
FRONT-END WEB DEVELOPMENT

SNACKS & DESIGN

TODAY!

TOM BUNTING & MICHAEL FISCHER

(GOOGLE SHEET IS PINNED IN SLACK)

FEWD

Q & A

!

FEWD

REVIEW

ANIMATION

TRANSITIONS

TRANSITIONS

- Provide a way to control animation speed when changing properties
- Instead of having property changes take effect immediately, you can have them take place over a period of time.

```
.example {  
  transition: [transition-property] [transition-duration] [transition-timing-function] [transition-delay];  
}
```

TRANSITIONS



PROPERTY

Which properties
to animate



DURATION

How long the
transition will last



**TIMING
FUNCTION**

How the transition
will run



DELAY

When the animation
will start

TRANSITIONS – TRANSITION-PROPERTY

PROPERTIES
TO ANIMATE

- ▶ Can specify a specific property to transition or "all" to transition all properties
- ▶ *Default:* all

```
div {  
  transition: opacity 0.5s;  
}
```

```
div {  
  transition: all 0.5s;  
}
```

```
div {  
  transition: height 0.5s;  
}
```

```
.example {  
  transition: [transition-property] [transition-duration] [timing-function] [transition-delay];  
}
```


TRANSITIONS – TRANSITION-DURATION

DURATION

- ▶ A time value, defined in seconds or milliseconds

```
div {  
  transition: all 0.5s;  
}
```

```
div {  
  transition: all 350ms;  
}
```

```
div {  
  transition: all 3s;  
}
```

```
.example {  
  transition: [transition-property] [transition-duration] [timing-function] [transition-delay];  
}
```

TRANSITIONS

TIMING FUNCTION

- Describes how a transition will proceed over its duration, allowing a transition to change speed during its course.
- Timing functions: ease, linear, ease-in, ease-out, ease-in-out

```
div {  
  transition: opacity 0.5s ease;  
}
```

```
div {  
  transition: opacity 0.5s ease-in-out;  
}
```

```
.example {  
  transition: [transition-property] [transition-duration] [timing-function] [transition-delay];  
}
```

transition timing: [W3 Schools](https://www.w3schools.com/css/css3_transitions.asp)

TRANSITIONS

DELAY

- ▶ Length of time before the transition starts

```
div {  
  transition: background-color 0.5s ease 2s;  
}
```

```
.example {  
  transition: [transition-property] [transition-duration] [timing-function] [transition-delay];  
}
```

MORE FUN WITH TRANSITIONS — CODROPS

Fun CSS button styles: [Creative buttons](#)

Icon hover effects: [Icon Hover Effects](#)

Modal dialogue effects (advanced): [Dialogue Effects](#)

ANIMATION

TRIGGERING TRANSITIONS

TRIGGERING TRANSITIONS

There are two ways to trigger CSS transitions:

1. Using the :hover CSS pseudo-class
2. Adding a class with jQuery

ANIMATION

TRANSFORMATIONS

TRANSFORM

transform: [W3 Schools](#)
transform-origin: [W3 Schools](#)

[lesson15_solutions/ [00] - Review Sidebar Lab]

RESPONSIVE BASICS

RESPONSIVE — REM/EM

EM

- *Relative* unit
- Sized based on the width of the letter “m”
- 1em = 100% font-size
- .5em = 50% font-size
- **Based on parent**

Parent { font-size:16px;}

Child {font-size:2em;}  Child's font size is 32px (200% x 16px)

REM

- "Root" em
- Same as em **except** based on the font-size of the html element

PIXELS AND EMS AND REMS, OH MY!!

| | RELATIVE? | BASED ON |
|-----|-----------|--------------|
| PX | absolute | |
| EM | relative | parent |
| REM | relative | html element |

