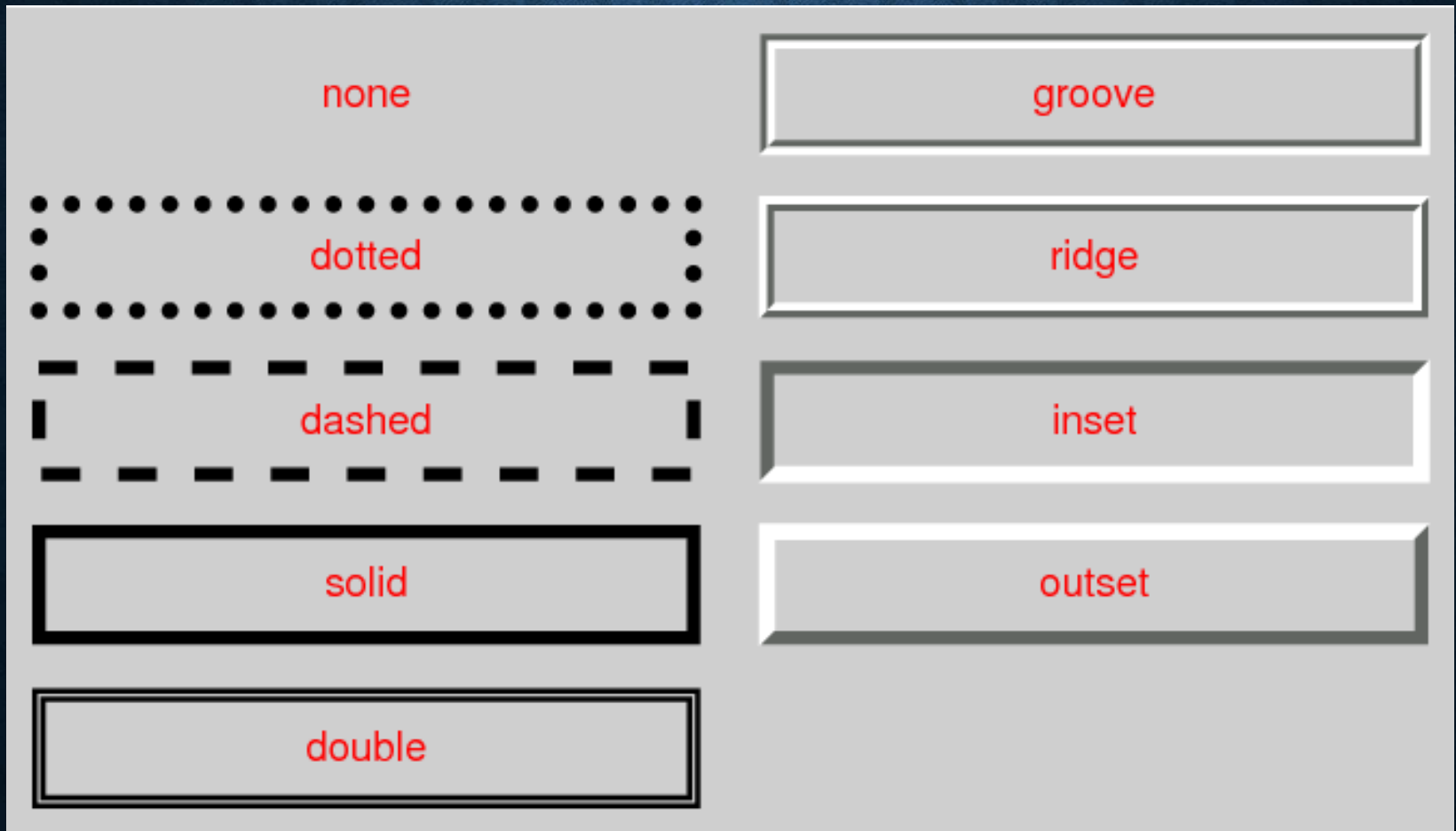


WEB TECHNOLOGY LECTURE - 04

CSS Border

CSS BORDER

- With the help of CSS **every element (not only table) can have borders.**



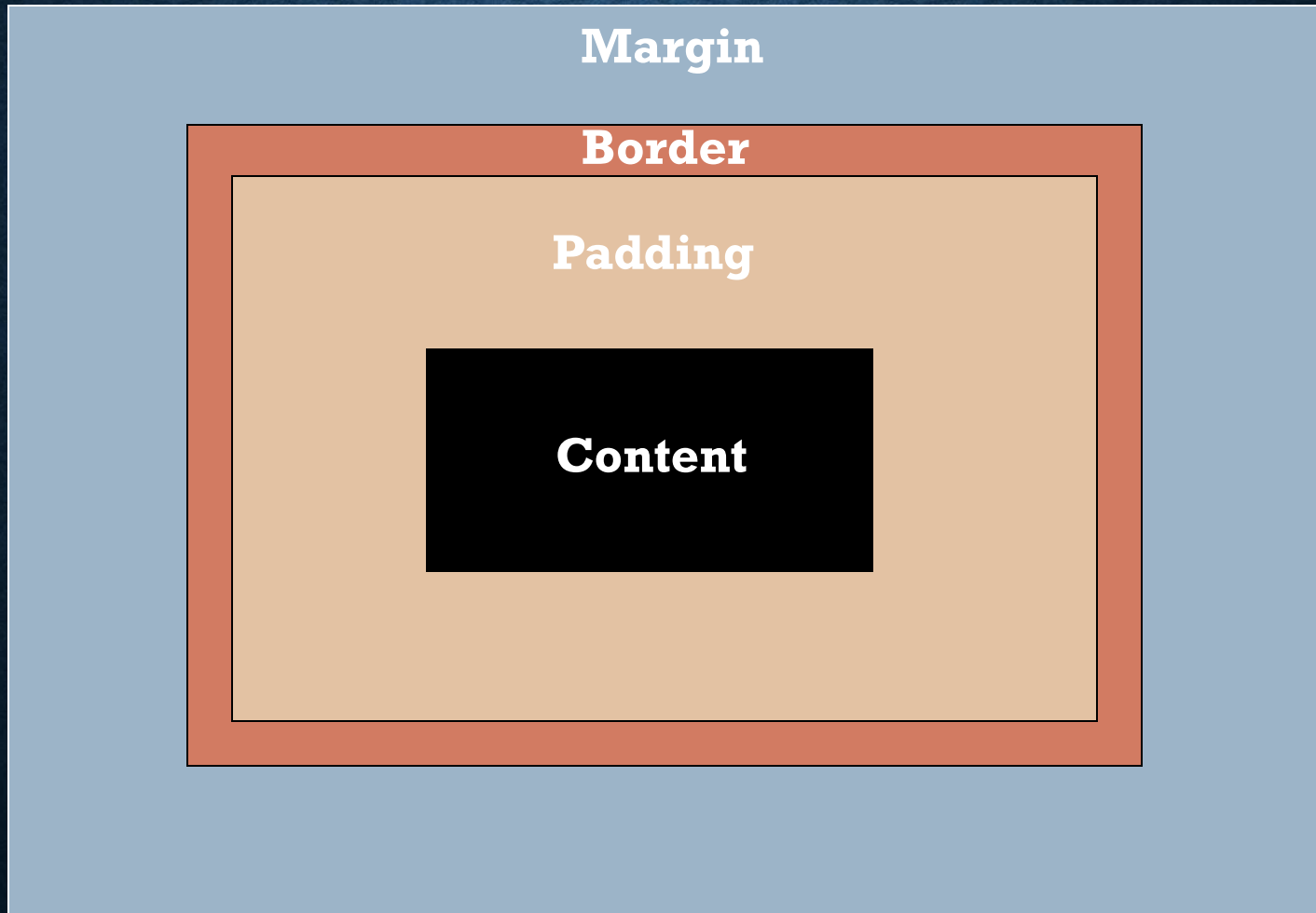
CSS BORDER

- There are numerous types of border styles.
- Example
 - ***border-style:** solid | dotted | dashed | double | groove etc;*
 - ***border-(top | right | bottom | left)-style:** solid | dotted | dashed | double | groove etc;*
 - ***border-width:** thin | thick | medium | px;*
 - ***border-color:** name | rgb(r,g,b) | #rrggbb*
 - ***border-radius:** px | 50%*

The CSS Box Model

THE CSS BOX MODEL

Every block element in CSS is effectively inside a box, and can have margins, paddings, borders and contents.

