Flexbox

- css display type design to help us craft css layouts quite easily.

- control the position, size, spacing of elements relative to their parent elements & each other.

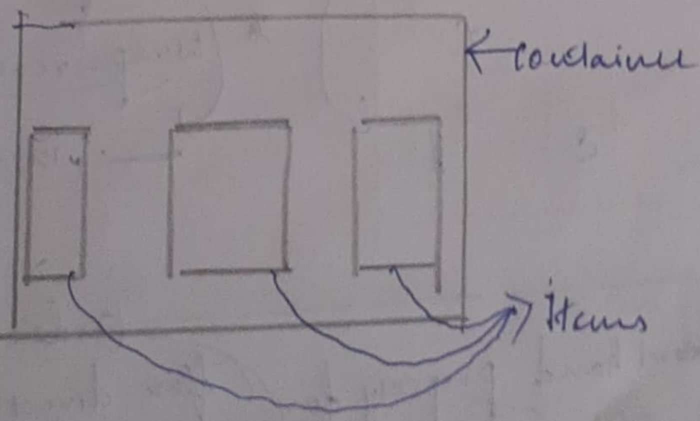
- makes it easier to design flexible responsive layout structure without using float or positioning.

- great for responsive designing.

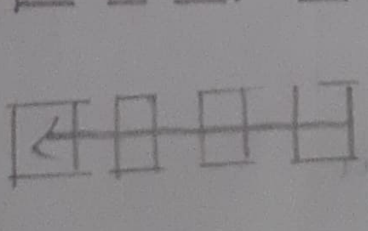
- has its own set of properties

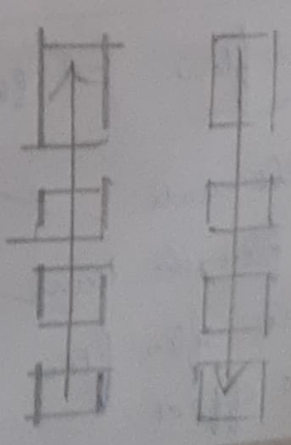
- defines formatting context which controls layout.

- is parent - child relationship.



- flex-direction
  - row
  - row-reverse
  - column
  - column-reverse





→ justify-content

→ flex-start

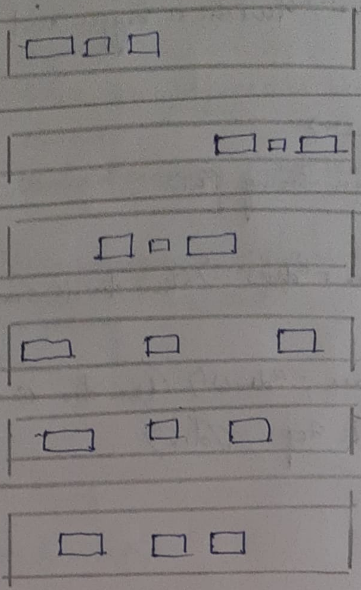
→ flex-end

→ center

→ Space-between

→ Space-around

→ Space-evenly



→ align-items

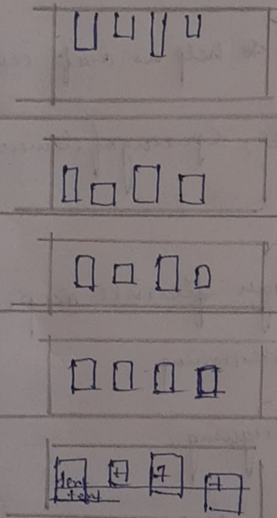
→ flex-start

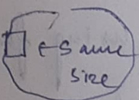
→ flex-end

→ center

→ Strech

→ baseline



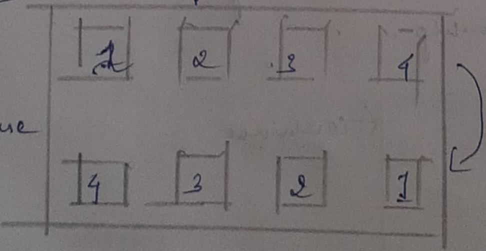


→ flexwrap → default

→ nowrap

→ wrap

→ wrap-reverse



* wrap that specifies flexible item will wrap if necessary.