THE BENEFIT OF USING RELATIVE UNITS

```
html { font-size: 16px; }
h1 { font-size: 33px; }
h2 { font-size: 28px; }
h3 { font-size: 23px; }
h4 { font-size: 19px; }
small { font-size: 13px; }
.box { padding: 20px; }

@media screen and (min-width: 1400px) {
  html { font-size: 20px; }
  h1 { font-size: 41px; }
  h2 { font-size: 35px; }
  h3 { font-size: 29px; }
  h4 { font-size: 24px; }
  small { font-size: 17px; }
  .box { padding: 25px; }
}
```

```
html { font-size: 16px; }
h1 { font-size: 2.074rem; }
h2 { font-size: 1.728rem; }
h3 { font-size: 1.44rem; }
h4 { font-size: 1.2rem; }
small { font-size: 0.833rem; }
.box { padding: 1.25rem; }

@media screen and (min-width: 1400px) {
  html { font-size: 18px; }
}
```

RELATIVE UNITS – CROSS BROWSER

```
html { font-size: 16px; }
h1 {
  font-size: 33px;
  font-size: 2.074rem;
}
h2 {
  font-size: 27px;
  font-size: 1.728rem;
}

@media screen and (min-width: 1400px) {
  html { font-size: 18px; }
}
```

~~**Tuesday April 5th**~~ - ~~Guest Speaker: Mig Reyes!~~ (plus drinks?)

Thursday April 7th - Form Design & Interaction

Sunday April 10th - Office Hours (*Eric*)

Tuesday April 12th - CSS Preprocessors (SASS & LESS)

Thursday April 14th - Launching your site! (Domain, hosting, etc)

Sunday April 17th - Office Hours (*Eric*)

Eric Boyer

Tuesday April 19th - Google Analytics, terminal basics, lab time

***** STUDENTS CHOICE? WORDPRESS? *****

Thursday April 21st - Terminal based tools + lab time

Saturday April 23rd - Office Hours (*Eric*)

Sunday April 24th - Office Hours (*Eric*)

Thursday April 26th - FINAL PROJECT PRESENTATIONS

Final Project / Homework Milestones

-

Sunday April 10th - Final Project Code - Milestone 1

Sunday April 17th - Final Project Code - Milestone 2

Sunday April 24th - Final Project Code - Milestone 3 *last minute tweaks*

Eric Boyer

SPROUT SOCIAL PANEL

Wed April 13th @ 6pm
(dinner provided)

131 S Dearborn St, Chicago, IL 60603

FORM DESIGN & INTERACTION

Eric Boyer

LEARNING OBJECTIVES

- › Be able to differentiate the different types of inputs and why/where we would use each
- › Explain how to group elements by name.

AGENDA



- Forms and Inputs
- Lab
- Form Validation
- Functional Forms


FORMS AND INPUTS

FORM BASICS

FORM BASICS

FORMS

How we get data from users

 **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

1. The user fills out the form and presses the submit button



FRONT-END WEB DEVELOPMENT

APPLY NOW

Where are you thinking of taking this course?

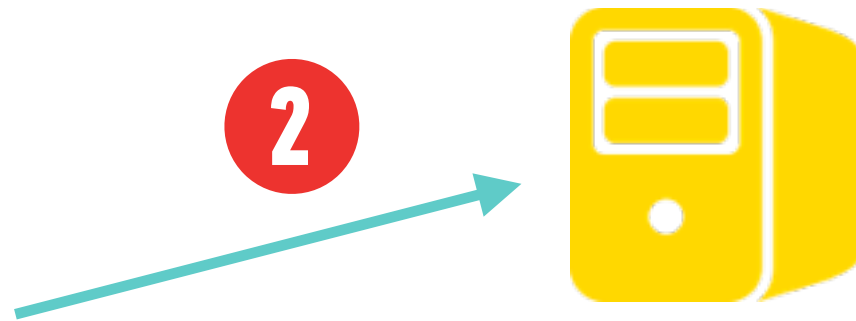
New York City

CONTINUE TO APPLICATION

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

FORMS

-
2. The **name** of each form field is sent to the server along with the **value** the user entered or selected