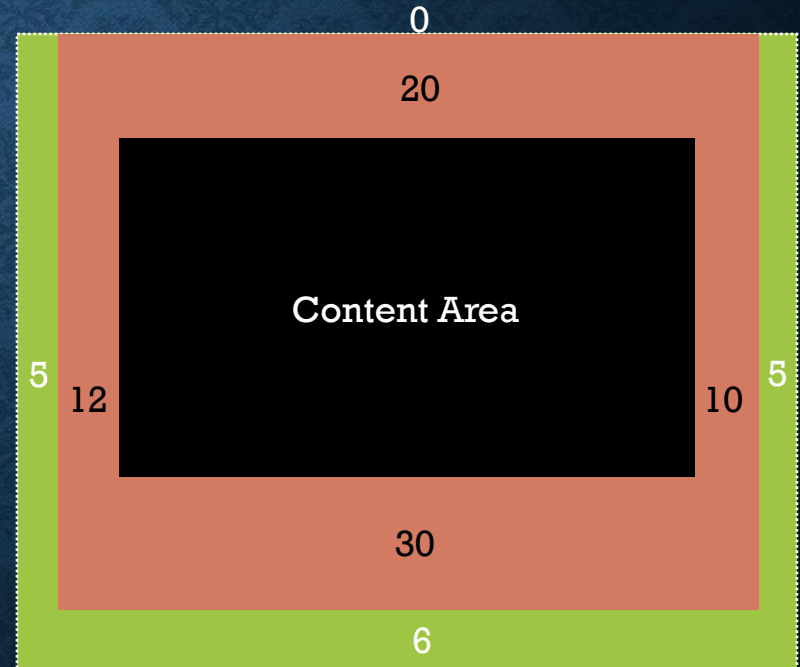


BOX MODEL TERMINOLOGY

- Every CSS box is divided into regions, consisting of:
 1. **Margin** - Clears an area around the border. The margin does not have a background color, and it is completely transparent
 2. **Border** - A border that lies around the padding and content. The border is affected by the background color of the box
 3. **Padding** - Clears an area around the content. The padding is affected by the background color of the box
 4. **Content** - The content of the box, where text and images appear

CSS SHORTHAND: MARGIN & PADDING

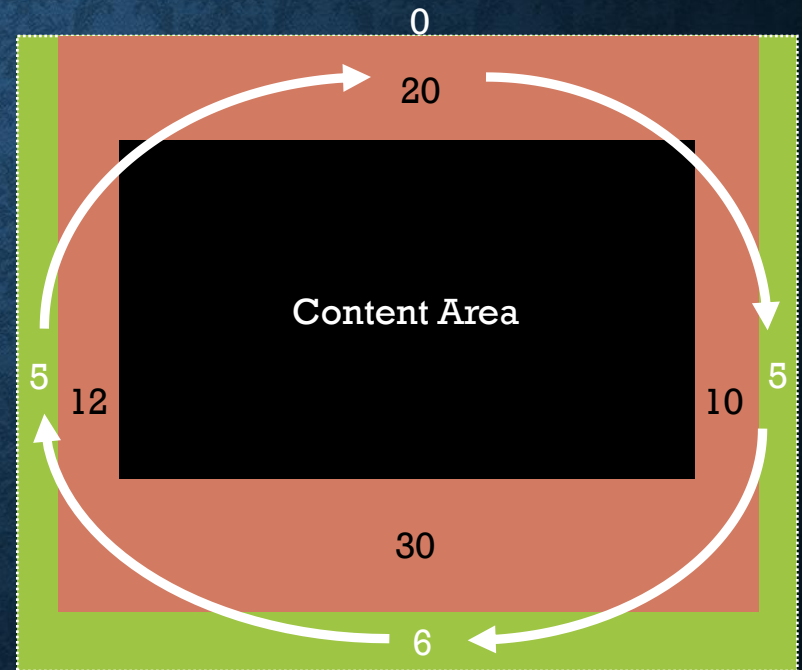
- For margin and padding (and others), CSS provides a number of **shorthand** properties that can save on writing lines and lines of code.
- Instead of writing this:
- ```
#container {
 margin-top: 0;
 margin-right: 5px;
 margin-bottom: 6px;
 margin-left: 5px;
 padding-top: 20px;
 padding-right: 10px;
 padding-bottom: 30px;
 padding-left: 12px;
}
```





