Topic	CSS and Flexbox
Class Description	**Students learn how to use flexbox to design the layout of a website. Students also learn the importance of keeping content and styling of a website separate. Students use a separate stylesheet for adding design to the portfolio website which they are creating.**
Class	**C50**
Class time	**45 mins**
Goal	● Create a separate CSS style sheet for the portfolio website. ● Use <div> blocks and flexbox to give a layout to the website. ● Use css selectors to add style elements to the website.

Resources Required	● Teacher Resources
	   ○ Laptop with internet connectivity
	   ○ Earphones with mic
	   ○ Notebook and pen
	
	● Student Resources
	   ○ Laptop with internet connectivity
	   ○ Earphones with mic
	   ○ Notebook and pen

Class structure	**Warm Up** **5 mins** **Teacher-led Activity** **10 min** **Student-led Activity** **20 min** **Wrap up** **5 min**

CONTEXT

- **Every html content is a box. These boxes can be laid down in a defined layout using flexbox.**
- **Importance of keeping style and content of a website separately**

Class Steps	Teacher Action	Student Action
Step 1: Warm Up (5 mins)	Hi. Welcome again. Remember, what we did in the last class?	We wrote some HTML content using html tags and added style to it.

		We learned how html elements can be styled.
	Remember, we are learning some HTML and CSS because it will help us in designing mobile apps using React. Today, we will be learning how to give layout to our website. Remember, last class we saw how every html content is a box and these boxes can be arranged using style to give any layout to our website. We will also be adding some styling to the website. Ready?	ESR: yes!
Teacher Initiates Screen Share	
<u>**CHALLENGE**</u> ● **Arrange the layout of different HTML elements using flexbox.**	
Step 2: Teacher-led Activity (15 min)	Teacher opens the activity from the last class or clones it from Github. Can you quickly review the code from the previous class? Review different html tags and what it is used for	The student reviews code from the previous class and explains it. Student explains the different html tags used in the document.

```
<> index.html ×

home > rajeev > Portfolio > <> index.html > 🔗 html > 🔗 body
    1    <!DOCTYPE HTML>
    2    <html>
    3        <head>
    4            <title>My Portfolio page</title>
    5            <style>
    6                h1 {
    7                    color: ■brown;
    8                }
    9
   10                * {
   11                    border: solid;
   12                }
   13            </style>
   14        </head>
   15
   16        <body>
   17            <h1>Student Name</h1>
   18            <a href="index.html">Projects</a>
   19            <a href="aboutme.html">About Me</a>
   20            |
   21            <h2>These are some games designed by me</h2>
   22            <p>Games use p5.js, p5.play and matter.js libraries in javascript</p>
   23
   24            <a href = "https://editor.p5js.org/whitehatjr/present/ePJrHCACM"><img src="https://via.placeholder.com/300"/></a>
   25            <p>Trex Runner Game</p>
   26
   27            <a href = "#"><img src="https://via.placeholder.com/300"/></a>
   28            <p>Name of game 2</p>
   29
   30            <a href = "#"><img src="https://via.placeholder.com/300"/></a>
   31            <p>Name of game 3</p>
```

		We used the <style> tag to add styling to the document. Typically, designers/developers try to create content and styling in two different documents so that content and design are independent of each other. This creates consistent design for different pages of a website easily. It also allows for a change in the design template of a website easily without affecting the content page.	Student listens.
		For example, look at these two- **Student Activity 1** and **Student Activity 2** - web designs. They contain exactly SAME content but completely different designs.	Student reviews the content and design for the **Student Activity 1 and 2.**

