

Internship Report – Frontend Dev(Week2)

Name: Zainab

Father Name: Assad Qayyum

Date: 5th July, 2025

Internship Domain: Front-end Intern

Task: CSS - Flexbox: display: flex, justify-content, align items & Mini Project

Task Overview: (Day6)

Today's task (Day 6 of Week 2) in my front-end internship was focused on learning two essential layout properties in CSS: Flexbox: display: flex, justify-content, align items.

Mini Project: Stylish product card or landing page

Content Covered:

CSS Flexbox:

- display:flex
- justify-content
- align items

Mini Project: Stylish product card or landing page

CSS Flexbox:

Flexbox (Flexible Box Layout) is a CSS layout model that allows you to arrange elements in a row or column and control how space is distributed between them — even when the screen size changes.

Flexbox makes it easier to design a flexible responsive layout structure, without using float or positioning.

1. Display: flex:

This is how you activate Flexbox on a container (parent element).

- Once you set display: flex, all direct child elements become flex items.
- By default, they are arranged horizontally (in a row).

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

2. Justify-content:

This property controls horizontal alignment (main axis) of flex items inside the container.

Common values:

flex-start (default)	Items start from the left
center	Items are centered horizontally
flex-end	Items are pushed to the right
space-between	Equal space between items, edges touching
space-around	Equal space around all items
space-evenly	Equal space between AND around all items

Example: Items will line up centered horizontally in the container.

```
.container {  
  display: flex;  
  justify-content: center;  
}
```

3. Align-items:

This property controls vertical alignment (cross axis) of items within the container.

Common values:

stretch (default)	Items stretch to fill container height
flex-start	Items align to the top
center	Items align to the center vertically
flex-end	Items align to the bottom
baseline	Aligns text baselines across items

Example: Items will line up vertically centered inside the flex container.

```
.container {  
  display: flex;  
  align-items: center;  
}
```

```
<style>  
.container {  
  display: flex;  
  justify-content: space-around;  
  align-items: center;  
  height: 100px;  
  background: lightgray;  
}  
  
.item {  
  background: cornflowerblue;  
  padding: 10px 20px;  
  color: white;  
  border-radius: 5px;  
}  
</style>  
  
<div class="container">  
  <div class="item">Item 1</div>  
  <div class="item">Item 2</div>  
  <div class="item">Item 3</div>  
</div>
```

Mini-Project:

This mini project showcases a stylish product card layout using core CSS concepts. It features three modern product cards—Mobile Phone, Laptop, and Headphones—designed with clean typography, flexible layouts, gradients, shadows, and interactive buttons. It includes

- Selectors: element, class, and ID
- Box Model: margin, padding, border, content
- Typography: fonts, weights, sizes
- Backgrounds, Borders, Shadows, and Gradients
- Positioning: static, relative
- Flexbox: display: flex, justify-content, align-items

Code:

Index.html:

```
index.html X # styles.css
index.html > html > head
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Stylish Product Cards</title>
5   <link rel="stylesheet" href="styles.css">
6 </head>
7 <body>
8
9   <div class="container">
10     <!-- Product Card 1 -->
11     <div class="card" id="card1">
12       
14         <h3>MOBILE PHONE</h3>
15         <p class="price">$269.99</p>
16         <p class="desc">Sleek design, high-resolution camera, and all-day battery life.</p>
17         <div class="btn-container">
18           <button class="btn">Add to Cart</button>
19           <button class="btn">Details</button>
20         </div>
21       </div>
22     </div>
23
24     <!-- Product Card 2 -->
25     <div class="card" id="card2">
26       
28         <h3>LAPTOP</h3>
29         <p class="price">$599.99</p>
30         <p class="desc">Powerful performance for work, gaming, and everyday multitasking.</p>
```

