

Document Object Model

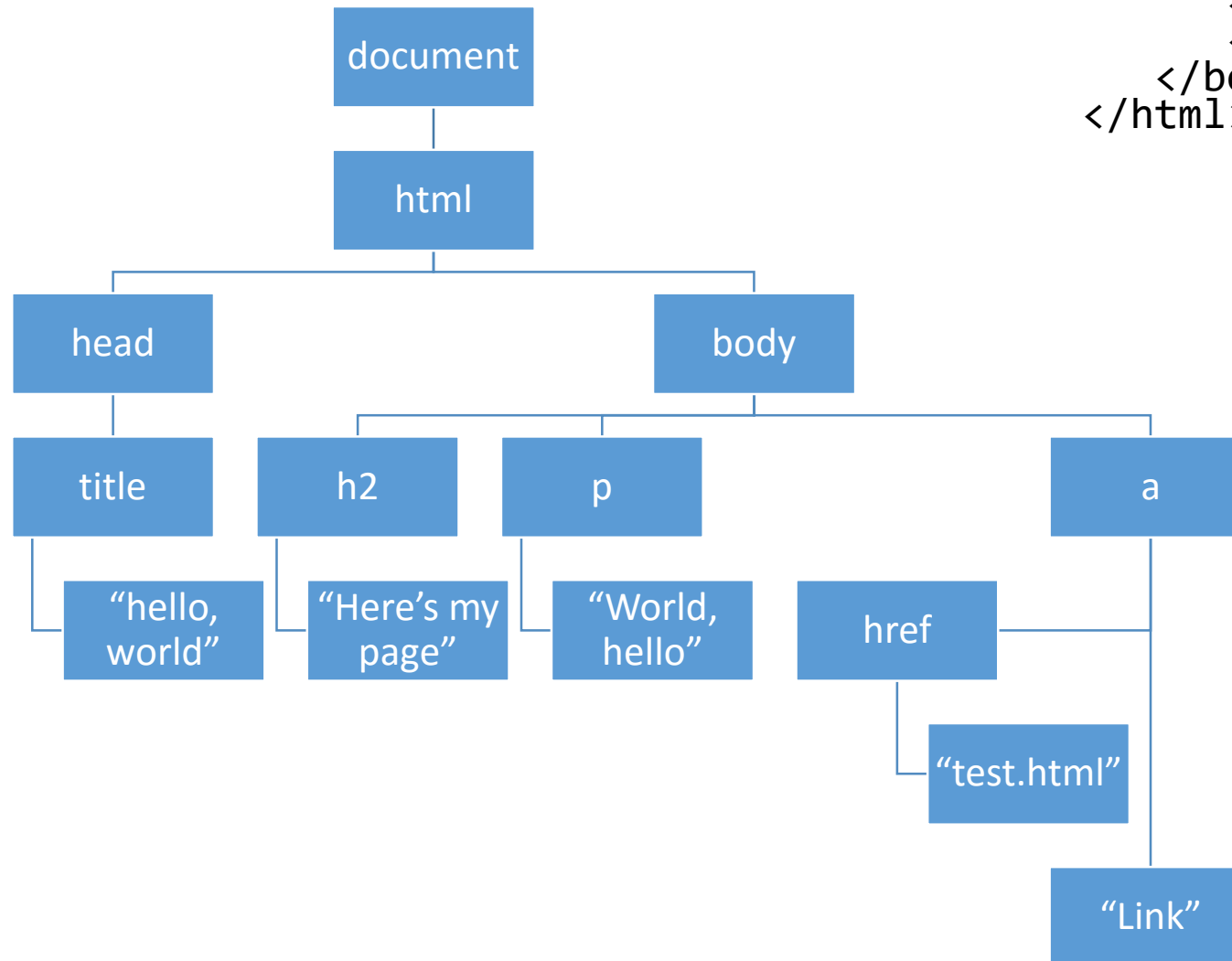
Document Object Model

- As we've seen, JavaScript objects are incredibly flexible, and can contain various fields, even when those fields are other objects.
- The *document object* is one way of employing this paradigm, whereby that object organizes the entire contents of a web page.
- By organizing an entire page into a JavaScript object, we can manipulate the page's elements programmatically.

