

Week 3: Web Development	1
1. CSS pseudo-class	1
2. CSS Positions Property	2
3. Working with Fonts	2
4. CSS visibility	2
5. CSS Float	2
6. CSS Outline	3
7. CSS Z-Index	4
10. CSS Overflow	4
11. Flex	5
12. Grid	5
13. Tailwind in html	6
14. Github	6
Lab Tasks	8
Task 1 : Flex and Grid	8
Task 2 : Apply the Covered Topics in this Prev Task	9

Week 3: Web Development

1. CSS pseudo-class

is used to define a special state of an element.

:hover:

Selects and styles an element when the user hovers over it.

:active:

Selects and styles an element when it is being activated (e.g., clicked).

:focus:

Selects and styles an element when it has focus, typically used with form elements.

:visited:

Selects and styles a link that has been visited by the user.

:first-child:

Selects the first child element of its parent.

:last-child:

Selects the last child element of its parent.

:nth-child(n):

Selects elements based on their position in the parent, where n can be a specific number, an equation, or a keyword.

2. CSS Positions Property

<https://css-tricks.com/almanac/properties/p/position/>

3. Working with Fonts

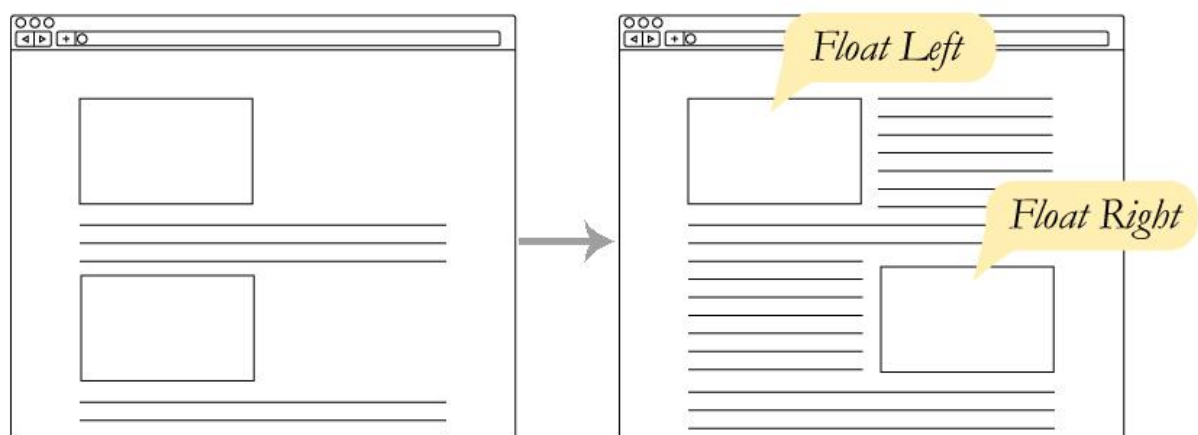
- Choose three different Google Fonts that you find appealing.
- Create an HTML page with three sections, each dedicated to one font.
- Apply a different font to each section using the font-family property in CSS.
- Experiment with font size, weight, and style.
- Include text in each section to showcase the chosen font.

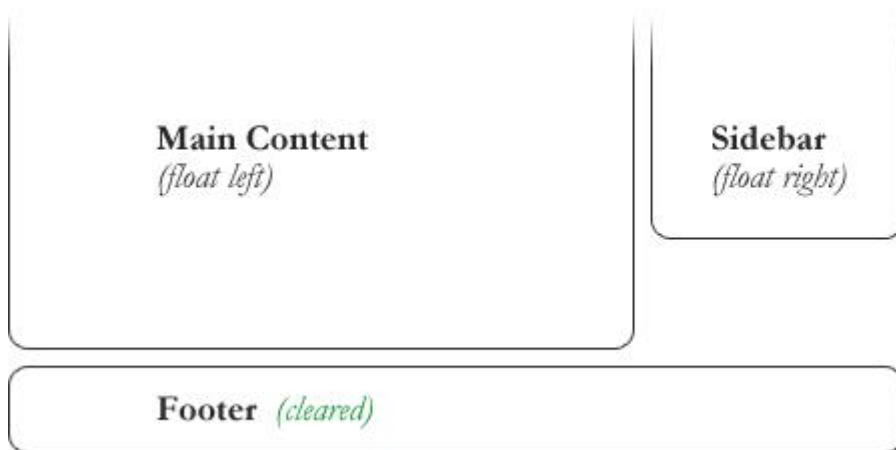
font-family: "Lucida Console", "Courier New",
monospace;