	Developers need to create two different and separate styling pages to create these two different designs. We call such styling page "CSS (cascading style sheet)."	
	We will be creating CSS for our portfolio.	
	Creating a CSS is quite simple.	Student follows the teacher and asks questions.
	1. Create an empty file in the same folder and call it style.css (or any other name. The extension .css is important) This is going to be our style sheet.	
	2. Copy the style content (inside style tags) to this style sheet. We are going to add more styling to our document in this and the next class.	
	3. This stylesheet needs to be linked to the html file. We do it using the <link/> tag.	
	Teacher shows how to create a stylesheet.	

```html
index.html ×
index.html > ...
1    <!DOCTYPE HTML>
2    <html>
3        <head>
4            <title>My Portfolio page</title>
5            <style>
6                h1 {
7                    color: brown;
8                }
9
10               * {
11                   border: solid;
12               }
13           </style>
14       </head>
15
16       <body>
17           <h1>Student Name</h1>
18           <a href="index.html">Projects</a>
19           <a href="aboutme.html">About Me</a>
20
21           <h2>These are some games designed by me</h2>
22           <p>Games use p5.js, p5.play and matter.js libraries in javascript</p>
23
24           <a href = "https://editor.p5js.org/whitehatjr/present/ePJrHCACM"><img src="https://via.pl
25           <p>Trex Runner Game</p>
26
27           <a href = "#"><img src="https://via.placeholder.com/300"/></a>
28           <p>Name of game 2</p>
29                              Follow link (ctrl + click)
30           <a href = "#"><img src="https://via.placeholder.com/300"/></a>
31           <p>Name of game 3</p>
```

```css
style.css ×    index.html
style.css > ...
1    h1 {
2        color: brown;
3    }
4
5    * {
6        border: solid;
7    }
```

```
# style.css          <> index.html ×

<> index.html > @ html > @ body > @ a
    1    <!DOCTYPE HTML>
    2    <html>
    3        <head>
    4            <title>My Portfolio page</title>
    5            <link rel = "stylesheet"
    6                  type = "text/css"
    7                  href = "style.css" />
    8        </head>
    9
   10        <body>
   11            <h1>Student Name</h1>
   12            <a href="index.html">Projects</a>
   13            <a href="aboutme.html">About Me</a>
   14
   15            <h2>These are some games designed by me</h2>
   16            <p>Games use p5.js, p5.play and matter.js libraries in javascript</p>
   17
   18            <a href = "https://editor.p5js.org/whitehatjr/present/ePJrHCACM"><img src="https://via.pl
   19            <p>Trex Runner Game</p>
   20
   21            <a href = "#"><img src="https://via.placeholder.com/300"/></a>
   22            <p>Name of game 2</p>
   23
   24            <a href = "#"><img src="https://via.placeholder.com/300"/></a>
   25            <p>Name of game 3</p>
   26
   27            <a href = "#"><img src="https://via.placeholder.com/300"/></a>
   28            <p>Name of game 4</p>
   29
   30            <a href = "#"><img src="https://via.placeholder.com/300"/></a>
```
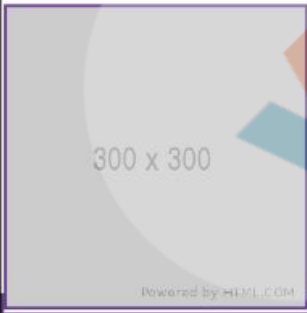

| | Remember, in the last class we had used * symbol to create a box around all the html content. All these boxes | Student listens. |

	are separate and independent of each other. We will be arranging these boxes to create a layout for our website. But before we go ahead and arrange these boxes, let's learn how to create a bigger box which can hold two smaller boxes. This will be useful for us when we arrange the boxes.	
	We create a big box using tags. All the html content we want inside the big box can be put in these div blocks. For example, let's say we want the website header name (inside h1 tag) and the menu items inside one big box. Let's create a div block which holds them. Teacher creates a div block to hold the website header name and the menu hyperlinks.	Student observes and asks questions.

```
# style.css        <> index.html ×
<> index.html > ⊗ html > ⊗ body > ⊗ div > ⊗ a
  5        <link rel = "stylesheet"
  6              type = "text/css"
  7              href = "style.css" />
  8    </head>
  9
 10    <body>
 11        <div>
 12            <h1>Student Name</h1>
 13            <a href="index.html">Projects</a>
 14            <a href="aboutme.html">About Me</a>
 15        </div>
 16
 17        <h2>These are some games designed by me</h2>
 18        <p>Games use p5.js, p5.play and matter.js libraries in javascript</p>
 19
 20        <a href = "https://editor.p5js.org/whitehatjr/present/ePJrHCACM"><img src="https://via.pl
 21        <p>Trex Runner Game</p>
 22
 23        <a href = "#"><img src="https://via.placeholder.com/300"/></a>
 24        <p>Name of game 2</p>
 25
 26        <a href = "#"><img src="https://via.placeholder.com/300"/></a>
 27        <p>Name of game 3</p>
 28
 29        <a href = "#"><img src="https://via.placeholder.com/300"/></a>
 30        <p>Name of game 4</p>
 31
 32        <a href = "#"><img src="https://via.placeholder.com/300"/></a>
 33        <p>Name of game 5</p>
```