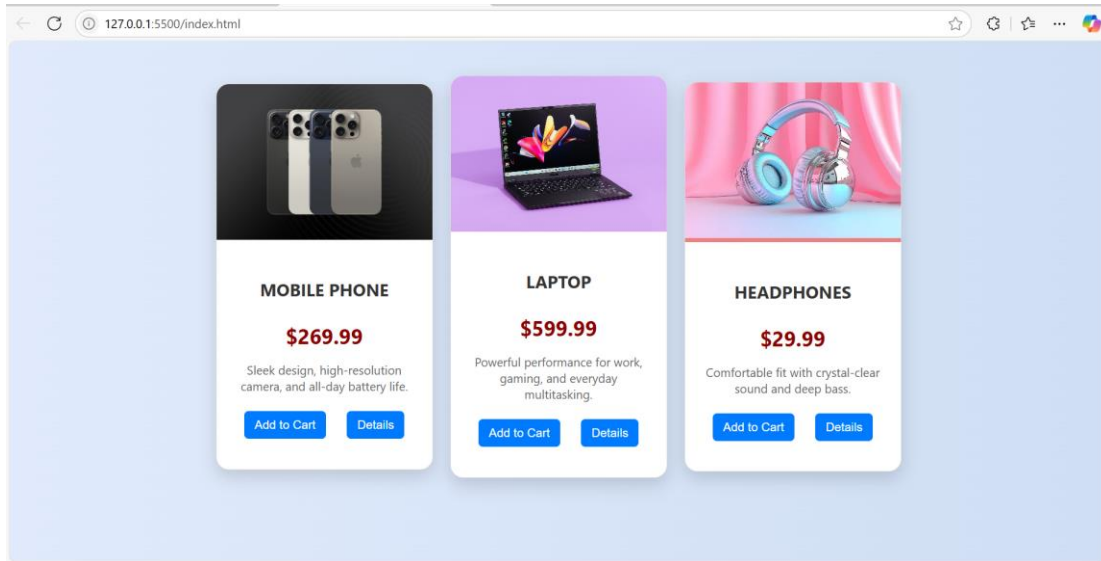
```
index.html # styles.css
index.html > html > body > div.container
2 <html>
7 <body>
9 <div class="container">
25 <div class="card" id="card2">
27 <div class="card-content">
30 <p class="desc">Powerful performance for work, gaming, and everyday multitasking.</p>
31 <div class="btn-container">
32 <button class="btn">Add to Cart</button>
33 <button class="btn">Details</button>
34 </div>
35 </div>
36 </div>
37
38 <!-- Product Card 3 -->
39 <div class="card" id="card3">
40 
41 <div class="card-content">
42 <h3>HEADPHONES</h3>
43 <p class="price">$29.99</p>
44 <p class="desc">Comfortable fit with crystal-clear sound and deep bass.</p>
45 <div class="btn-container">
46 <button class="btn">Add to Cart</button>
47 <button class="btn">Details</button>
48 </div>
49 </div>
50 </div>
51 </div>
52 </body>
53 </html>
54
Ln 51, Col 9 Spaces: 4 UTF-8 CRLF {} HTML Port: 5500
```

Styles.css:

```
index.html # styles.css X
# styles.css > .price
1 /* Body styling */
2 body {
3   font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
4   margin: 0;
5   padding: 0;
6   background: linear-gradient(to right, #e0eafc, #cfdef3);
7 }
8
9 /* Flexbox container */
10 .container {
11   display: flex;
12   justify-content: center;
13   align-items: center;
14   flex-wrap: wrap;
15   padding: 40px;
16   gap: 20px;
17 }
18
19 /* Product Card */
20 .card {
21   background-color: #fff;
22   border-radius: 15px;
23   width: 250px;
24   box-shadow: 0 8px 20px rgba(0, 0, 0, 0.15);
25   overflow: hidden;
26   border: 1px solid #ddd;
27   position: relative;
28   transition: transform 0.3s ease;
29 }
30
Ln 57, Col 17 Spaces: 4 UTF-8 CRLF {} CSS Port: 5500
```

```
index.html # styles.css X
# styles.css > .price
20 .card {
21 }
22
23 .card:hover {
24   transform: scale(1.05);
25 }
26
27 /* Image */
28 .product-image {
29   width: 100%;
30   height: 180px;
31   object-fit: cover;
32 }
33
34 /* Card content */
35 .card-content {
36   padding: 20px;
37   text-align: center;
38 }
39
40 /* Typography */
41 .card-content h3 {
42   font-size: 20px;
43   margin-bottom: 10px;
44   color: #333;
45 }
46
47 .price {
48   font-size: 24px;
49   color: darkred;
50 }
51
52 Ln 57, Col 17 Spaces: 4 UTF-8 CRLF {} CSS Port: 5500
```

```
index.html # styles.css
# styles.css > .desc
55 .price {
56   font-size: 24px;
57   color: darkred;
58   font-weight: bold;
59   margin-bottom: 10px;
60 }
61
62 .desc {
63   font-size: 14px;
64   color: #666;
65   margin-bottom: 20px;
66 }
67
68 /* Button styles */
69 .btn-container {
70   display: flex;
71   justify-content: space-around;
72   margin-bottom: 15px;
73 }
74
75 .btn {
76   background: #007bff;
77   color: #fff;
78   padding: 8px 12px;
79   border: none;
80   border-radius: 5px;
81   cursor: pointer;
82   font-size: 14px;
83 }
84
85 .btn:hover {
86   background: #0056b3;
87 }
88
89 Ln 66, Col 2 Spaces: 4 UTF-8 CRLF {} CSS Port: 5500
```



Conclusion:

In short, today's task helped me understand some new concepts in CSS: Flexbox: display:flex, justify-content, align items.

Then a mini project- Stylish product card. This helped me apply all the core CSS concepts I've learned so far in a practical way. From layout and styling to interactivity and responsiveness, it demonstrates how CSS brings structure and design together in modern web interfaces.