**Current Version : CSS3**

1. **What is the difference between CSS and CSS3 ?**

|  |  |
| --- | --- |
| **CSS** | **CSS3** |
| Css does not support media query. | Css3 supports media query. |
| Css can not split into veried modules. | Css3 can be easily split into veried modules. |
| Css does not support by all new browsers. | Css3 supported by all new browsers. |
| Css is not mobile friendly. | Css3 is mobile friendly. |

1. **How to display name by css file using css property ?**

By using after and before css property we can show name in html file**.**

<!DOCTYPE html>

<html>

<head>

<style>

p::after {

content: "Rajkumar";

color:red;

}

</style>

</head>

<body>

<p>My name is Donald</p>

<p>I live in Ducksburg</p>

</body>

</html>

1. **What is the CSS position ?**

There are five different position values:

* relative
* absolute
* fixed
* sticky
* static

**relative: div(any tag) jaha par hota hai wahi se(apni actual position se) left,right,top aur bottom count hota hai.**

**absolute: sirf position absolute dene se uski z-index badh jati hai, aur left,right,top, bottom dene se screen se ( browser ke top se) counting hoti hai.**

**relative and absolute: parent div ko position relative aur child ko position absolute dekar, agar child div ko top,bottom,left,right dene se ye parent div ke andar hi move hota hai i.e. parent div se hi count hota hai.**

**fixed: position fixed karne se uski position fix ho jati hai, scroll-up or scroll-down karne par bhi uski position change nahi hoti hai.**

**sticky: same as position fixed. position:sticky;top:50px; scroll-down karne par jab top se 50px ki margin rah jayegi tab ye fix jayegi iski position change nahi hogi.**

**Ye sirf chrome, firefox ,safari, is me hi support karta hai.**

**static: isse position change nahi hoti hai. Ye default position hoti hai.**

1. **What are the responsive break points ?**
2. For Mobile:

@media only screen and (min-device-width: 360px) and (max-device-width: 767px){ }

For Landscape Mode:

@media screen and (orientation:landscape) and (min-device-width: 360px) and (max-device-width: 767px){ }

1. For ipad:

@media only screen and (min-device-width: 768px) and (max-device-width: 1024px){ }

For Landscape Mode:

@media screen and (orientation:landscape) and (min-device-width: 768px) and (max-device-width: 1024px){ }

1. For ipad pro:

@media only screen and (min-device-width: 1025px) and (max-device-width: 1365px){ }

For Landscape Mode:

@media screen and (orientation:landscape) and (min-device-width: 1024px) and (max-device-width: 1365px){ }

1. Another screen:

@media only screen and (min-device-width: 1366px) and (max-device-width: 1440px){ }

1. Another screen:

@media (min-device-width: 1796px){ }

1. Another screen:

@media (min-device-width: 2560px){ }

1. Another screen:

@media (min-device-width: 2652px){ }

1. **What is the cross browser testing / compatibility ?**

Jab ham web page create karte hai tab ek hi browser par check karte hue banate hai, but hamari webpage

sabhi browser par proper show hona chahiye . ise alag-alag browser me check karna chahiye . ise hi cross

browser testing/compatibility kahte hai.

//for Mozilla-Firefox

@-moz-document url-prefix()

{

div{color:red;}

}

1. **What is the difference between “resetting” and “normalizing” in css ?**

Jab ham webpage ki css set nahi karthe hai to browser default css provide karta hai aur har browser ka css alag-alag hota hai islilye har browser me alag-alag show hoti hai.

**Resetting**:

Isme ham browser ke sare default css remove(unset) kar dete hai aur apni css apply kar dete hai. ise resetting kahte hai. fir sare browser me ek design show hoti hai.

i.e.Isme ham sari css ko unset kar dete hai to sabhi browser me hamari webpage ek jaisa show hota hai.

**Normalizing :**

isme ham browser ke default css remove(unset) nahi karte hai apna khud ka css property set kar dete hai.

fir sabhi browser me hamari webpage ek jaisa show hota hai.