FRONT-END WEB DEVELOPMENT

APPLY NOW

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

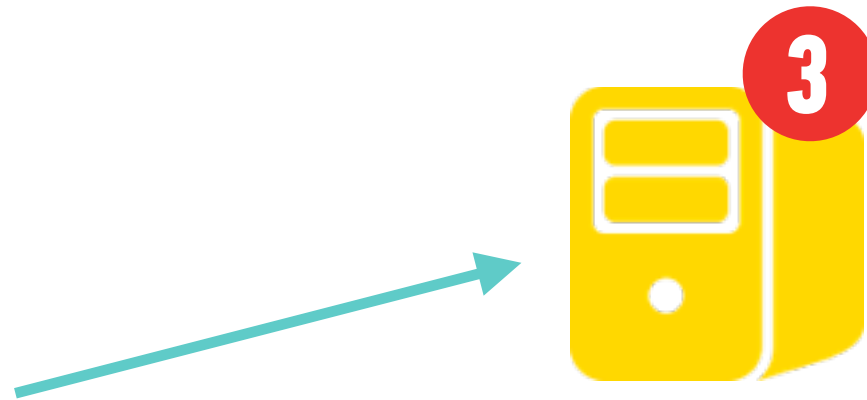
Where are you thinking of taking this course?

New York City

CONTINUE TO APPLICATION

FORMS

3. The server processes the data using a language such as PHP, C# or Java. It may also store the information in a database



FRONT-END WEB DEVELOPMENT
APPLY NOW

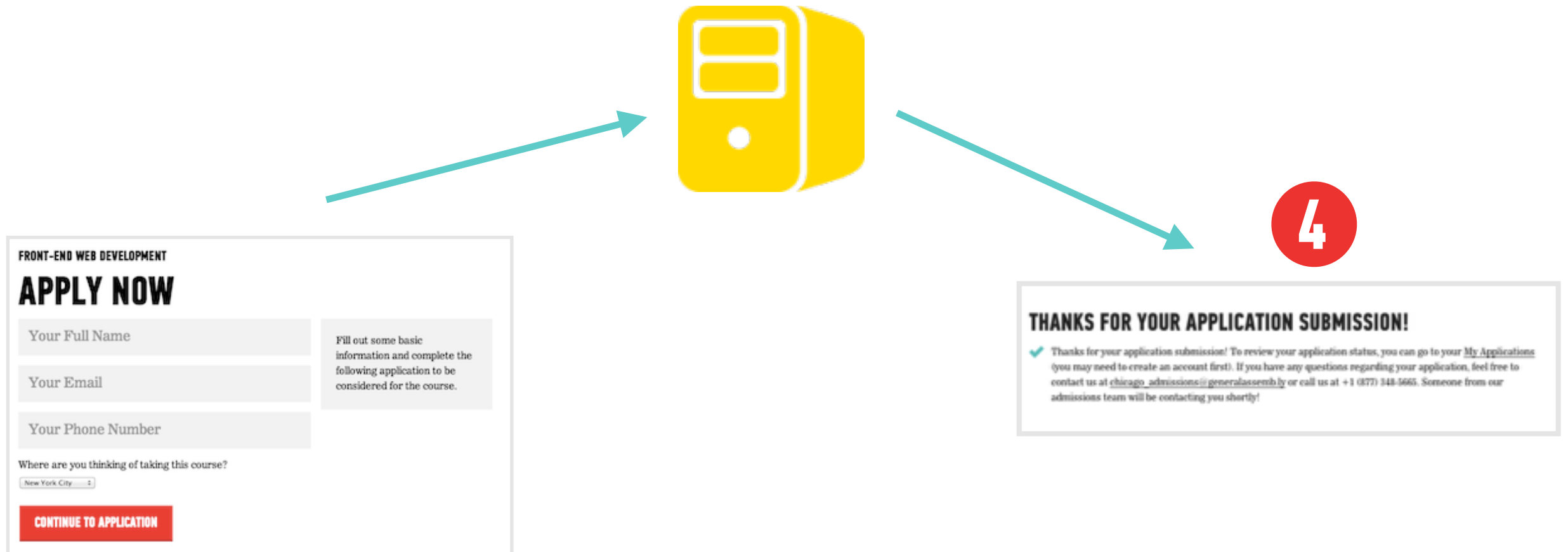
Where are you thinking of taking this course?