Document Object Model

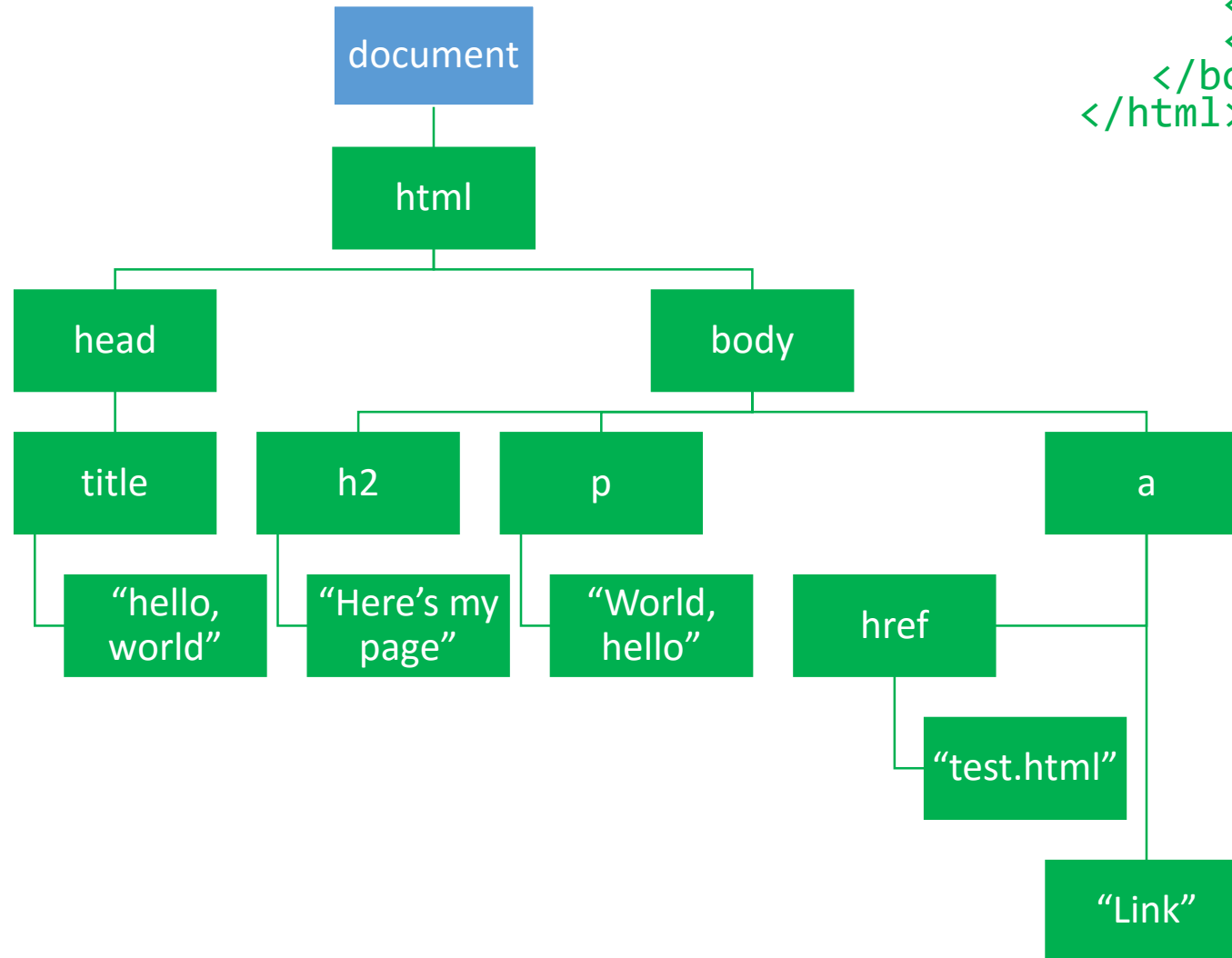
```
<html>
  <head>
    <title>Hello, world</title>
  </head>
  <body>
    <h2>Here's my page</h2>
    <p>World, hello</p>
    <a href="test.html">Link</a>
  </body>
</html>
```