1. **What is the Limitation / Disadvantage of css ?**

Disadvantages of CSS are given below:

* We can’t select parent tag.
* Some selectors behaviour are different in a different browser.

1. **What is the difference between SASS and SCSS ?**

**SASS vs SCSS**

* SASS is based on indentation and SCSS is not.
* SASS uses .sass extension while SCSS uses .scss extension.
* SASS doesn’t use curly brackets and semicolons while SCSS uses it.

**SASS Syntax:**

$**font**-**color**: #fff

$bg-color: #00f

#box

color: $font-color

background: $bg-color

**SCSS Syntax:**

$**font**-**color**: #fff;

$bg-color: #00f;

#box

{

color: $font-color;

background: $bg-color;

}

1. **What is a CSS Preprocessor ? What are SASS, LESS and STYLUS ?**

CSS Preprocessor is a tool that is used to extend the basic functionality of default vanilla CSS through its own scripting language. It helps to use logical syntax like – variables, functions, mixins, code nesting, inheritance, etc.

**SASS**: Sass is an acronym for “Syntactically Awesome Style Sheets”. SASS can be written in two different syntaxes using SASS or SCSS. SASS doesn’t use curly brackets and semicolons. SASS uses ‘$’ symbol to define variable.

**SASS Syntax:**

$**font**-**color**: #fff

$bg-color: #00f

#box

color: $font-color

background: $bg-color

**LESS:** LESS is an acronym for “Leaner Stylesheets”. LESS is easy to add in any javascript projects by using NPM or less.js file. LESS uses curly brackets and semicolons. LESS uses ‘@’ symbol to define variable.

**LESS Syntax:**

**@font-color**: #fff;

**@bg-color**: #00f;

#box

{

**color**: @font-color;

**background**: @bg-color;

}

**Stylus:** Stylus offers a great deal of flexibility in writing syntax, supports native CSS as well as allows omission of colons, semicolons and curly brackets. It doesn’t use $ or @ symbol to define variables.

/\* STYLUS SYNTAX WRITTEN LIKE NATIVE CSS \*/

**font**-**color**= #fff;

bg-**color** = #00f;

#box

{

**color**: font-color;

**background**: bg-color;

}

/\* OR \*/

/\* STYLUS SYNTAX WITHOUT CURLY BRACES \*/

**font**-**color**= #fff;

bg-**color** = #00f;

#box

**color**: font-color;

**background**: bg-color;

**Note:** SASS and LESS both are browser compatible (auto generate css properties for different browsers).

1. **What is the VH / VW (Viewport height / viewport width) in CSS ?**

Both are CSS unit. It is used to provide height and width in percentage with respect to the viewport. It is mainly used in responsive design techniques.

1. **What is the difference between inline, inline-block and block elements in css ?**

**Inline Elements:** Inline elements don't start a new line, they appear on the same line. We can not use height, width, margin and padding property on inline elements.

Example: <a>, <span> , <strong> tags.

**Inline Block Elements:** Inline-block elements are similar to inline elements, except We can use height, width, margin and padding property on inline block elements.

<!DOCTYPE html**>**

**<html>**

**<head>**

**<style>**

span.b {

   display: inline-block;

   width: 100px;

  height: 100px;

  padding: 5px;

   color:white;

  border: 1px solid blue;

  background-color: blue;

}

**</style>**

**</head>**

**<body>**

**<div>**This is an example of inline-block element with a span colored as blue **<span** class="b"**>**Inline-Block**</span>**  **</div>**

**</body>**

**</html>**

**Block Element:** Block elements always start a new line or new block.

Example: <div>, <p>, <h1> etc.

1. **What are Pseudo elements and Pseudo classes ?**

**Pseudo-elements** :

It is used to create items that do not normally exist in the document tree.

For example :

* ::after
* ::before
* ::first-letter
* ::first-line
* ::selection

In the below example, the color will appear only on the first line of the paragraph.

**p**: :first-line {

color: #ffOOOO;

**font-variant**: small-caps;

}

**Pseudo-classes** :

It select regular elements but under certain conditions .

For example:

* :hover
* :visited
* :active
* :focus
* :link

Example of the pseudo-class, In the below example, the color applies to the anchor tag when it’s hovered.

