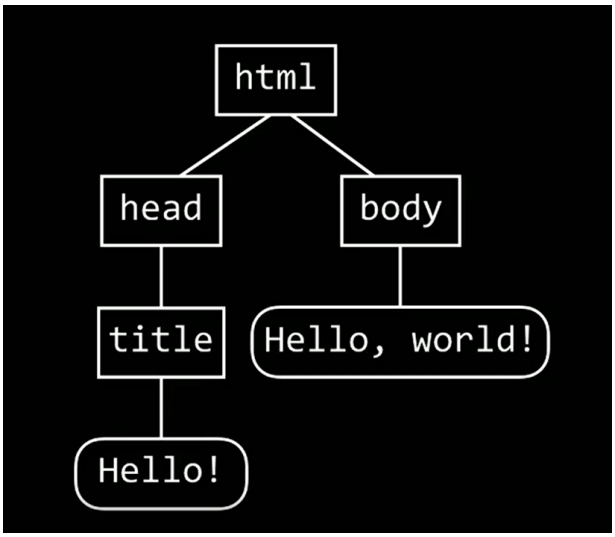
IT 105 – Principles of Programming

Day 11

1. Just a reminder, when you save your pages make sure they are text, but that they have the `.html` extension and not `.txt`. As you modify your pages to add functionality, be sure to save separate copies.
2. Previously, our webpages have been ***static***. They haven't changed unless their associated source `.html` file is saved and the page is reloaded in a browser. Now, we will start to make our pages ***dynamic***, where they respond to user interaction.
3. Some examples of user interaction on a webpage are:
    a. Clicking.
    b. Mouse-over.
    c. Resizing the window.
    d. Entering text.
4. We will use ***JavaScript*** (JS) to respond to user interaction. By the way, it's NOT Java, but the syntax is similar. The difference is we won't do much/any Object-Oriented-Programming (OOP).
5. In JavaScript, we will end up using each of the four basic types of control structures in most programming languages:
    a. Assignment statements (example: = sign).
    b. Selection / Conditional statements (example: if).
    c. Iteration statements (example: for or while).
    d. Function or method calls (example: alert).
6. First, consider the following simple HTML code (http://jcsites.juniata.edu/faculty/kruse/it105/inClass/IntroDOM.html):

```
<!DOCTYPE html>
<html lang="en">
   <head>
      <title>Hello!</title>
   </head>
   <body>
      Hello, world!
   </body>
</html>
```

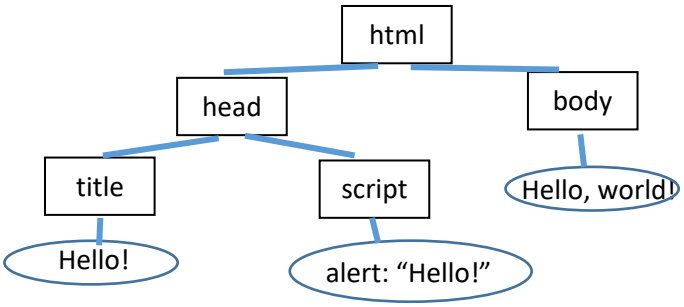
7. Let's now view this code as a tree. The `<head>` and `<body>` elements are nested within the `<html>` element, `<title>` is within `<head>`, and so on:



8. This model of representing HTML elements is called the ***Document Object Model*** (DOM). What makes JavaScript so powerful is that it can modify elements of the DOM. This means our web pages can now be dynamic, and change based on user input.
9. Let's update our simple HTML code with some Javascript (http://jcsites.juniata.edu/faculty/kruse/it105/inClass/js01.html):

```
<!DOCTYPE html>
<html lang="en">
   <head>
      <title>Hello!</title>
      <script>
         alert("Hello!");
      </script>
   </head>
   <body>
      Hello, world!
   </body>
</html>
```

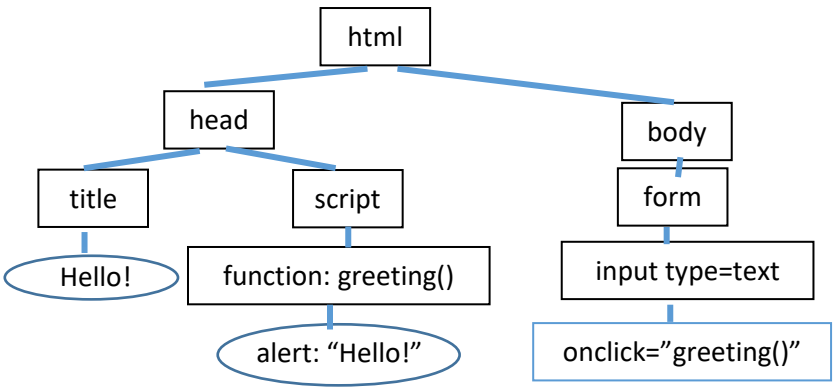
10. It doesn't do much, just print out "Hello" as soon as the page is rendered.
11. Here is the DOM for this page:



12. Let's update, adding a `form` element to our simple HTML code as well as some Javascript: (http://jcsites.juniata.edu/faculty/kruse/it105/inClass/js02.html):

```
<!DOCTYPE html>
<html lang="en">
   <head>
      <title>Hello!</title>
      <script>
         function greeting()
         {
           alert("hey")
         }
      </script>
   </head>
   <body>
      <form name=inclass>
        <input type=button value="Press This"
                              onclick="greeting();">
      </form>
   </body>
</html>
```

13. Here is the DOM for this page:



14. Here are some tips on coding webpages with Javascript:
    a. First, focus on rendering the page and any form elements.
    b. Next, add the first Javascript function and get it functioning properly.
    c. Then, iterate thru, adding one more Javascript function at a time.
    d. Generally it is much easier to debug smaller segments of code.