



WEEK 8 / LESSON 15:

# FORMS & INPUTS

Leonie Dunnett

# LEARNING OBJECTIVES

- Differentiate the different types of inputs and why/where we would use each
- Explain how to group elements
- Perform pseudo-styling of input elements that the browser won't let us directly style

# TODAY'S SCHEDULE

- Forms overview
- Forms Lab

# FORMS OVERVIEW

Getting data from users

The screenshot shows a web form for General Assembly. The header is black with the General Assembly logo and a 'Sign In' link. The main heading is 'FRONT-END WEB DEVELOPMENT' followed by 'APPLY NOW' in large, bold letters. The form contains three text input fields: 'Your Full Name', 'Your Email', and 'Your Phone Number'. To the right of these fields is a grey box with the text: 'Fill out some basic information and complete the following application to be considered for the course.' Below the input fields is a label 'Where are you thinking of taking this course?' followed by a dropdown menu showing 'New York City'. At the bottom is a red button with the text 'CONTINUE TO APPLICATION'.

GA GENERAL ASSEMBLY Sign In

FRONT-END WEB DEVELOPMENT

**APPLY NOW**

Your Full Name

Your Email

Your Phone Number

Fill out some basic information and complete the following application to be considered for the course.

Where are you thinking of taking this course?

New York City

**CONTINUE TO APPLICATION**

# FORMS

Wrapper for data collection elements:

- Text fields
- Dropdowns
- Radio Buttons
- Checkboxes
- Buttons
- etc

## **A FORM TELLS THE PAGE:**

- Where to send the data
- How to send it
- What is being sent

Forms can be used purely on the client side, or can send data to the server where it is acted upon or stored in a database.

# FORM TAG

`<form> </form>`

Available Attributes:

- **method:** post, get
- **action:** url to send data to
- **enctype:** multipart/form-data if uploading files

```
<form action="register.php" method="post" enctype="multipart/form-data">  
  <!--Data collection elements go here-->  
</form>
```

# INPUTS

- Place between `<form>` `</form>` tags
- Attributes:
  - `type`: text, submit, password, email, checkbox, button, radio, file, etc
  - `value`
  - `placeholder`
  - `name` (used server side)


**Note:** font-family for an input is not inherited  
(can lead to sizing issue on Mac vs PC due to different default font)



# INPUTS: TEXT

Use value to set initial text

```
<input type="text">
```

 GENERAL ASSEMBLY Sign In

FRONT-END WEB DEVELOPMENT

**APPLY NOW**

Where are you thinking of taking this course?

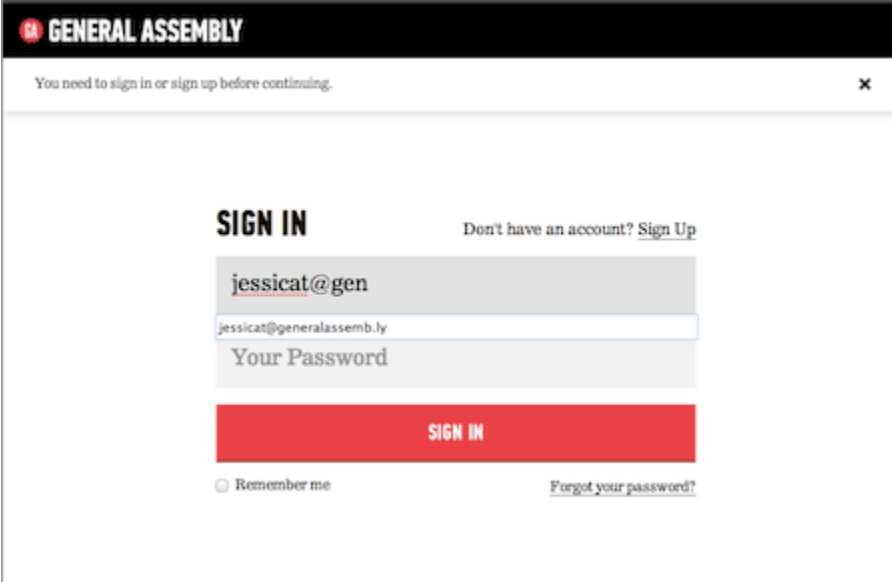
CONTINUE TO APPLICATION

Fill out some basic information and complete the following application to be considered for the course.

