# CSS Layout and the Box Model

Skills Bootcamp in Front-End Web Development

Lesson 1.3

WELCOME

# Today’s Objectives

By the end of class, students will be able to:

- Set an element as an inline or block element using the CSS **`display`** property.
- Determine how an element is positioned in an HTML document using the CSS `position` property.
- Identify the box model and its role in web design and layout.

# Stoke Curiosity

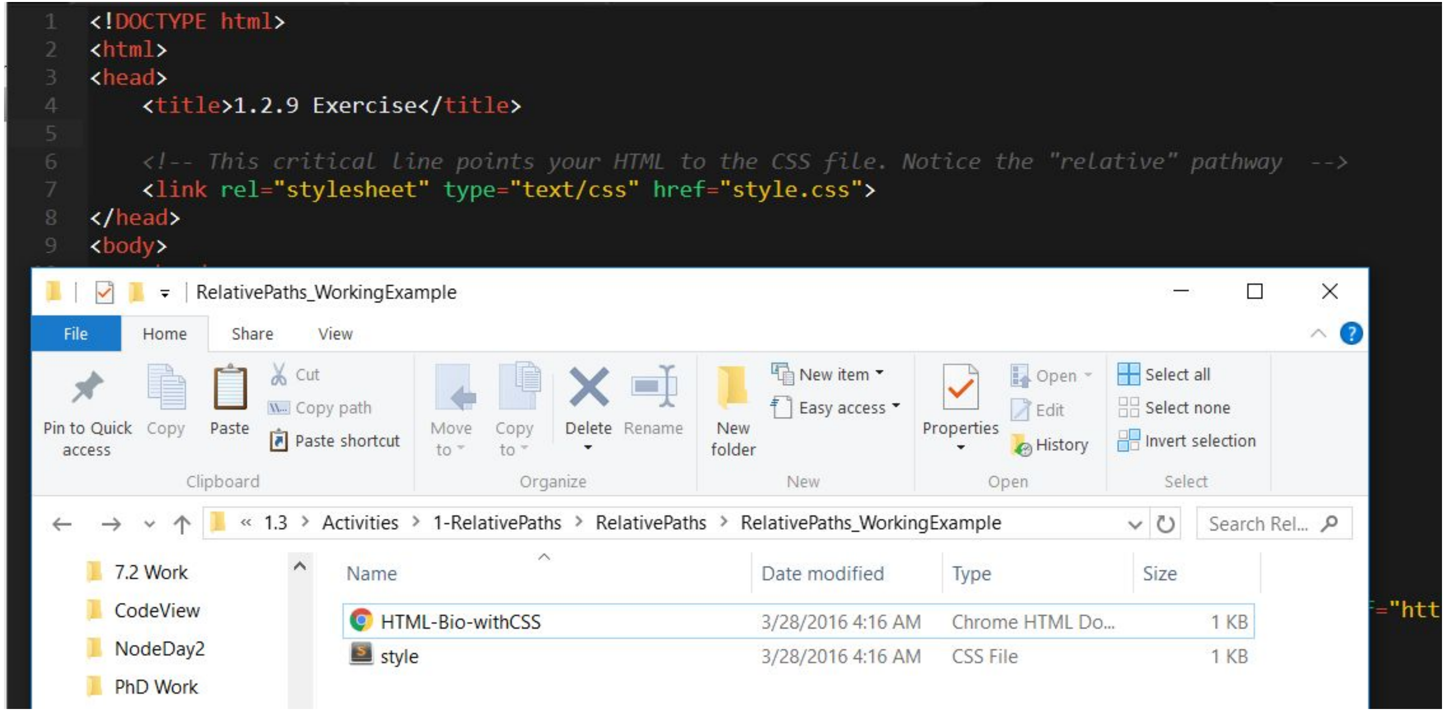
# Instructor Demonstration

## Chrome Inspector

# Relative File Paths

# Relative File Paths

**Relative file paths** connect us with other files in our working directory. In this case, style.css is in the same folder as our HTML document.



```
<!DOCTYPE html>
<html>
<head>
    <title>1.2.9 Exercise</title>

    <!-- This critical line points your HTML to the CSS file. Notice the "relative" pathway  -->
    <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
```

| Name | Date modified | Type | Size |
|---|---|---|---|
| HTML-Bio-withCSS | 3/28/2016 4:16 AM | Chrome HTML Do... | 1 KB |
| style | 3/28/2016 4:16 AM | CSS File | 1 KB |

# Instructor Demonstration

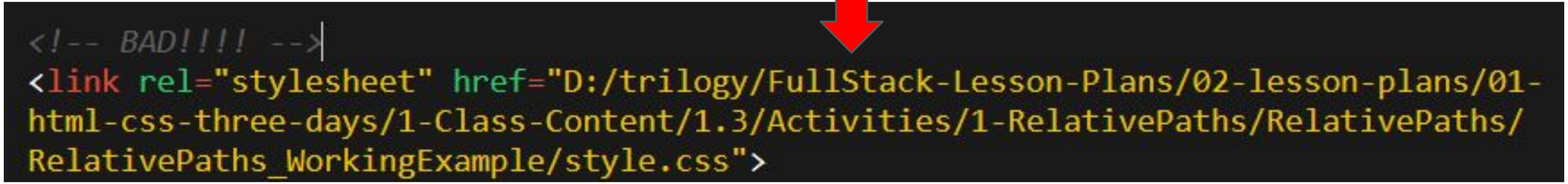
## Relative File Paths

# Absolutely No Absolute Paths

Always use relative file paths!

- If you deploy websites without relative file paths, **all of your links will fail**.
- The same will happen if you move your project from one folder to another.
- Remember, there is no such thing as a C: drive on the internet.

