FORMS & BOOTSTRAP

WEEKLY OVERVIEW

WEEK 4	Forms & Bootstrap / Intro to JavaScript
WEEK 5	jQuery Basics / jQuery Part II

FORM BASICS

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.
- Integrate Bootstrap to create a multi-column layout

AGENDA

- Review
- Forms & Inputs
- Lab
- Bootstrap

HOMEWORK REVIEW

HOMEWORK — GROUP DISCUSSION

EXERCISE

TYPE OF EXERCISE

- Groups of 2-3

TIMING

10 min

1. Pick someone to take notes for your group.
2. Share 1 thing you're excited about being able to accomplish. Focus on the positives!
3. Have each person in the group note 1 thing they found challenging for the assignment and make note. Discuss as a group how you think you could solve that problem.
4. Discuss the bonus tasks (hover effect with smooth transition, fixed background, sticky nav, image hover, jump links, masonry layout for gallery). If anyone in your group was able to tackle one of these, share!

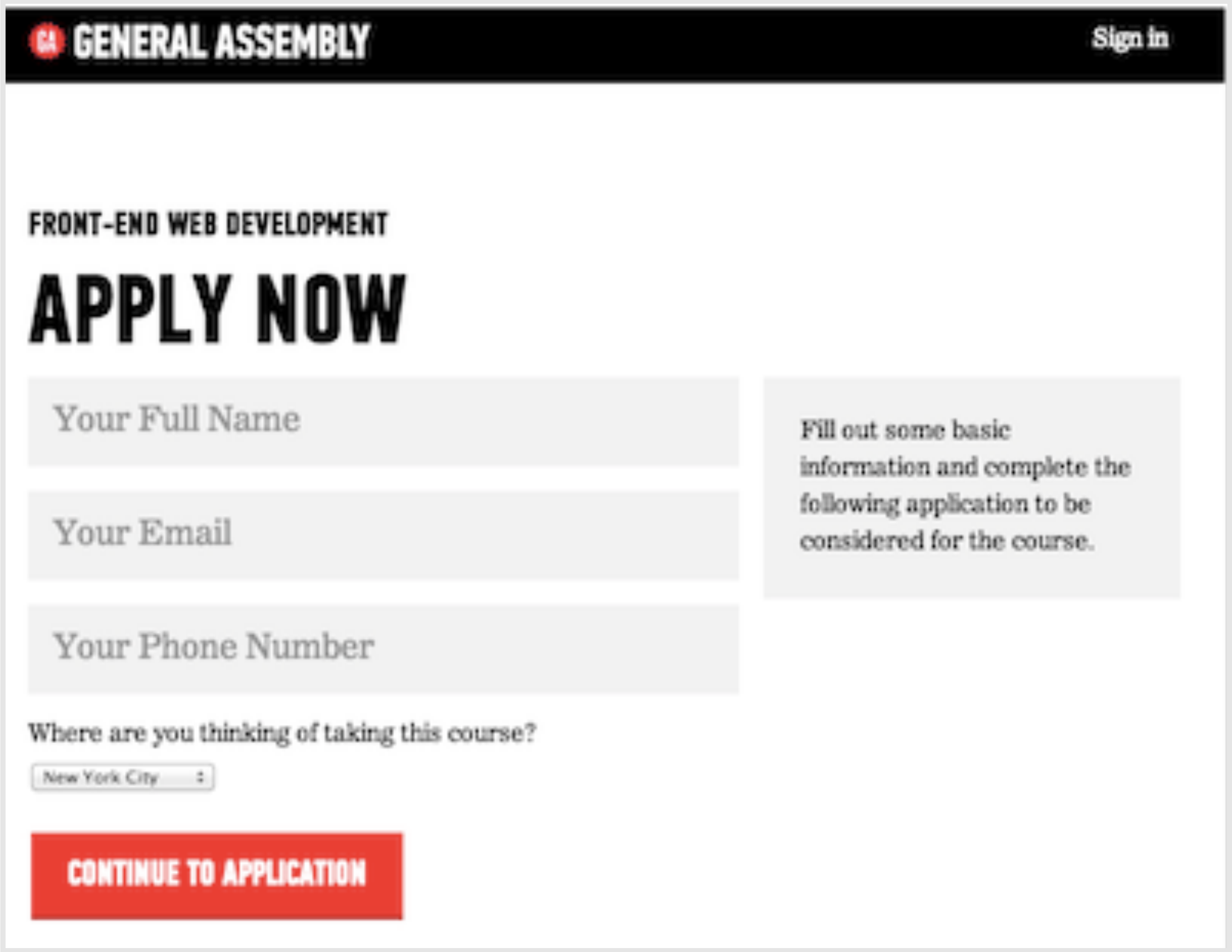
FORM BASICS

FORMS AND INPUTS

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

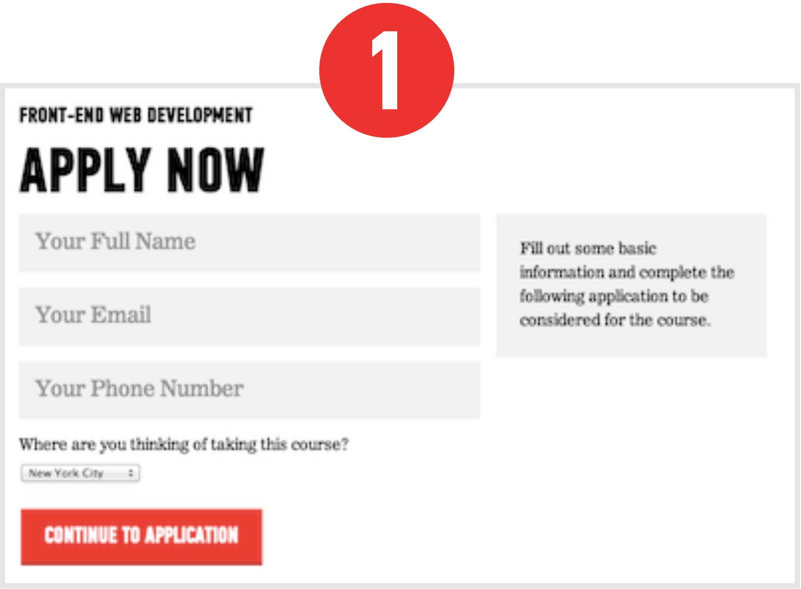
Fill out some basic information and complete the following application to be considered for the course.

Where are you thinking of taking this course?

New York City

FORMS

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



FRONT-END WEB DEVELOPMENT

APPLY NOW

Your Full Name

Your Email

Your Phone Number

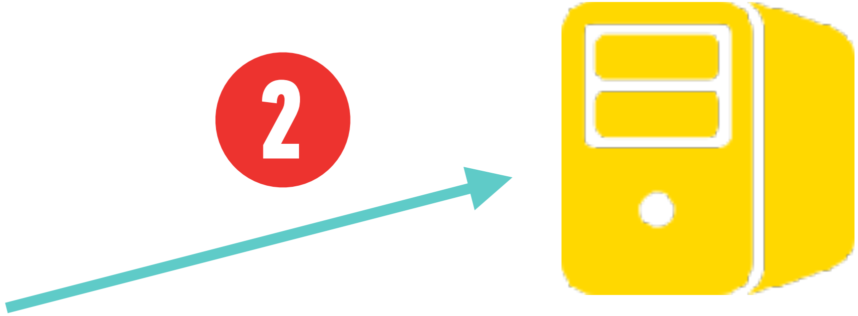
Fill out some basic information and complete the following application to be considered for the course.

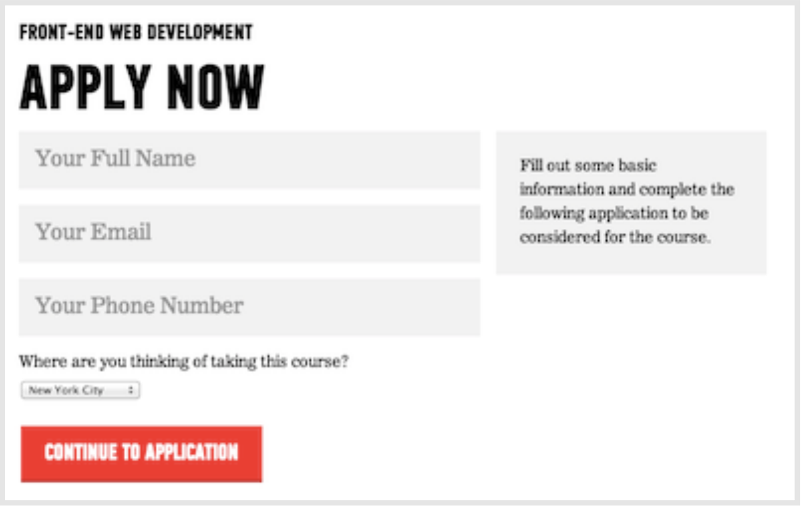
Where are you thinking of taking this course?