Document Object Model



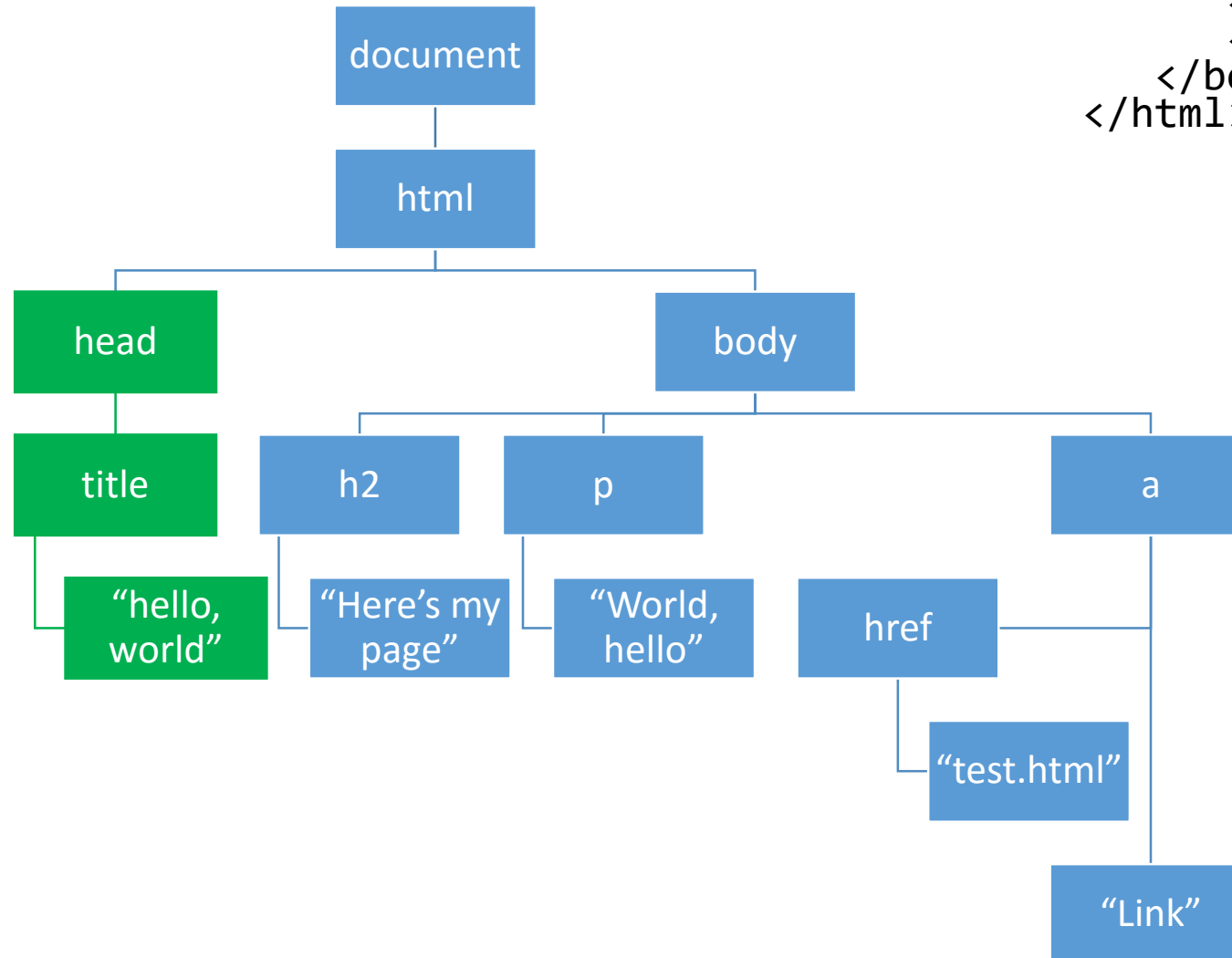
```
<html>
  <head>
    <title>Hello, world</title>
  </head>
  <body>
    <h2>Here's my page</h2>
    <p>World, hello</p>
    <a href="test.html">Link</a>
  </body>
</html>
```

Document Object Model



```
<html>
  <head>
    <title>Hello, world</title>
  </head>
  <body>
    <h2>Here's my page</h2>
    <p>World, hello</p>
    <a href="test.html">Link</a>
  </body>
</html>
```