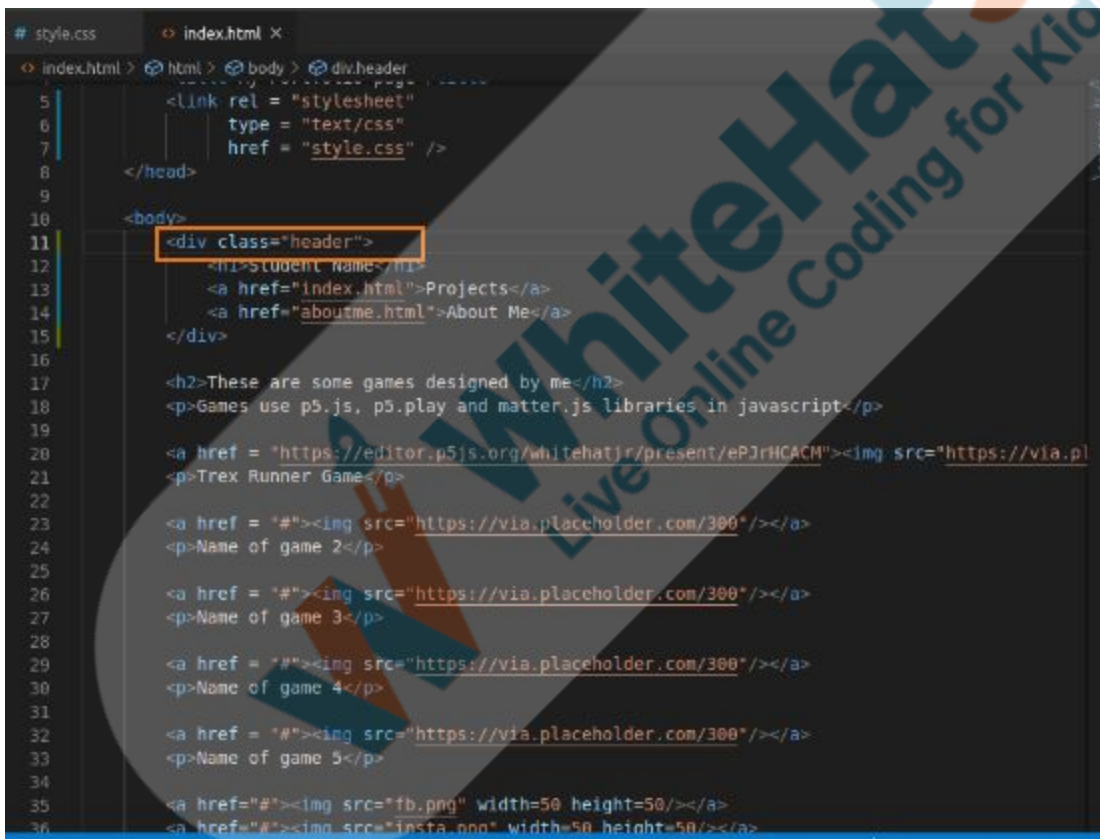|	You can see a bigger box holding the header name and the menu items. We normally give a class name to	Student observes and asks questions.

each div block. It helps when we are styling it.

Let's call our div block "header" - since it will be at the top of our website.

Class can be styled in stylesheet using .classname selector.

Here, header div class can be styled using .header selector.

```
# style.css  ×      <> index.html
# style.css > t5 .header
1    h1 {
2        color: ■brown;
3    }
4
5    * {
6        border: solid;
7    }
8
9    .header{
10       |
11   }
```

	Do you want to add boxes using div blocks on your own? And then I will help you lay them out using flexbox! Ready?	 Yes!
	Teacher Stops Screen Share	
	Now it's your turn. Please share your screen with me.	

ACTIVITY
- **Arrange the layout of the different HTML elements using flexbox.**
- **Use css selectors to add some style to the document.**

