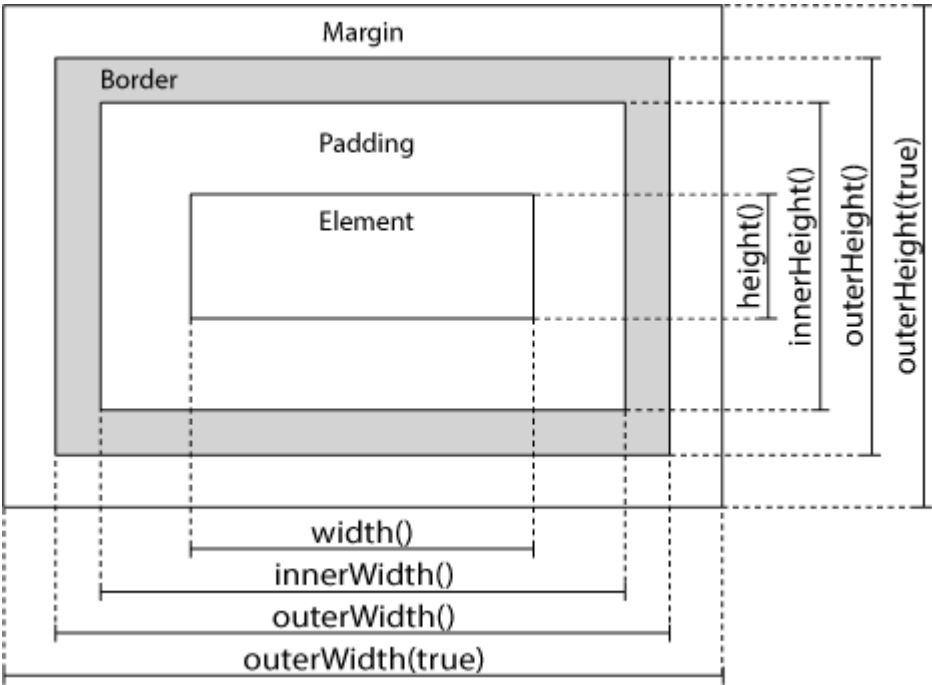


5. Working with Dimensions using jQuery and JavaScript

The Bootstrap framework's default 12-column grid uses percentage-based width dimensions to create Bootstrap's fluid grids. But in the end, when rendered in a browser, all page and element dimensions are calculated by the browser using pixel (px) dimension units. The jQuery library gives us access to the complete dimensions for each and every box in the CSS Box Model diagram of an HTML element – also measured in pixels.

In this example, we will interactively display the width and height of a page element when the user clicks on that element. Our choices for measuring both width and height include four different possible boxes, as shown in the diagram below.



jQuery Methods in Use

Here we look at the use of the jQuery library's built-in **width()**, **height()**, **outerWidth()** and **outerHeight()** methods.

<code>\$('#myitem').width();</code>	Returns the width in pixels of the content box of element with <i>id</i> ="myitem".
<code>\$('#myitem').height();</code>	Returns the height in pixels of the content box of element with <i>id</i> ="myitem".
<code>\$('#myitem').outerWidth();</code>	Returns the width in pixels of the visible element (without margins) for element with <i>id</i> ="myitem".
<code>\$('#myitem').outerWidth(true);</code>	Returns the width in pixels of the entire element <u>including margins</u> for element with <i>id</i> ="myitem".
<code>\$('#myitem').outerHeight();</code>	Returns the height in pixels of the visible element (without margins) for element with <i>id</i> ="myitem".
<code>\$('#myitem').outerHeight(true);</code>	Returns the width in pixels of the entire element <u>including margins</u> for element with <i>id</i> ="myitem".

Example





Our example allows the user to click on any div element to insert that element's width and height as content into the element itself.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>jQuery Example: Working with Dimensions</title>
  <!-- load bootstrap css via cdn -->
  <link rel="stylesheet" href="https://netdna.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css">
  <!-- load jquery js via cdn -->
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
  <!-- load bootstrap js via cdn -->
  <script src="https://netdna.bootstrapcdn.com/bootstrap/3.3.6/js/bootstrap.min.js"></script>
</head>
<body>
  <div class="container">
    <div class="row">
      <div class="col-xs-12">
        <h2>Display width and height in pixels</h2>
      </div>
    </div>
    <div class="row">
      <div class="col-xs-2">2</div>
      <div class="col-xs-4">4</div>
      <div class="col-xs-6">6</div>
    </div>
  </div>
  <!-- custom javascript using jquery to handle dynamic search filtering -->
  <script>
    // create event handler for user clicking on any div
    $("div").click(
      function() {
        // get width of element that was clicked using $(this)
        var w = $(this).width();
        // get height of element that was clicked using $(this)
        var h = $(this).height();
        // replace contents with dimensions
        $(this).html( w + "px wide " + h + "px high" );
        // ensure click doesn't replace document content with returned results
        return false;
      }
    );
  </script>
</body>
</html>
```

Experiment with the different jQuery methods for returning different box widths and heights.

Learn more about the jQuery context selector **\$(this)** – a special shorthand syntax for the **find()** method – in the jQuery API documentation for the **find()** method at <https://api.jquery.com/find/> [_https://api.jquery.com/find/](https://api.jquery.com/find/).