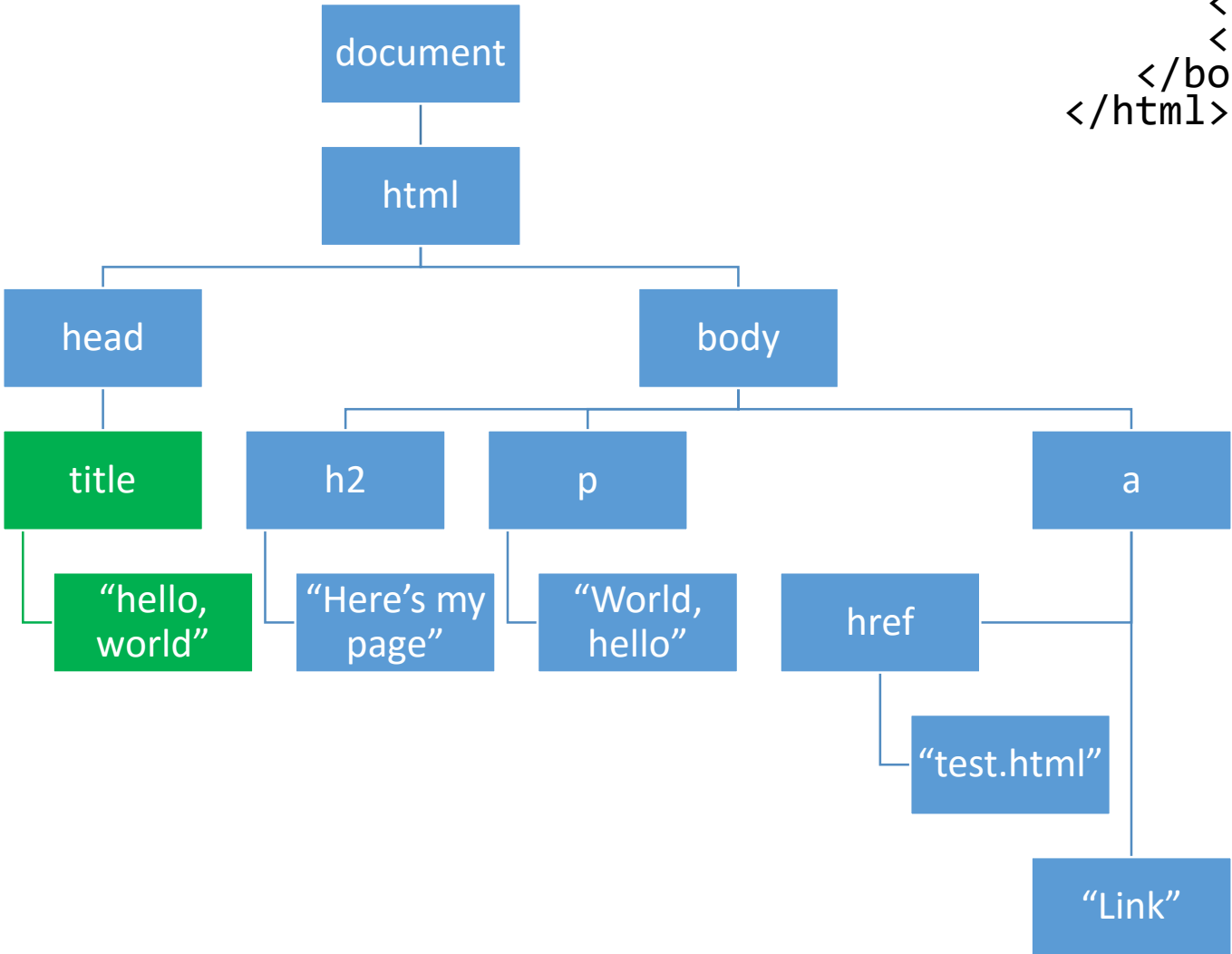
Document Object Model



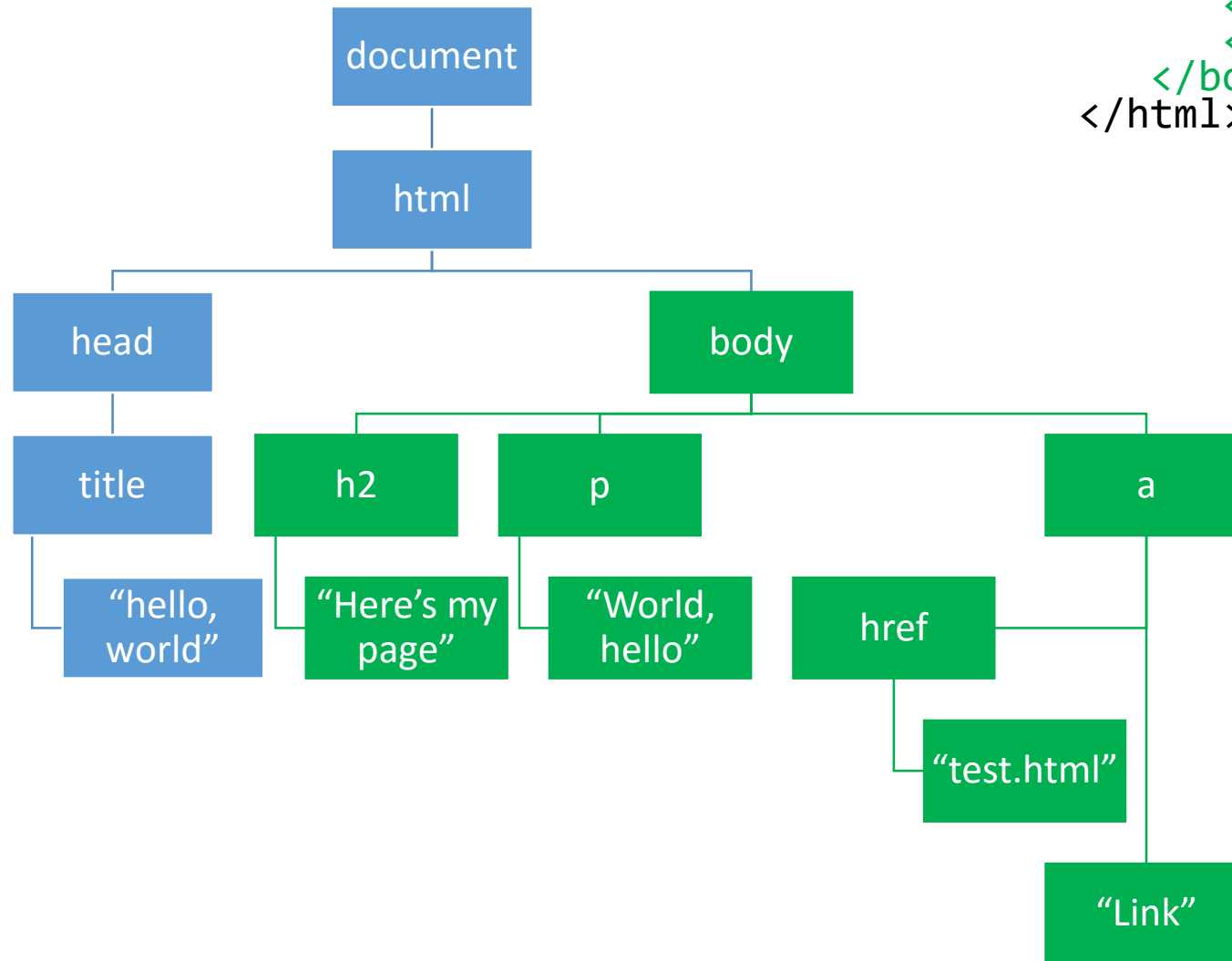
```
<html>
  <head>
    <title>Hello, world</title>
  </head>
  <body>
    <h2>Here's my page</h2>
    <p>World, hello</p>
    <a href="test.html">Link</a>
  </body>
</html>
```