New York City

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

Your Full Name

Your Email

Your Phone Number

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

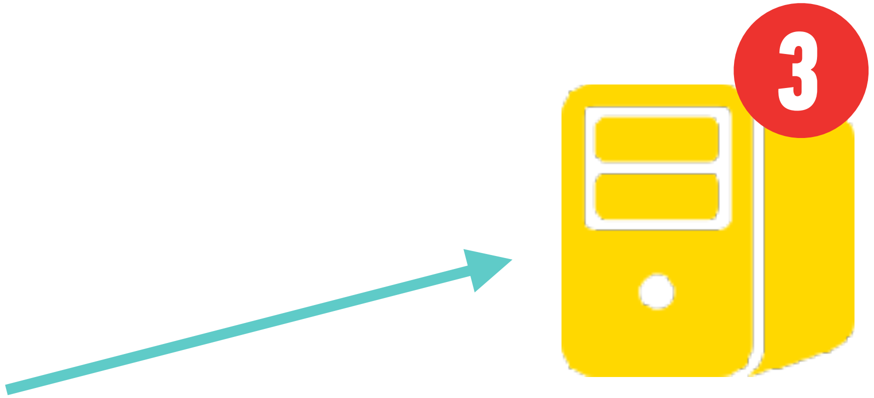
Where are you thinking of taking this course?

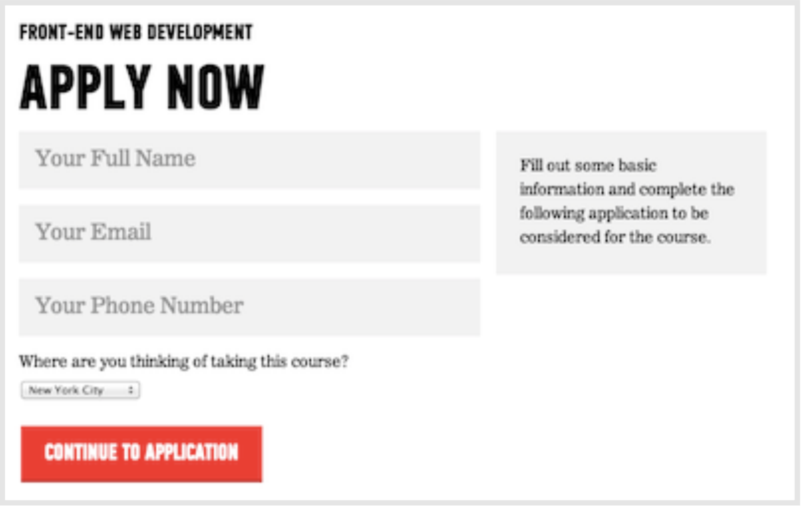
New York City

CONTINUE TO APPLICATION

FORMS

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

Your Full Name

Your Email

Your Phone Number

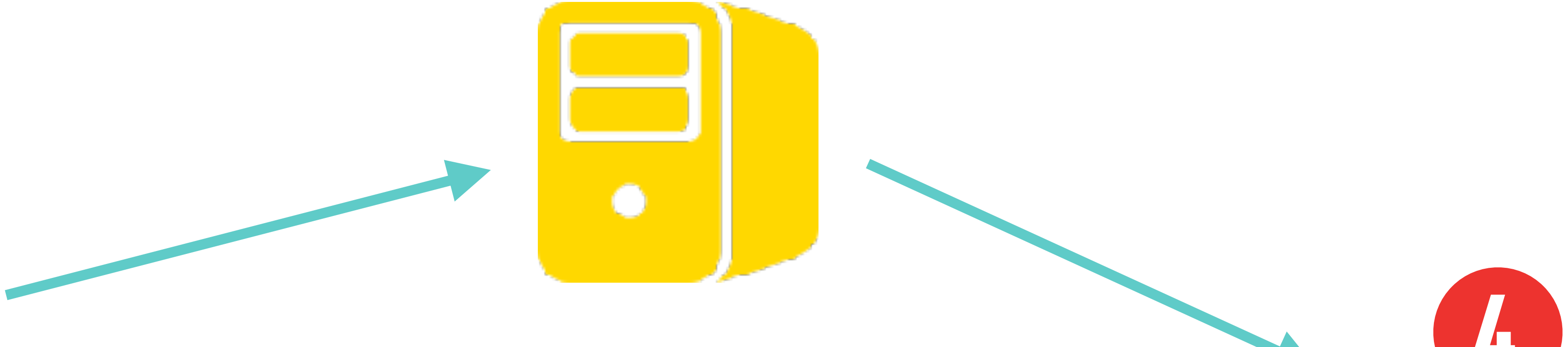
Fill out some basic information and complete the following application to be considered for the course.

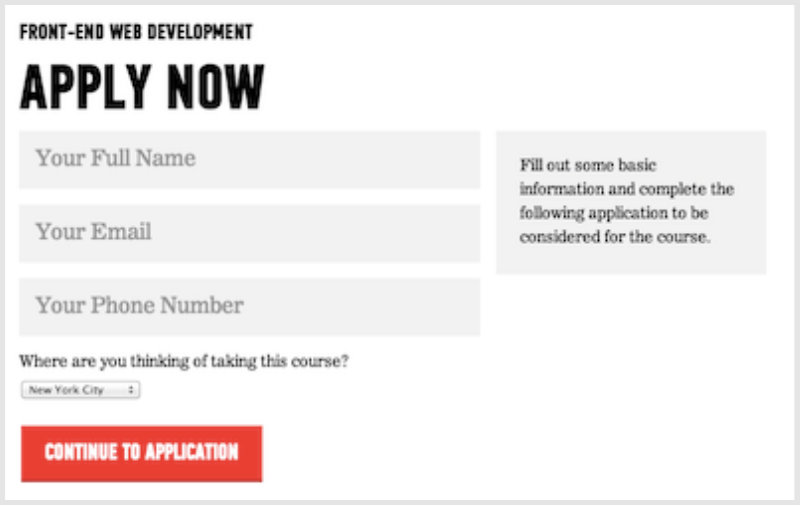
Where are you thinking of taking this course?

New York City

FORMS

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





FRONT-END WEB DEVELOPMENT

APPLY NOW

Your Full Name

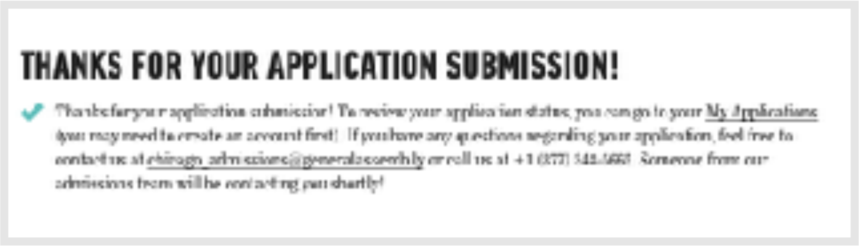
Your Email

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



THANKS FOR YOUR APPLICATION SUBMISSION!

✓ Thanks for your application submission! To review your application status, you can go to your My Applications (you may need to create an account first). If you have any questions regarding your application, feel free to contact us at chicago_admissions@generalassembly or call us at +1 (877) [illegible]. Someone from our admissions team will be contacting you shortly!