4. CSS visibility

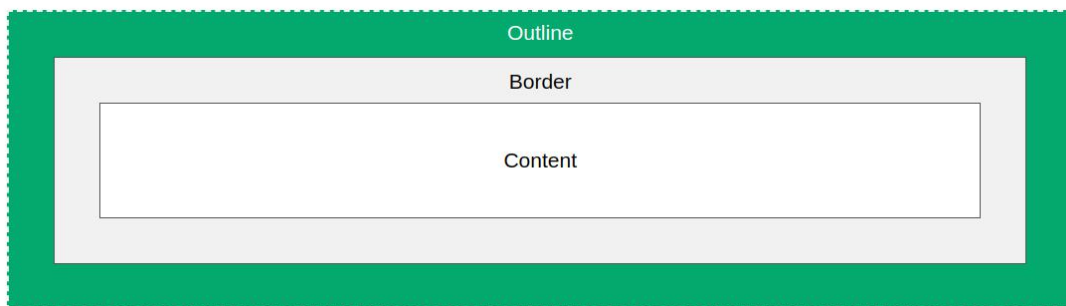
visible Default value. The element is visible
hidden The element is hidden (but still takes up space)

5. CSS Float





6. CSS Outline



CSS: Outline vs Border

blog.kritikapattalam.com

@KritikaPattalam

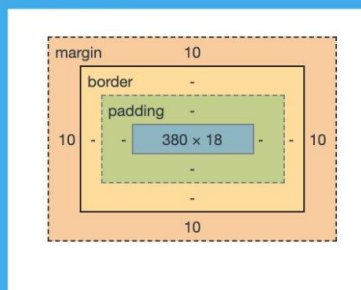
- Outlines never take up space, as they are drawn outside of an element's content. Whereas borders take up space.

```

1 <div style="width: 400px">
2   <div class="outline-div">hello</div>
3   <div class="border-div">border</div>
4 </div>
5 <style>
6   .outline-div,
7   .border-div {
8     background: #ccc;
9     margin: 10px;
10  }
11  .outline-div {
12    outline: 1px solid red;
13  }
14  .border-div {
15    border: 1px solid red;
16  }
17 </style>

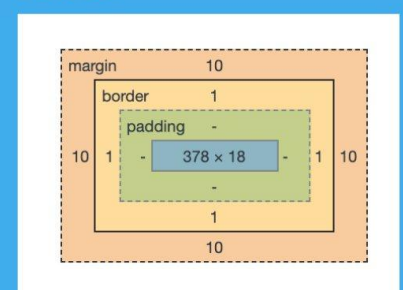
```

Outline:



Parent div = margin + width

Border:



Parent = margin + border + width

- Outline: Parent(400) = marginLeft(10) + width(380) + marginRight(10)
- Border: Parent(400) = marginLeft(10) + borderLeft(1) + width(380) + borderRight(1) + marginRight(10)

An outline is a line drawn outside the element's border.

7. CSS Z-Index



8. CSS Display

inline Displays an element as an inline element

block Displays an element as a block element

contents Makes the container disappear, making the child elements children of the element the next level up in the DOM

flex Displays an element as a block-level flex container

grid Displays an element as a block-level grid container

inline-block Displays an element as an inline-level block container. The element itself is formatted as an inline element, but you can apply height and width values

none The element is completely removed

9. CSS Icons

SVG, Images

10. CSS Overflow

visible - Default. The overflow is not clipped. The content renders outside the element's box

hidden - The overflow is clipped, and the rest of the content will be invisible

scroll - The overflow is clipped, and a scrollbar is added to see the rest of the content

