# BIPONNE BRIS

# Help Us Help You!

### Send **Madeline** a Slack message if:

- You know in advance that you will be missing a class or multiple classes
- You're sick, overloaded at work, or something comes up and you can't make it to class last minute
- You will be 10 or more minutes late to class
- You need to leave early
- You will be submitting your homework late

# Send Sarah or Madeline (or both) a Slack message if:

- You are having trouble completing your homework and need some assistance
- You're struggling with the material and need some extra help
- You've finished the homework assignment and would like some bonus tasks

# **RESPONSIVE BASICS**

# **LEARNING OBJECTIVES**

- Describe responsive design.
- Apply media queries to achieve a responsive layout.

# **WEEKLY OVERVIEW**

WEEK 4 Responsive Design / CSS Positioning WEEK 5 Form Basics / Final Project Lab WEEK 6 JS Intro / JS Functions

# **AGENDA**

Review

Responsive — Layout Design

Responsive — Media Queries

Lab

# **RESPONSIVE BASICS**

# REVIEW

# **HOMEWORK** — GROUP DISCUSSION



#### **TYPE OF EXERCISE**

▶ Groups of 3 - 4

#### **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 (background image, font awesome icons). If anyone in your group was able to tackle one of these, share!

#### **REVIEW QUESTIONS**

#### **HTML & CSS REVIEW:**

- ▶ When would we use divs in our HTML?
- ▶ What do classes and IDs do? When would we want to use classes and IDs?
- ▶ What do padding and margin allow us to do? What is the difference between the two?
- ▶ You're trying to add margin to the bottom of some anchors and it's not being applied. What else do we need to add in our CSS to apply margin to the top or bottom of anchors?

#### **CSS GRID:**

- ▶ What should our HTML structure look like to set up CSS grid?
- ▶ How do we initiate CSS Grid? Name the property. Which element would we set this property on (the grid container or grid items)?
- ▶ What else do we need to add in our CSS in order to set up columns?
- ▶ How can we determine how much space goes between each column?

# **FEWD**

# REVIEW

#### DIMENSION – A KEY DIFFERENCE BETWEEN INLINE AND BLOCK ELEMENTS

# If you try to add dimension to an inline element:

- ▶ Padding and margin will only apply to the *left and right*
- ▶ Width and height will have no effect



# DISPLAY — INLINE-BLOCK

Setting an inline element's display to inline-block will allow us to set a width and height, and apply padding and margin to the top and bottom.

#### **BEFORE:**

# My Link

height width

# **BEFORE:**

My Link

```
height
```

```
a {
    display: inline-block;
}
```

# **CSS GRID**

# **Grid Container**

# **CSS GRID**

# **DISPLAY**

# To initiate the "Grid Magic"

```
.container {
  display: grid;
}
```

# **GRID-TEMPLATE-COLUMNS**

```
.container {
   display: grid;
   grid-template-columns: 3fr 1fr;
   grid-gap: 20px;
}
```

