

Custom Attributes

In this lesson, we will study about creating custom attributes in HTML.
Let's begin!

WE'LL COVER THE FOLLOWING



- Listing 5-12:Exercise-05-09/index.html

HTML5 adds many new elements and changes the definition of old ones to support semantic web development.

An important step toward this paradigm is that HTML5 allows using custom attributes. This is very important because you can use them to provide your own semantic to your own markup building blocks.

To understand the way HTML5 defines and utilizes custom attributes, take a look at Listing 5-12.

Listing 5-12:Exercise-05-09/index.html


```
<!DOCTYPE html>
<html>
<head>
  <title>Custom attributes</title>
</head>
<body>
  <h3>Product details</h3>
  <div id="computer"
    data-product-cpu="4th generation Intel Core i5"
    data-product-ssd="64/128/256/512 GB">
    <h4>Surface 2 Pro</h4>
    <p>CPU: <span id="s2ProCpu"></span></p>
    <p>SSD: <span id="s2ProSsd"></span></p>
    <button onclick="getDetails('computer',
      's2ProCpu', 's2ProSsd')">
      Get Details
    </button>
  </div>
  <script>
    function getDetails(prod, label1, label2) {
      var prodNode = document.getElementById(prod);
```

```
var l1 = document.getElementById(label1);
var l2 = document.getElementById(label2);
l1.innerText =

    prodNode.getAttribute('data-product-cpu');
l2.innerText = prodNode.dataset.productSsd;
}
</script>
</body>
</html>
```

Here, the `<div>` element with the `"computer"` identifier defines two custom attributes, `data-product-cpu` and `data-product-ssd`. When the user clicks the `Get Details` button, the `getDetails()` function reads the attribute values and displays them on the screen.

The first attribute is read with the `getAttribute()` function, while the second one uses the `dataset` property of the `<div>` node. This property-based style uses naming conventions to translate the `'data-product-ssd'` attribute name to `'productSsd'`.

 **NOTE:** Do not feel intimidated if it is not clear for you how `getDetails()` does its magic. In [Chapter 8](#) and [9](#) you will learn the basics of JavaScript and you'll understand the mechanisms behind this code.

In the *next lesson*, we will paint the canvas in HTML.

Stay tuned! :)