<!DOCTYPE html>

<html lang="en" dir="ltr">

  <head>

    <meta charset="utf-8">

    <link rel="stylesheet" type="text/css" href="style.css">

    <link href="https://fonts.googleapis.com/css?family=Rubik" rel="stylesheet">

    <title>Textarea element</title>

  </head>

  <body>

    <section id="overlay">

      <img src="https://content.codecademy.com/courses/web-101/unit-6/htmlcss1-img\_burger-logo.svg" alt="Davie's Burgers Logo" id="logo">

      <hr>

      <form action="submission.html" method="POST">

        <h1>Create a burger!</h1>

          <section class="protein">

          <label for="patty">What type of protein would you like?</label>

          <input type="text" name="patty" id="patty">

        </section>

        <hr>

        <section class="patties">

          <label for="amount">How many patties would you like?</label>

          <input type="number" name="amount" id="amount">

        </section>

        <hr>

        <section class="cooked">

          <label for="doneness">How do you want your patty cooked</label>

          <br>

          <span>Rare</span>

          <input type="range" name="doneness" id="doneness" value="3" min="1" max="5">

          <span>Well-Done</span>

        </section>

        <hr>

        <section class="toppings">

          <span>What toppings would you like?</span>

          <br>

          <input type="checkbox" name="topping" id="lettuce" value="lettuce">

          <label for="lettuce">Lettuce</label>

          <input type="checkbox" name="topping" id="tomato" value="tomato">

          <label for="tomato">Tomato</label>

          <input type="checkbox" name="topping" id="onion" value="onion">

          <label for="onion">Onion</label>

        </section>

        <hr>

        <section class="cheesy">

          <span>Would you like to add cheese?</span>

          <br>

          <input type="radio" name="cheese" id="yes" value="yes">

          <label for="yes">Yes</label>

          <input type="radio" name="cheese" id="no" value="yes">

          <label for="no">No</label>

        </section>

        <hr>

        <section class="bun-type">

          <label for="bun">What type of bun would you like?</label>

          <select name="bun" id="bun">

            <option value="sesame">Sesame</option>

            <option value="potatoe">Potato</option>

            <option value="pretzel">Pretzel</option>

          </select>

        </section>

        <hr>

        <section class="sauce-selection">

          <label for="sauce">What type of sauce would you like?</label>

          <input list="sauces" id="sauce" name="sauce">

          <datalist id="sauces">

            <option value="ketchup"></option>

            <option value="mayo"></option>

            <option value="mustard"></option>

          </datalist>

        </section>

        <hr>

        <section class="extra-info">

          <label for="extra">Anything else you want to add?</label>

          <br>

          <textarea id="extra" name="extra" rows="3" cols="40"></textarea>

        </section>

        <hr>

        <section class="submission">

          <!--Add your code below-->

          <input type="submit" value="Submit">

        </section>

      </form>

    </section>

  </body>

</html>

In this lesson we went over:

* The purpose of a <form> is to allow users to input information and send it.
* The <form>‘s action attribute determines where the form’s information goes.
* The <form>‘s method attribute determines how the information is sent and processed.
* To add fields for users to input information we use the <input> element and set the type attribute to a field of our choosing:
  + Setting type to "text" creates a single row field for text input.
  + Setting type to "password" creates a single row field that censors text input.
  + Setting type to "number" creates a single row field for number input.
  + Setting type to "range" creates a slider to select from a range of numbers.
  + Setting type to "checkbox" creates a single checkbox that can be paired with other checkboxes.
  + Setting type to "radio" creates a radio button that can be paired with other radio buttons.
  + Setting type to "text" and adding the list attribute will pair the <input> with a <datalist> element if the list of <input> and the id of <datalist> are the same.
  + Setting type to "submit" creates a submit button.
* A <select> element is populated with <option> elements and renders a dropdown list selection.
* A <datalist> element is populated with <option> elements and works with an <input> to search through choices.
* A <textarea> element is a text input field that has a customizable area.
* When a <form> is submitted, the name of the fields that accept input and the value of those fields are sent as name=value pairs.

Using the <form> element in conjunction with the other elements listed above allows us to create sites that take into consideration the wants and needs of our users. Take the opportunity to take what you’ve learned, and apply it!

<!DOCTYPE html>

<html lang="en" dir="ltr">

  <head>

    <meta charset="utf-8">

    <title>Sign Up Page</title>

    <link rel="stylesheet" href="style.css" type="text/css">

    <link href="https://fonts.googleapis.com/css?family=Fjalla+One" rel="stylesheet">

  </head>

  <body>

    <section class="overlay">

      <h1>Sign Up</h1>

      <p>Create an account:</p>

      <form action="submission.html" method="GET">

        <label for="username">Username:</label>

        <br>

        <input id="username" name="username" type="text" required minlength="3" maxlength="15">

        <br>

        <label for="pw">Password:</label>

        <br>

        <!--Add the pattern attribute to the input below-->

        <input id="pw" name="pw" type="password" required minlength="8" maxlength="15">

        <br>

        <input type="submit" value="Submit">

      </form>

    </section>

  </body>

</html>

* Client-side validations happen in the browser before information is sent to a server.
* Adding the required attribute to an input related element will validate that the input field has information in it.
* Assigning a value to the min attribute of a number input element will validate an acceptable minimum value.
* Assigning a value to the max attribute of a number input element will validate an acceptable maximum value.
* Assigning a value to the minlength attribute of a text input element will validate an acceptable minimum number of characters.
* Assigning a value to the maxlength attribute of a text input element will validate an acceptable maximum number of characters.
* Assigning a regex to pattern matches the input to the provided regex.
* If validations on a <form> do not pass, the user gets a message explaining why and the <form> cannot be submitted.

These quick checks help ensure that input data is correct and safe for our servers. It also helps give users immediate feedback on what they need to fix instead of having to wait for a server to send back that information.