**VERY, VERY BAD**



```
<!-- BAD!!!! -->
<link rel="stylesheet" href="D:/trilogy/FullStack-Lesson-Plans/02-lesson-plans/01-
html-css-three-days/1-Class-Content/1.3/Activities/1-RelativePaths/RelativePaths/
RelativePaths_WorkingExample/style.css">
```

# Activity: Relative File Paths Activity

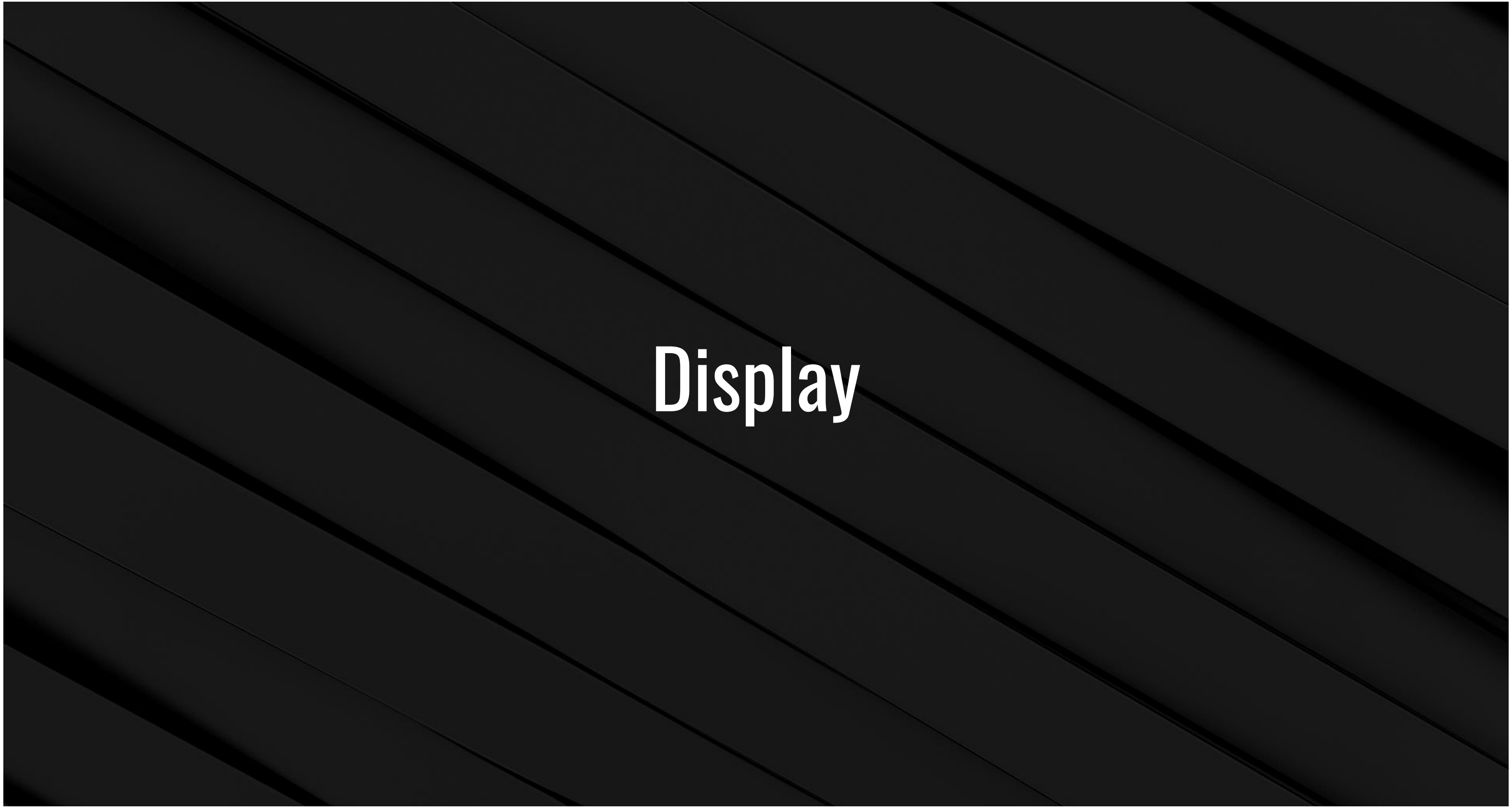
Instructions:

- Unzip the folder provided to you (or copy and paste the contents outside).
- Then modify each of the four `html-bio.html` pages such that they can access the CSS inside their folder. **Don't move the CSS file, and don't move the HTML file.**
- Use relative linking to make it work!
- **Hint:** If you need some reading material on relative linking, you can use [https://css-tricks.com/quick-reminder-about-file-paths/](https://css-tricks.com/quick-reminder-about-file-paths/).

Suggested Time:

15 minutes

# Time’s Up! Let’s Review.



# Display

# Instructor Demonstration

## Display

# Partner Activity: Display Activity

In this activity, you’ll work with a partner to resolve issues within the given code

Suggested Time:

10 minutes

# Time’s Up! Let’s Review.

Break

# Box Model

**CSS box model:** Every HTML element you create is represented as a rectangular box, with the box's content, padding, border, and margin built up around one another like the layers of an onion!