* wrap-reverse → in reverse order.

→ flex-flow → Short hand property for (flex-direction, flex-wrap.) Order 1 2

→ If the element are not flexible items then flex flow property has no effect.

- align-content :- used to align the flex lines.
  - stretch - default
  - center
  - all combination of properties of
    - justify content - align items.

- justify-content :- Aligns the flexible container items when the item do not use all available space on main axis (horizontally)
  (pg no 168)

- Perfect centering (pg no 171 eg3)
  Set both 1) justify content
  2) align items } center & flex item will be centred perfectly.

---

flexbox html

Eg 1)

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      .flex-container{
          display:flex; ✓
        background color: blue;
      }
    .flex-container > div{
          background-color: Dioheat;
          margin: 10px;
          padding: 20px;
           font-size: 30px;
          }
    </style>
  </head>
  <body>
      <div class: "flex-container">
          <div> 1 </div>
          <div> 2 </div>
          <div> 3 </div>
      </div>
    <!-- <p> direct child elements of the flexible container
          automatically becomes flexible items </p> -->
    </body>
</html>
```

To eg 2}

flex-wrap.html

```
<!DOCTYPE html>
    <html>
        <head>
          <style>
            .flex-container-1{
                display:flex;
                flex-wrap:wrap;
               background-color: □Blue;
             }
          .flex-container-2{
              display:flex;
            flex-wrap: nowrap;
             background-color: □Blue;
            }
        .flex-container-1>div{
              background-color: □red;
                Width: 100px;
               Margin: 10px;
                text-align:center;
                line-height: 75px;        ) Pgnono
                  font-size: 30px;
              }
        .flex-container-2>div{
             background-color: □red;
               Width: 100px;
             Margin:10px;
         text-align: center;
          line-height; 75px;
             font-size: 30px;
          }
      </style>
    <head>
<body>
   <h3 Style ="text-align:center">Try resicing the browser window </h3>
   <h3> flex-wrap: wrap; </h3>
      <div class ="flex-container-1">
              <div> 1 </div>
              <div> 2 </div>
              <div>3 <div>
              <div> 4 </div>
              <div> 5 </div>
               <div> 6 </div>
               <div> 7 </div>
              <div> 8 </div>
              <div> 9 </div>
              <div> 11 <div>
          <d ~~ 1A ~~ </d>
               ~~ 12 ~~
  </div>     ~~ 0 ~~      }
```

```
</h2>
<br><br>

<h3> flex-wrap : nowrap; </h3>
 <div class = "flex-container-2">
      <div> 1 </div>
      <div> __ </div>

       <div> 5 </div>
  </div>
</body>
  </html>
```

Eg3) : perfect-centering.html

```
<!DOCTYPE html>
  <html>
    <head>
      <style>
        .flex-container {
             display : flex;
             justify-content : center;
              align-items : center;
             height : 300px;
             background-color : □ blue;
           }
        .flex-container > div {
             background-color : □ black;
              color : □ white;
             width : 150px;
             height : 100px;
             text-align : center;
           }
      </style>
   </head>
    <body>
       <div class = "flex-container">
            <div> PERFECT CENTERING </div>
       </div>
    </body>
 </html>
```

→) flex-items : properties

(Can refer the diagram in pg no 167)

- order → Specify the order of flex items
- flex-grow → Specify how much flex item grow relative to rest of flexitem.
  - → default: 0
- flex-shrink → vice versa of (flex-grow)
  - → default: 1
- flex-basis → Specify initial length of flex item.
- flex → Short hand property : flex grow flex shrink flex basis.
- align-self
  - → Specifies alignment for the selected item inside flexible container.
  - → Overrides the default alignment set by container's align-items property.