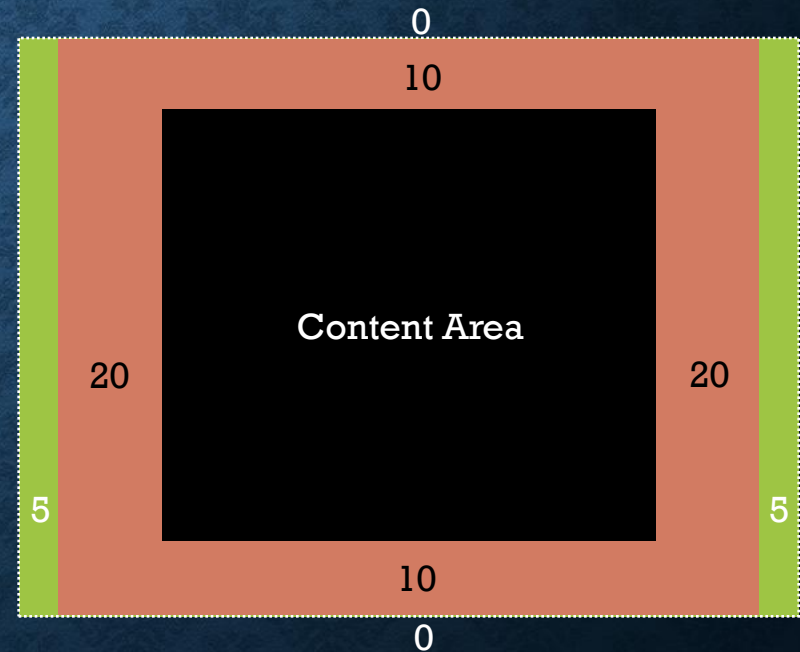
# CSS SHORTHAND: MARGIN & PADDING

- ...Its much easier to write this:
- `#container {  
padding: 20px 10px 30px 12px;  
margin: 0px 5px 6px 5px;  
}`
- The sequence **order** is always clockwise,  
starting from the top



# CSS SHORTHAND: MARGIN AND PADDING

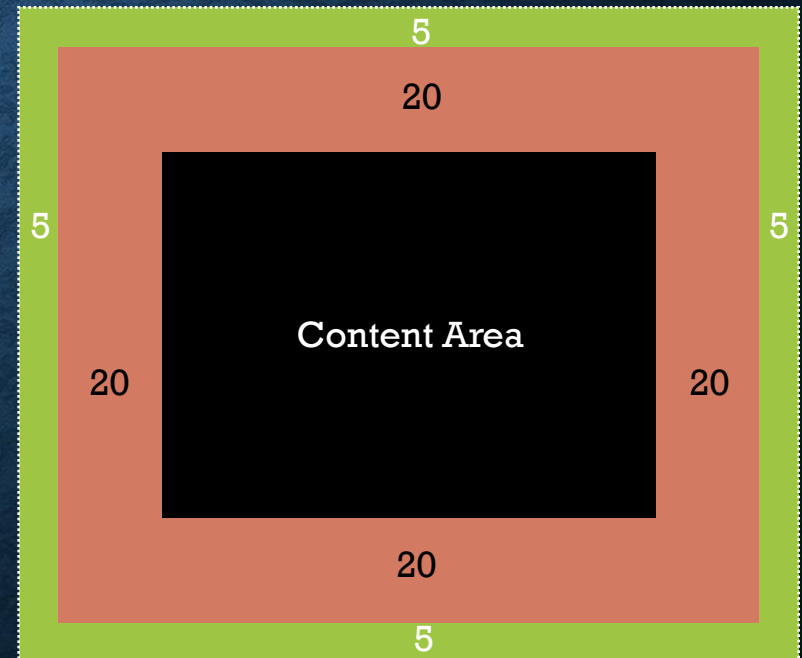
- And you can apply two values, example:
- `#container {  
padding: 10px 20px;  
margin: 0px 5px;  
}`
- The first value is applied to the **top and bottom**
- The second value is applied to the **left and right**





# CSS SHORTHAND: MARGIN AND PADDING

- You can also apply just one value, example:
- `#container {  
padding: 20px;  
margin: 5px;  
}`
- Which will apply the value specified equally on all 4 sides