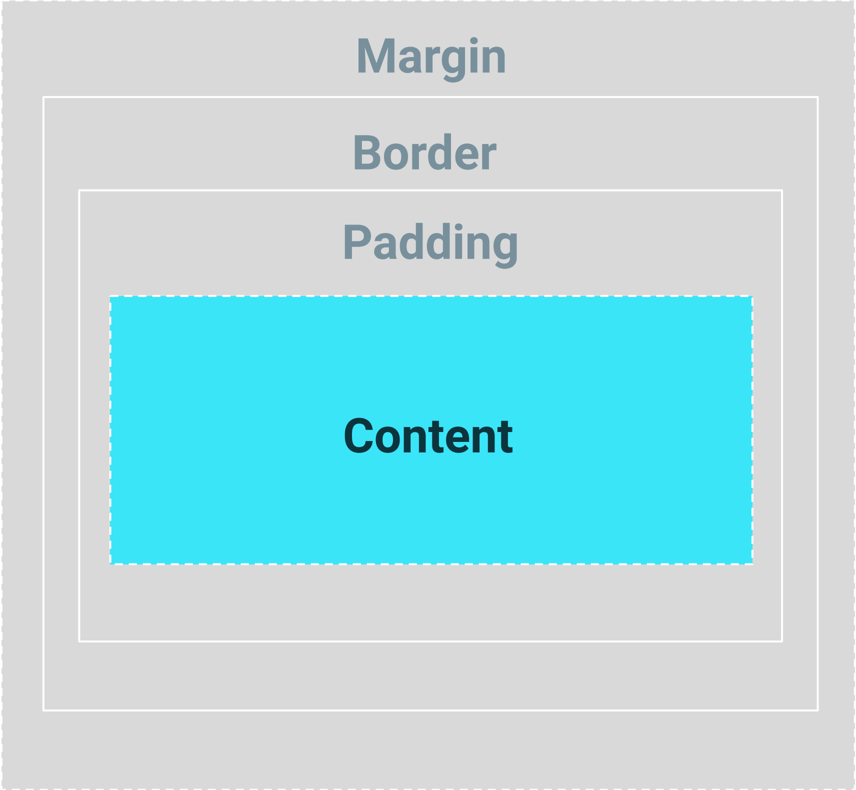
# The CSS Box Model

**Content:** This is what's inside a container, such as text and images.

**Padding:** Padding is used to expand the space *inside* a box, between our content and our border.

**Border:** A border surrounds your content and distinguishes it from other elements around it.

**Margin:** Margin is used to create additional space *outside* of your border. Margin spacing increases the distance between your box and other elements on the page.



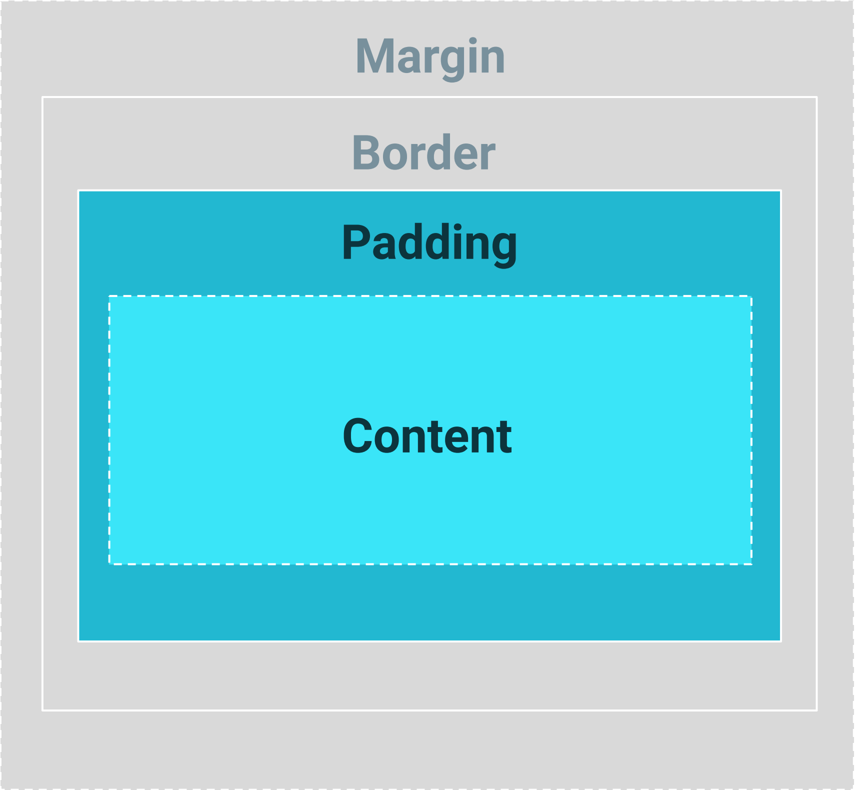
# The CSS Box Model

**Content:** This is what's inside a container, such as text and images.

**Padding:** Padding is used to expand the space *inside* a box, between our content and our border.

**Border:** A border surrounds your content and distinguishes it from other elements around it.

**Margin:** Margin is used to create additional space *outside* of your border. Margin spacing increases the distance between your box and other elements on the page.