**Grid Container** 



# **ACTIVITY**



#### **KEY OBJECTIVE**

▶ Review HTML/CSS Layouts

#### TYPE OF EXERCISE

Individual/Partner

#### **TIMING**

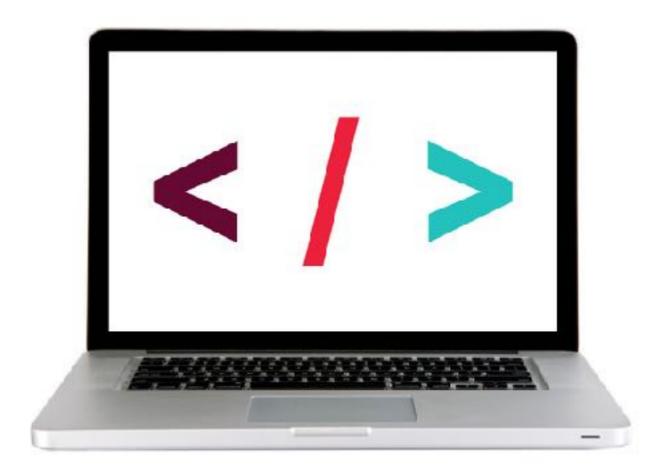
15 *min* 

1. Use HTML and CSS to recreate albums\_1.png

#### **RESPONSIVE BASICS**

# RESPONSIVE — LAYOUT DESIGN

# **LET'S TAKE A CLOSER LOOK**



https://generalassemb.ly/

# **RESPONSIVE DESIGN**

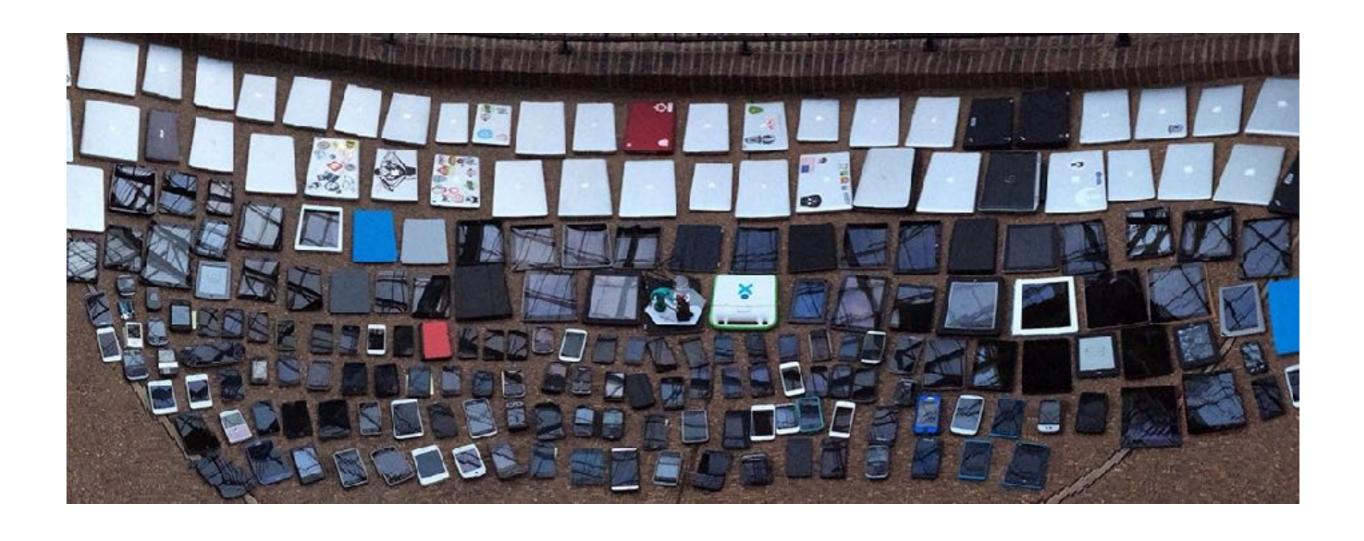
"Day by day, the number of devices, platforms, and browsers that need to work with your site grows. Responsive web design represents a fundamental shift in how we'll build websites for the decade to come."

- Jeffrey Veen

# **RESPONSIVE DESIGN**



# **RESPONSIVE DESIGN**



# **SLACK RESPONSE**

- ▶ What makes a site responsive?
- What are some considerations we might want to make when updating the layout of a page for mobile?

#### **RESPONSIVE BASICS**

# RESPONSIVE — MEDIA QUERIES

# **ACTIVITY**



#### **KEY OBJECTIVE**

Review Grid / Preview Media Queries

#### **TYPE OF EXERCISE**

Individual/Partner

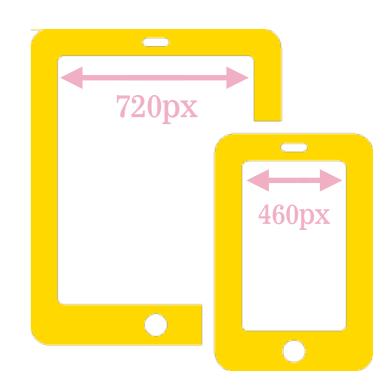
#### **TIMING**

15 *min* 

1. Look at this <u>Codepen</u> with a partner and talk through steps 1 - 3.

# **MEDIA QUERIES**

With media queries, we can allow most of our styles to remain the same, while we make **small tweaks for specific formats.** 



# **ADDING MEDIA QUERIES**

```
@media screen and (max-width: 600px) {
    section {
        grid-template-columns: repeat(2, 1fr);
    }
}
```

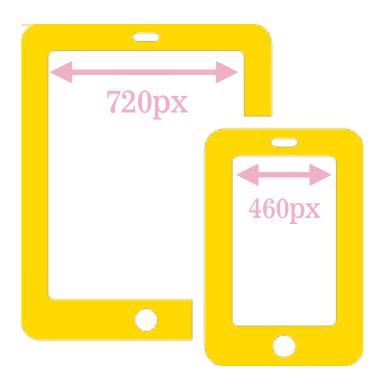
# **MEDIA QUERIES** — SYNTAX

# STYLES FOR COMPUTER SCREENS @media 'screen' and (max-width: 600px) { section { grid-template-columns: repeat(2, 1fr); **STYLES FOR PRINT** @media print { /\* All your print styles go here \*/ header, footer, nav { display: none;

# **MEDIA QUERIES** — SYNTAX

#### MAX-WIDTH: SCREENS 600PX OR LESS

```
@media screen and (max-width: 600px) {
     section {
         grid-template-columns: repeat(2, 1fr);
             MIN-WIDTH: SCREENS 600PX OR MORE
@media screen and (min-width: 600px) {
      section {
          grid-template-columns: repeat(2, 1fr);
```



