

CSS

Important Note for Students:

This list of questions and answers is like a helpful guide for your upcoming interview. It's designed to give you an idea of what to expect and help you get ready.

But remember:

1. **Variety of Questions:** The same questions can be asked in many different ways, so don't just memorise the answers. Try to understand the concept behind each one.
 2. **Expect Surprises:** There might be questions during your interview that are not on this list. It's always good to be prepared for a few surprises.
 3. **Use This as a Starting Point:** Think of this material as a starting point. It shows the kind of questions you might encounter, but it's always good to study beyond this list during your course.
-

Founder of CSS: Håkon Wium Lie (1996)

1. What is css?

In publishing and graphic design, Lorem ipsum is a placeholder text commonly used to demonstrate the visual form of a document or a typeface without relying on meaningful content. Lorem ipsum may be used as a placeholder before the final copy is available.

- CSS stands for **Cascading Style Sheets (Full Form)**
- CSS describes how HTML elements are to be displayed on screen, paper, or in other media.
- CSS saves a lot of work. It can control the layout of multiple web pages all at once.
- CSS is used to define styles for your web pages, including the design, layout and variations in display for different devices and screen sizes.

2. How many types of CSS?

There are 3 types of css

- Inline Css
- Internal Css
- External Css

3. Inline CSS :

Inline CSS is the technique to define the single element with the insert style sheets in an HTML document.

Here is the basic syntax example with output:

INPUT :

```
<!DOCTYPE html>
<html>
  <head>
    <title>Inline css</title>
  </head>
  <body>
    <h1 style="color:blue;">This is a heading</h1>
  </body>
</html>
```

OUTPUT :

We can apply Inline CSS on this heading.

4. Internal CSS :

Internal CSS is one of the most widely used CSS forms for changing, styling, and modifying the unique styles of a single web page. You can use the internal CSS by integrating the <style> element in the <head> section of a HTML web page.

Internal CSS can be applied to the whole web page but not on multiple web pages and you can style several web pages by using the same code on every page.

Here is the basic syntax example with output:

INPUT :

```
<!DOCTYPE html>
<html>
  <head>
    <title>Internal CSS</title>
    <style>
      body{
        background-color: powderblue;
      }
      h1{
        color: blue;
      }
      p{
        color: red;
      }
    </style>
  </head>
  <body>
    <h1>This is a heading</h1>
    <p>This is a paragraph.</p>
  </body>
</html>
```

OUTPUT :

This is a heading

This is a paragraph.

5. External CSS :

The external style sheet is generally used when you want to make changes on multiple pages. It is ideal for this condition because it facilitates you to change the look of the entire web site by changing just one file.

It uses the <link> tag on every pages and the <link> tag should be put inside the head section
Here is the basic syntax example with output:

INPUT:

HTML file (index.html)

```
<!DOCTYPE html>
<html>
  <head>
    <title>Internal CSS</title>
    <link rel="stylesheet" href="style.css">
  </head>
  <body>
    <h1>This is a heading</h1>
    <p>This is a paragraph.</p>
  </body>
</html>
```

CSS file (style.css)

```
body {
  background-color: lightblue;
}
h1 {
  color: blue;
}
p {
  color:red;
}
```

OUTPUT:

This is a heading

This is a paragraph.

6. CSS Selectors

CSS selectors are used to "find" (or select) the HTML elements you want to style. We can divide CSS selectors into five categories:

- Simple selectors (select elements based on tag, id, class)
- Combinator selectors (select elements based on a specific relationship between them)
- Pseudo-class selectors (select elements based on a certain state)
- Pseudo-elements selectors (select and style a part of an element)
- Attribute selectors (select elements based on an attribute or attribute value)

