSSelector`.
- `document.querySelectorAll("aCSSSelector")`
  - All matching `aCSSSelector`.



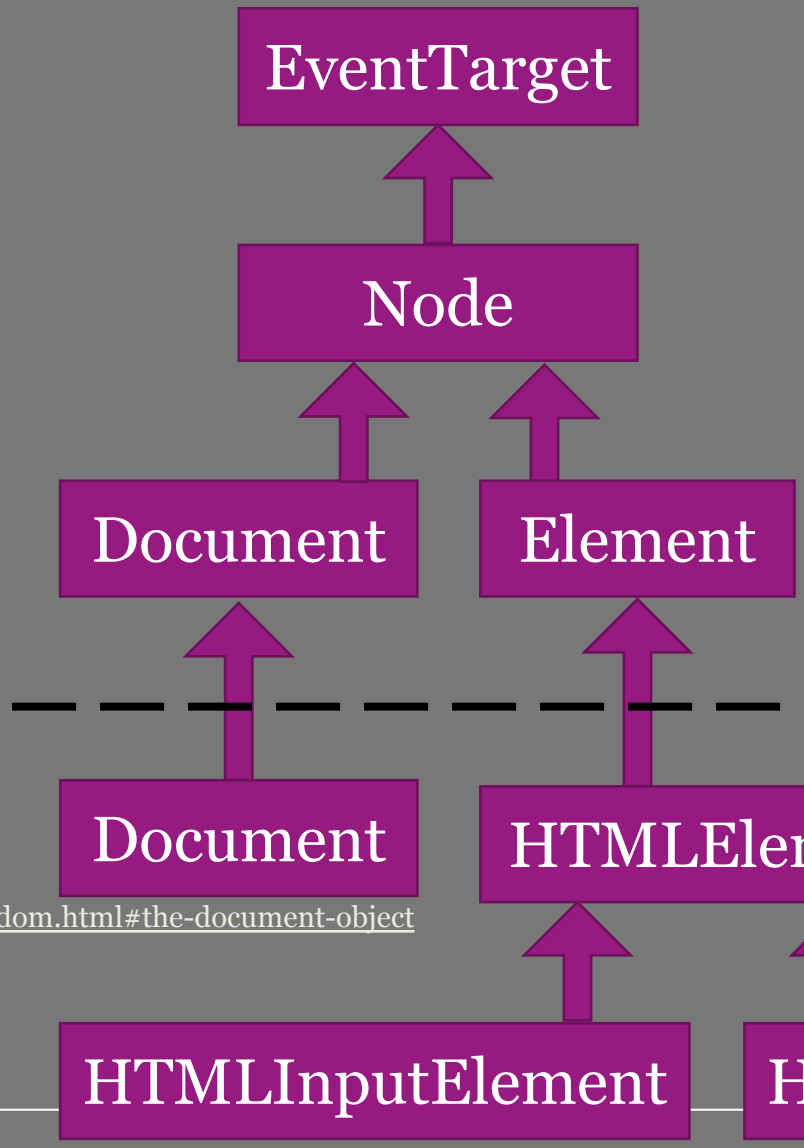
THE DOCUMENT OBJECT MODEL

Some common properties for the objects representing elements:

- `theObject.innerHTML`
  (what's inside the element).
- `theObject.innerText`
  (what's inside the element - HTML code).
- `theObject.classList`
  (the classes of the element).
  - `theObject.classList.add("newClass")`
  - `theObject.classList.remove("oldClass")`
  - `theObject.classList.toggle("theClass")`



THE INTERFACES



DOM Specification
https://dom.spec.whatwg.org

HTML Specification
https://html.spec.whatwg.org

https://html.spec.whatwg.org/multipage/dom.html#the-document-object

https://html.spec.whatwg.org/multipage/tables.html#the-table-element

EXAMPLE

```
<!DOCTYPE html>
<html>
  <head>
    <title>Hi!</title>
  </head>
  <body>
    <p id="p">Boring text.</p>
  </body>
</html>
```

```
const p = document.getElementById("p")
p.innerText = "Fun text!"
```

EXAMPLE

```
<!DOCTYPE html>
<html>
  <head>
    <title>Hi!</title>
    <script>

    </script>
  </head>
  <body>
    <p id="p">Boring text.</p>
  </body>
</html>
```

Executed before the browser have read the code in `<body>`!

```
const p = document.getElementById("p")
p.innerText = "Fun text!"
```

EXAMPLE

```
<!DOCTYPE html>
<html>
  <head>
    <title>Hi!</title>
    <script>


    </script>
  </head>
  <body>
    <p id="p">Boring text.</p>
  </body>
</html>
```

```
function changeText(){
  const p = document.getElementById("p")
  p.innerText = "Fun text!"
}
document.addEventListener(
  "DOMContentLoaded",
  changeText
)
```

EXAMPLE

```
<!DOCTYPE html>
<html>
  <head>
    <title>Hi!</title>
    <script src="file.js"></script>
  </head>
  <body>
    <p id="p">Boring text.</p>
  </body>
</html>
```

index.html

```
function changeText(){
  const p = document.getElementById("p")
  p.innerText = "Fun text!"
}
document.addEventListener(
  "DOMContentLoaded",
  changeText
)
```

file.js

EXAMPLE

```
<!DOCTYPE html>
<html>
  <head>
    <title>Hi!</title>
    <script src="file.js"></scr
  </head>
  <body>
    <button>0</button>
  </body>
</html>
```


```
document.addEventListener(
  "DOMContentLoaded",
  function(){
    const b = document.body.firstElementChild
    b.addEventListener(
      "click",
      function(){
        const oldCount = parseInt(b.innerHTML)
        const newCount = oldCount + 1
        b.innerHTML = newCount
      }
    )
  }
)
```

MORE ABOUT EVENTS

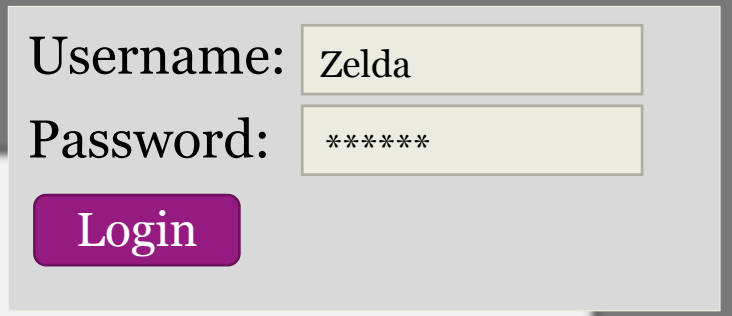
There exists many of them:

- https://developer.mozilla.org/en-US/docs/Web/Events

Not all elements support all events, but some common:

- `DOMContentLoaded` (for the `document` object).
- `keydown`, `keypress`, `keyup` (for elements that can have focus).
- `click`, `mousemove`, `mouseenter` (for elements that are shown).
- `reset`, `submit` (for `<form>`).

EXAMPLE




```
<form id="form">
  Username: <input id="username" type="text"><br>
  Password: <input id="password" type="password"><br>
  <input type="submit" value="Login">
</form>
```

```
document.addEventListener("DOMContentLoaded", function(){
  const form = document.getElementById('form')
  form.addEventListener("submit", function(e){
    e.preventDefault()
    const username = document.getElementById('username').value
    const password = document.getElementById('password').value
  })
})
```

TYPICAL STRUCTURE

```
document.addEventListener("DOMContentLoaded", function(){
  // 1. Change what the page looks like in the beginning.
  // 2. Add listeners for different events on the elements.
})
```