
Tables and Forms.

Objectives:

By the end of this chapter, you should be able to:

- Build semantic HTML tables using HTML5 elements
- Build forms using various inputs and understand their specific attributes

Building Semantic HTML Tables

Although they are not as popular as they used to be (entire web pages used to be designed just using tables!), it's still quite valuable to know how to build tables. They are great for displaying certain kinds of data!

`<table>` - This tag encompasses the entire table and wraps all of your HTML code for a table

`<tr>` - The table row element defines a row of cells which can include `td` and/or `th` tags

`<th>` - The table header tag defines a cell as the "header" for group of cells. This is traditionally placed inside of a `thead` tag.

`<td>` - The table data tag defines a cell of a table that contains some data.

`<thead>` - The table head tag defines a set of rows which define the columns of a table

`<tbody>` - From [MDN](#):

The HTML Table Body Element (`<tbody>`) defines one or more `<tr>` element data-rows to be the body of its parent `<table>` element (as long as no `<tr>` elements are immediate children of that table element.) In conjunction with a preceding `<thead>` and/or `<tfoot>` element, `<tbody>` provides additional semantic information for devices such as printers and displays. Of the parent table's child elements, `<tbody>` represents the content which, when longer than a page, will most likely differ for each page printed; while the content of `<thead>` and `<tfoot>` will be the same or similar for each page printed. For displays, `<tbody>` will enable separate scrolling of the `<thead>`, `<tfoot>`, and `<caption>` elements of the same parent `<table>` element.

`<tfoot>` - The table foot tag defines a set of rows summarizing the columns of the table.

Here is an example of what a semantic table looks like in HTML:

```
<table>
  <thead>
    <tr>
```

```

        <th>heading</th>
        <th>heading</th>
        <th>heading</th>
    </tr>
</thead>
<tbody>
    <tr>
        <td>content</td>
        <td>content</td>
        <td>content</td>
    </tr>
    <tr>
        <td>content</td>
        <td>content</td>
        <td>content</td>
    </tr>
</tbody>
<tfoot>
    <tr>
        <td>content</td>
        <td>content</td>
        <td>content</td>
    </tr>
</tfoot>
</table>

```

Building HTML Forms

Forms are another essential part of HTML and will be used quite a bit when we get to server-side programming. We use forms to transmit data to other pages and servers.

From [MDN](#):

HTML Forms are one of the main points of interaction between a user and a web site or application. They allow users to send data to the web site. Most of the time that data is sent to the web server, but the web page can also intercept it to use it on its own.

An HTML Form is made of one or more widgets. Those widgets can be text fields (single line or multi line), select boxes, buttons, check boxes, or radio buttons. Most of the time, those widgets are paired with a label that describes their purpose.

To get started with a form, we need a `<form></form>` tag and an `action` attribute (where the form should go when submitted). We will use `#` as the value of this attribute to signify that the form should not go anywhere. Forms also have another essential attribute called `method` which specifies how the form should be submitted (but don't worry about this one, we'll talk much more about it later).

```
<form action="#"></form>
```

Like MDN says, forms need to have widgets (or inputs) which can be paired with a label to describe their purpose. What does that mean, exactly? Let's take a look at this example form:

```
<form action="#">
  <div>
    <label for="username">Username</label>
    <input type="text" id="username">
  </div>
  <div>
    <label for="password">Password</label>
    <input type="password" id="password">
  </div>
  <div>
    <label>Favorite Instructor</label>
    <label for="elie">Elie</label>
    <input type="radio" name="favorite_instructor" id="elie">
    <label for="matt">Matt</label>
    <input type="radio" name="favorite_instructor" id="matt">
    <label for="tim">Tim</label>
    <input type="radio" name="favorite_instructor" id="tim">
  </div>
  <div>
    <label>Favorite Programming Language(s)</label>
    <label for="ruby">Ruby</label>
    <input type="checkbox" name="favorite_language" value="Ruby"
id="ruby">
    <label for="javascript">JavaScript</label>
    <input type="checkbox" name="favorite_language" value="JavaScript"
id="javascript">
    <label for="python">Python</label>
    <input type="checkbox" name="favorite_language" value="Python"
id="python">
    <label for="go">Go</label>
    <input type="checkbox" name="favorite_language" value="Go" id="go">
  </div>
  <div>
    <label>Additional Comments</label>
    <textarea name="" id="" cols="30" rows="10"></textarea>
  </div>
  <div>
    <input type="submit" value="">
  </div>
</form>
```

You can see that some of these inputs have `name` attribute. While we will not be discussing this in great detail for now (we will come back to this later), this input is necessary if we are trying to collect form

values to be transmitted to a server/another page.