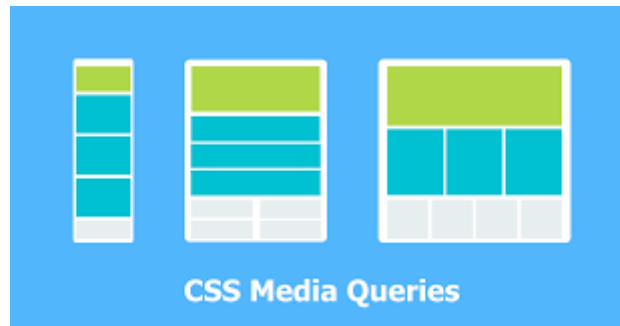
7. Simple selectors :

- **CSS id Selector:** The id attribute is a unique identifier that is used to specify the document. It is used by CSS and JavaScript to perform a certain task for a unique element. In CSS, the id attribute is written using the # symbol followed by id
- **CSS Class Selector:** CSS syntax contains a selector, and a class is exactly that. It is needed to stylize HTML elements – including changing colors, fonts, or the size of a text. If you want to use a class, use a full stop (.) followed by the class name in a style block. Next, use a bracket called a declaration block that contains the property to stylize the element, such as text color or text size.

8. What is @media?

We can use @media for responsive web pages in every device like mobile, tablet, and laptop.

Example:



9. What Is a @Keyframes in CSS?

A keyframe in animation and filmmaking is a drawing that defines the starting and ending points of any smooth transition. ... A sequence of keyframes defines which movement the viewer will see, whereas the position of the keyframes on the film, video, or animation defines the timing of the movement.

10. Why do we use CSS?

We use CSS because of the following reasons:

- CSS saves time: You can write CSS once and reuse the same sheet on multiple HTML pages.
- Easy Maintenance: To make a global change simply change the style, and all elements in all the webpages will be updated automatically.
- Search Engines: CSS is considered a clean coding technique, which means search engines won't have to struggle to "read" its content.
- Superior styles to HTML: CSS has a much wider array of attributes than HTML, so you can give a far better look to your HTML page in comparison to HTML attributes.
- Offline Browsing: CSS can store web applications locally with the help of an offline cache. Using of this we can view offline websites.

11. Text Properties :

- Font Family : poppins, roboto
- Font-size : 16px
- Color : red;
- Font-weight : 100 to 900 (normal to bold)
- Text-align : left, center, right, justify

- Line-height : space between 2 line
- Word-spacing : space between 2 word
- Letter-spacing : space between 2 letter
- Text-decoration : underline, line-through, overline, none
- Font-style : italic, bold
- Text-transform : uppercase, lowercase, capitalise
- Text-indent : use for space before the paragraph

12. What is @font-face in css?

@font-face is a CSS rule that allows you to input your own font to appear on a website even when the particular font is not installed on the visitor's computer. The most important thing about this rule is that it opened a whole new world for designers. You are welcome to use just about any font you like and as many as you like. However, in the nature of design principles I do urge you to use a max of three fonts per design.

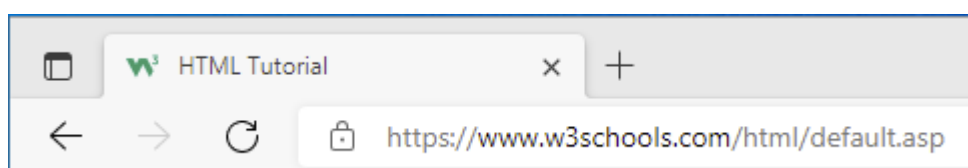
13. What is Favicon and how to attach it?

A favicon is a small image displayed next to the page title in the browser tab. You can use any image you like as your favicon.