Step 3: Student-Led Activity (15 min)	Guide the student to create and link a stylesheet on their own page.	The student creates a stylesheet and links it to his/her html page for the portfolio website.
	The <a> tags themselves can be put inside a div block with the class name "menu".	The student creates the div block called "menu" with the two <a> tags inside them.

```
# style.css        <> index.html ×
<> index.html > 🔗 html > 🔗 body > 🔗 div.site-description
  5        <link rel = "stylesheet"
  6                type = "text/css"
  7                href = "style.css" />
  8      </head>
  9
 10    <body>
 11        <div class="header">
 12            <h1>Student Name</h1>
 13            <div class="menu">
 14                <a href="index.html">Projects</a>
 15                <a href="aboutme.html">About Me</a>
 16            </div>
 17        </div>
 18
 19        <div class="site-description">
 20          <h2>These are some games designed by me</h2>
 21          <p>Games use p5.js, p5.play and matter.js libraries in javascript</p>
 22        </div>
 23
 24        <a href = "https://editor.p5js.org/whitehatjr/present/ePJrHCACM"><img src="https://via.p
 25        <p>Trex Runner Game</p>
 26
 27        <a href = "#"><img src="https://via.placeholder.com/300"/></a>
 28        <p>Name of game 2</p>
 29
 30        <a href = "#"><img src="https://via.placeholder.com/300"/></a>
 31        <p>Name of game 3</p>
 32
 33        <a href = "#"><img src="https://via.placeholder.com/300"/></a>
 34        <p>Name of game 4</p>
```

	Now let's learn how to arrange these boxes to create a layout which we want. We will be using flexbox to create the layout. Look at the **Student Activity 3** to see how to use flex.	Student goes through the **student activity 3.**
	The bigger box is called the parent. The boxes inside the bigger box are called the children. Let's style the box (div block) named 'header'. We will be using ".header" as a css selector to add styling to this box.	The student adds a .header selector in the css stylesheet.

```
# style.css  ×    <> index.html
# style.css > 🎨 .header
  1    h1 {
  2        color: ■brown;
  3    }
  4
  5    * {
  6        border: solid;
  7    }
  8
  9    .header{
 10        │
 11    }
```

	We can add a simple background to our 'header class div' to check if our styling works.	The student adds a light background-color to the header class div.

```
# style.css  ×    <> index.html
# style.css > 🎨 .header
  1    h1 {
  2        color: ■brown;
  3    }
  4
  5    * {
  6        border: solid;
  7    }
  8
  9    .header{
 10        background-color: ■cadetblue;
 11    }
```

| | '.header' class is the parent class containing h1 tag and another div class called 'menu'. We will use flexbox to arrange the boxes inside the header class.

Let's use "display: flex" to tell the browser that we are going to be using the flexbox. | The student adds "display: flex" property inside the '.header' selector. |
|---|---|---|

```css
# style.css    <> index.html
# style.css > .header
1   h1 {
2       color: brown;
3   }
4
5   * {
6       border: solid;
7   }
8
9   .header{
10      background-color: cadetblue;
11      display: flex;
12  }
```

| | Now, we can use "justify-content" property inside the parent div to align the inside boxes horizontally.

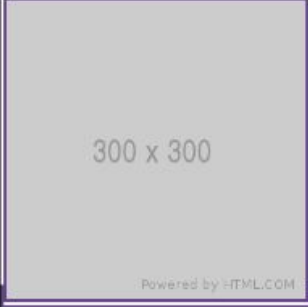If you want the header name towards left and menu to the right of the web page, you could use "space-between" | The student adds "justify-content:space-between" inside the '.header' selector.

Student runs the code and sees the output. |
|---|---|---|

```css
# style.css    <> index.html
# style.css > .header
1   h1 {
2       color: brown;
3   }
4
5   * {
6       border: solid;
7   }
8
9   .header{
10      background-color: cadetblue;
11      display: flex;
12      justify-content: space-between;
13  }
```