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

CONTINUE TO APPLICATION

FORMS

4. The server creates a new page to send back to the browser based on the information received.



FORMS

Form controls live inside the <form element>

```
<form action="http://www.example.com/login.php" method="post">  
  <!--Data collection elements go here-->  
</form>
```

FORMS

Form attributes:

```
<form action="http://www.example.com/login.php" method="post">  
  <!--Data collection elements go here-->  
</form>
```

FORMS

Form attributes:

ACTION (REQUIRED)

Where to send the data (URL)



```
<form action="http://www.example.com/login.php" method="post">  
  <!--Data collection elements go here-->  
</form>
```


FORMS

Form attributes:

ACTION (REQUIRED)

Where to send the data (URL)

METHOD (WILL USUALLY HAVE)

How to send it (post or get)



```
<form action="http://www.example.com/login.php" method="post">  
  <!--Data collection elements go here-->  
</form>
```

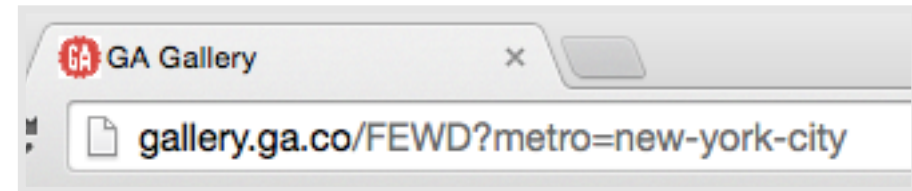
FORMS — METHODS

POST

- Data is not shown in URL
- Can contain sensitive data
- No size limitations
- Adds information to, or deletes info from a database

GET

- Short forms (such as search fields)
- Appended to URL in name/value pairs
- Never use for sensitive info!!!
- Useful for form submissions when user wants to bookmark results



```
<form action="http://www.example.com/login.php" method="post">  
  <!--Data collection elements go here-->  
</form>
```

FORM BASICS

GETTING INFORMATION FROM USER

GETTING INFO — INPUTS

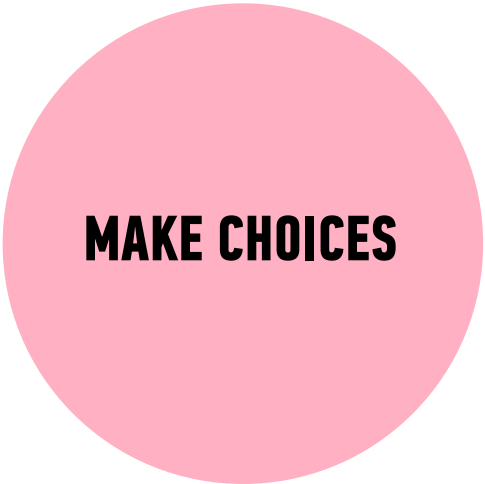
Place any inputs between `<form>` `</form>` tags

Attributes:

- **type** — text, submit, password, email, checkbox, button, radio, file, etc.
- **name, value** — The name attribute is sent to the user along with the value the user selects.
- **placeholder** — For text inputs - hint for what user should enter in field

Note: For a complete spec see [MDN](#)

FORM



Where are you thinking of taking this course?



NAME/VALUE PAIRS

- Information is sent from the browser to the server using name/value pairs.

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

| NAME | VALUE |
|----------|-------------------|
| username | =eboyer@gmail.com |

```
<input type="radio" name="color" value="red" label="Red">  
<input type="radio" name="color" value="green" label="Green" checked="checked">
```



A visual representation of the radio button code. It shows two radio buttons. The first is labeled 'Red' and is unselected. The second is labeled 'Green' and is selected, indicated by a filled circle and a dashed border around the label.

| NAME | VALUE |
|-------|--------|
| color | =green |

INPUTS — TEXT

ADD TEXT

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

Your Full Name

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

Your Email

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

.....

