

Hand Note of Module: 08

8.1: Introduction to website layout:

What is website layout:

A website layout is a pattern (or framework) that defines a website's structure. It has the role of structuring the information present on a site both for the website's owner and for users. It provides clear paths for navigation within webpages and puts the most important elements of a website front and center.

8.2: Flexbox, flex-direction, justify-content, align-items:

What is Flexbox?

Flexbox is a layout model that allows elements to align and distribute space within a container. Using flexible widths and heights, elements can be aligned to fill a space or distribute space between elements, which makes it a great tool to use for responsive design systems.

Justify-content and align-items:

ধরুন আমার একটি সেকশন এর ভিতর ৮ টি ডিভ আছে দুই লাইনে এখন আমি আমার সেকশনটাকে ডিসপ্লে ফ্লেক্স করে দিলে আমি আমার সেকশনের ভিতরের ডিভগুলোকে হরিজেন্টালি এবং ভারটিকেলি এলাইন করতে পারবো। হরিজেন্টালি এলাইন করার জন্য Justify-content ব্যবহার করা হয়ে থাকে এবং ভারটিকেলি এলাইনের জন্য align-items ব্যবহার করা হয়ে থাকে। জাস্টিফাই কনটেন্ট এবং এলাইন আইটেম এর কতগুলো সাব প্রপার্টি আছে। এর মধ্যে উল্লেখ যোগ্য কয়েকটি হলো।

Start: start এর মাধ্যমে হরিজেন্টালি বামদিকে এবং ভারটিকেলি উপরের দিকে এলাইন করা হয়। তবে হরিজেন্টালি করার জন্য জাস্টিফাই কনটেন্ট এবং ভারটিকেলি করার জন্য এলাইন আইটেম ব্যবহার করা হয়।

Center: center এর মাধ্যমে হরিজেন্টালি মাঝ বরাবর এবং ভারটিকেলি মাঝ বরাবর এলাইন করা হয়। তবে হরিজেন্টালি করার জন্য জাস্টিফাই কনটেন্ট এবং ভারটিকেলি করার জন্য এলাইন আইটেম ব্যবহার করা হয়।

Space-between: space-between এর মাধ্যমে হরিজেন্টালি কনটেইনারের ভিতরে সমান গ্যাপ থাকবে এবং যদি দুইটা ডিভ থাকে কনটেইনারের ভিতরে তাহলে একটা বামে বরাবর থাকবে এবং অপরটা ডান বরাবর থাকবে আর মধ্যের জায়গাটুকু ফাকা থাকবে। একইভাবে ভারটিকেলি যদি দুইটা থাকে তাহলে একটা কনটেইনারের একদম উপরে থাকবে এবং অপরটা একদম নিচে বরাবর থাকবে আর মধ্যের ফাকা অংশ থাকবে। তবে হরিজেন্টালি করার জন্য জাস্টিফাই কনটেন্ট এবং ভারটিকেলি করার জন্য এলাইন আইটেম ব্যবহার করা হয়।

Space-around: ধরি আমাদের তিনটা ডিভ আছে আমরা যদি হরিজেন্টালি একটার থেকে আরেকটার দূরত্ব এবং ডানদিকের স্ক্রিন থেকে প্রথম ডিভের দূরত্ব ও বামদিকের স্ক্রিন থেকে শেষ ডিভের দূরত্ব সমান রাখতে চাই তাহলে space-around ব্যবহার করতে হবে। একই ভাবে আমরা যদি ভারটিকেলি একটার থেকে আরেকটার দূরত্ব এবং উপরের স্ক্রিন থেকে প্রথম ডিভের দূরত্ব ও নিচের স্ক্রিন থেকে শেষ ডিভের দূরত্ব সমান রাখতে চাই তাহলে space-around ব্যবহার করতে হবে। তবে হরিজেন্টালি করার জন্য জাস্টিফাই কনটেন্ট এবং ভারটিকেলি করার জন্য এলাইন আইটেম ব্যবহার করা হয়।