Student Name

These are some games designed by me

Games use p5.js, p5.play and matter.js libraries in javascript

300 x 300

Powered by HTML.COM

Trex Runner Game

style.css   ^   Show all

	Amazing, isn't it! How simple it is!	The student creates a div called "grid" class to hold all the game images and their names.
	Let's arrange the other boxes on the page to create a great layout for the portfolio website.	
		Each item in the grid contains one image and the name of the game is put inside a div named "item".
	Do not worry about the styling too much right now. We will focus more on styling in our next class.	
		Student adds css to layout the grid using flexbox properties by referencing **Student Activity 3**.
	Let's create the layout for the grid of images for the games.	

```html
            <div class = "grid">
                <div class="item">
                    <a href = "https://editor.p5js.org/whitehatjr/present/ePJrHCACM"><img src="https:
                    <p>Trex Runner Game</p>
                </div>

                <div class="item">
                    <a href = "#"><img src="https://via.placeholder.com/300"/></a>
                    <p>Name of game 2</p>
                </div>

                <div class="item">
                    <a href = "#"><img src="https://via.placeholder.com/300"/></a>
                    <p>Name of game 3</p>
                </div>

                <div class="item">
                    <a href = "#"><img src="https://via.placeholder.com/300"/></a>
                    <p>Name of game 4</p>
                </div>

                <div class="item">
                    <a href = "#"><img src="https://via.placeholder.com/300"/></a>
                    <p>Name of game 5</p>
                </div>

                <div class="item">
                    <a href = "#"><img src="https://via.placeholder.com/300"/></a>
                    <p>Name of game 6</p>
```

```css
 1   h1 {
 2       color: white;
 3   }
 4
 5   /* * {
 6       border: solid;
 7   } */
 8
 9   .header{
10       background-color: cadetblue;
11       display: flex;
12       justify-content: space-between;
13   }
14   |
15
16   .grid{
17       display: flex;
18       flex-wrap: wrap;
19       justify-content: space-evenly;
20   }
21
22   .footer{
23       display: flex;
24       justify-content: flex-end;
25   }
26
27   p,h2{
28       text-align: center;
29   }
```
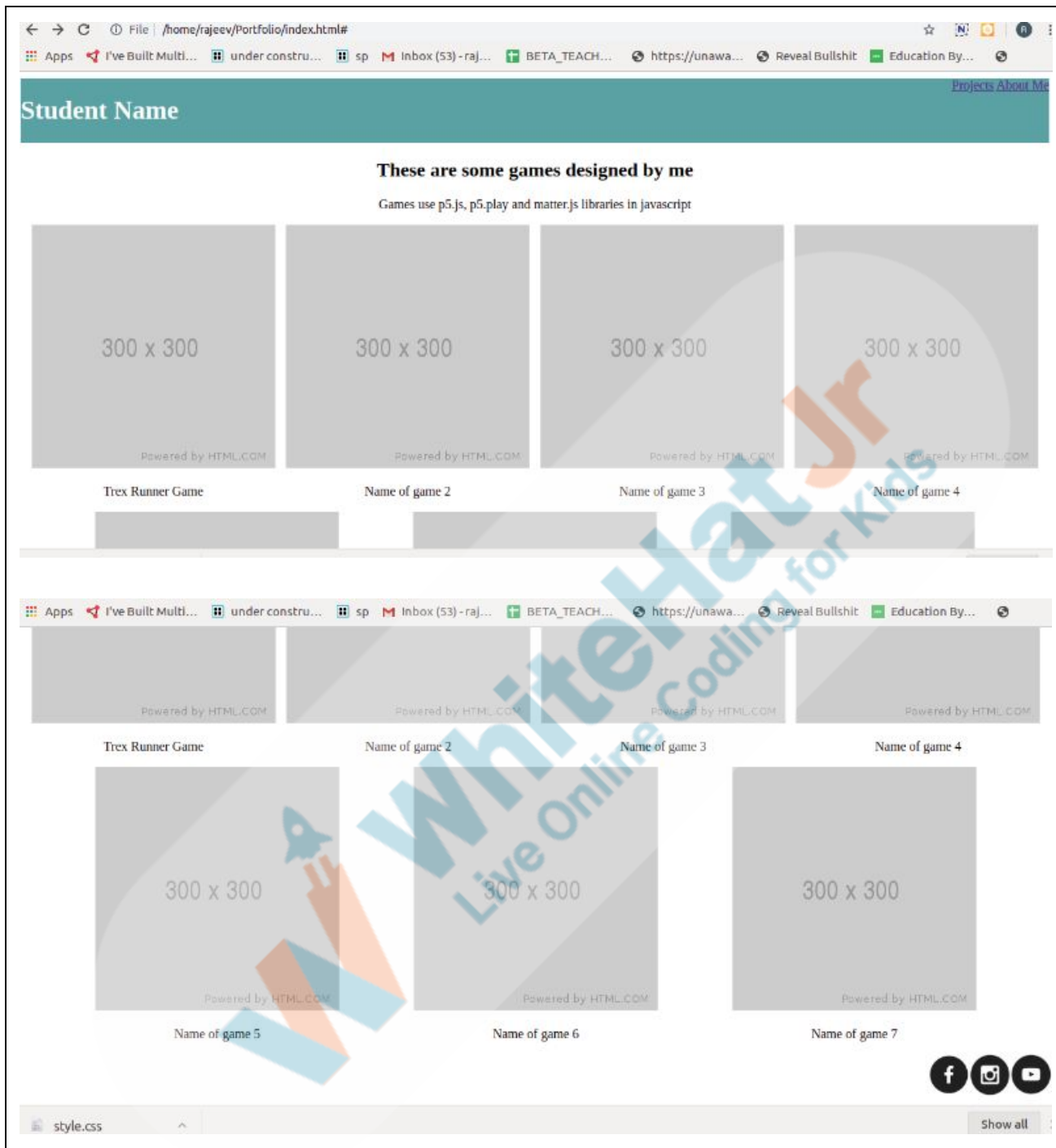
		Let's create a div to hold the social media icons.	The student creates a div class with the name "footer". He/She uses the flexbox property to align the footer to the right.


```
# style.css        <> index.html ×
<> index.html > ⊘ html > ⊘ body > ⊘ div.footer
52
53
54              <div class="item">
55                  <a href = "#"><img src="https://via.placeholder.com/300"/></a>
56                  <p>Name of game 7</p>
57              </div>
58
59          </div>
60
61          <div class="footer">
62              <a href="#"><img src="fb.png" width=50 height=50/></a>
63              <a href="#"><img src="insta.png" width=50 height=50/></a>
64              <a href="#"><img src="youtube.png" width=50 height=50/></a>
65          </div>
66
67      </body>
68  </html>
```



```
# style.css  ●    <> index.html
# style.css > ...
 1   h1 {
 2       color: ▉white;
 3   }
 4
 5   /* * {
 6       border: solid;
 7   } */
 8
 9   .header{
10       background-color: ▉cadetblue;
11       display: flex;
12       justify-content: space-between;
13   }
14
15
16   .grid{
17       display: flex;
18       flex-wrap: wrap;
19       justify-content: space-evenly;
20   }
21
22   .footer{
23       display: flex;
24       justify-content: flex-end;
25   }
26
27   p,h2{
28       text-align: center;
29   }
```