**Can also carry a maxlength attribute to limit the number of characters the user may enter*

LABELS

Information about the input field should be put in a <label> tag:

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

To tie the two together:

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

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

ACTIVITY



EXERCISE

KEY OBJECTIVE

- ▶ Identify input types, add styles to a form

TYPE OF EXERCISE

- ▶ Individual/partner

TIMING

45 min

1. Review contact_form.png
2. Write html for contact form
3. Style the form
4. Optional: add responsive styles (see contact_form_responsive.png)
5. Super optional: Add validation (see later slides)

** You will need to look up the textarea element*

SELECT AND OPTION

MAKE CHOICES

```
<select name="referral">
  <option value="friend">Friend</option>
  <option value="instructor">Instructor</option>
  <option value="online">Online</option>
</select>
```

Where are you thinking of taking this course?

Chicago

- Atlanta
- Austin
- Boston
- ✓ Chicago
- Hong Kong
- London
- Los Angeles
- Melbourne
- New York City
- San Francisco
- Seattle
- Sydney
- Washington D.C.

CHECKBOXES AND RADIO BUTTONS

MAKE CHOICES

```
<input type="checkbox" name="store_credentials">
```


☐ Remember me

RADIO BUTTONS

MAKE CHOICES

Radio buttons are grouped together by their name attribute

```
<input type="radio" name="color" value="red" label="Red">  
<input type="radio" name="color" value="green" label="Green" checked="checked">
```



SUBMIT FORM

**SUBMIT**

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

Continue

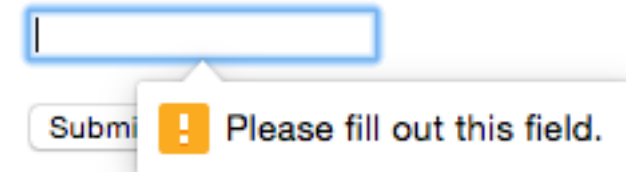
FORM BASICS

VALIDATION

VALIDATION

- ▶ You've probably seen forms on the web that give users messages if the form control has not been filled out correctly.
- ▶ Traditionally, validation has been performed using Javascript.
- ▶ HTML5 also introduced browser-based form validation.

```
<input type="text" name="fullname" required />
```



VALIDATION

- ▶ For more substantial validation, it is highly recommended that you use a validation library, such as [Parsley](#).
- ▶ To add parsley validation:
 1. Add jQuery to your project
 2. Add the parsley.js file to your project after you've included jQuery

```
<script src="js/jquery-2.1.3.min.js"></script>  
<script src="js/parsley.js"></script>
```