FORMS

- Form controls live inside the `form` element.
- The entire form (all the fields you want to be submitted together) go within the `form` element.

```
<form>
  <!--Data collection elements go here-->
</form>
```

FORM BASICS

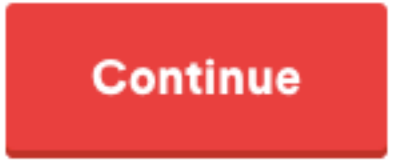
GETTING INFORMATION FROM USERS

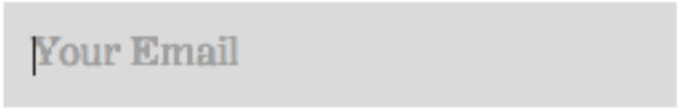
FORM — TYPES OF INPUTS

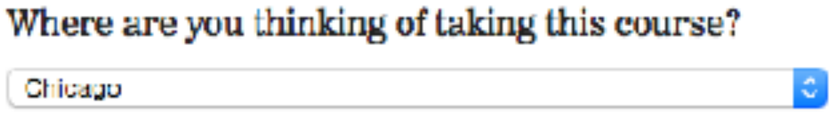
SUBMIT

ADD TEXT

MAKE CHOICES






Where are you thinking of taking this course?

Chicago

GETTING INFO — INPUTS

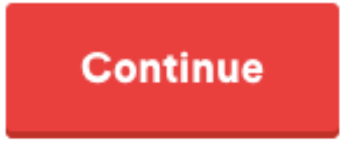
- Allow us to get content from users.
- Should live inside our form tags.

```
<form>
    <input type="text" name="fullName" placeholder="Enter your full name">
    <button type="submit">Continue</button>
</form>
```

SUBMITTING A FORM

★

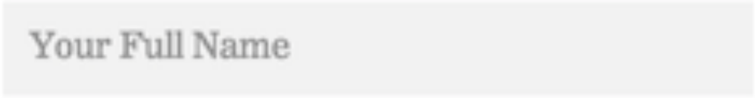
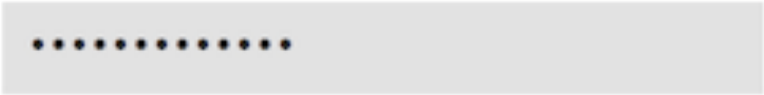
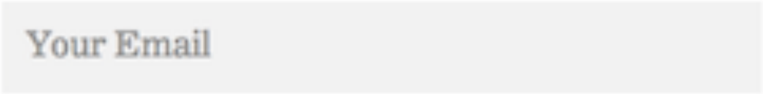
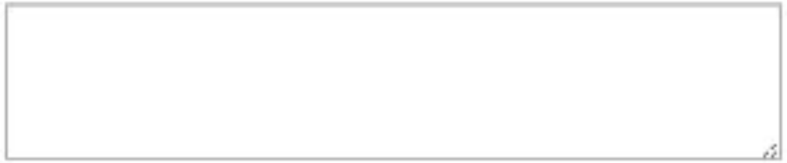
```
<button type="submit">Continue</button>
```



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

TEXT INPUTS — DESCRIPTIONS

ADD TEXT

FIELD TYPE	HTML	ATTRIBUTES	WIDGET
Plain Text	`<input type="text">`	*type, name, id, placeholder*	 Your Full Name
Password	`<input type="password">`	*type, name, id, placeholder*	 ••••••••••••••
Email	`<input type="email">`	*type, name, id, placeholder*	 Your Email
Text Area	`<textarea></textarea>`	*name, id, placeholder*	

TEXT INPUTS – ATTRIBUTES

ADD TEXT

ATTRIBUTE	HTML EXAMPLE	DESCRIPTION
type	`type="text"`	The type of data that is being input. *Examples: text, email, password, checkbox, radio*
name	`name="username"`	The key used to describe this data in the HTTP request.
id	`id="message"`	The unique identifier for the element (will match the label)
placeholder	`placeholder="Full Name"`	A "prompt" for an input letting the user know what to enter into the field.

ALL TOGETHER NOW! EXAMPLES

ADD TEXT

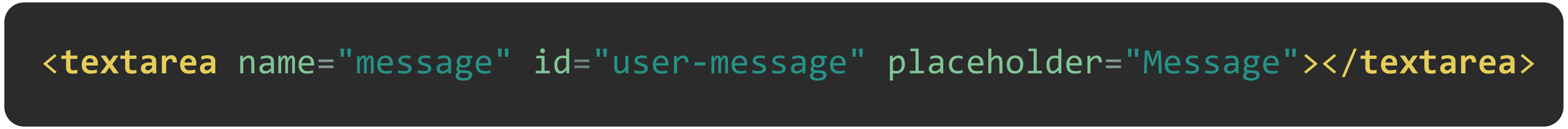
```
<input type="text" name="username" placeholder="Your Full Name" id="full-name">
```

```
<input type="email" name="email-address" placeholder="Your Email" id="email-address">
```

```
<input type="password" name="user-password" id="password">
```

ALL TOGETHER NOW! EXAMPLES

ADD TEXT



```html
<textarea name="message" id="user-message" placeholder="Message"></textarea>
```

STYLING INPUTS

Want an easy way to style inputs in your CSS without adding a bunch of ids or classes to each one? The attribute selector is just the ticket!

```css
input[type="text"] {
   border: 1px solid grey;
}
```

```css
input[type="password"] {
   border: 1px solid grey;
}
```

ACTIVITY

EXERCISE

KEY OBJECTIVE

- Identify input types, add styles to a form

TYPE OF EXERCISE

- Individual/partner

TIMING

15 min

1. In the contact_form folder, review contact_form.png
2. Write html for contact form
3. Style the form

BONUS

- Style the placeholder text. You'll need to Google this! "Style placeholder text CSS"
- Add a hover effect to the button!

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">
```

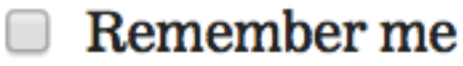
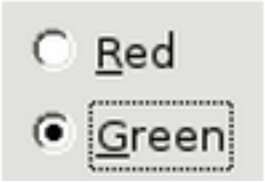
To tie the two together:

```
<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)

INPUTS — MAKING CHOICES

MAKE CHOICES

FIELD TYPE	HTML	ATTRIBUTES	WIDGET
Checkbox	`<input type="checkbox">`	*type, name, id, checked*	 ☐ Remember me
Radio	`<input type="radio">`	*type, name, id, value, checked*	 ○ Red ● Green

TEXT INPUTS – ATTRIBUTES

MAKE CHOICES

ATTRIBUTE	HTML EXAMPLE	DESCRIPTION
type	`type="text"`	The type of data that is being input. *Examples: text, email, password, checkbox, radio*
name	`name="username"`	The key used to describe this data in the HTTP request.
id	`id="message"`	The unique identifier for the element (will match the label)
value	`value="Green"`	For radios, the value that should be sent to the server

ALL TOGETHER NOW! CHECKBOXES

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

RADIO BUTTONS

MAKE CHOICES

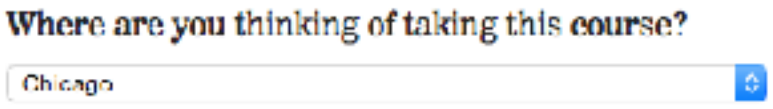
We can connect radios together by giving them the *same* **name** *attribute*

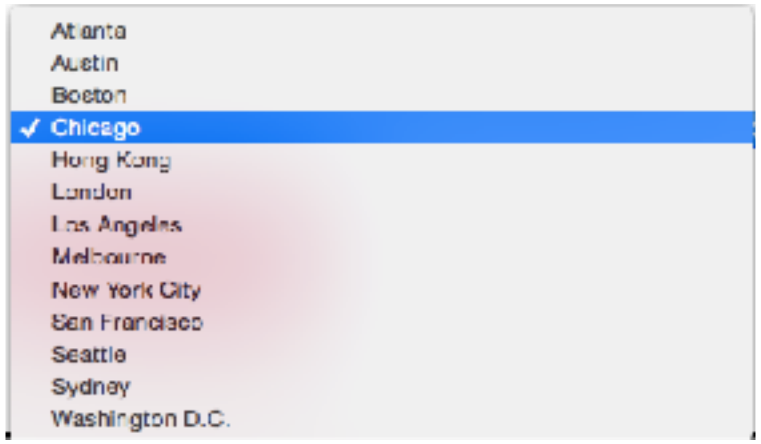
```
<input type="radio" name="color">
<input type="radio" name="color" checked="checked">
```