	See how easy it is to create a simple layout for a website using flexbox. Without adding much of styling, we could create a simple and beautiful portfolio webpage! In the next class, we will add more styling to the web page to make it look super professional.	The student is excited about the prospect.
	Teacher Guides Student to Stop Screen Share	
	FEEDBACK ● **Encourage the student to make reflection notes in the markdown format.** ● **Complement the student for her/his effort in the class.** ● **Encourage the student to create layout and styling for the "About Me" page on the website.**	
Step 4: **Wrap-Up** **(5 min)**	Let's quickly summarize what we learned in today's class	We learned: 1) how to use div blocks to hold other html elements. 2) how to assign class name to div blocks and add style properties to them. 3) create a css sheet and link it to an html file. 4) use flexbox and flexbox properties to create a layout for a website.
	You get a "hats off." Amazing! Super cool. I am looking forward to the next class.	Make sure you have given at least 2 Hats Off during the class for: Creatively Solved Activities +10 Great Question +10

		Strong Concentration +10
Project Overview	**Origami Website Design Part 2** **Goal of the Project:** Today, you have learnt to use the CSS and flexbox. In this project, you will have to use different CSS properties and flexbox properties to make the website look more aesthetic and professional. **Story:** Honey has an origami studio. Now she is creating a website to explain step-by-step how origami is made. I am very excited to see your project solution and I know you both will do really well. Bye Bye!	Students engage with the teacher over the project.
	Teacher Clicks ✖ **End Class**	
Additional Activities	Can you create the layout for the "about me" page using flexbox and flexbox properties? You can use the same stylesheet but you will have to link it to the "about me" page.	The student links the style.css to "about me" html page. He/She designs the layout for the about me page.
	Encourage the student to write reflection notes in their reflection journal using markdown.	The student uses the markdown editor to write

	Use these as guiding questions: ● What happened today? - Describe what happened - Code I wrote ● How did I feel after the class? ● What have I learned about programming and developing games? ● What aspects of the class helped me? What did I find difficult?	her/his reflection in a reflection journal.

Activity	Activity Name	Links
Teacher Activity 1	Previous Class Code	https://github.com/whitehatjr/portfolio
Teacher Activity 2	Reference Code	https://github.com/whitehatjr/StudentPortfolio1.0
Student Activity 1	Zen Garden 1	http://www.csszengarden.com/221/
Student Activity 2	Zen Garden 2	http://www.csszengarden.com/219/
Student Activity 3	Flexbox Reference	https://css-tricks.com/snippets/css/a-guide-to-flexbox/