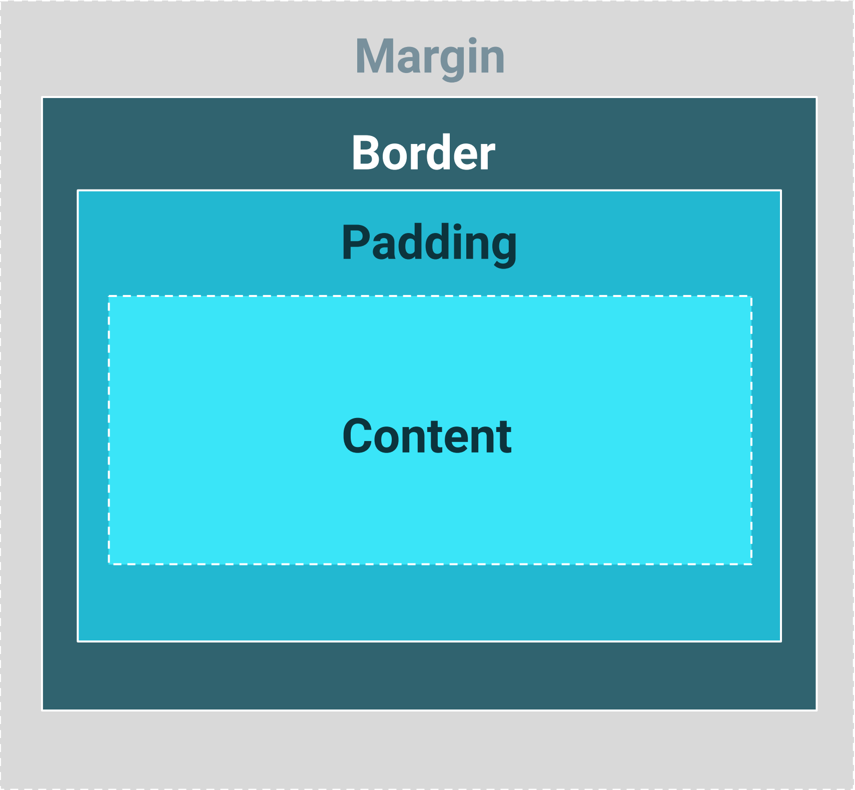
# The CSS Box Model

**Content:** This is what's inside a container, such as text and images.

**Padding:** Padding is used to expand the space *inside* a box, between our content and our border.

**Border:** A border surrounds your content and distinguishes it from other elements around it.

**Margin:** Margin is used to create additional space *outside* of your border. Margin spacing increases the distance between your box and other elements on the page.



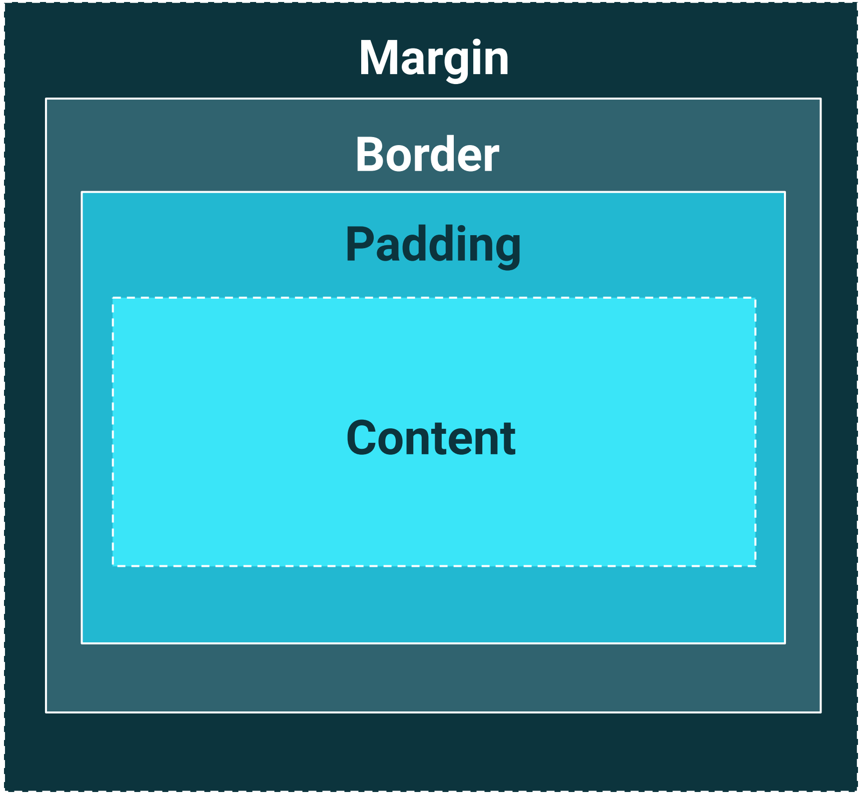
# The CSS Box Model

**Content:** This is what's inside a container, such as text and images.

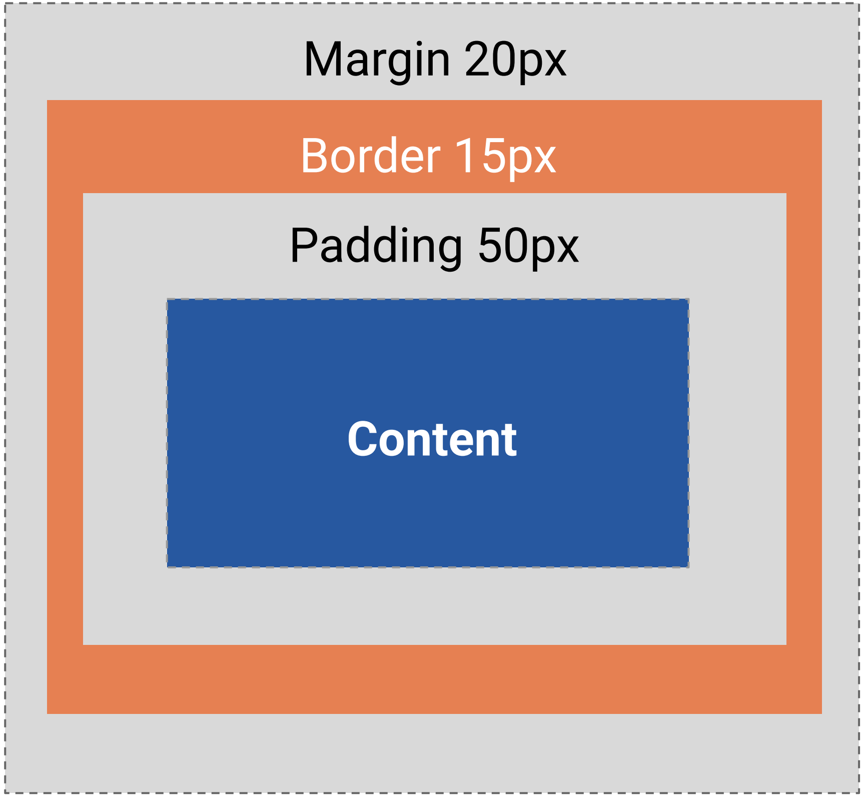
**Padding:** Padding is used to expand the space *inside* a box, between our content and our border.