INPUTS — MAKING CHOICES

MAKE CHOICES

FIELD TYPE	HTML	ATTRIBUTES	WIDGET

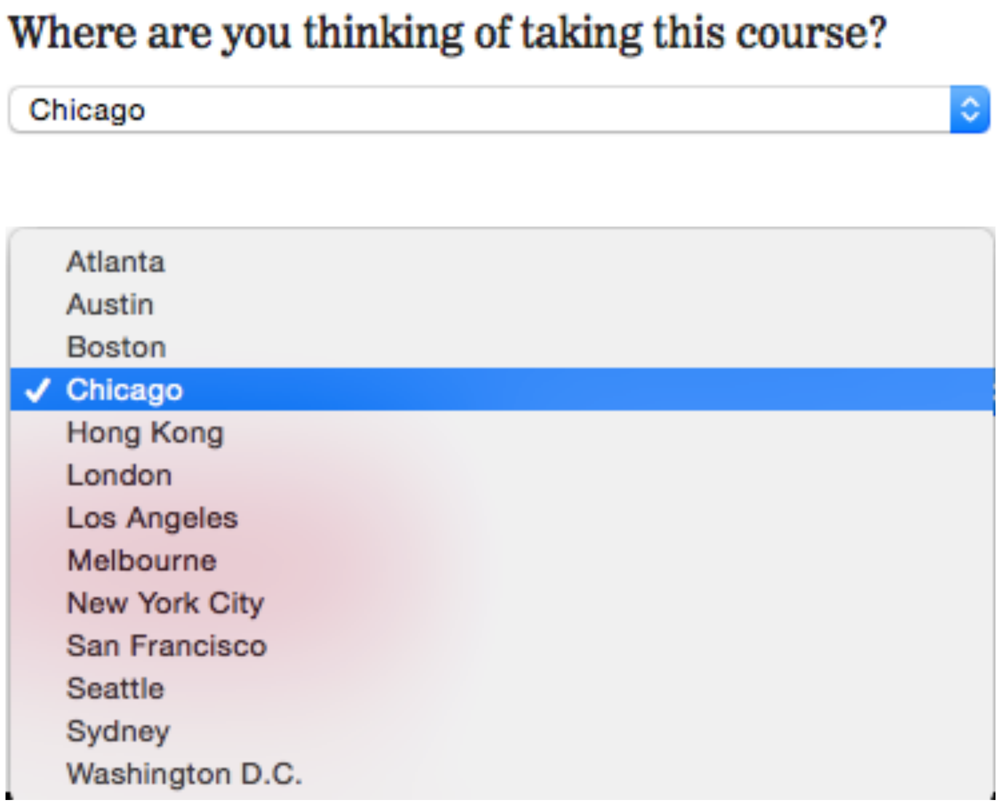
Select	`<select name="referral">`
	`  <!-- options go here -->`
	`</select>`
		name, id	
			Where are you thinking of taking this course?
			Chicago

Option	`<option>Friend</option>`	*value*	 Atlanta Austin Boston ✓ Chicago Hong Kong London Los Angeles Melbourne New York City San Francisco Seattle Sydney Washington D.C.

ALL TOGETHER NOW! SELECT AND OPTIONS

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?

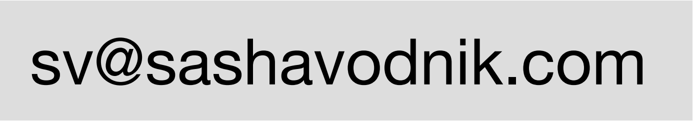
Chicago

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

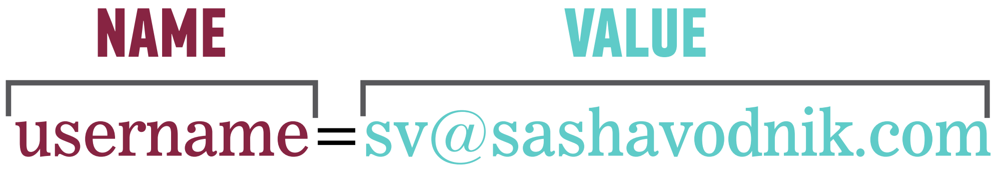
NAME/VALUE PAIRS

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

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



sv@sashavodnik.com

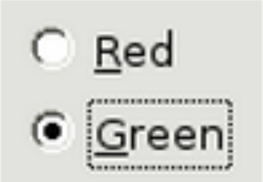


NAME

VALUE

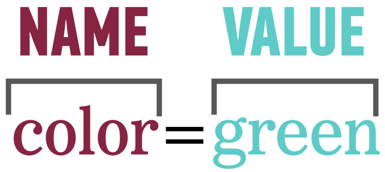
username=sv@sashavodnik.com

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



○ Red

● Green



NAME

VALUE

color=green

LAB

ACTIVITY

EXERCISE

KEY OBJECTIVE

- Identify input types, add styles to a form

TYPE OF EXERCISE

- Individual/partner

TIMING

5 min 1. In the application_form folder, review application_form.png and discuss with a partner

45 min 2. Write HTML for the form

3. Style the form with CSS. Focus on getting the form centered and getting the information on the right rows, and then add other styles if you have time.

BONUS

- Research and implement the fieldset element

GRID SYSTEMS

INTEGRATING BOOTSTRAP

LAYOUT GRIDS

- Composition in any visual art is the placement or arrangement of visual elements — how they are organized on a page.
- Many designers use a grid structure to help them position items on a page.

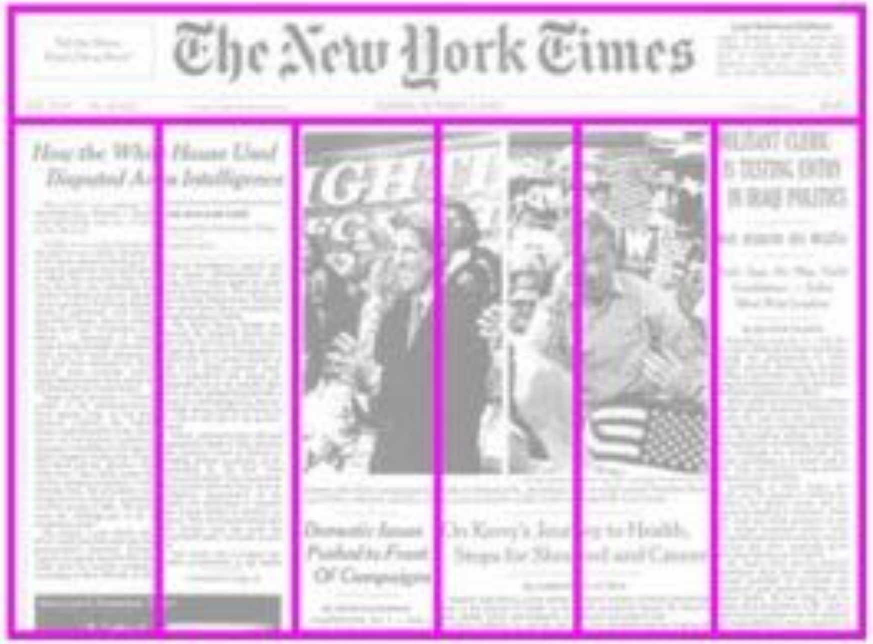
The New York Times

How the White House Used Disputed Arms Intelligence

MILITANT CLERIC IS TESTING ENTRY IN IRAQI POLITICS

Domestic Issues Pushed to Front Of Campaigns

On Kerry's Journey to Health, Stops for Shrapnel and Cancer



BENEFITS OF GRID SYSTEM LAYOUTS

While grids might seem like a restriction at first, they have many benefits including:

- Creating continuity between different pages
- Helping users predict where they can find information on various pages
- Making it easier to add new content to the site in a consistent way
- Helping designers/developers collaborate in a consistent way

CSS FRAMEWORKS

CSS FRAMEWORKS PROVIDE CODE FOR COMMON TASKS, SUCH AS:

- Creating responsive layout grids
- Styling forms
- Styling UI elements, etc.

