

Form Fields: Text Zones

In this lesson, you will add JavaScript functions to handle the form data entered by the user.

WE'LL COVER THE FOLLOWING ^

- Access Input Values
- Handling Focus

Access Input Values

A text field allows a user to input text on single or multiple lines. You have two options for defining text fields: a single-line text field is defined in HTML as `<input type="text">`, and a multi-line text input field will be defined via `<textarea>` instead. Here's the extract from the above code that lets users input a username

Output

HTML

```
<html>
<head>
</head>
<body>



  <label for="username">Username</label>:
  <input type="text" name="username" id="username" required>
  <span id="usernameHelp"></span>

</body>
</html>
```



In JavaScript, you can access the `value` of a text field by using the value property of the corresponding DOM element. By setting a new value for this property, you'll modify the value shown in the text field. The following

example adds the value “MyCoolUsername” to the text field.

Output
JavaScript
HTML
<pre><html> <head> </head> <body> <label for="username">Username</label>: <input type="text" name="username" id="username" required> </body> </html></pre>
 

Handling Focus

When a text zone is selected, it becomes the focused area of the form. You may have noticed field borders turning blue or other effects when you’re accessing a particular input area. This helps you know where you are in the form. A user clicking on a text field (or tabbing down into it) kicks off a focus event. Additionally, a focus event triggers a blur event on the field that previously had the focus.

You can use these events to show the user tips related to the current text field, as in the following example:

Output
JavaScript
HTML
<pre><html> <head> </head> <body> <label for="username">Username</label>: <input type="text" name="username" id="username" required> </pre>

```
</body>
</html>
```



By selecting the `username` input field, you'll see a helpful message in the HTML `` defined specifically for that purpose and initially empty. From JavaScript code, you can modify the input target by calling the `focus()` (to give focus) and `blur()` methods (to remove it) on a DOM element.

Output

JavaScript

HTML

```
<html>
<head>
</head>
<body>

  <label for="username">Username</label>:
  <input type="text" name="username" id="username" required>
  <span id="usernameHelp"></span>

</body>
</html>
```



Multi-line text fields (`<textarea>` tags) work similarly to `<input>` tags.

You'll learn how to validate text that a user inputs (to make sure it fits certain criteria) in the later chapters.