CSS Interview Questions

1. What is CSS?

CSS (Cascading Style Sheets) is a style sheet language used for describing the look and formatting of a document written in HTML or XML. It helps in styling web pages by applying rules to elements.

2. What are the different types of CSS?

- **Inline CSS**: Styles applied directly within HTML elements using the style attribute.
- **Internal CSS**: Styles defined within the <style> tag inside the <head> section of the HTML document.
- **External CSS**: Styles written in an external file with a .css extension, linked to the HTML using the <link> tag.

3. What is the Box Model in CSS?

The CSS box model describes how elements are rendered in a web page, including:

- **Content**: The content of the box where text and images appear.
- **Padding**: Clears an area around the content (inside the box).
- **Border**: A border surrounding the padding (optional).
- **Margin**: Clears an area outside the border (creates space between elements).

4. Explain the different CSS selectors.

- **Universal Selector** (*): Selects all elements.
- **Element Selector**: Selects elements by tag name, e.g., div.
- **Class Selector** (.classname): Selects all elements with a specific class.
- **ID Selector** (#idname): Selects an element by its ID.
- **Attribute Selector**: Selects elements based on an attribute, e.g., input[type="text"].

5. What is the difference between class and id selectors in CSS?

- **Class selector** can be used on multiple elements.
- **ID selector** must be unique within a page and can be used only once.

6. What are pseudo-classes in CSS?

Pseudo-classes are used to define the special state of an element, e.g.:

- :hover: When a user hovers over an element.
- :active: When an element is being clicked.
- :focus: When an element is focused, like a form input.

7. What are pseudo-elements in CSS?

Pseudo-elements allow you to style parts of an element. Some common pseudo-elements:

- ::before: Inserts content before an element.
- ::after: Inserts content after an element.
- ::first-letter: Styles the first letter of an element.

8. What is the difference between visibility: hidden and display: none?

- visibility: hidden: Hides the element, but it still takes up space in the layout.
- display: none: Completely removes the element from the layout, so it doesn't take up any space.

9. **How do you center a block element horizontally?**

Code-

```
 div {
  width: 50%;
  margin: 0 auto;
}
```

The margin: 0 auto makes the element horizontally centered when its width is less than its container's width.

10. What is Flexbox in CSS?

Flexbox is a layout model that allows items in a container to align and distribute space. It's designed for layout flexibility and can handle both vertical and horizontal alignments.

11. What are the differences between relative, absolute, fixed, and sticky positioning in CSS?

- **Relative**: Positioned relative to its normal position.
- **Absolute**: Positioned relative to the nearest positioned ancestor.
- **Fixed**: Positioned relative to the browser window.
- **Sticky**: Toggles between relative and fixed depending on the scroll position.

12. What is z-index and how does it work?

z-index is used to specify the stack order of elements. Elements with a higher z-index appear on top of those with a lower z-index. It only works for elements with a positioned value (absolute, relative, fixed, or sticky).

13. Explain the float and clear properties in CSS.

- **float**: Moves an element to the left or right, allowing other content to wrap around it.
- **clear**: Specifies that no floating elements are allowed on the left or right (or both) of an element.

14. What are media queries in CSS?

Media queries allow you to apply CSS rules based on device characteristics like screen size, orientation, and resolution, making it a key tool for responsive design.

Code-

```
@media (max-width: 600px) {

  body {

    background-color: lightblue;

  }

}
```

15. What is the difference between em and rem units in CSS?

- **em**: Relative to the font-size of the parent element.
- **rem**: Relative to the root element’s font-size (html).

16. What is the overflow property in CSS?

overflow controls what happens to content that is too big to fit in an element’s box. Possible values:

- **visible**: The content overflows.
- **hidden**: The overflow is hidden.
- **scroll**: Scrollbars are added to see the overflowing content.
- **auto**: Scrollbars are added only if necessary.

17. **What are grid systems in CSS?**

CSS Grid Layout is a two-dimensional system for layout that works with rows and columns. It offers control over both horizontal and vertical layout.