INPUT :

```
<!DOCTYPE html>
<html>
  <head>
    <title>Internal CSS</title>
    <link rel="icon" href="image/(icon image name )">
    <link rel="stylesheet" href="style.css">
  </head>
  <body>
    <h1>This is a heading</h1>
    <p>This is a paragraph.</p>
  </body>
</html>
```

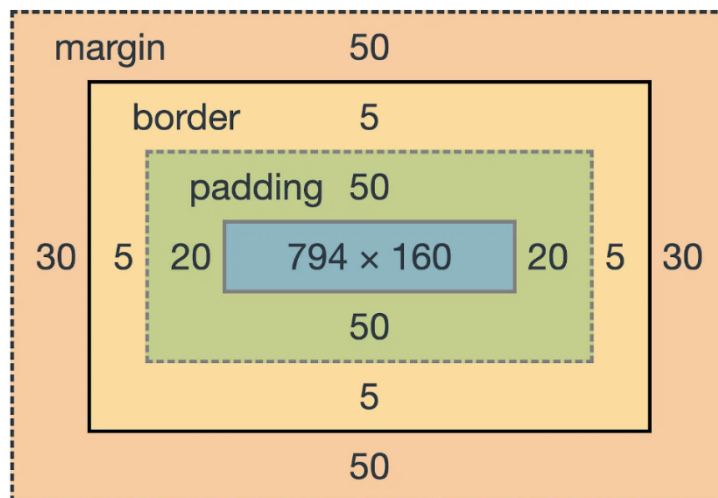
OUTPUT :



14. What is the box model in CSS?

The box model in CSS is the way in which the browser calculates the size and position of elements on a web page. The model consists of a content box, surrounded by padding, a border, and a margin. The size of the content box is determined by the width and height properties of the element. The padding, border, and margin add extra space around the element. The width and height properties include the padding, border, and margin, so the total size of the element is the sum of all four properties.

Example :



15. What's the difference between margin, padding, and border in css?

The css box model refers to the design and layout of the boxes that make up the content of a web page. Margin is the space outside of the box, padding is the space inside the box, and border is the line around the box.

16. What is padding?

Padding is the space that's inside the element between the element and the border. Padding goes around all four sides of the content and you can target and change the padding for each side (just like a margin).

17. What is the margin?

Margin is the space between the border and the next element of your design. Think of the space outside the border and between it and the other elements. This is all the margin. Margin goes around all four sides of the content and you can target and change the margin for each side.

18. What are the different CSS border properties?

CSS border properties allow us to set the style, color, and width of the border.

- **Border Style:** The border-style property specifies the type of border. None of the other border properties will work without setting the border style.
- **Border Width:** Border width sets the width of the border. The width of the border can be in px, pt, cm or thin, medium and thick.
- **Border Color:** This property is used to set the color of the border. Color can be set using the color name, hex value, or RGB value. If the color is not specified, the border inherits the color of the element itself.
- **Border : 1px solid black;**

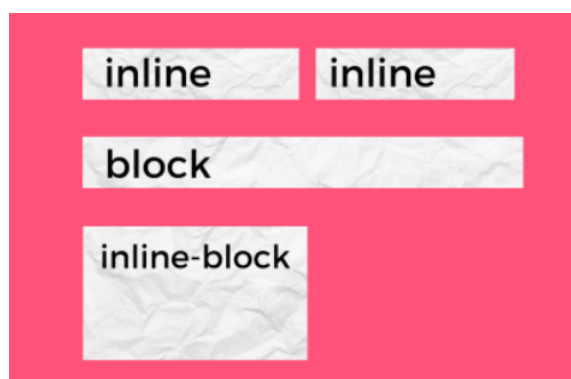
19. Display property?

display: none | inline | block | inline-block;

So, the difference between **display: "none"**; and **visibility: "hidden"**;, right from the name itself we can tell the difference as **display: "none"**, completely gets rid of the tag, as it had never existed in the HTML page whereas **visibility: "hidden"**;, just makes the tag invisible it will still be on the HTML page occupying space it's just invisible.

20. Difference between inline, inline block and block?

- **Inline** The element doesn't start on a new line and only occupies just the width it requires. You can't set the width or height.
- **Inline-block** It's formatted just like the inline element, where it doesn't start on a new line. BUT, you can set width and height values.
- **Block** The element will start on a new line and occupy the full width available. And you can set width and height values.