/\* mouse over link \*/

**a**:hover {

**color**: #FFOOFF;

}

1. **How do you specify units in css ? What are the different ways to do it ?**

There are different ways to specify units in CSS like px, em,rem, pt, percentage (%),vh and vw.

1. **Does margin-top or margin-bottom have an effect on inline elements ?**

No, it doesn’t affect the inline elements. Inline elements flow with the contents of the page.

1. **What property is used for changing the font face ?**

We can use the font-family property for achieving this.

For example:

  p {

font-family: "Times New Roman", Times, serif;

}

1. **What are the differences between responsive design and adaptive design ?**

|  |  |
| --- | --- |
| **Responsive Design** | **Adaptive Design** |
| It adjusts its content and width according to the device. | According to the device, it loads the content of the web page. |
| Designers have to work less . | Designers have to work more . |
| If there is any new layout of the screen comes into the market than the content is automatically adjusted according to the them. | If there is any new layout of the screen comes into the market than Designers have to create completely a new page. |
| Responsive Design works well for larger sites . | Adaptive Design works well for smaller sites . |

1. **How many types of selectors in css ?**

Some selectors:

Id selector , class selector , universal selector ( for all html tags ), element selector, attribute selector

Ex:

#, . , \*{} , div {} , input[type=text]{}

### The meaning of the given list of selectors goes as follows:

### • **div, p:** it will select all div tags and all p tags.

### • **div p :** it will select all p tags that are inside div tags.

### • **div ~ p :** it will select all p tags that are outside div tags.

### • **div > p :** it will select all p tag which has div an immediate/first parent tag.

### • **div + p :** it will select first p tag after the div tag.

1. **How to show a particular div in center vertically and horizontally ?**

**1st method:**

.welcome

{

position: absolute;

left: 50%;

top: 50%;

transform: translate(-50%, -50%);

border: 1px solid;

height:200px;

width:200px;

}

**2nd method:**

.welcome

{

position: absolute;

margin: auto;

top: 0;

right: 0;

bottom: 0;

left: 0;

width: 100px;

height: 100px;

border: 1px solid;

}

1. **How to center align a div inside another div ?**

**Read first 2**

**(1) Centering with Transform**

**HTML:**

**<div class="parent"><div class="child">your content</div></div>**

**CSS:**

**.parent {**

**position: relative;**

**width: 500px;**

**height: 500px;**

**}**

**.child {**

**position: absolute;**

**top: 50%;**

**left: 50%;**

**transform: translate(-50%,-50%);**

**width: 200px;**

**height: 200px;**

**}**

**(2) Centering with Flexbox**

**HTML:**

**<div class="parent"><div class=" child ">your content</div></div>**

**CSS:**

**.parent {**

**display: flex;**

**justify-content: center;**

**align-items: center;**

**}**

**(3) Centering with Grid**

**HTML:**

**<div class=”parent”>**

**<div id=”child”>vertical aligned text<br />some more text here</div>**

**</div>**

**CSS:**

**.parent {**

**display: grid;**

**place-content: center;**

**}**

**(4) Centering with Table(Ignore):**

**HTML:**

**<div class=”outer”><div class=”inner”>your content</div></div>**

**CSS:**

**.outer {**

**display: table-cell;**

**width: 500px;**

**height: 500px;**

**vertical-align: middle;**

**text-align: center;**

**}**

**.inner {**

**display: inline-block;**

**width: 200px;**

**height: 200px;**

**}**

1. **What are the different types of css framework ?**
   1. Animate
   2. Ant Design
   3. Bass css
   4. Blaze UI
   5. Bulma
   6. Bootstrap
   7. Chota
   8. Css
   9. Css Wand
   10. External css
   11. Fomentic UI
   12. Foundation
   13. Lightweight css frameworks
   14. Materialize
   15. Milligram
   16. MVPPicnic css
   17. Pure
   18. Semantic UI
   19. Skeleton
   20. Spectre
   21. Tachyons
   22. Tailwind
   23. Uikit
2. **Read the code and concept of SASS ?**