Document Object Model

```
<html>  
  <head>  
    <title>Hello, world</title>  
  </head>  
  <body>  
    <h2>Here's my page</h2>  
    <p>World, hello</p>  
    <a href="test.html">Link</a>  
  </body>  
</html>
```

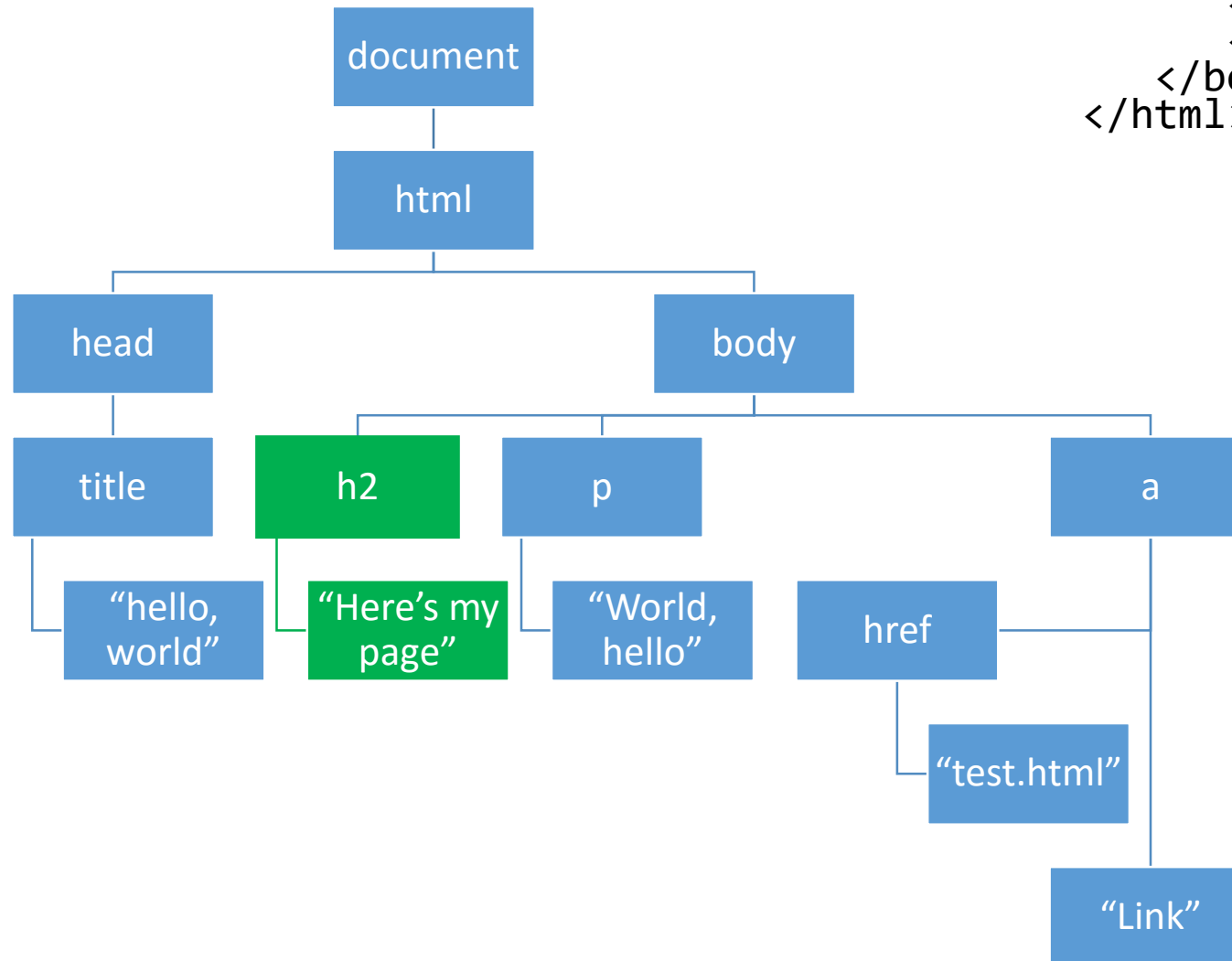