21. Explain all border style property?

None, dotted, double, solid, dashed, groove, ridge, inset, outset, hidden

22. What is the border radius?

The border-radius CSS property rounds the corners of an element's outer border edge. You can set a single radius to make circular corners, or two radii to make elliptical corners.

23. Why do we use max-width in container?

The max-width property in CSS is used to define the maximum width of an element. The value of the width cannot be larger than the value by max-width. If the content is larger than the max-width then it will go to the next line and if the content is smaller than the max-width then it has no effect.

24. What is the difference between max-width and min-width?

With min-width, styles START and continue forever as long as min-width is met, and no max-width is specified. Max-width is the maximum width at which a style will continue to be applied. After that, the style will STOP being applied. (Have to be ordered from largest to smallest to work properly, regular styles first).

25. What is a container?

A container is a fundamental concept in web development and design. It refers to an HTML element that wraps and holds other elements or content within a webpage. Containers are often used to organise and style content, and they play a crucial role in creating responsive and structured layouts.

26. What is the background property in CSS used for?

The background property in CSS is used to set various background-related styles for an element, such as color, image, position, and repetition. It allows you to control the visual appearance of the background of an HTML element.

27. How can you set the background image of an HTML element using CSS?

To set the background image of an HTML element using CSS, you can use the background-image property.

For example:

```
.element {  
    background-image: url('path/to/image.jpg');  
}
```

28. Background properties with value ?

- Background-color : black, #000, rgba(0,0,0,0.5);
- Background-size : cover, contain;
- Background-attachment: scroll, fixed;
- Background : linear-gradient(79deg, rgba(203,194,18,1) 0%, rgba(52,207,187,1) 47%, rgba(60,97,105,1) 100%);
- Background-repeat : no-repeat;
- Background-position : center, top center, 20% 50%;
- Background-image : url('../img/xyz.png');
- Background-blend-mode : darken, lighten, brightness, normal, contrast

29. What does the CSS float property do?

The float property is used to make an element float to the left or right within its parent container, allowing other content to wrap around it.

30. How do you clear the float in CSS?

To clear a float, you can use the clear property. Adding clear: both; to an element will prevent it from floating next to any floated elements.

31. What are the common issues with using float in layouts?

Common issues with using float include layout collapsing, overlapping content, and unequal column heights. These can be mitigated using clear fix techniques or newer layout methods like Flexbox or Grid.

32. What does the CSS object-fit property do?

The object-fit property is used to control how an element, such as an image or video, resizes and fits within its container.

33. How many values can the object-fit property take, and what do they do?

The object-fit property can take five values: fill, contain, cover, none, and scale-down.

- fill: The element fills the container, ignoring aspect ratio.
- contain: The element maintains aspect ratio and fits inside the container.
- cover: The element maintains aspect ratio and covers the entire container, possibly cropping parts of the element.

- none: The element retains its original size, and any overflowing content will be visible outside the container.
- scale-down: The element's size is scaled down to fit inside the container if it's larger than the contain value, otherwise, it behaves like none.
- **object-fit :**



- **object-position :**



34. What is a box shadow in CSS?

A box shadow is a visual effect that adds a shadow behind an element in CSS, making it appear elevated or floating.

35. What properties are used to create a box shadow?

The box-shadow property is used to create a box shadow effect. It takes values for horizontal offset, vertical offset, blur radius, spread radius, and the shadow color.

36. How do you apply a box shadow to an element?

You can apply a box shadow to an element using the box-shadow property in your CSS code. For example:

```
.box { box-shadow: 2px 2px 4px #000000; }
```

This will create a box shadow with a horizontal offset of 2 pixels, a vertical offset of 2 pixels, a blur radius of 4 pixels, and a black color.

37. Can an element have multiple box shadows?

Yes, an element can have multiple box shadows by separating them with commas in the box-shadow property. For example

```
.box { box-shadow: 2px 2px 4px #000000, -2px -2px 4px #ffffff; }
```

This will create two box shadows, one with a black shadow and another with a white shadow.