#### **DESIGNING RESPONSIVELY**

Changing the number of columns:

```
section {
    grid-template-columns: repeat(4, 1fr);
@media screen and (max-width: 600px) {
     section {
         grid-template-columns: repeat(2, 1fr);
@media screen and (max-width: 400px) {
     section {
         grid-template-columns: repeat(1, 1fr);
```

### MEDIA QUERIES — ORDER

- 1. Place your media queries at the bottom of your stylesheet so they can override code that comes before them.
- 2. If you have multiple media queries, they should go from largest to smallest (max-width: 600 would be placed above max-width: 400)

```
@media screen and (max-width: 600px) {
    section {
        grid-template-columns: repeat(2, 1fr);
    }
}

@media screen and (max-width: 400px) {
    section {
        grid-template-columns: repeat(1, 1fr);
    }
}
```

# **ACTIVITY**



#### **KEY OBJECTIVE**

▶ Apply media queries to achieve a responsive layout.

#### TYPE OF EXERCISE

Individual / Partner

#### **TIMING**

25 min

- 1. Use media queries to update the number of columns for the screensizes noted in the album\_2 images.
- 2. Make sure to add your media queries at the bottom of your stylesheet!!!

# **DESIGNING RESPONSIVELY**

We may also want to show or hide content for different screen widths.

To hide content:

```
a {
    display: none;
}
```

To show content:

```
a {
   display: inline-block;
}

a {
   display: block;
}
```

# **DESIGNING RESPONSIVELY**

Showing and hiding content for mobile:

```
.hamburger {
    display: none;
}

@media screen and (max-width: 600px) {
    .hamburger {
        display: block;
    }
}
```

# **ACTIVITY**



#### **KEY OBJECTIVE**

▶ Apply media queries to achieve a responsive layout.

#### TYPE OF EXERCISE

Individual / Partner

#### **TIMING**

25 min

- 1. First recreate the layout in albums\_3.png for screens 660px and less
- 2. Then recreate the layout in albums\_4.png for screens 660px and less

# **DESIGNING RESPONSIVELY**

To update the size of text:

```
h1{
    font-size: 50px;
}

@media screen and (max-width: 600px) {
    h1{
        font-size: 30px;
    }
}
```

# **ACTIVITY**



#### **KEY OBJECTIVE**

▶ Apply media queries to achieve a responsive layout.

#### TYPE OF EXERCISE

Individual / Partner

#### **TIMING**

25 *min* 

- 1. Change the font-size of the h1 to 40px for screens 500px and narrower.
- 2. Change the font-size for the paragraph in the header to 17px for screens 500px and narrower.

#### **DESIGNING RESPONSIVELY**

It's also important to think about what a site will look like on really wide screens!

To make sure that content doesn't stretch to be too wide:

```
main {
    max-width: 1000px;
    margin: 0 auto;
}
```

# **ACTIVITY**



#### **KEY OBJECTIVE**

▶ Apply media queries to achieve a responsive layout.

#### **TYPE OF EXERCISE**

Individual / Partner

#### **TIMING**

25 min

1. Try it out in this **CodePen!** 

#### **TURN AND CHAT**

- ▶ What are media queries? What can we change inside media queries?
- ▶ Where should we add media queries? Why is this order important?
- I have the two following media queries. How should these be rearranged? Can anything be removed?

```
@media screen and (max-width: 400px) {
    .hamburger {
          display: block;
    }
    section {
          grid-template-columns: repeat(1, 1fr);
    }
}

@media screen and (max-width: 600px) {
    .hamburger {
          display: block;
    }
    section {
          grid-template-columns: repeat(2, 1fr);
    }
}
```

#### **VIEWPORT META TAG — AN IMPORTANT NOTE!!**

- ▶ The viewport meta tag controls how a webpage is displayed on a mobile device.
- Without the tag, mobile devices will assume you want the full desktop experience and will set the viewport width at 980px (iOS)

#### **DEVICE-WIDTH**

▶ This tells the browser "My Website adapts to your width"

#### INITIAL-SCALE

Sets the initial zoom level and prevents default zooming

<meta name="viewport" content="width=device-width, initial-scale=1">

# **RESPONSIVE BASICS**

# LAB

### **ACTIVITY**



#### **KEY OBJECTIVE**

Practice applying media queries to achieve a responsive layout.