auto - Similar to scroll, but it adds scrollbars only when necessary

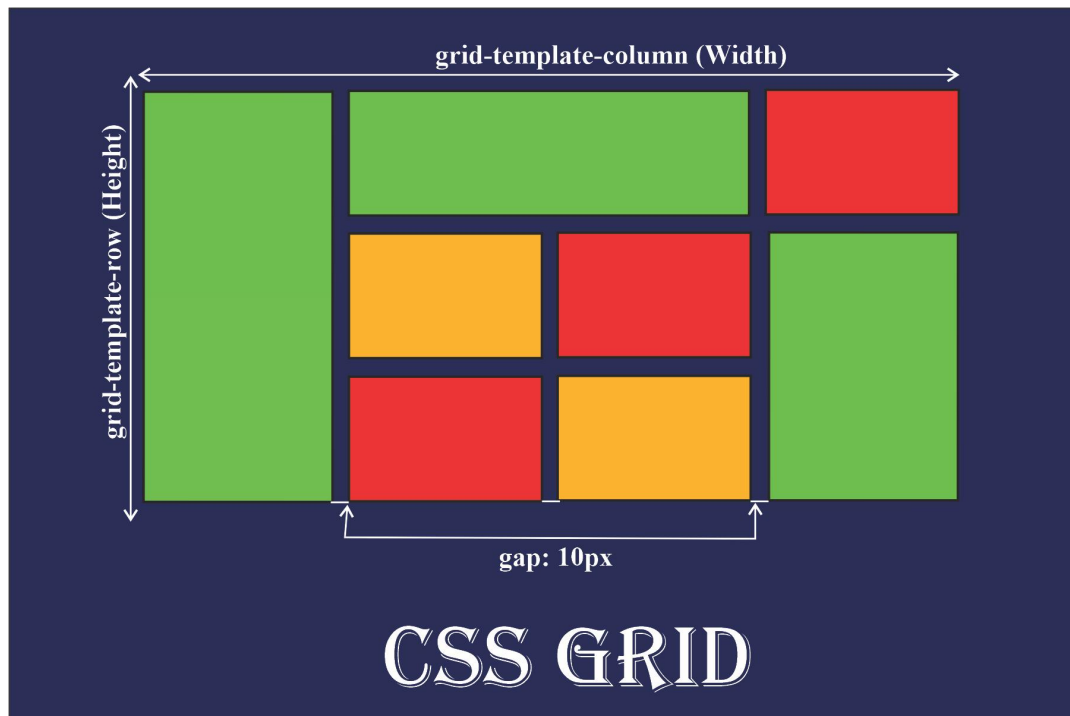
11. Flex

<https://css-tricks.com/snippets/css/a-guide-to-flexbox/>



12. Grid

<https://css-tricks.com/snippets/css/complete-guide-grid/>



13. Tailwind in html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Your HTML Page</title>
  <link
href="https://cdn.jsdelivr.net/npm/tailwindcss@2.2.19/dist/tailwind.min.css"
rel="stylesheet">
</head>
<body>
  <h1 class="text-4xl font-bold text-blue-600">Hello, Tailwind!</h1>
</body>
</html>
```

14. Github

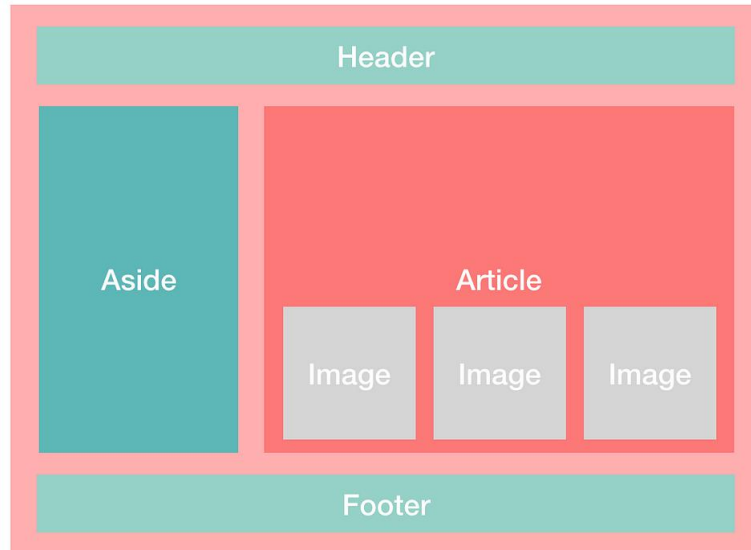