**Border:** A border surrounds your content and distinguishes it from other elements around it.

**Margin:** Margin is used to create additional space *outside* of your border. Margin spacing increases the distance between your box and other elements on the page.



# The CSS Box Model



```
div {
    background-color: navy blue;
    padding: 50px;
    border: 15px orange;
    margin: 20px;
}
```

The box model applies to all elements, whether they are text, images, div sections, etc.

# Partner Activity: Box Model Activity

Work with a partner to implement the following user story:

As a developer, I want to use the CSS box model properties to position four boxes inside a frame

Suggested Time:

10 minutes

# Time’s Up! Let’s Review.

# Positioning

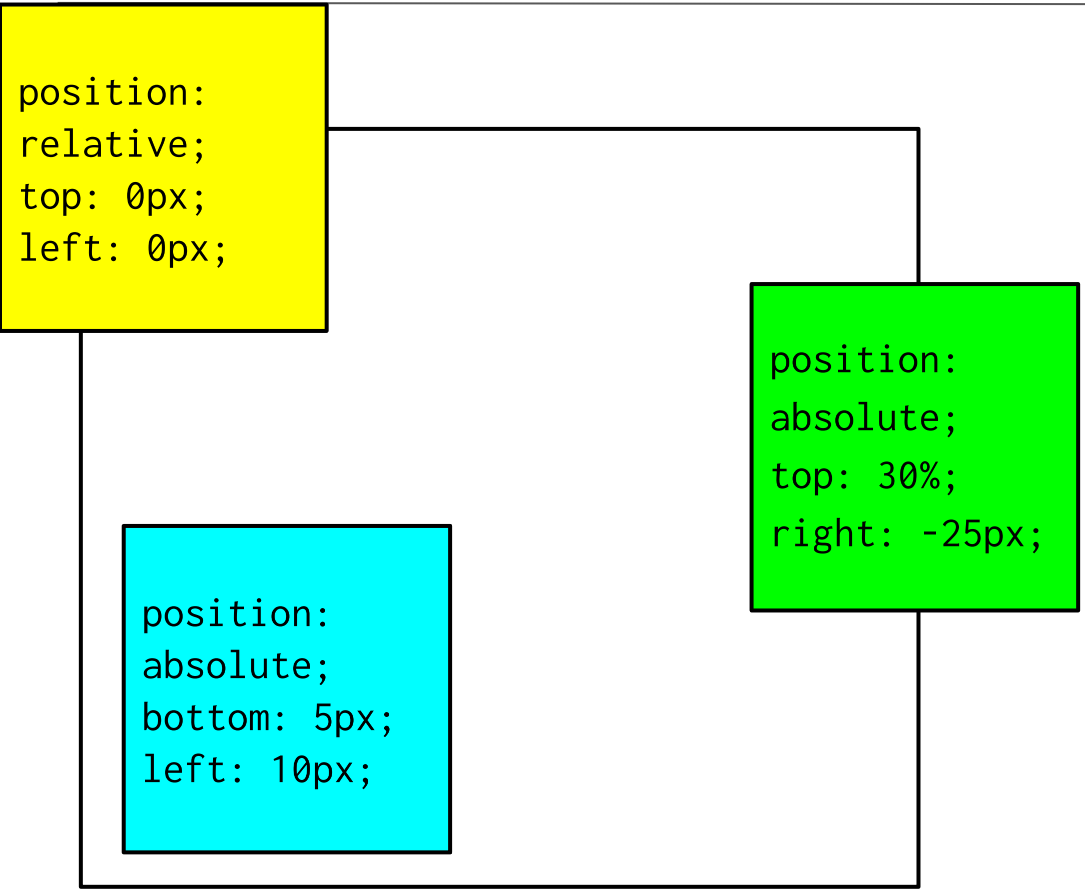
# Position: Static

Positions an element according to the normal flow of the document.

```
.myClass {
position: static;
}
```

**Static is the default position property applied.**

# position: static;



position:
relative;
top: 0px;
left: 0px;

position:
absolute;
top: 30%;
right: -25px;

position:
absolute;
bottom: 5px;
left: 10px;

