Introduction

In this lesson you are going to learn how to create tables using the `table` tag and other related tags. Open up the *"tables-begin.html"* file in the editor and in the browser to prepare for this lesson. You can see that there is already some content inside: it's a simple shopping cart.

Many years ago the `table` tag was the only option for making complex layouts for websites. This was before the web standards movement and the wide adoption of CSS. Nowadays the `table` element continues to be perfectly valid and useful but for its original purpose: to format tabular data.

This shopping cart is a good example of tabular data, so let's see how to you use HTML semantically to format the table.

Creating the table and its headings

You can see that the text is already formatted like a simple table in the document. However if we open this in the browser you will see that the content appears all in one line.

So let's start building a table by adding the `table` tag:

```
<table>
    Shopping cart

    Product     Price      Quantity       Subtotal
    Product 1   $10.00     2              $20.00
    Product 2   $5.00      1              $5.00
    Shipping and Handling                 $5.00
    Discount                              10%
                                          - $3.00
    Grand total                           $27.00
</table>
```

This tag will not do much by itself, so the next step is creating a table row with the tag `tr`:

```
<table>
    <tr></tr>
    [...]
</table>
```

In HTML, we create the rows and then fill in the columns. The first column in our table can be considered the heading for the cells below, so we can employ the `th` tag to create table headings, one for each column.

```
<table>
    <tr>
        <th>Product</th>
        <th>Price</th>
        <th>Quantity</th>
        <th>Subtotal</th>
    </tr>
</table>
```

Notice that I've put some simple CSS styles in the document so the table has a border to make each cell easier to see:

```
<style>
    table, th, td {
        border: 1px solid #666;
        border-collapse: collapse;
```

```
    }
</style>
```

We can make the mark up better by using a `thead` tag around the `tr` to show that this is the header of the table:

```
<table>
    <thead>
        <tr>
            <th>Product</th>
            <th>Price</th>
            <th>Quantity</th>
            <th>Subtotal</th>
        </tr>
    </thead>
</table>
```

Another thing we can do to improve this code is to use a scope attribute on the `th` tag:

```
<table>
    <thead>
        <tr>
            <th scope="col">Product</th>
            <th scope="col">Price</th>
            <th scope="col">Quantity</th>
            <th scope="col">Subtotal</th>
        </tr>
    </thead>
</table>
```

This means that these cells are headings of the columns below it.

For the body of the table, which comes next, we can include the `tbody` tag:

```
<table>
    <thead>
        <tr>
            <th scope="col">Product</th>
            <th scope="col">Price</th>
            <th scope="col">Quantity</th>
            <th scope="col">Subtotal</th>
        </tr>
    </thead>
    <tbody>
        [...]
    </tbody>
</table>
```

Using tr and td tags to create table cells

Next, we have two more rows each with four columns of content:

```
<tbody>
    <tr>
        <td>Product 1</td>
        <td>$10.00</td>
        <td>2</td>
        <td>$20.00</td>
    </tr>
    <tr>
        <td>Product 2</td>
        <td>$5.00</td>
        <td>1</td>
        <td>$5.00</td>
    </tr>
</tbody>
```

In these table rows, we are employing another tag, `td`, that is used for generic table cells.

Using the rowspan and colspan attributes

The next line is a bit different: the shipping and handling columns are occupying more than one column. If you just place the content in `tds` like this:

```
<tr>
        <td>Shipping and Handling</td>
        <td>$5.00</td>
</tr>
<tr>
    <td>Discount</td>
    <td>10%</td>
</tr>
```

and reload the page in the browser, you will see that the row is incomplete because it is lacking two columns. So we want the shipping and handling to span three columns, which means one actual column plus two additional ones. This problem can be solved by using the `colspan` attribute:

```
<tr>
    <td colspan="3">Shipping and Handling</td>
    <td>$5.00</td>
</tr>
<tr>
    <td colspan="3">Discount</td>
    <td>10%</td>
</tr>
```

Now those columns are spanning three columns like intended.

The next line is a bit different as well. Note that the discount column should occupy two rows as well as three columns like the column above it. Let's start with the basic markup:

```
<tr>
    <td colspan="3">Discount</td>
    <td>10%</td>
</tr>
<tr>
    <td>- $3.00</td>
</tr>
<tr>
    <td colspan="3">Grand total</td>
    <td>$27.00</td>
</tr>
```

This will work but suppose you wanted to remove this line and make the discount sales span two rows as well as three columns. We can do that with the `rowspan` attribute:

```
<tr>
    <td  rowspan="2" colspan="3">Discount</td>
    <td>10%</td>
</tr>
<tr>
    <td colspan="3"></td>
    <td>- $3.00</td>
</tr>
```

If you now reload the page you'll see that the table is broken. That is because we still have td on the next line occupying the space that should be reserved for the discount cell. So just remove it:

```
<tr>
    <td  rowspan="2" colspan="3">Discount</td>
    <td>10%</td>
</tr>
<tr>
    <td colspan="3">- $3.00</td>
</tr>
```

For the grand total, the process is the same as it was for shipping and handling:

```
<tr>
```

```
    <td  rowspan="2" colspan="3">Discount</td>
    <td>10%</td>
</tr>
<tr>
    <td colspan="3">- $3.00</td>
</tr>
<tr>
    <td colspan="3">Grand total</td>
    <td>$27.00</td>
</tr>
```

The caption tag

The table is almost complete. However we still have the "Shopping cart" title which we can be incorporated in the table. For that we can use a `caption` tag right after the opening `table` tag:

```
<table>
    <caption>Shopping cart</caption>
    <thead>
    […]
</table>
```

In the browser you can see that the table's complete and properly marked up.