PROS:

- Save developers from repeatedly writing code for the same tasks
- Tested across various browsers
- Useful for rapid website development

CONS:

- Often require developers to use class names in HTML that only control the presentation of the page (rather than describe content)
- Often contain more code than you need for your particular page, creating “bloat”

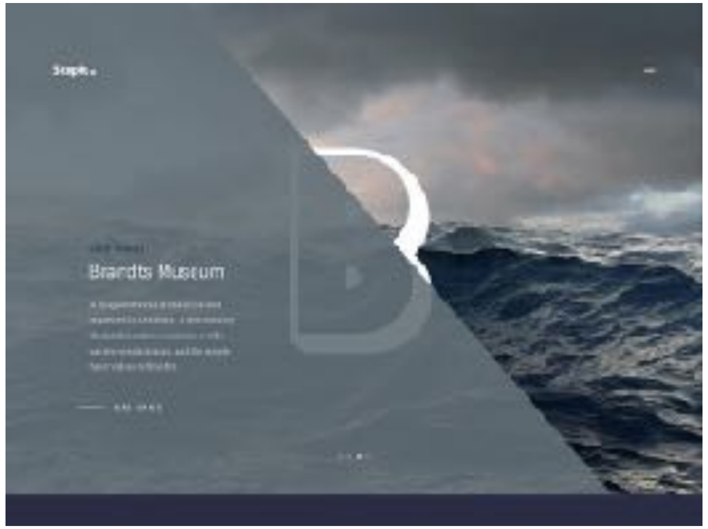
HELLO BOOTSTRAP!

- **Bootstrap** is *“the most popular HTML, CSS and JS framework for developing responsive, mobile first projects on the web.”*

FEATURES:

1. Grid system
2. Forms/buttons
3. Navigation
4. Tabs and pills
5. Alerts/error messages
6. Modals
7. And much more

SITES THAT USE BOOTSTRAP



Brandts Museum

BE GOOD WITH YOUR MONEY
STAY ON TOP OF IT ALL
AND HAVE YOUR BEST YEAR YET

See understand and do more with your finances everyday.

All-in-one? Done

Bills? Paid

Credit? Checked

BMW STAND 360° – GENEVA INT'L MOTORSHOW

DISCOVER THE BMW STAND IN 360°

Responsive HTML5/CSS3

And many, many more: Bootstrap Expo

GETTING STARTED

GETTING STARTED

1. Download Bootstrap here
2. Once downloaded, unzip and open folder in Finder. You'll see something like this:

```
bootstrap/
├── css/
│   ├── bootstrap.css
│   ├── bootstrap.css.map
│   ├── bootstrap.min.css
│   ├── bootstrap-theme.css
│   ├── bootstrap-theme.css.map
│   └── bootstrap-theme.min.css
├── js/
│   ├── bootstrap.js
│   └── bootstrap.min.js
└── fonts/
    ├── glyphicons-halflings-regular.eot
    ├── glyphicons-halflings-regular.svg
    ├── glyphicons-halflings-regular.ttf
    ├── glyphicons-halflings-regular.woff
    └── glyphicons-halflings-regular.woff2
```

3. We'll then need to add the files we want to use to our project and include those files in our HTML.

BOOTSTRAP

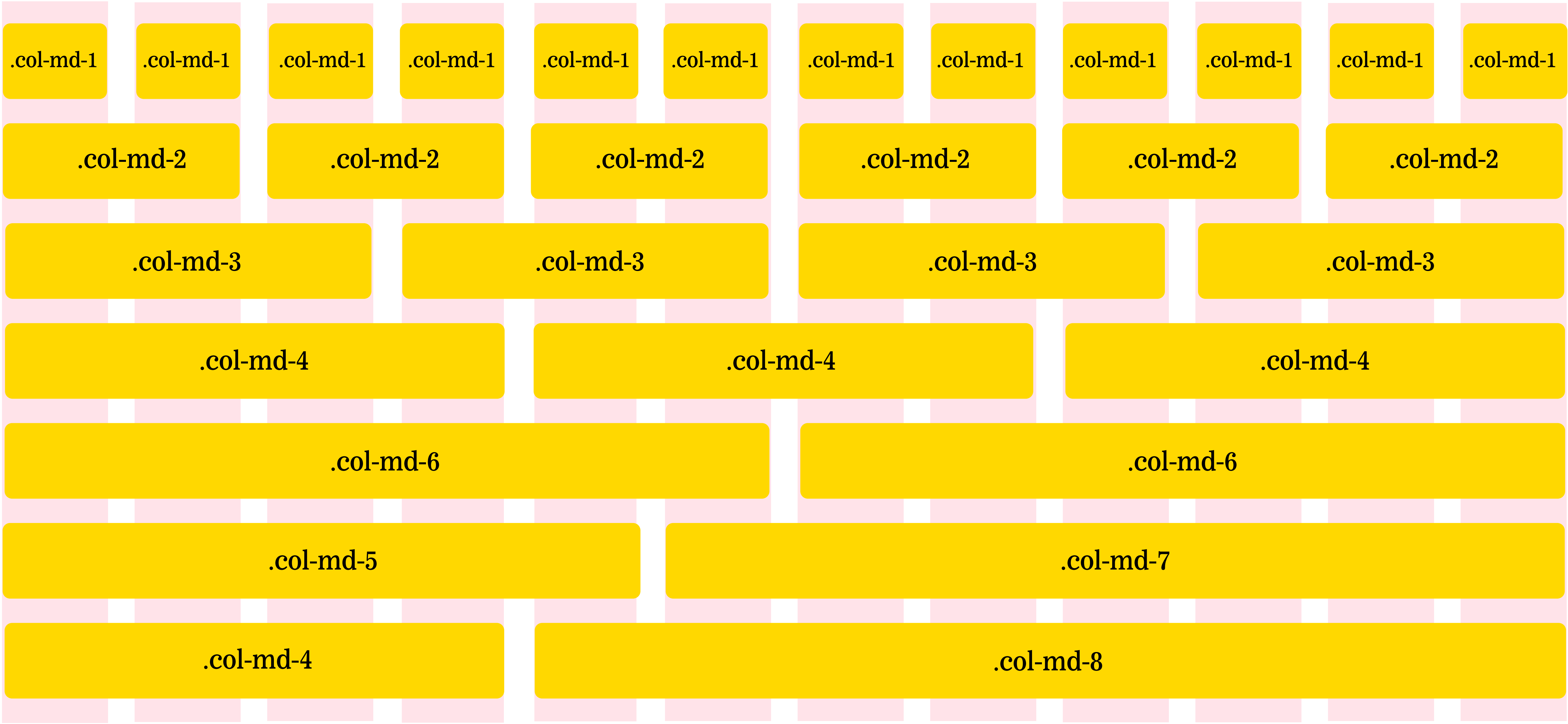
GRID SYSTEM

BOOTSTRAP — GRID SYSTEM

- Bootstrap includes a responsive, mobile-first fluid grid system that appropriately scales up to **12 columns** as the device or viewport size increases.
- Page layouts are created through a series of rows and columns that house your content

BOOTSTRAP — GRID SYSTEM

- It includes predefined classes for easy layout options.



.col-md-1 .col-md-1 .col-md-1 .col-md-1 .col-md-1 .col-md-1 .col-md-1 .col-md-1 .col-md-1 .col-md-1 .col-md-1 .col-md-1