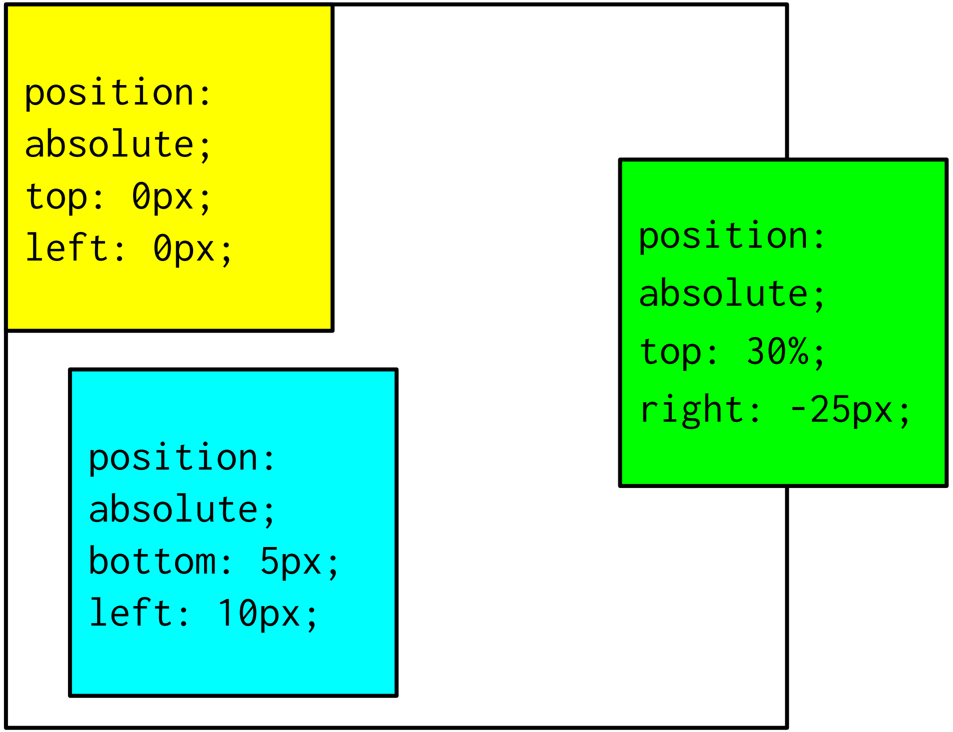
```
position: relative;
top: 0px; left: 0px;

position: absolute;
bottom: 5px; left: 10px;

position: absolute;
top: 30%; right: -25px;
```

# Position: Relative

Positions the element relative to the normal position it would otherwise have (e.g., if it were left as a static element). Used to offset an element from its default, based on any values assigned (top, right, bottom, left).

```
.myClass {
position: relative;
}
```

# position: relative;



```
position:
absolute;
top: 0px;
left: 0px;
```

```
position:
absolute;
top: 30%;
right: -25px;
```

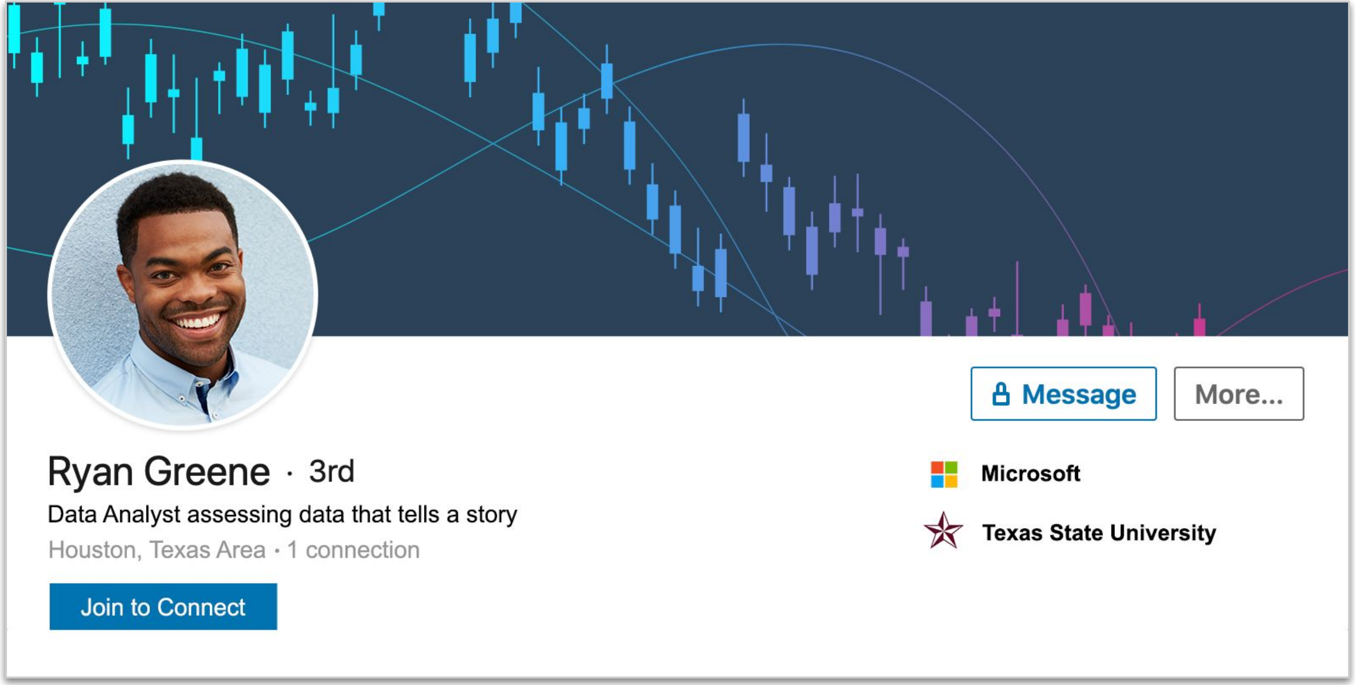
```
position:
absolute;
bottom: 5px;
left: 10px;
```

```
position: absolute;
top: 0px; left: 0px;

position: absolute;
bottom: 5px; left: 10px;

position: absolute;
top: 30%; right: -25px;
```

Switching the boxes to relative will nudge the boxes in relation to their “original” location.

# Position: Relative

An example of a relative element is the container for any profile picture that overlaps other HTML elements. Think of the container for the profile picture for LinkedIn.



