

Using JQuery

Until recently, sliders were not part of the HTML standard and were not natively supported by browsers. HTML 5 extends the range of input types to include a standard slider, and a number of software libraries include sliders that offer greater functionality. In this exercise you will explore the use of both the standard slider and a more sophisticated one from the JQuery JavaScript library.

First, create a web-page that includes a `range` input element, e.g.:

```
<input type='range'>
```

View your web-page in a browser. The element should appear as a slider, with a handle that can be dragged.

Add a text-box to your web-page. This will be used to display the current value of the slider. Give the text-box a suitable `id` so that you can access it from a script.

Add an `onchange` event-handler to the `input` element. This should call a function and pass `this.value` as a parameter.

Create a JavaScript function to receive this value and display it in the text-box. View your web-page again - you should find that dragging the slider changes the displayed value.

When this is working, try customising the slider. By default it will return values between 0 and 100, but you can change this by adding `min` and `max` properties. You can also determine where the handle is positioned by setting the `value` property.

Save your web-page, then create another page which uses the JQuery JavaScript library to create a slider. First, you will need to create a link to the online JQuery code and the style-sheet. Add the following lines of code to the head of your web page:

```
<script src="https://code.jquery.com/jquery-3.3.1.min.js"></script>
<script src="http://code.jquery.com/ui/1.12.1/jquery-ui.js"></script>
<link rel="stylesheet"
      href="http://code.jquery.com/ui/1.10.0/themes/base/jquery-ui.css">
```

Next, create a `<div>` element to act as the slider. There is no need to make it look like a slider - that will be done by JQuery. All that is necessary is to give it a suitable `id`, e.g., `id = 'slider'`.

Now initialise the `<div>` using the following code:

```
$( "#slider" ).slider({ value: [ 50 ] });
```

The code `$()` performs selection in JQuery: it will obtain a reference to the element whose `id` is `slider`. The element will then be set-up as a slider with an initial setting of 50.

Make sure this code runs each time the page loads, e.g., by placing it in a function that is called by an `onload` event.

View your web-page in a browser. The `<div>` should appear as a slider with a draggable handle. View the page in different browsers and see how the slider differs in appearance.

Add CSS to position the slider wherever you wish on the page, and to set its width, etc..

The next stage is to recover the setting of the slider each time it changes. Add a text-box to your page, then add the following code to your initialisation function:

```
$( "#slider" ).on(
  { slide:
    function() {
      var setting = $( "#slider" ).slider( "option", "value" );
      document.getElementById( 'textBox' ).value = setting;
    }
  }
);
```

where `textBox` is the `id` of your text-box. This code creates a function that is called whenever the slider is moved (`$("#slider").on({ slide:})`). The function obtains the new setting of the slider (its `value`) and displays it in the text-box. Note that the function doesn't have a name - it doesn't need one.

View your web-page in a browser. When the slider is moved, the new value should appear in the text-box.