.col-md-2 .col-md-2 .col-md-2 .col-md-2 .col-md-2 .col-md-2

.col-md-3 .col-md-3 .col-md-3 .col-md-3

.col-md-4 .col-md-4 .col-md-4

.col-md-6 .col-md-6

.col-md-5 .col-md-7

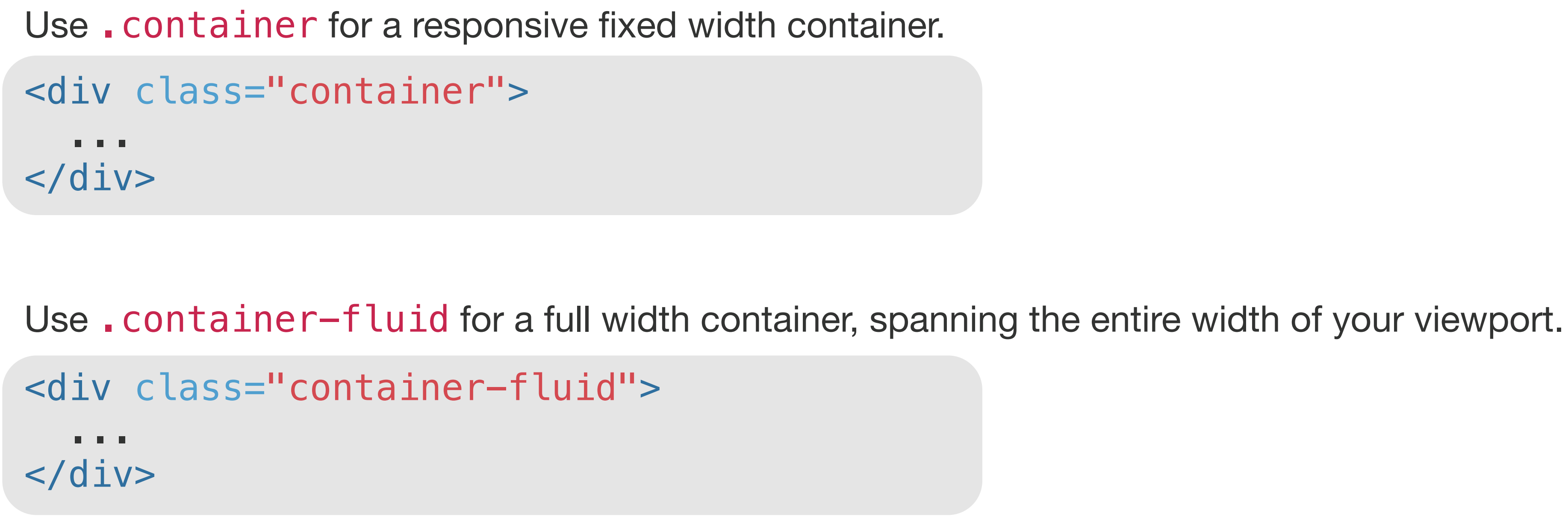
.col-md-4 .col-md-8

BOOTSTRAP — SETTING UP GRID

1. Add containing element to page
2. Add rows to create horizontal groups of columns
3. Place content within columns, specifying the number of twelve available columns you wish to span.

STEP 1 — CONTAINING ELEMENT

- Bootstrap requires a containing element to wrap site contents and house our grid system. You may choose one of two containers to use in your projects:



Use `.container` for a responsive fixed width container.

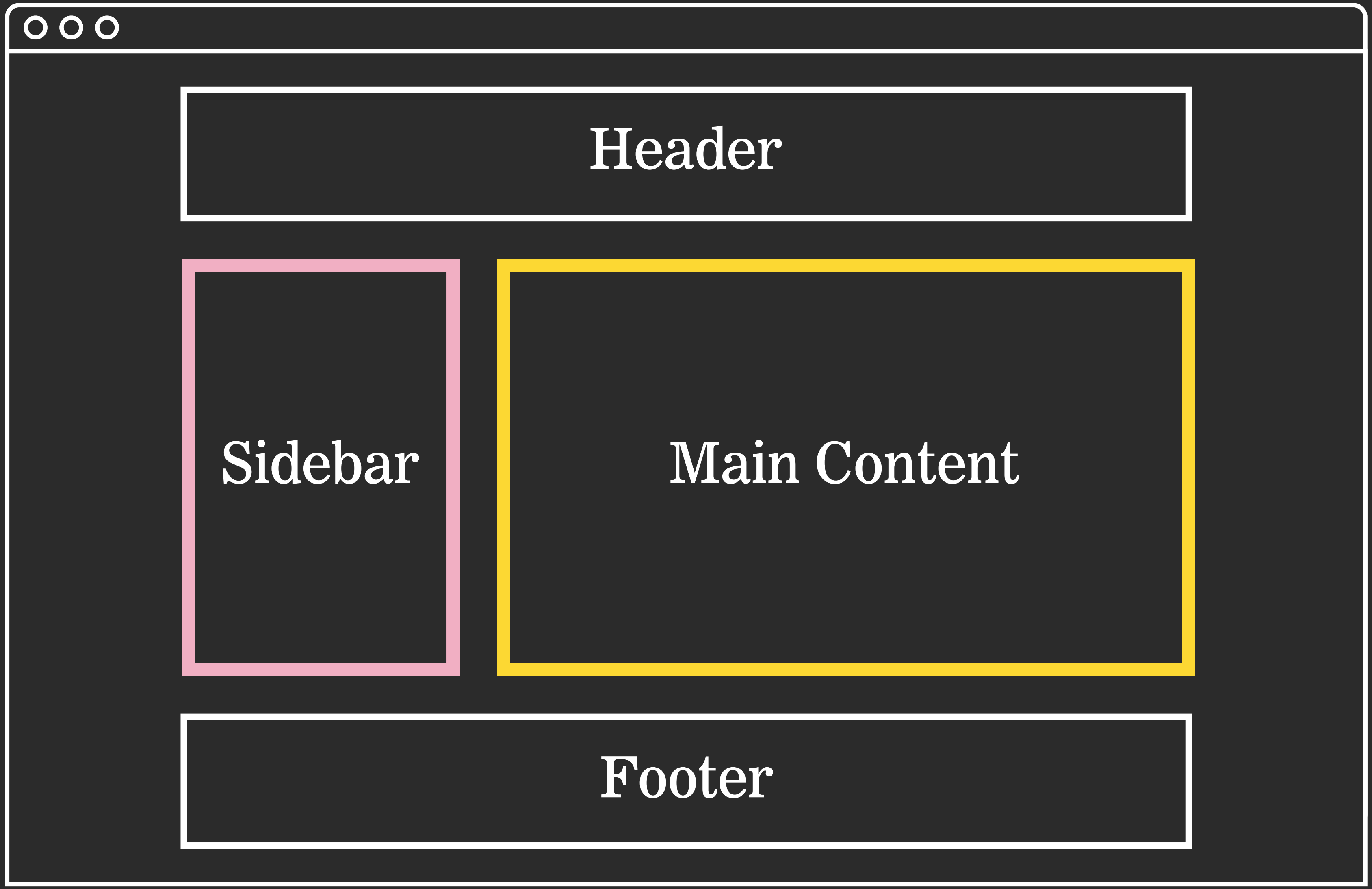
```
<div class="container">
  ...
</div>
```

Use `.container-fluid` for a full width container, spanning the entire width of your viewport.

```
<div class="container-fluid">
  ...
</div>
```