Space-between: আমরা যদি হরিজেন্টালি এবং ভারটিকেলি আমাদের কনটেইনারের ভিতরের প্রত্যেকটা আইটেমকে অনেকটা সমান মার্জিন দিয়ে রাখার মত রাখতে চাই তাহলে আমাদেরকে Space-between ব্যবহার করতে হবে। ধরি আমরা হরিজেন্টালি Space-between করে দিলাম তাহলে আমাদের আইটেম গুলো অনেকটা লেফট এবং রাইটে একই মার্জিন দেওয়ার মত সরবে। অর্থাৎ এক্ষেত্রে যদি আমাদের তিনটা ডিভ থাকে তাহলে প্রথমটা থেকে দ্বিতীয়টার দূরত্ব যত হবে স্ক্রিন থেকে প্রথমটার দূরত্ব তার অর্ধেক হবে। তবে হরিজেন্টালি করার জন্য জাস্টিফাই কনটেন্ট এবং ভারটিকেলি করার জন্য এলাইন আইটেম ব্যবহার করা হয়।

এছাড়াও আরো কয়েকটি উল্লেখযোগ্য সাব প্রপার্টি আছে যেগুলো হলো:

End, flex-start, flex-end

Flex-grow: 1; এর কাজ হচ্ছে ধরি আমাদের কনটেইনারের ভিতর ২টা ডিভ আছে এবং ডিফল্টভাবে প্রত্যেকটা ডিভের উইথ ২০০ পি.এক্স আছে। এক্ষেত্রে আমরা যদি একটি ডিভকে দিয়ে দেই তাহলে ঐ ডিভটি কনটেইনারের ভিতরের অবশিষ্ট জায়গা দখল করবে এবং অপর ডিভটি তার জন্য নির্দারন করা ২০০ পি.এক্স এই সীমাবদ্ধ থাকবে।

Hand Note of Module: 08

8.3: Create login form and flexible nav using flexbox:

flex-direction: row;	<table><tr><td>Item 1</td><td>Item 2</td><td>Item 3</td></tr></table>	Item 1	Item 2	Item 3						
Item 1	Item 2	Item 3								
flex-direction: row-reverse;	<table><tr><td>Item 3</td><td>Item 2</td><td>Item 1</td></tr></table>	Item 3	Item 2	Item 1						
Item 3	Item 2	Item 1								
flex-direction: column;	<table><tr><td></td><td>Item 1</td><td></td></tr><tr><td></td><td>Item 2</td><td></td></tr><tr><td></td><td>Item 3</td><td></td></tr></table>		Item 1			Item 2			Item 3	
	Item 1									
	Item 2									
	Item 3									
flex-direction: column-reverse;	<table><tr><td></td><td>Item 3</td><td></td></tr><tr><td></td><td>Item 2</td><td></td></tr><tr><td></td><td>Item 1</td><td></td></tr></table>		Item 3			Item 2			Item 1	
	Item 3									
	Item 2									
	Item 1									

8.4: CSS Grid layout template column grid gap:

Display: grid; [ডিসপ্লে গ্রিড হবে]

Grid-template-columns: 200px 200px 200px; [২০০ পি.এক্স এর তিনটা কলাম হবে]

Grid-template-columns: repeat (4, 200px); [২০০ পি. এক্স এর চারটা কলাম হবে]

Grid-template-columns: repeat (6, 1fr); [ছয়টা কলাম হবে একই মাপের]

Grid-template-columns: repeat (4, 1fr) 100px; [প্রথমে চারটা কলাম হবে একই মাপের তার ১০০ পি.এক্স এর আরো একটা কলাম হবে।]

Grid-gap: 20 px; [গ্রিড কলাম থেকে গ্রিড কলাম এবং গ্রিড র থেকে গ্রিড ও এর গ্যাপ ২০ পি.এক্স হবে]

8.5: Grid-template-rows:

Grid-template-columns এর মতই ফাস্ট Grid-template-rows হবে ভাটিকেলি।

Hand Note of Module: 08

8.6: Explore Responsiveness using media query and flexbox:

প্রেইন সি. এস. এস দিয়ে ওয়েবসাইট রেসপনসিভ করার জন্য মিডিয়া কুয়েরি ব্যবহার করা হয়। এর সিনটেক্স হলো:

```
@media (condition) { style role}

@media only screen and (max-width: 600px) {
  body {
    background-color: lightblue;
  }
}

@media only screen and (min-width: 900px) {
  body {
    background-color: green;
  }
}
```

8.7: Create a photo Album using relative position and flexbox:

Flex-wrap: wrap; [এর মাধ্যমে ফ্লেক্স আইটেমগুলোকে আটো রেসপনসিভ করা যায়]

8.8: Apply transition transform in your photo album:

Just recap old modules (transition and transform)

8.9: All CSS properties, everything you need to know about CSS.

Recourses: <https://www.tutorialrepublic.com/>