# INPUTS: EMAIL

Allows browser to autofill field

```
<input type="email">
```



The screenshot shows the General Assembly website's sign-in page. At the top, a black header contains the 'GENERAL ASSEMBLY' logo. Below it, a message states 'You need to sign in or sign up before continuing.' with a close button. The main content area is titled 'SIGN IN' and includes a link for users without an account. The email input field is highlighted, showing two autofill suggestions: 'jessicat@gen' and 'jessicat@generalassemb.ly'. Below the email field is a password field labeled 'Your Password'. A red 'SIGN IN' button is positioned below the password field. At the bottom, there is a 'Remember me' checkbox and a 'Forgot your password?' link.

**GENERAL ASSEMBLY**

You need to sign in or sign up before continuing. ✕

**SIGN IN** Don't have an account? [Sign Up](#)

jessicat@gen

jessicat@generalassemb.ly

Your Password

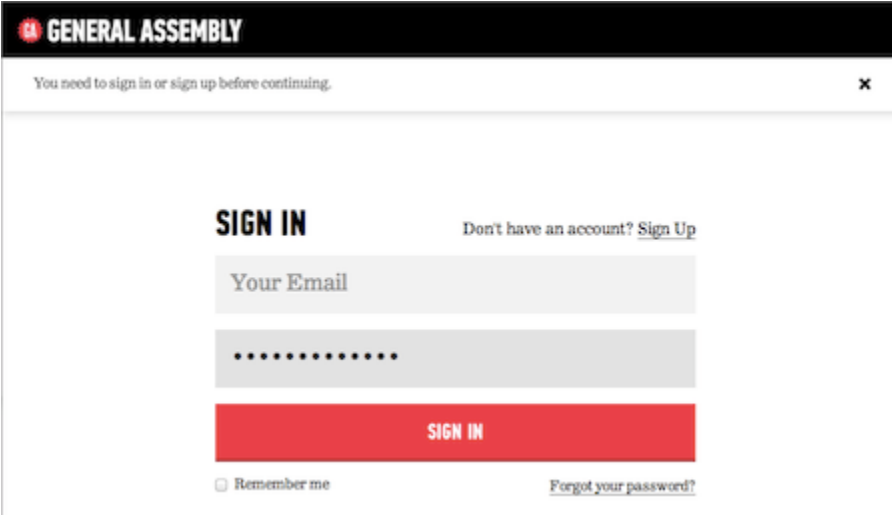
**SIGN IN**

☐ Remember me [Forgot your password?](#)

# INPUTS: PASSWORD

Hides characters as typed

```
<input type="password">
```



The screenshot shows a login interface for 'GENERAL ASSEMBLY'. At the top, a black header contains the logo and name. Below it, a white banner with a close button (X) states: 'You need to sign in or sign up before continuing.' The main content area has a 'SIGN IN' heading. To the right of the heading is a link: 'Don't have an account? [Sign Up](#)'. Below the heading are two input fields: the first is labeled 'Your Email' and the second is a password field filled with dots. A red 'SIGN IN' button is positioned below the password field. At the bottom left, there is a checkbox labeled 'Remember me'. At the bottom right, there is a link: '[Forgot your password?](#)'.

# INPUT BUTTONS

## Form submit:

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

The value attribute is the button text.