Ryan Greene · 3rd

Data Analyst assessing data that tells a story

Houston, Texas Area · 1 connection

Microsoft

Texas State University

# Position: Absolute

Absolute elements are removed from the document flow and are placed in absolute position based either on their parent container or the overall document body.

```
.myClass {
position: absolute;
}
```

# Absolute Positioning vs. Relative Positioning

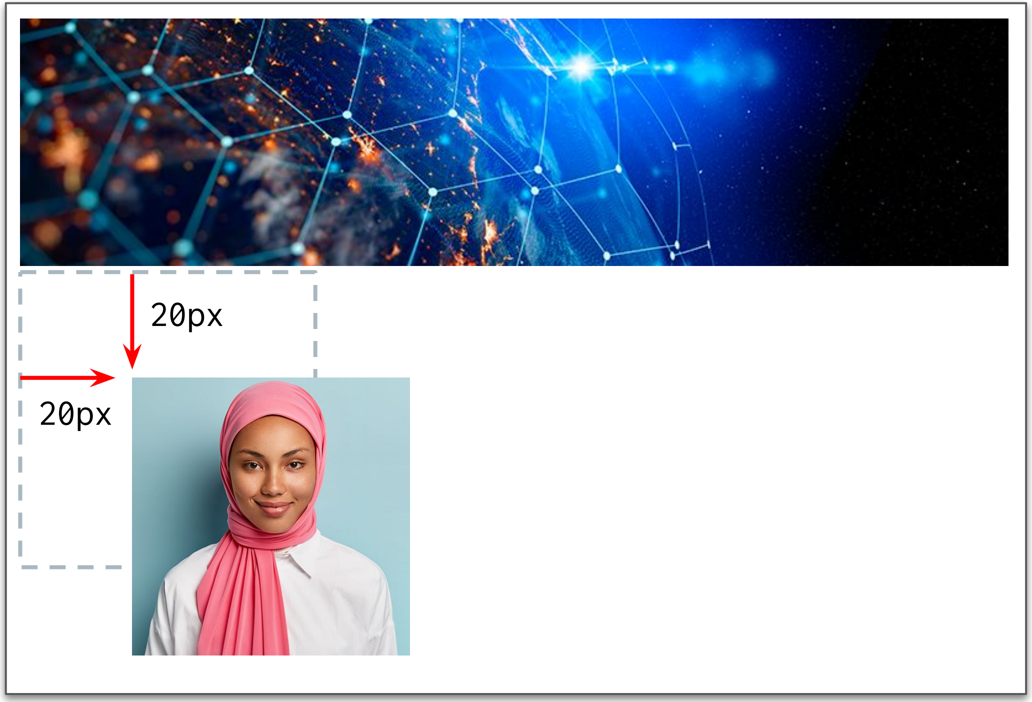
## Relative

Relative positioning is just like stating no positioning at all! The left, right, top, and bottom attributes “nudge” elements out of their normal layout.

## Absolute

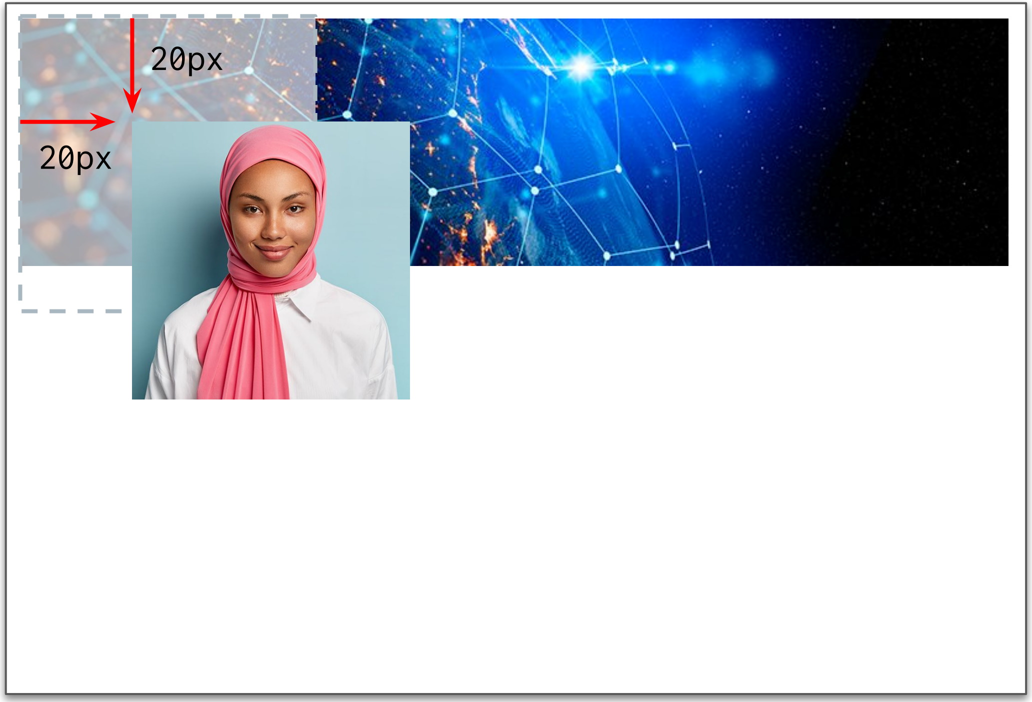
Absolute positioning allows you to place your element *precisely* where you want it—and it won’t budge.