38. How can you make a box shadow appear only on specific sides of an element?

You can control the box shadow on specific sides of an element using the inset keyword in the box-shadow property. For example:

```
.box { box-shadow: 2px 2px 4px #000000 inset; }
```

This will create an inset shadow on the top and left sides of the element.

Box shadows are a versatile way to add depth and visual appeal to your web design, allowing you to create various effects and make elements stand out on the page.

39. What is the CSS position property used for?

The position property is used to control the positioning of an element within its containing element or the viewport.

40. How many values can the position property take, and what do they mean?

The position property can take four values: static, relative, absolute, and fixed.

- static: Default value, the element follows the normal document flow.
- relative: The element is positioned relative to its normal position.
- absolute: The element is positioned relative to its closest positioned ancestor.
- fixed: The element is positioned relative to the viewport and remains fixed even during scrolling.

- Sticky : The position: sticky; value is used to create elements that behave like relative until they reach a specified offset, then they become fixed until the scrolling reaches a defined end point.

41. What is the z-index property used for in combination with position?

The z-index property is used to control the stacking order of elements with a position value other than static. It determines which elements appear in front of or behind other elements.

42. Define z-index.

This is one of the most frequently asked CSS interview questions. Z-index is used to specify the stack order of elements that overlap each other. Its default value is zero and can take both negative and positive values. A higher z-index value is stacked above the lower index element. It takes the following values- auto, number, initial, and inherit.



43. What is CSS overflow?

The CSS overflow controls the big content. It tells whether to clip content or to add scroll bars.

The overflow contains the following property:

- Visible: The content is not clipped and is visible outside the element box.
- Hidden: The overflow is clipped and the rest of the content is invisible.
- Scroll: The overflow is clipped but a scrollbar is added to see the rest of the content. The scrollbar can be horizontal or vertical.
- Auto: It automatically adds a scrollbar whenever it is required
- Overflow-x and Overflow-y: This property specifies how to change the overflow of elements. x deals with horizontal edges and y deals with vertical edges.

44. Differentiate between absolute and relative in CSS.

This table compares various courses offered by Simplilearn, based on several key features and details. The table provides an overview of the courses' duration, skills you will learn, additional benefits, among other important factors, to help learners make an informed decision about which course best suits their needs.

44. What are ::before and ::after in CSS?

::before and ::after are pseudo-elements in CSS that allow you to insert content before and after an element's content, respectively.

45. What is the syntax for using ::before and ::after?

The syntax is as follows:

```
selector::before {  
  content: "Content to be inserted before";  
}  
  
selector::after {  
  content: "Content to be inserted after";  
}
```

46. What are pseudo-elements in CSS?

Pseudo-element in CSS is used to add style to specified parts of an element. Example: Using style before or after an element.

```
selector::pseudo-element {  
  property:value;  
}
```

- **::before Pseudo-element:** It is used to add some CSS property before an element when that element is called.
- **::after Pseudo-element:** It is used to add some CSS property after an element when that element is called.

In CSS, the before and after pseudo-elements are used to insert content before or after an element's content, respectively, without the need to modify the HTML markup. They are part of the CSS content module and are commonly used in conjunction with the content property.

Here's a brief explanation of Before:

::before: This pseudo-element allows you to insert content before the content of the selected element. It is often used to add decorative elements or icons to an element.

```
.selector::before {  
  content: "→";  
}
```

Here's a brief explanation of After:

- **::after:** This pseudo-element allows you to insert content after the content of the selected element. Like ::before, it is used for decorative or additional content.

```
.selector::after {  
  content: "←";  
}
```

- In these examples, the CSS rules add a right-pointing arrow before the element with the class .selector and a left-pointing arrow after it, respectively.
- Remember that ::before and ::after are pseudo-elements, so their content is not part of the actual HTML structure and will not affect accessibility or be accessible by JavaScript. They are purely presentational additions through CSS.