#### TYPE OF EXERCISE

Individual/Partner

#### **TIMING**

5 min

1. Spend 5 minutes sketching out the responsive version of the Monument Landing page with a group

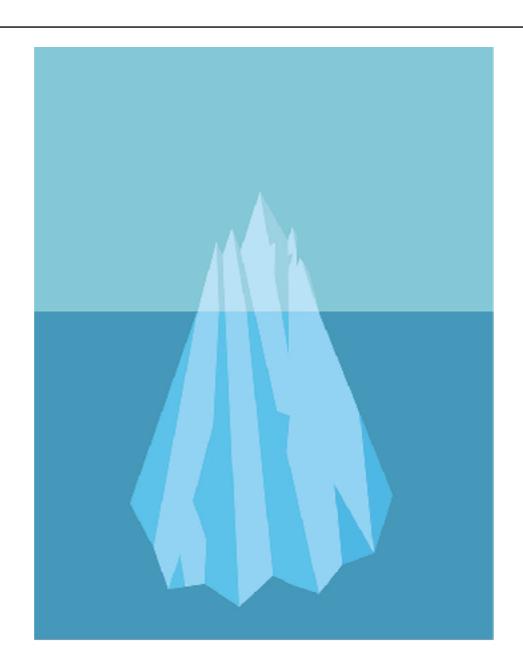
20 min

2. Write media queries at the bottom of your CSS file to create a responsive version of the Monument Landing Page.

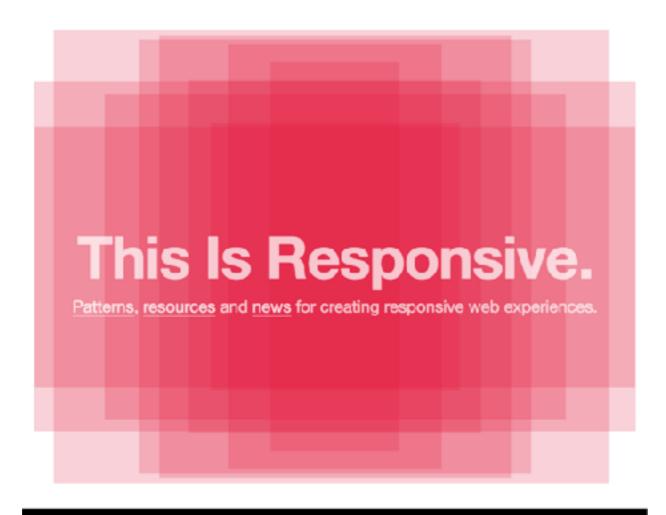
#### **RESPONSIVE BASICS**

# MORE RESOURCES

# **MORE RESOURCES**



### MORE RESOURCES — THIS IS RESPONSIVE



### MORE RESOURCES — MEDIA QUERIES

diagrani.45



#### FROONT







#### International Polar Foundation

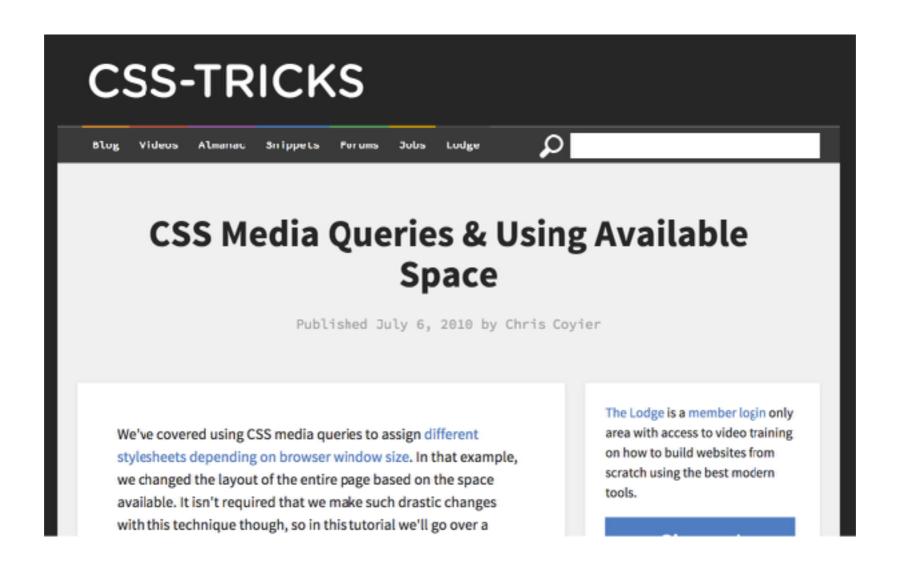






0

#### MORE RESOURCES — MEDIA QUERIES



# **WEEKLY OVERVIEW**

WEEK 4 Responsive Design / CSS Positioning WEEK 5 Form Basics / Final Project Lab WEEK 6 JS Intro / JS Functions

## **RESPONSIVE BASICS**

# **LEARNING OBJECTIVES**

- Describe responsive design.
- Apply media queries to achieve a responsive layout.

#### **HTML BASICS**

# EXIT TICKETS