# position: relative;



```
position: relative;
top: 20px;
left: 20px;
```

# position: absolute;



```
position: absolute;
top: 20px;
left: 20px;
```

# Position: Fixed

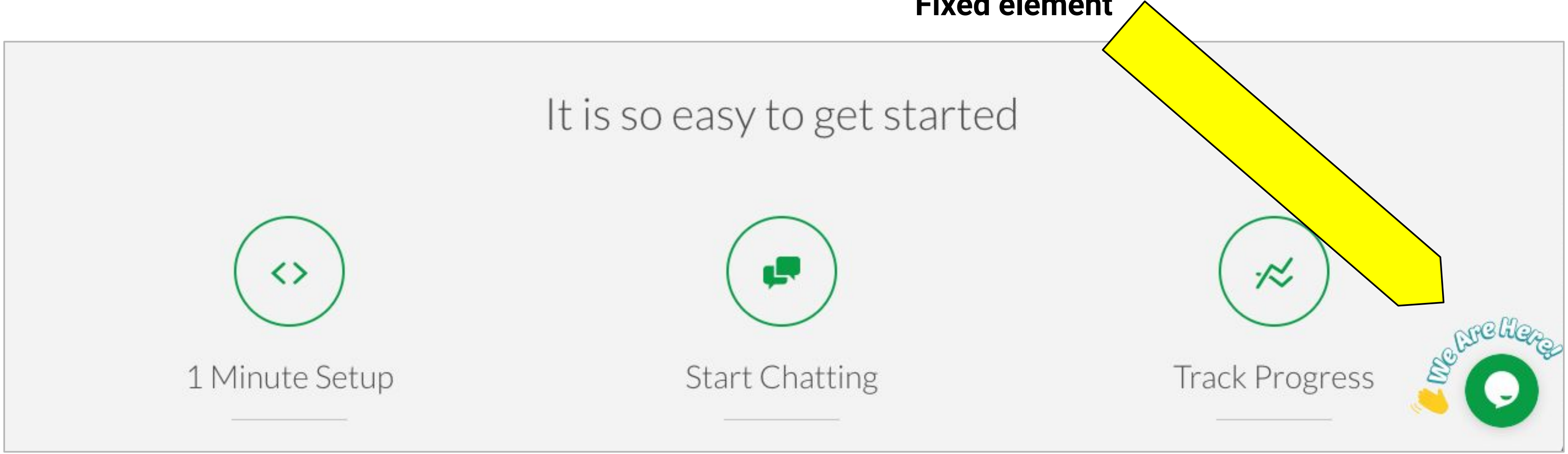
The element is removed from the normal document flow. It will always stay where you put it, even if a user scrolls down the page.

```css
.myClass {
position: fixed;
}
```

# Position: Fixed

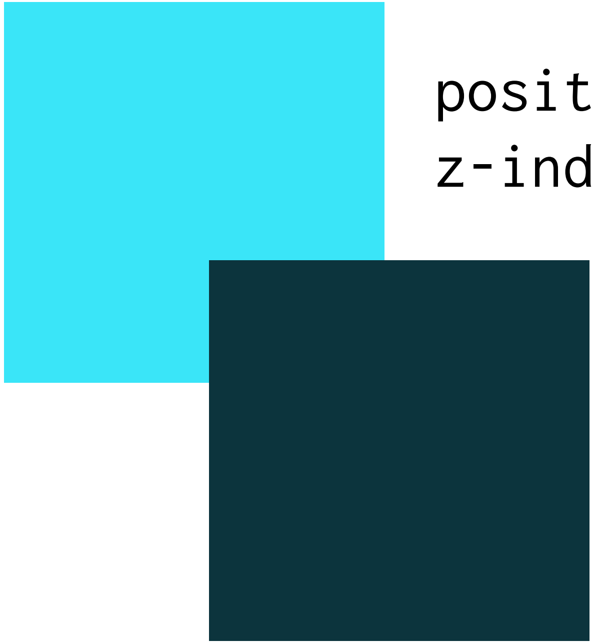
An example of a fixed element might be a “Contact us” button that follows you as you scroll.

Fixed element



# Layering with z-Index

The z-index property allows you to layer elements on top of each other.



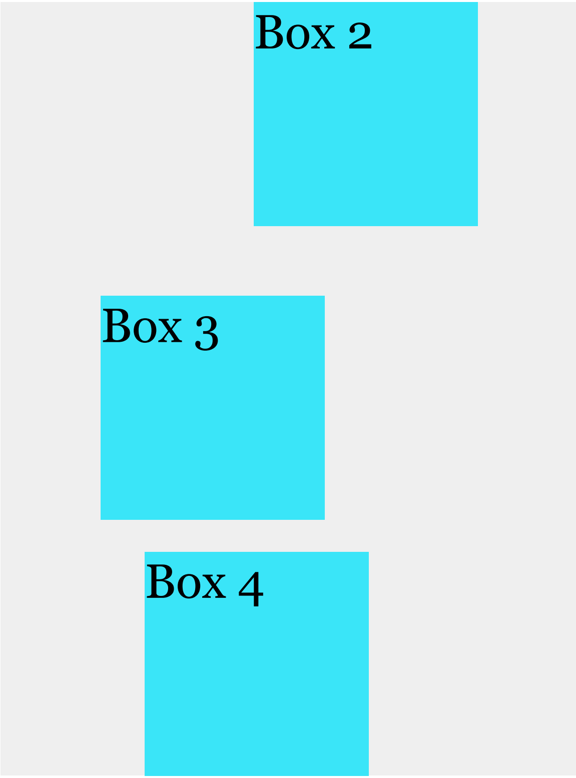
```
position: absolute;
z-index:1;
```

```
position: absolute;
z-index:2;
```

# Hiding Things

Display: none allows you to hide elements from view.

This will become useful in later sections, when we'll hide and reveal specific HTML elements of our choosing.



# Instructor Demonstration

## Positioning

# Partner Activity: Positioning Activity

Work with a partner to implement the following user story:

As a developer, I want to use the CSS position property to change the layout of my page.

Suggested Time:

10 minutes

# Time's Up! Let's Review.

# Mini-Project: Landing Page

# Instructor Demonstration

## Mini-Project: Landing Page

# Mini-Project: Landing Page

Instructions:

- Unzip the folder provided to you (or copy and paste the contents outside).
- Follow the provided instructions to complete this project

Suggested Time:

30 minutes

# Time’s Up! Let’s Review.

# Questions?

# RECAP

# The End