47. What are pseudo-classes in CSS?

A Pseudo class in CSS is used to define the special state of an element. It can be combined with a CSS selector to add an effect to existing elements based on their states. For Example, changing the style of an element when the user hovers over it, or when a link is visited. All of these can be done using Pseudo Classes in CSS.

Note that pseudo-class names are not case-sensitive.

```
selector: pseudo-class{  
  property: value;  
}
```

There are many Pseudo-classes in CSS but the ones which are most commonly used are as follows:

- **:hover Pseudo-class:** This pseudo-class is used to add a special effect to an element when our mouse pointer is over it. The below example demonstrates that when your mouse enters the box area, its background color changes from yellow to orange
- **:focus Pseudo-class:** This pseudo-class is used to select an element that is currently focussed by the user. It works on user input elements used in forms and is triggered as soon as the user clicks on it. In the following example, the background color of the input field which is currently focused changes.

48. What is CSS Transition?

- A CSS transition is the simplest way to create animations on your site. You can do some really cool stuff without a line of JavaScript, and they are really easy to learn.
- Many moons ago, if web developers wanted to animate their websites, all they really had were gifs and marquee text. Now we have CSS transitions - and all is well with the world.
- And what's really awesome is, to get up and running you really only need to know about two properties - CSS transition-duration and CSS transition-property.

49. What is CSS transition duration?

- transition-duration **is the CSS property that controls how long the transition from one state to another should take**. So if you want to change the background color of an element, transition-duration will make that change happen gradually, fading smoothly from one color to the other.
- The default value for transition-duration is 0s - the element will just snap instantly from the first state to the second. So in a sense, transition-duration is the master transition property - without it, nothing works.

50. What is the transition timing function in css?

The transition-timing-function property specifies the speed curve of the transition effect.

- Transition-timing-function : ease;
- Transition-timing-function : ease-in;
- Transition-timing-function : ease-out;
- Transition-timing-function : ease-in-out;
- Transition-timing-function : linear;
- Transition-timing-function : cubic-bezier;

51. How do you apply an element using transform?

- You can rotate an element using the rotate function within the transform property:

```
.rotate-element {  
  transform: rotate(45deg);  
}
```

- You can scale an element using the scale function within the transform property:

```
.scale-element {  
  transform: scale(1.5);  
}
```

- You can translate an element using the translate function within the transform property:

```
.translate-element {  
  transform: translate(50px, 30px);  
}
```

- You can skew an element using the skew function within the transform property:

```
.skew-element {  
  transform: skew(180deg);  
}
```

52. What is browser support?

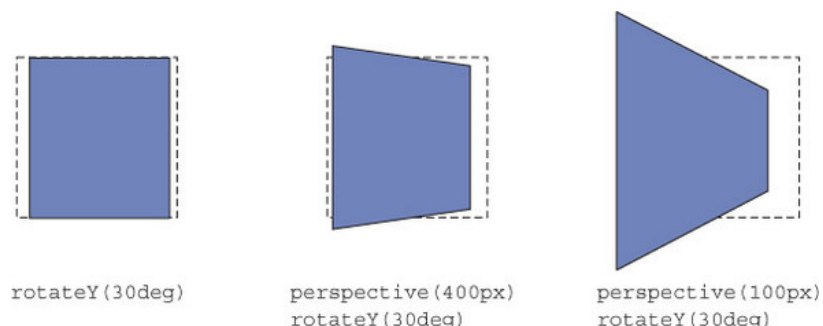
Browser support refers to how well a web browser is compatible with and correctly displays the various technologies and features used to create and style websites, such as HTML, CSS, and JavaScript. Different browsers may have varying levels of support for these technologies, which can impact how a website looks and functions across different browsers and devices.