3. Add the data-parsely-validate attribute to your form tag

```
<input data-parsley-validate/>
```

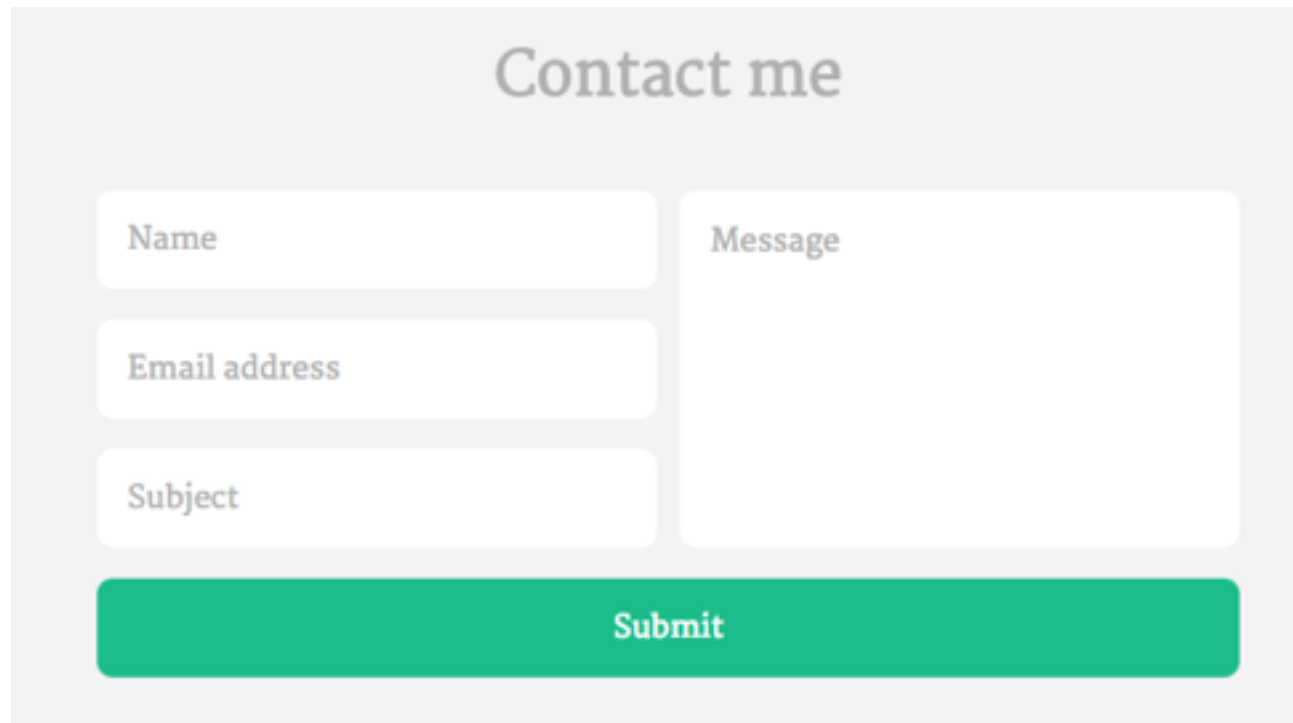
4. Add the required attribute to any fields you want to be required.

```
<input type="text" name="fullname" required />
```


CONTACT FORM

BUILDING A FUNCTIONAL CONTACT FORM

1. I recommend [FormSpree](#) for contact forms that send you an email without having to use PHP.
2. Your site will need to be hosted on a server in order to test emails.