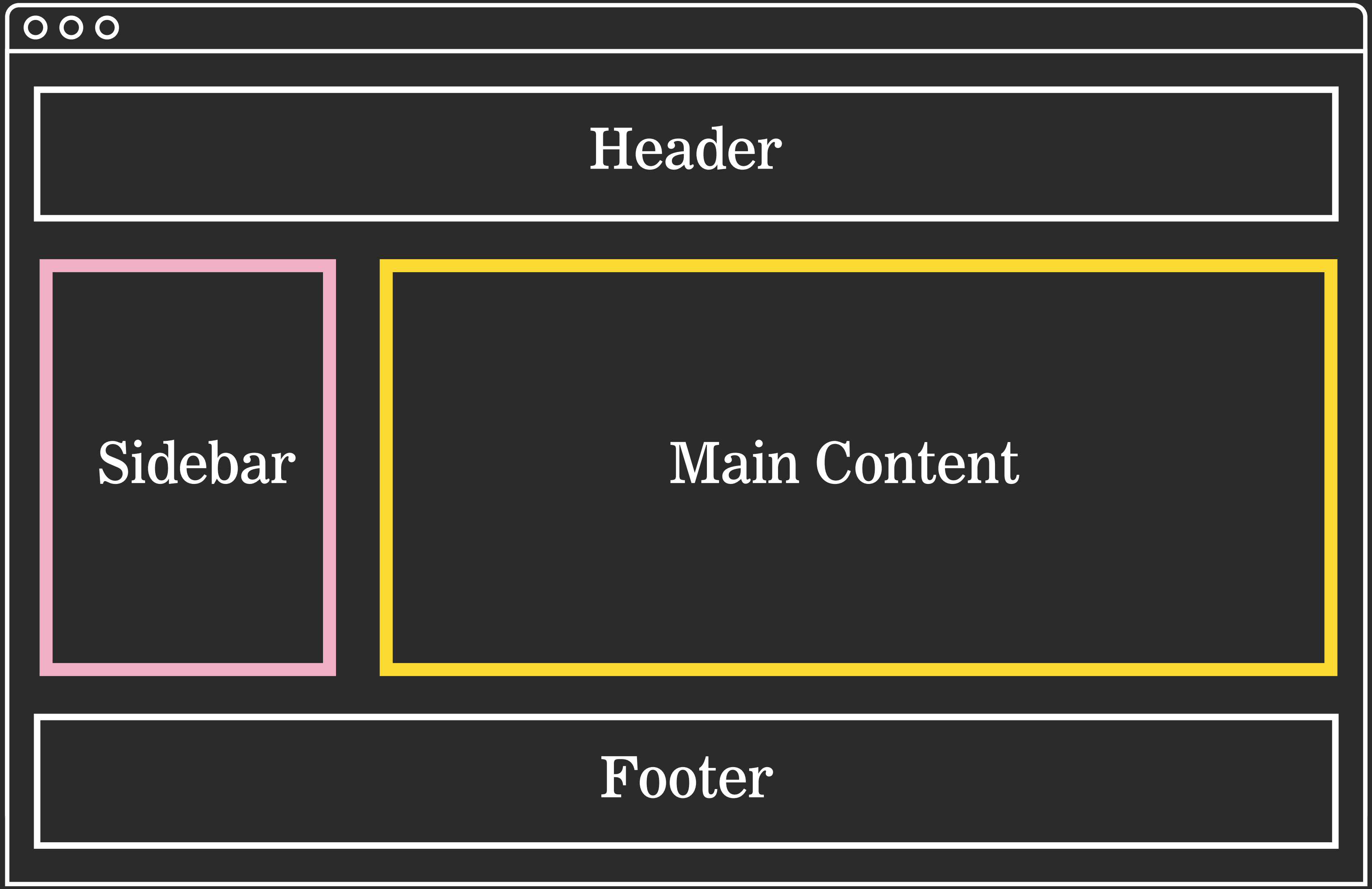
FIXED WIDTH LAYOUT

Fixed width layouts do not change size as the user increases/ decreases width of browser window



FLUID LAYOUT

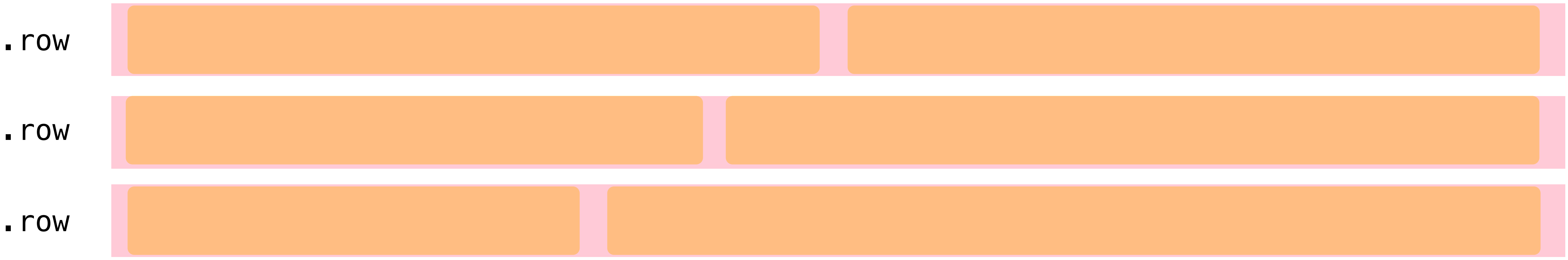
Fluid layouts stretch and contract as the user increases/ decreases the size of their browser window



STEP 2 — ADD ROWS

- Use rows to create horizontal groups of columns.
- Only columns may be immediate children of rows.

```
<div class="row">
    (columns)
</div>
```



STEP 3 — ADD COLUMNS — STACKED TO HORIZONTAL

- Content should be placed within columns.

```
<div class="row">
  <div class="col-md-5">.col-md-5</div>
  <div class="col-md-7">.col-md-7</div>
</div>
<div class="row">
  <div class="col-md-4">.col-md-4</div>
  <div class="col-md-4">.col-md-4</div>
  <div class="col-md-4">.col-md-4</div>
</div>
```

ACTIVITY

EXERCISE

KEY OBJECTIVE

- Get practice creating rows and columns.

TYPE OF EXERCISE

- Individual/Partner

LOCATION

- Starter Code > rows_columns

TIMING

10 min

1. Take a look at grid.png.
2. Recreate the grid in your HTML.

ACTIVITY

EXERCISE

KEY OBJECTIVE

- Understand how Bootstrap works

TYPE OF EXERCISE

- Partner

TIMING

4 min

1. With a partner, use Chrome's inspect element to look at the different styles that Bootstrap is adding. See if you can find them in the Bootstrap stylesheet!

RESPONSIVE IMAGES

RESPONSIVE IMAGES

- Images can be made responsive-friendly via the addition of the .img-responsive class. This applies `max-width: 100%;`, `height: auto;` and `display: block;` to the image so that it scales nicely to the parent element.

```
<img src="..." class="img-responsive" alt="Responsive image">
```

[Codepen Example]

LAB

ACTIVITY

EXERCISE

KEY OBJECTIVE

- Get practice making an image responsive.

TYPE OF EXERCISE

- Individual/Partner

LOCATION

- Starter Code > rows_columns

TIMING

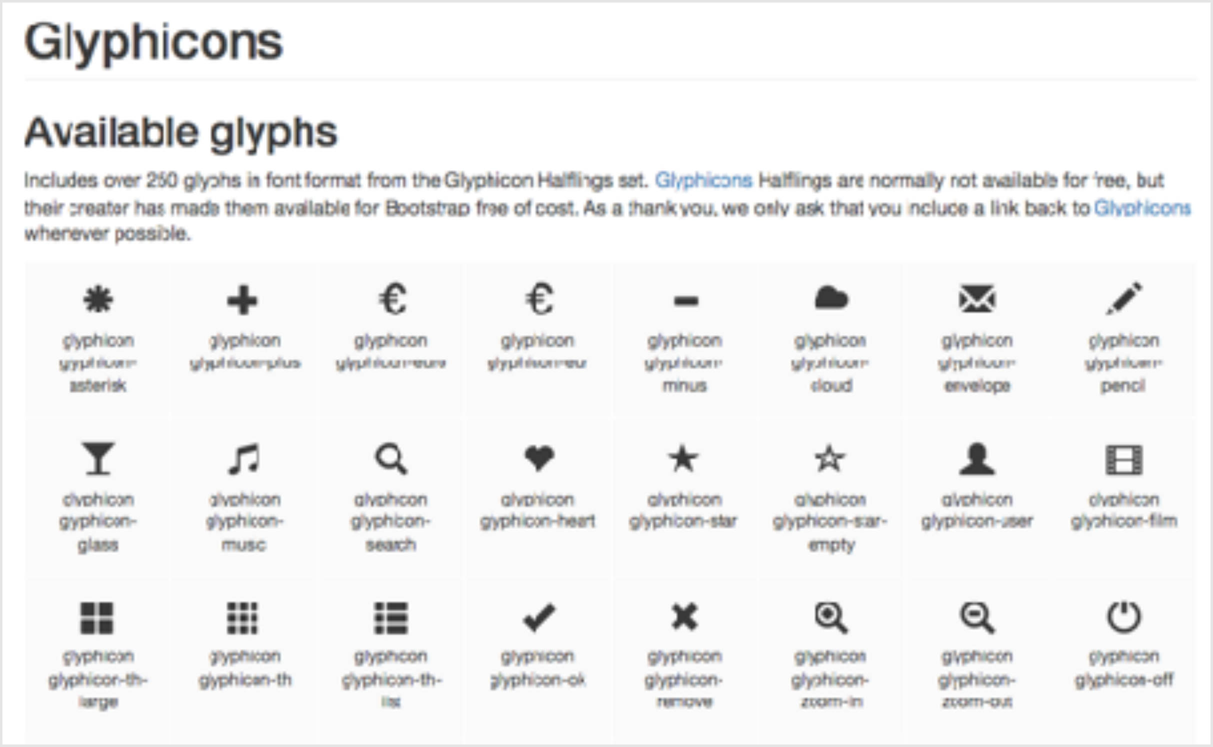
10 min	1. Look at part 2 in index.html

BOOTSTRAP

COMPONENTS

GLYPHICONS

- Includes over 250 icons that are free for use!