Document Object Model



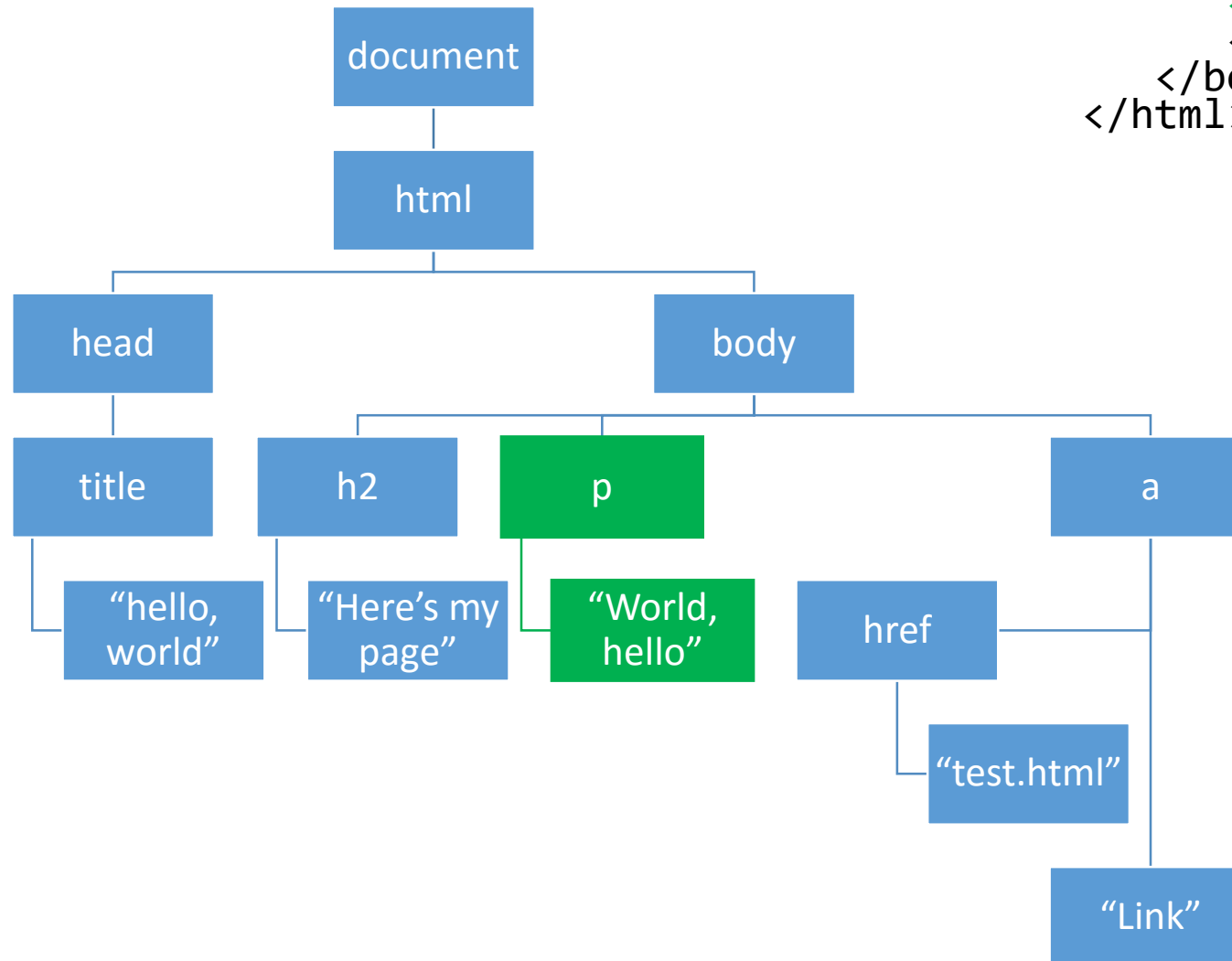
```
<html>
  <head>
    <title>Hello, world</title>
  </head>
  <body>
    <h2>Here's my page</h2>
    <p>World, hello</p>
    <a href="test.html">Link</a>
  </body>
</html>
```