Contact me

Name

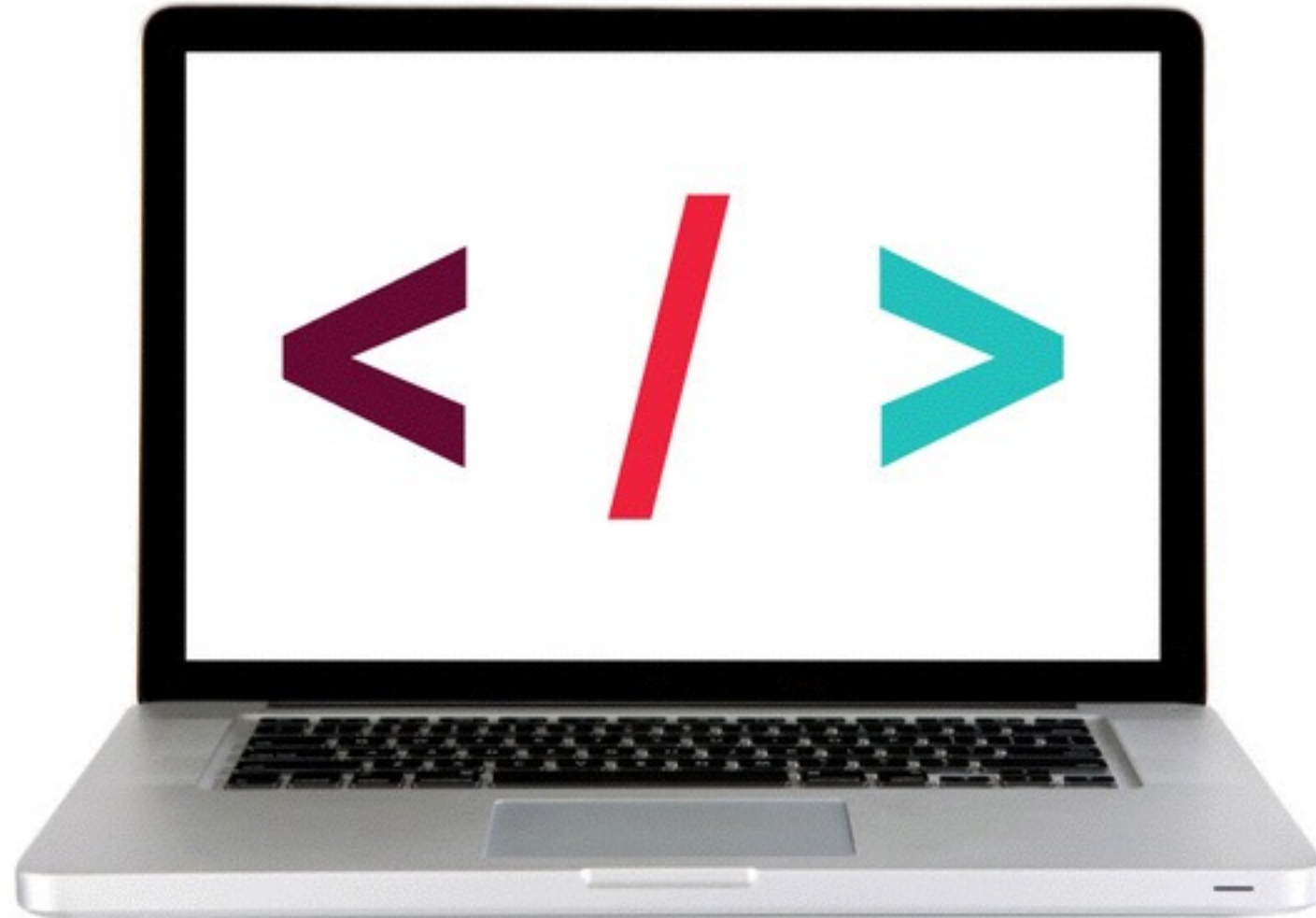
Email address

Subject

Message

Submit

LET'S TAKE A CLOSER LOOK



LEARNING OBJECTIVES

- › Be able to differentiate the different types of inputs and why/where we would use each
- › Explain how to group elements by name.

FINAL PROJECTS

HOMEWORK

Final Project / Homework Milestones

-

Sunday April 10th - Final Project Code - Milestone 1

Sunday April 17th - Final Project Code - Milestone 2

Sunday April 24th - Final Project Code - Milestone 3 *last minute tweaks*

Eric Boyer

FEEDBACK GROUPS

DUE 11:59PM, SUNDAY APRIL 10TH

- Final Project Code - Milestone 1
- Draft of HTML/CSS
- Create a “milestone1” branch in your class repository
- Make your code updates in your branch
- Push your branch to Github and create a “Pull Request” to your “master” branch

Let's create the branch together.

FEEDBACK GROUPS

FINAL PROJECT CODE – MILESTONE 1 (FOLDER STRUCTURE)

- Final Project
 - Assets
 - wireframe.jpg
 - proposal.txt
 - Site
 - index.html

FEEDBACK GROUPS

FINAL PROJECT CODE – MILESTONE 1 (CODE IDEAS)

- Use HTML5 structural elements. To describe layout (header, footer, nav, section)
- Demonstrate uses of classes and IDs (OR NAW?)
- Use the appropriate tags to describe content (h1, h2, p)
- Use a custom font with Google Fonts
- Demonstrate layout techniques. (floats, display: inline-block, percentage based layouts, etc)

FRONT-END WEB DEVELOPMENT

SNACKS & DESIGN

TUESDAY APRIL 12TH

JON ILER & JIM HOWES

(GOOGLE SHEET IS PINNED IN SLACK)

RESPONSIVE BASICS

EXIT TICKETS