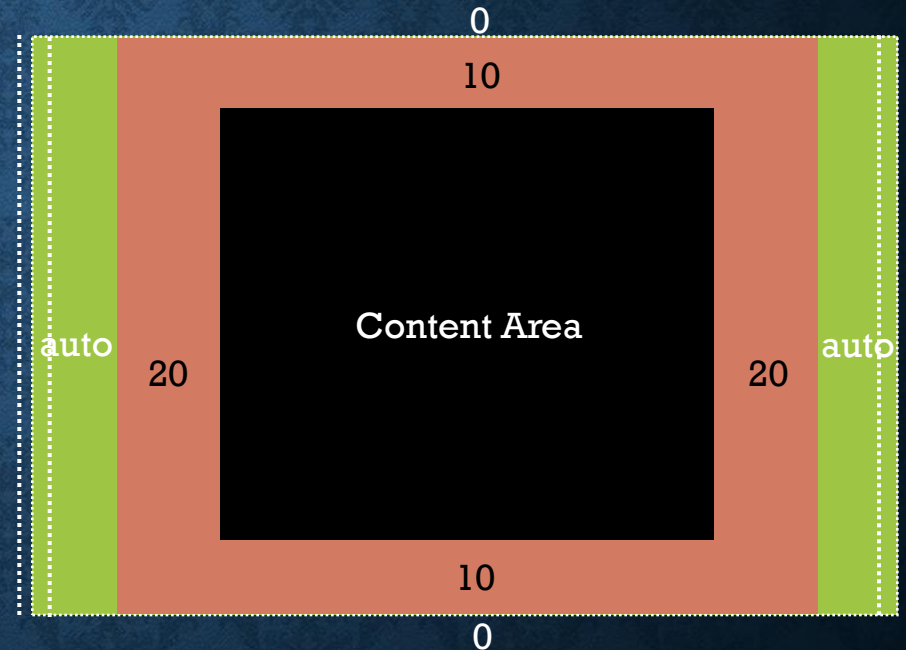


# MARGIN AND PADDING: AUTO

- A useful value to remember is 'auto':

- `#container {  
padding: 10px 20px;  
margin: 0px auto;  
}`

- Usually applied to the **left & right areas** of the margin property, auto is useful for **centering a block element** in the **browser window** (for relative positioned element)





# BOX MODEL TYPE

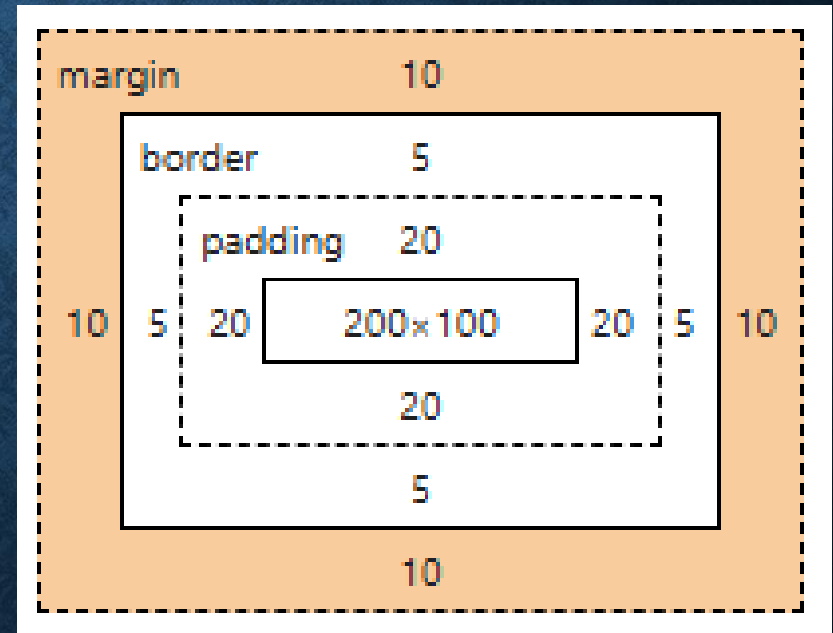
- Box Model Types:
- **Standard Box Model** (default)
- Alternative Box Model:

**CSS: `box-sizing: border-box;`**

# STANDARD BOX MODEL

- The width and height properties apply **only to the content**.
- Total width** = **width** + padding + border + margin
- Total height** = **height** + padding + border + margin

```
.box {
 width: 200px;
 height: 100px;
 padding: 20px;
 border: 5px solid black;
 margin: 10px;
}
```

