jQuery: Events and the DOM Tree

CISC 282

March 1-5, 2021

Event Handling

- jQuery offers a few shortcuts for event handling
  - The following are member functions for jQuery objects

- **on(*eventName, function*)**
  - Registers *function* to handle *eventName* on matching elements
  - Alternative to **addEventListener(…)** in JavaScript

- *eventName*(*function*)
  - Further shortcut for **on(*eventName, function*)**
  - Available for most events
    - *e.g.*, click

The **`ready`** Event
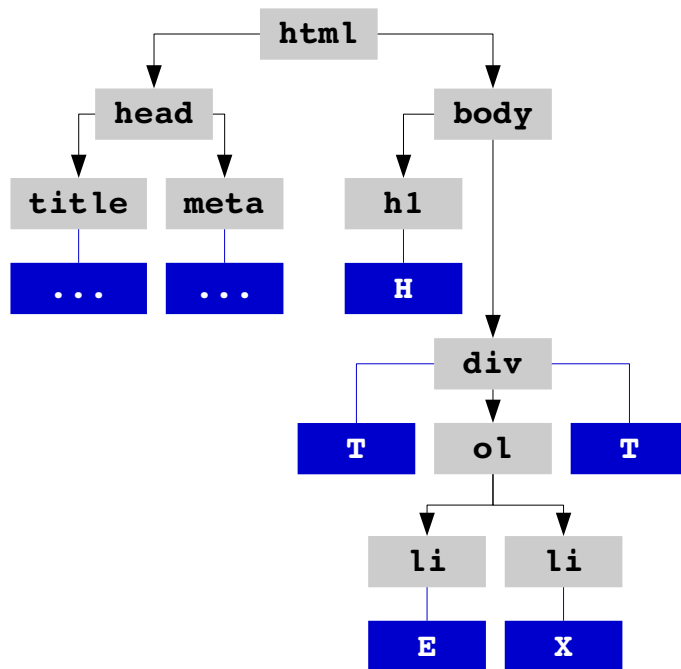
$(document).ready(*function*)

- Parallel to **`DOMContentLoaded`** event for **`document`** in JS

- Alternative to **`$(window).load(function)`**

- Called once all HTML elements have been parsed
  - Does **not** include load time for images and CSS

The DOM Tree

- A tree representation of the HTML document
  - Root is the **`html`** tag
  - All internal nodes represent HTML elements and their properties

- Can manipulate and traverse the tree in JavaScript and jQuery
  - Doing so in JS is trickier
    - Each element often has more than one tree node for itself, text, *etc.*
    - Not a one-to-one relationship between elements and nodes
  - Working with the DOM in jQuery is simpler
    - Simulates a one-to-one element-node relationship

The DOM Tree

```
<html>
    <head>
        <title>...</title>
        <meta>...</meta>
    </head>

    <body>
        <h1>H</h1>
        <div>T
            <ol>
                <li>E</li>
                <li>X</li>
            </ol>
            T
        </div>
    </body>
</html>
```

DOM Manipulation With jQuery

Method	Description
prepend(...), **append(...)**	Adds parameter* to the beginning/end of an element
before(...), **after(...)**	Adds parameter* before/after an element
remove()	Removes all elements on which this is called
remove(*selector*)	Removes all elements on which this is called with the additional *selector* applied

- ***** the parameter can be many things
  - A String with HTML, a jQuery Object, a JavaScript **Element**, *etc.*

DOM Traversal With jQuery

Method	Returns
parent(*selector*)	The parents of all the elements represented by the object (the *selector* filter is optional)
siblings(*selector*)	The siblings of all the elements represented by the object (the *selector* filter is optional)
children(*selector*)	The children of all the elements represented by the object (the *selector* filter is optional)
find(*selector*)	The descendants of all the elements represented by the object, filtered by *selector*