Lets Code!

HTML Forms

Html Forms

Web forms are one of the main points of interaction between a user and a web site or application.

The need for forms

Forms allow users to enter data, which is generally sent to a web server for processing and storage, or used on the client-side to immediately update the interface in some way (for example, add another item to a list, or show or hide a UI feature).

Input types

Type	Description
`<input type="text">`	Displays a single-line text input field
`<input type="radio">`	Displays a radio button (for selecting one of many choices)
`<input type="checkbox">`	Displays a checkbox (for selecting zero or more of many choices)
`<input type="submit">`	Displays a submit button (for submitting the form)
`<input type="button">`	Displays a clickable button

Name Attribute

Notice that each input field must have a name attribute to be submitted.

If the name attribute is omitted, the value of the input field will not be sent at all.

Labels

The `<label>` tag defines a label for many form elements.

The `<label>` element is useful for screen-reader users, because the screen-reader will read out loud the label when the user focus on the input element.

The `<label>` element also help users who have difficulty clicking on very small regions (such as radio buttons or checkboxes) - because when the user clicks the text within the element, it toggles the radio button/checkbox.

Submit Button

The `<input type="submit">` defines a button for submitting the form data to a form-handler.

The form-handler is typically a file on the server with a script for processing input data.

The form-handler is specified in the form's action attribute.

Validation

Before submitting data to the server, it is important to ensure all required form controls are filled out, in the correct format. This is called client-side form validation, and helps ensure data submitted matches the requirements set forth in the various form controls.

Main reasons for validation:

- We want to get the right data, in the right format.
- We want to protect our users' data.
- We want to protect ourselves.

Javascript vs built-in

There are two different types of client-side validation that you'll encounter on the web:

Built-in form validation

Built-in form validation uses HTML5 form validation features, which we've discussed in many places throughout this module. This validation generally doesn't require much JavaScript. Built-in form validation has better performance than JavaScript, but it is not as customizable as JavaScript validation.

JavaScript validation

JavaScript validation is coded using JavaScript. This validation is completely customizable, but you need to create it all (or use a library).

JavaScript validation

- **required:** Specifies whether a form field needs to be filled in before the form can be submitted.
- **minlength and maxlength:** Specifies the minimum and maximum length of textual data (strings)
- **min and max:** Specifies the minimum and maximum values of numerical input types
- **type:** Specifies whether the data needs to be a number, an email address, or some other specific preset type.
- **pattern:** Specifies a regular expression that defines a pattern the entered data needs to follow.

JavaScript validation

At the end of the day, we want to validate client data all the way through the application process. From client to server, and possibly from server to data store