Document Object Model



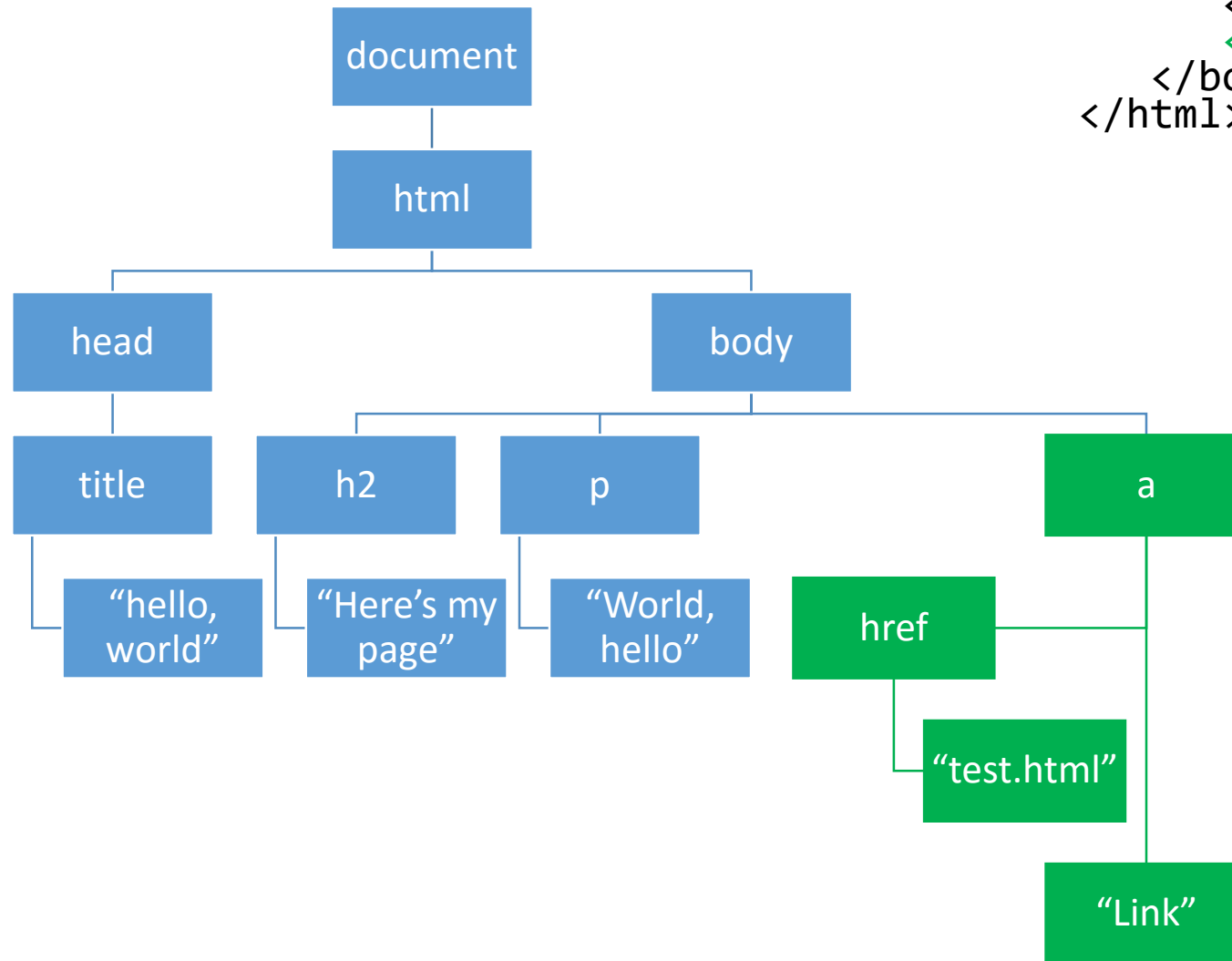
```
<html>
  <head>
    <title>Hello, world</title>
  </head>
  <body>
    <h2>Here's my page</h2>
    <p>World, hello</p>
    <a href="test.html">Link</a>
  </body>
</html>
```

Document Object Model



```
<html>
  <head>
    <title>Hello, world</title>
  </head>
  <body>
    <h2>Here's my page</h2>
    <p>World, hello</p>
    <a href="test.html">Link</a>
  </body>
</html>
```

Document Object Model



```
<html>
  <head>
    <title>Hello, world</title>
  </head>
  <body>
    <h2>Here's my page</h2>
    <p>World, hello</p>
    <a href="test.html">Link</a>
  </body>
</html>
```

Document Object Model

- The document object itself, as well as all of the objects contained within it, have a number of *properties* and a number of *methods* that can be used to drill down to a very specific piece of your website.
- By resetting those properties or calling certain methods, the contents of our web pages can change without us needing to refresh the page.