Defaults to 'submit' in chrome, 'submit query' in IE

**Form button:** `<input type="button">`

**File upload:** `<input type="file">`

# BUTTON BUTTONS

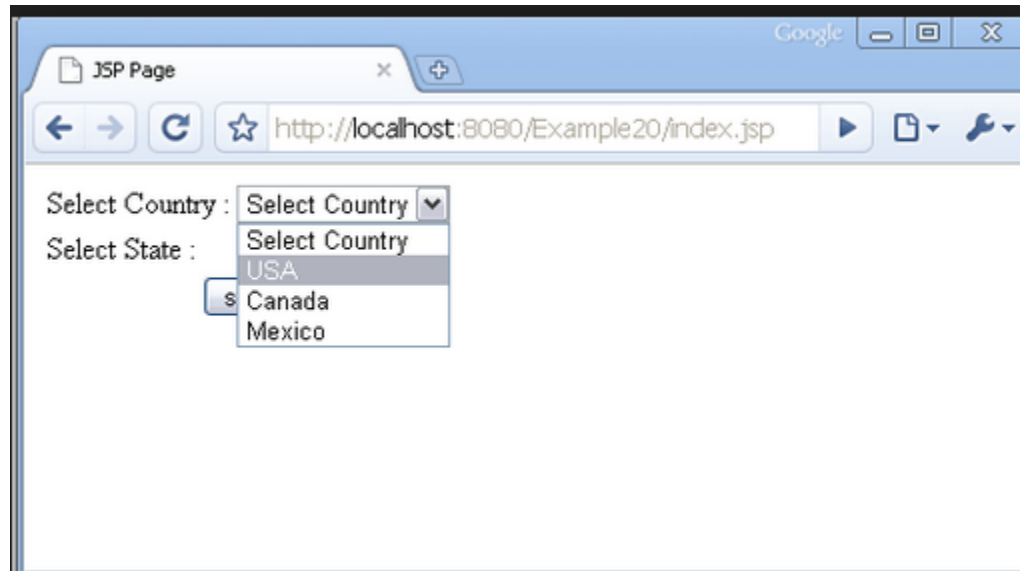
```
<button type="submit">Submit</button>
```

A button can be of three types: **submit**, **reset**, or **button**.

- a **submit** button - sends the form's data to the web page defined by the action attribute of the **<form>** element
- **reset** all the form widgets to their default (bad UX)
- a button **button** does... nothing!
  - amazingly useful to build custom buttons with JS

Button VS Input tags

# SELECT AND OPTION



```
<select name="country">
  <option value="" selected>Select Country</option>
  <option value="usa">USA</option>
  <option value="canada">Canada</option>
  <option value="mexico">Mexico</option>
</select>
```

# LABELS

Description of input field should be put in a `<label>` tag

To tie the two together choose one of these methods:

```
<label>Name: <input type="text" name="yourName"></label>
```

OR:

```
<label for="yourName">Name:</label>  
<input type="text" name="yourName" id="yourName">
```

Note: Clicking the label text in either case places the focus in the input field (great for radio buttons)

# FIELDSET/LEGEND

Fieldsets are a wrapper for grouped form elements

Eg: First, middle, last name text fields

Legend goes inside fieldset

(Defines the grouping term for the fieldset)

```
<fieldset>
  <legend>Your Name</legend>
  <input type="text" name="first_name">
  <input type="text" name="middle_name">
  <input type="text" name="last_name">
</fieldset>
```



# STYLING OPTIONS

Checkboxes, file upload inputs, radio buttons  
can't be styled directly

One option:

```
/*  Opacity 0 on the element
/*  Set it's height and width to define clickable area
/*  Set the height and width of it's parent to be the same as the input
    (don't forget position relative)
/*  Style the parent */
```

Another option:

```
/*  Hide the element,
/*  style a corresponding label how you want element to appear */
```

# RESOURCES

- Shay Howe: [Building Forms](#)
- MDN: [My First HTML Form](#)
- W3Schools: [Forms](#) / [Elements](#), [Input Types](#) / [Attributes](#)
- HTML5 Rocks: [HTML5 Forms](#) / [Client Side Validation](#)
- Webdesign.com: [Form Validation with CSS3/HTML5](#)
- **Mailchimp**: [build custom forms & collect form data](#)

# ADDITIONAL RESOURCES

- Browser Compatibility:
  - Can I use - [HTML5 Form Features](#)
  - Wufoo - [The Current State of HTML5 Forms, Example HTML5 Form with Feature Testing](#)
- Bootstrap: [Forms](#) / [Buttons](#)
  - [Bootstrap Forms / CSS Reference](#) via W3Schools
- [parsleyjs.org](http://parsleyjs.org) - a JS form validation library



# REGISTRATION FORM

Relaxr, Citipix or Final Project  
(Assignment 8)