Firefox (-moz-), Chrome, Safari,opera (-webkit-)

- **Chrome:** Generally strong support for modern CSS features.
- **Firefox:** Good support for a wide range of CSS specifications.
- **Safari:** Good CSS support, especially on macOS and iOS.
- **Edge:** Solid support due to Chromium engine adoption.
- **Internet Explorer (IE):** Phased out; limited CSS support.
- **Mobile Browsers:** Good support on mobile platforms.

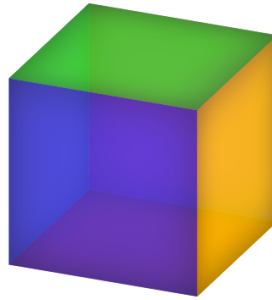
53. What is perspective?

In CSS (Cascading Style Sheets), perspective is a property that is used in conjunction with 3D transformations to create a sense of depth and three-dimensionality in a web page element. It defines the distance at which the viewer is positioned from the object being transformed. This property is primarily used with the transform property to apply 3D transformations like rotations and translations. It helps achieve more realistic and immersive visual effects on web elements.



54. What is the transform style?

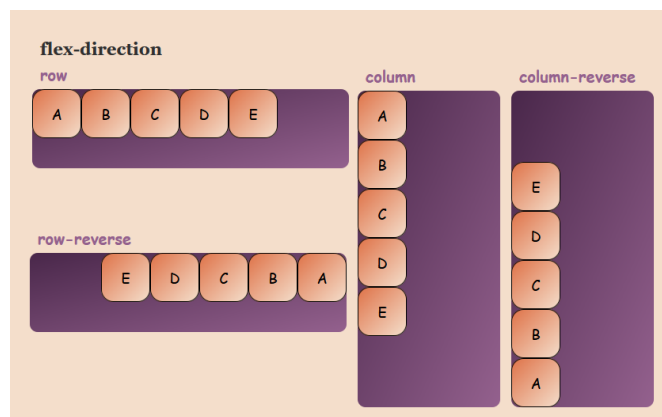
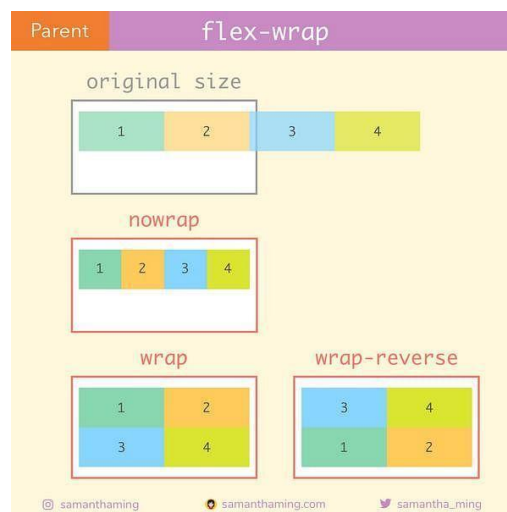
- In CSS, the transform-style property is used to define how child elements of a 3D transformed element are displayed in relation to the parent's 3D space.
- preserve-3d: Child elements maintain their individual 3D transformations and are displayed in their own 3D space within the parent's 3D environment.
- In summary, transform-style determines how child elements interact with the 3D transformations of their parent element in CSS.