Document Object Model

- **DOM Properties**

DOM Property	Description
innerHTML	Holds the HTML inside a set of HTML tags.
nodeName	The name of an HTML element or element's attribute.
id	The "id" attribute of an HTML element.

Document Object Model

- **DOM Properties**

DOM Property	Description
innerHTML	Holds the HTML inside a set of HTML tags.
nodeName	The name of an HTML element or element's attribute.
id	The "id" attribute of an HTML element.
parentNode	A reference to the node one level up in the DOM.

Document Object Model

- **DOM Properties**

DOM Property	Description
<code>innerHTML</code>	Holds the HTML inside a set of HTML tags.
<code>nodeName</code>	The name of an HTML element or element's attribute.
<code>id</code>	The "id" attribute of an HTML element.
<code>parentNode</code>	A reference to the node one level up in the DOM.
<code>childNodes</code>	An array of references to the nodes one level down in the DOM.

Document Object Model

- **DOM Properties**

DOM Property	Description
<code>innerHTML</code>	Holds the HTML inside a set of HTML tags.
<code>nodeName</code>	The name of an HTML element or element's attribute.
<code>id</code>	The "id" attribute of an HTML element.
<code>parentNode</code>	A reference to the node one level up in the DOM.
<code>childNodes</code>	An array of references to the nodes one level down in the DOM.
<code>attributes</code>	An array of attributes of an HTML element.

Document Object Model

- **DOM Properties**

DOM Property	Description
<code>innerHTML</code>	Holds the HTML inside a set of HTML tags.
<code>nodeName</code>	The name of an HTML element or element's attribute.
<code>id</code>	The "id" attribute of an HTML element.
<code>parentNode</code>	A reference to the node one level up in the DOM.
<code>childNodes</code>	An array of references to the nodes one level down in the DOM.
<code>attributes</code>	An array of attributes of an HTML element.
<code>style</code>	An object encapsulating the CSS/HTML styling of an element.

Document Object Model

- **DOM Methods**

DOM Method	Description

Document Object Model

- **DOM Methods**

DOM Method	Description
<code>getElementById(id)</code>	Gets the element with a given ID below this point in the DOM.

Document Object Model

- **DOM Methods**

DOM Method	Description
<code>getElementById(id)</code>	Gets the element with a given ID below this point in the DOM.
<code>getElementsByTagName(tag)</code>	Gets all elements with the given tag below this point in the DOM.

Document Object Model

- **DOM Methods**

DOM Method	Description
<code>getElementById(id)</code>	Gets the element with a given ID below this point in the DOM.
<code>getElementsByTagName(tag)</code>	Gets all elements with the given tag below this point in the DOM.
<code>appendChild(node)</code>	Add the given node to the DOM below this point.

Document Object Model

- **DOM Methods**

DOM Method	Description
<code>getElementById(id)</code>	Gets the element with a given ID below this point in the DOM.
<code>getElementsByTagName(tag)</code>	Gets all elements with the given tag below this point in the DOM.
<code>appendChild(node)</code>	Add the given node to the DOM below this point.
<code>removeChild(node)</code>	Remove the specified child node from the DOM.

Document Object Model

- **DOM Methods**

DOM Method	Description
<code>getElementById(id)</code>	Gets the element with a given ID below this point in the DOM.
<code>getElementsByTagName(tag)</code>	Gets all elements with the given tag below this point in the DOM.
<code>appendChild(node)</code>	Add the given node to the DOM below this point.
<code>removeChild(node)</code>	Remove the specified child node from the DOM.

Document Object Model

- If we start from document, we can get to any piece of our web page that we choose, through careful use of DOM properties and methods.

Document Object Model

- **jQuery**
- Because DOM manipulation is so common with JavaScript, and because the JavaScript to do so can get quite lengthy, people wanted alternatives.
- jQuery is a popular open-source library, released in 2006, that is designed to simplify client-side scripting (such as DOM manipulations).

Document Object Model

- **jQuery**

```
document.getElementById( 'colorDiv' ).style.backgroundColor = 'green'
```

Document Object Model

- **jQuery**

```
document.getElementById('colorDiv').style.backgroundColor = 'green'
```

Document Object Model

- **jQuery**

```
$( '#colorDiv' ).css( 'background-color', 'green' );
```

Document Object Model

- **jQuery**

<https://api.jquery.com>