Glyphicons

Available glyphs

Includes over 250 glyphs in font format from the Glyphicon Halflings set. Glyphicons Halflings are normally not available for free, but their creator has made them available for Bootstrap free of cost. As a thank you, we only ask that you include a link back to Glyphicons whenever possible.

✱ glyphicon glyphicon-asterisk	✚ glyphicon glyphicon-plus	€ glyphicon glyphicon-euro	€ glyphicon glyphicon-eur	− glyphicon glyphicon-minus	☁ glyphicon glyphicon-cloud	✉ glyphicon glyphicon-envelope	✎ glyphicon glyphicon-pencil
Y glyphicon glyphicon-glass	♫ glyphicon glyphicon-music	🔍 glyphicon glyphicon-search	♥ glyphicon glyphicon-heart	★ glyphicon glyphicon-star	☆ glyphicon glyphicon-star-empty	👤 glyphicon glyphicon-user	🎞 glyphicon glyphicon-film
▦ glyphicon glyphicon-th-large	▦ glyphicon glyphicon-th	☰ glyphicon glyphicon-th-list	✓ glyphicon glyphicon-ok	✖ glyphicon glyphicon-remove	🔍 glyphicon glyphicon-zoom-in	🔍 glyphicon glyphicon-zoom-out	⏻ glyphicon glyphicon-off

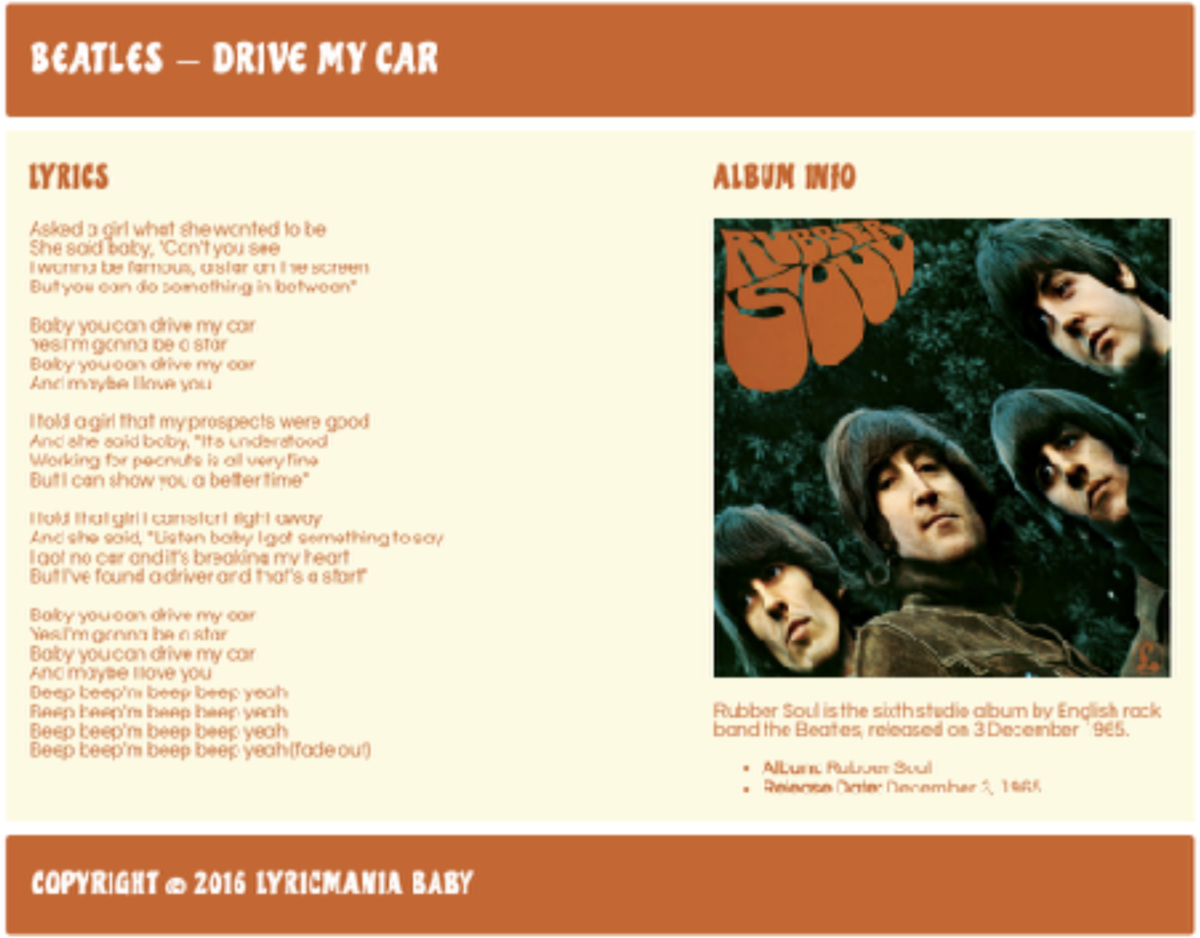
```
<span class="glyphicon glyphicon-heart"></span>
```

♥

Codepen Example

MINI-LAB!

LET'S TAKE A CLOSER LOOK



BEATLES – DRIVE MY CAR

LYRICS

Asked a girl what she wanted to be
She said baby, 'Can't you see
I wanna be famous, a star on the screen
But you can do something in between"

Baby you can drive my car
Yes I'm gonna be a star
Baby you can drive my car
And maybe I love you

I told a girl that my prospects were good
And she said baby, "It's understood
Working for peanuts is all very fine
But I can show you a better time"

I told that girl I can start right away
And she said, "Listen baby I got something to say
I got no car and it's breaking my heart
But I've found a driver and that's a start"

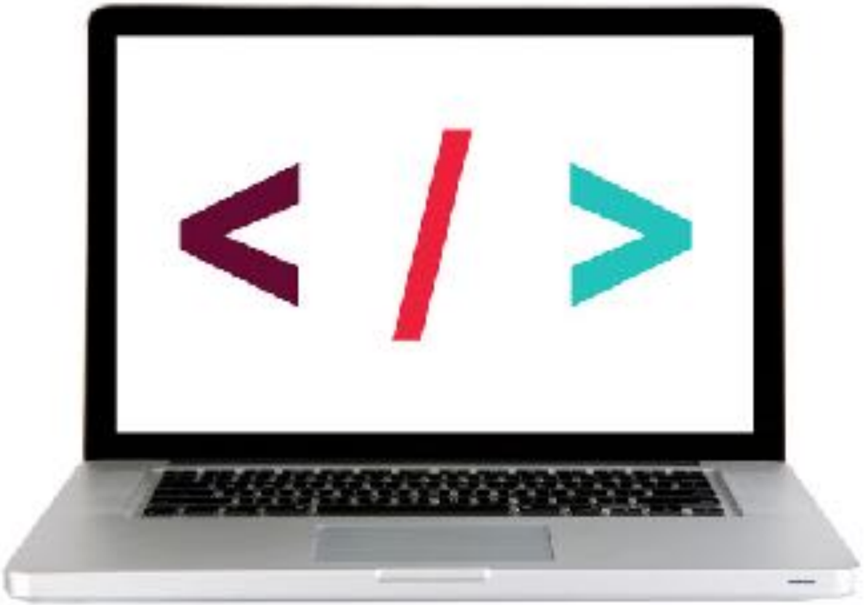
Baby you can drive my car
Yes I'm gonna be a star
Baby you can drive my car
And maybe I love you
Beep beep'm beep beep yeah
Beep beep'm beep beep yeah
Beep beep'm beep beep yeah
Beep beep'm beep beep yeah(fade out)

ALBUM INFO

Rubber Soul is the sixth studio album by English rock band the Beatles, released on 3 December 1965.

- **Album:** Rubber Soul
- **Release Date:** December 3, 1965

COPYRIGHT © 2016 LYRICMANIA BABY



ACTIVITY

EXERCISE

KEY OBJECTIVE

- Get practice creating rows and columns.

TYPE OF EXERCISE

- Individual/Partner

LOCATION

- Starter Code > columns_lab

TIMING

10 min 1. Follow steps 1 - 4 in index.html

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.
- Integrate Bootstrap to create a multi-column layout

WEEKLY OVERVIEW

WEEK 4	Forms & Bootstrap / Intro to JavaScript
WEEK 5	jQuery Basics / jQuery Part II

EXIT TICKETS!