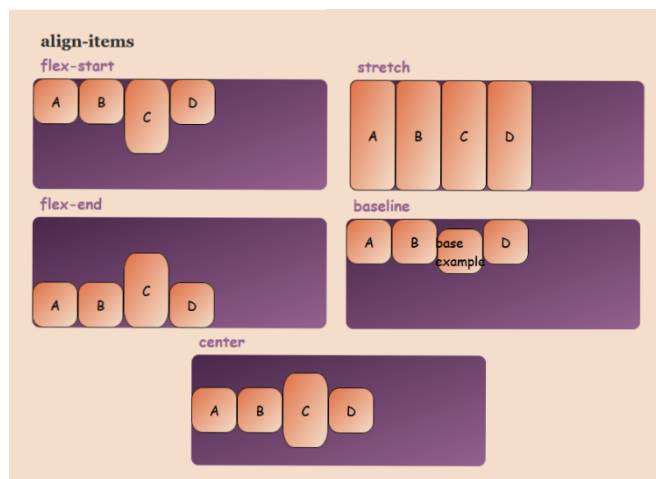
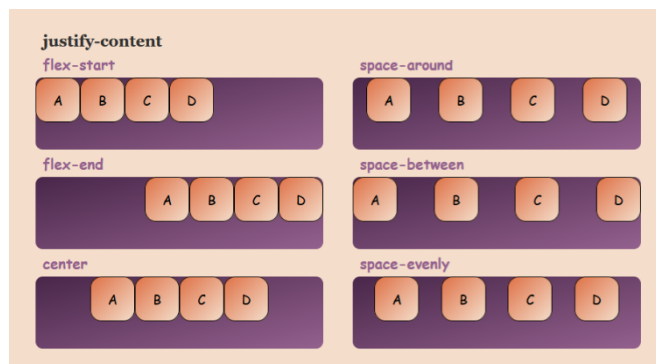
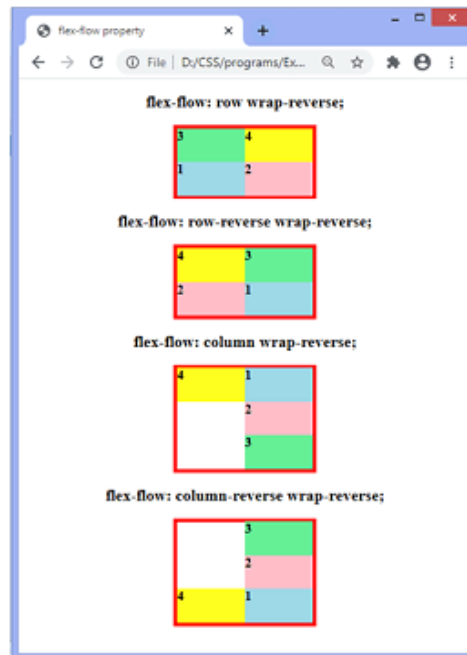


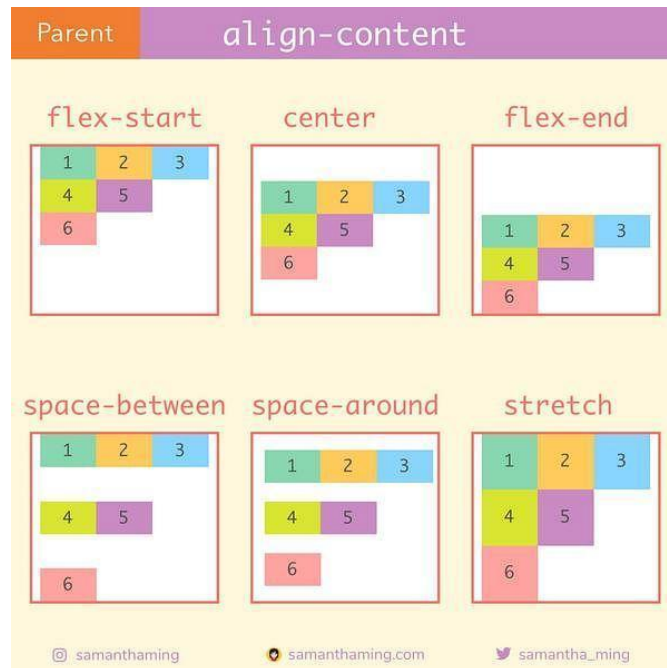
55. What is Flex Or Flexbox?

Display: Flex is a CSS property that creates a flexible layout system, allowing you to align and arrange elements within a container easily. It's part of the Flexbox layout model and simplifies complex layouts by providing control over item distribution and alignment.

Display: Flex (Parent Property):







Display: Flex (Child Property):