###1.Clone Repos

###2.Commit and Push

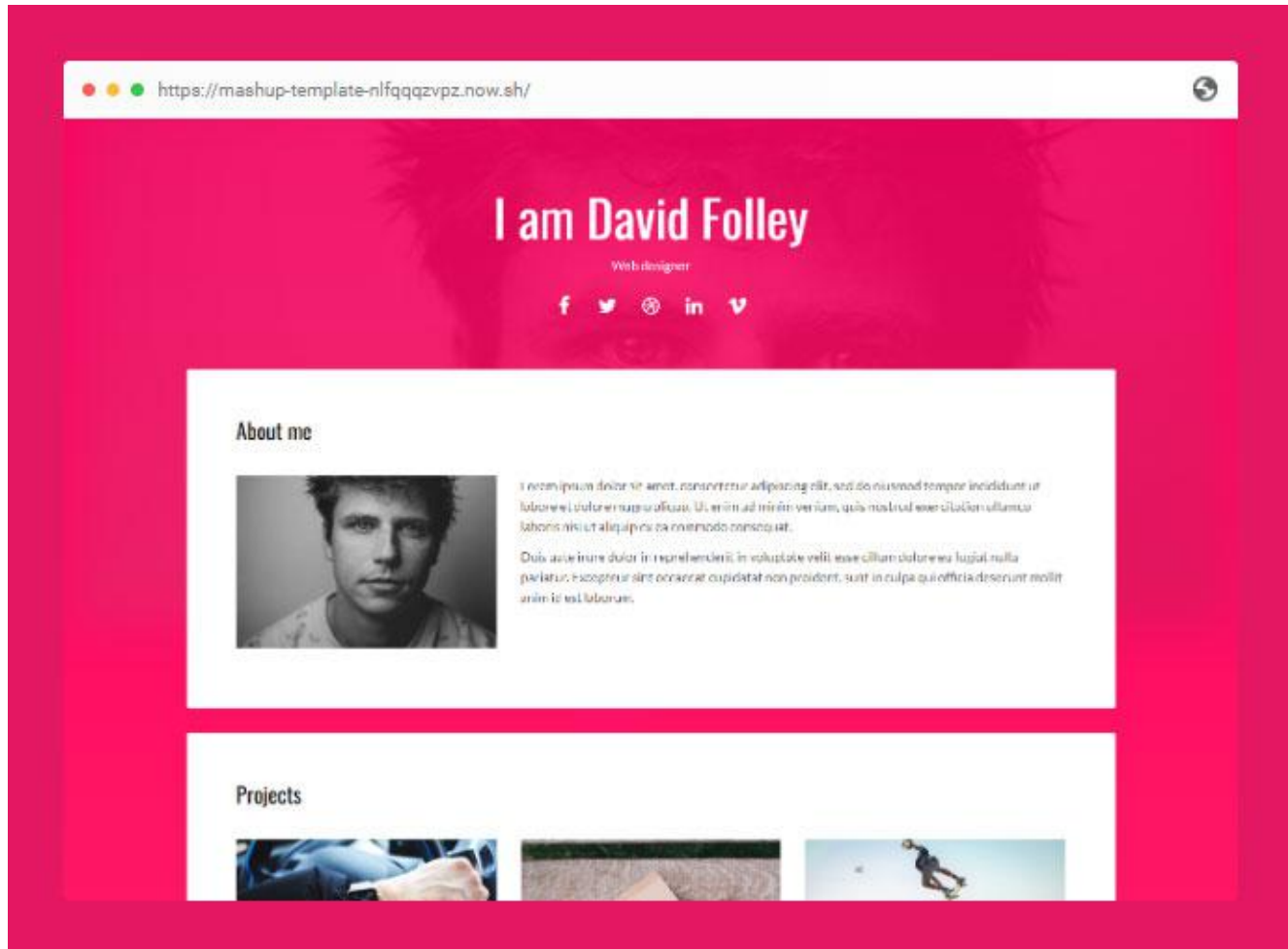
###3.Add Readme to Project and Images to it

Lab Tasks

Task 1 : Flex and Grid



Task 2 : Apply the Covered Topics in this Prev Task



Objective:

Create a personal portfolio page that showcases your skills, projects, and contact information.

Requirements:

Design a layout for your portfolio using HTML, CSS, Pseudo classes .

Apply CSS to style the portfolio, ensuring a clean and visually appealing design.

Include sections such as an introduction, skills, projects with images, and a contact form.