

CSS Flexbox

Theory Assignment 3

Q.1 What is CSS Flexbox, and how is it useful for layout design?

Explain the terms flex-container and flex-item.

Ans: CSS Flexbox is a layout model in CSS that makes it easier to design and build responsive web layouts. Instead of using float or positioning hacks, Flexbox allows you to easily align, distribute, and space elements within a container – even when their sizes are unknown or dynamic.

How Flexbox is Useful:

1. Easier alignment: Center items vertically and horizontally with just a few lines of CSS.
2. Responsive layouts: Adjust items automatically based on screen size.
3. Flexible sizing: Items can grow, shrink, or stay fixed as needed.

Key Terms:

1) Flex-container: This is the parent element that holds all the items you want to layout with Flexbox. You make an element a flex container by adding this CSS:

```
CSS
display: flex;
```

Once this is set, all direct children of that element become flex items.

2) Flex-item: These are the children inside the flex container. They automatically become flexible and follow the rules of Flexbox, such as how they align, grow, or shrink based on the container and other items.

```
html

<div class="flex-container">
  <div class="flex-item">Item 1</div>
  <div class="flex-item">Item 2</div>
  <div class="flex-item">Item 3</div>
</div>
```

```
css

.flex-container {
  display: flex;
  justify-content: space-between;
  align-items: center;
}

.flex-item {
  padding: 10px;
  background-color: lightblue;
}
```

In this example:

- The container arranges the items in a row.
- Items are spaced evenly.
- Items are vertically centered.

Q.2 Describe the properties justify-content, align-items, and flex-direction used in Flexbox.

Ans:

Justify-content:

This property controls how flex items are spaced along the main axis.

Common values:

- Flex-start: Items are packed at the start.
- Flex-end: Items are packed at the end.
- Center: Items are centered in the container.
- Space-between: Items are spaced evenly, first time at the start and last at the end.
- Space-around: Items have equal space around them.
- Space-evenly: Items have equal space between all elements, including edges.

```
CSS
justify-content: center;
```

This will center all items horizontally inside the container.

Align-items:

This property controls how flex-items are aligned along the cross axis.

Common values:

- Stretch: Items stretch to fill the container's height.
- Flex-start: Items align at the top of the container.
- Flex-end: Items align at the bottom.
- Center: Items are centered vertically.

```
CSS
align-items: flex-start;
```

This aligns all items to the top of the container.

Flex-direction:

This sets the direction of the main axis, i.e., how the flex items are laid out.

Common values:

- Row: Items are placed left to right in a row.
- Row-reverse: Items are placed right to left.
- Column: Items are placed top to bottom.
- Column-reverse: Items are placed bottom to top:

CSS

```
flex-direction: column;
```

This stacks items vertically like a column.