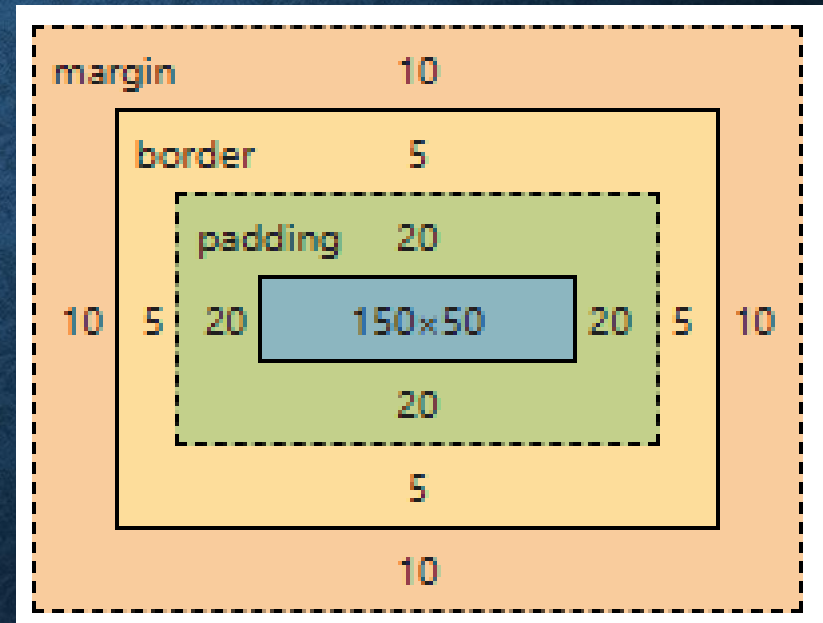




# BORDER-BOX MODEL

- The **width** and **height include** content, padding, and border (**excluding margin**)
- Makes layout calculations **easier**.

```
.box {
 width: 200px;
 height: 100px;
 padding: 20px;
 border: 5px solid black;
 margin: 10px;
 box-sizing: border-box;
}
```



Use Chrome **DevTools** to **inspect and debug** the CSS.

# OVERFLOW AND CURSOR

- Changing box size may not fit its content properly. To control this, we can use overflow
  - Syntax: *Overflow: visible | hidden | scroll | auto;*
    1. **Visible (default)**: content is visible outside of the borders
    2. **Hidden**: content is hidden outside of the borders
    3. **Scroll**: content is hidden outside of the border but can be explored using a scrolling handle
    4. **Auto**: scrolling handle appears automatically whenever needed.
- Overflow is different from white-space you have learned in text formatting in the previous lecture, how?*
- **cursor**: *text | pointer | progress | grab | wait | zoom-in | crosshair etc.*  
*// to change cursor type*



# A Simple Design Challenge

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices nec congue eget, auctor vitae massa. Fusce luctus vestibulum augue ut aliquet. Mauris ante ligula, facilisis sed ornare eu, lobortis in odio. Praesent convallis urna a lacus interdum ut hendrerit risus congue. Nunc sagittis dictum nisi, sed ullamcorper ipsum dignissim ac. In at libero sed nunc venenatis imperdiet sed ornare turpis. Donec vitae dui eget tellus gravida venenatis. Integer fringilla congue eros non fermentum. Sed dapibus pulvinar nibh tempor porta. Cras ac leo purus. Mauris quis diam velit.



- **The normal flow:** Block elements are stacked, one above the other. Inline elements are placed next to each other
- How do we achieve the above design where one element wraps another?

# FLOAT: LEFT | RIGHT | NONE

Hello World.  
This is Md.  
Arman  
Hossain.  
Floating left.

This is first paragraph wrapping the float box. It started just after the float box's border and margin. This is the default behaviour.

This is the second paragraph. Still wrapping the float box. However, It I want to make it appeared as normal flow, i.e., in another line

```
p{
margin: 10px;
}
.myfloat{
float: left;
width: 100px;
border: 3px solid black;
margin: 5px;
}
```

```
1 <h2>Float Left</h2>
2
3 <p class="myfloat">Hello World. This is Md. Arman
 Hossain. Floating left. </p>
4
5 <p>This is first paragraph wrapping the float box. It
 started just after the float box's border and
 margin. This is the default behaviour. </p>
6 <p>This is the second paragraph. Still wrapping the
 float box. However, It I want to make it appeared
 as normal flow, i.e., in another line</p>
```

Clear: none | left | right | both; // To restore the normal flow (second paragraph)



**End**