Code-

```
.container {

  display: grid;

  grid-template-columns: 1fr 1fr 1fr;

}
```

18. How do you apply custom fonts in CSS?

You can use the @font-face rule to apply custom fonts:

Code-

```
@font-face {

  font-family: 'MyFont';
  src: url('myfont.woff2') format('woff2');
}
```

19. What is the position: sticky property in CSS?

position: sticky toggles between relative and fixed positioning depending on the scroll position.

20. Explain the transition property in CSS.

transition allows you to change property values smoothly (over a given duration):

Code-

```
button {
  transition: background-color 0.3s ease-in-out;
}
```

21. What is the transform property in CSS?

transform allows you to rotate, scale, translate, or skew elements.

Code-

```
div {
  transform: rotate(45deg);
}
```

22. How do you create animations in CSS?

CSS animations can be defined using @keyframes:

Code-

```
@keyframes example {
  from { background-color: red; }
  to { background-color: yellow; }
}
div {
  animation: example 2s infinite;
}
```

23. What are the different display properties in CSS?

- **block**: Element takes up the full width available.
- **inline**: Element takes up only as much width as it needs.
- **inline-block**: Like inline, but respects width and height properties.

- **none**: Hides the element.

24. Explain the calc() function in CSS.

calc() allows you to perform calculations in CSS:

code-

```
div {

  width: calc(100% - 50px);

}
```

25. What is the opacity property in CSS?

opacity controls the transparency of an element. It ranges from 0 (completely transparent) to 1 (completely opaque).

26. How do you create responsive designs using CSS?

Responsive design is achieved using:

- **Flexible layouts** (using percentages or flexible units like vw, vh).
- **Media queries** to apply styles based on screen width.
- **Fluid grids** (Flexbox or Grid).

27. What is vh and vw in CSS?

- **vh**: Represents 1% of the viewport height.
- **vw**: Represents 1% of the viewport width.

28. Explain min-width and max-width properties in CSS.

- **min-width**: Sets the minimum width an element can be.
- **max-width**: Sets the maximum width an element can be.

29. What is outline in CSS?

The outline property draws a line around the element but does not affect layout, unlike border.

30. Explain the difference between inline, inline-block, and block elements.

- **Inline**: Doesn't start on a new line, and width/height properties are ignored.
- **Block**: Starts on a new line and takes up the full width available.
- **Inline-block**: Behaves like inline but respects width/height.

31. What is the difference between nth-child() and nth-of-type()?

- **:nth-child()** selects elements based on their position among all child elements.
- **:nth-of-type()** selects elements based on their position among siblings of the same type.

32. How do you vertically center an element in CSS?

Using Flexbox:

Code-

```
.container {

  display: flex;

  justify-content: center;

  align-items: center;

}
```

33. What is a CSS Preprocessor?

A CSS preprocessor (e.g., SASS, LESS) allows you to write more maintainable and structured CSS. It introduces features like variables, nesting, and mixins.

34. What is the purpose of CSS variables (Custom Properties)?

CSS variables allow you to store values that can be reused throughout a stylesheet:

Code-

```
:root {

  --main-color: #3498db;

}

div {

  color: var(--main-color);

}
```

35. How do you add shadows to elements in CSS?

Using box-shadow for elements and text-shadow for text:

Code-

```
box-shadow: 10px 10px 5px #888888;

text-shadow: 2px 2px 2px #333;
```

36. What is the difference between visibility: hidden and opacity: 0?

- visibility: hidden: The element is hidden but still occupies space.
- opacity: 0: The element is transparent but still visible in terms of layout and interaction.

37. What is the flex-grow property in Flexbox?

flex-grow defines how much an item will grow relative to other items inside a flex container.

38. How does the order property work in Flexbox?

order allows you to change the order of flex items without altering the HTML structure.

39. What is the use of the pointer-events property in CSS?

The pointer-events property specifies whether an element can be the target of mouse events. Setting pointer-events: none disables all interaction.

40. What is the difference between CSS Grid and Flexbox?

- **Flexbox**: Best for 1-dimensional layouts (either a row or a column).
- **Grid**: Best for 2-dimensional layouts (both rows and columns).

41. How do you create a CSS triangle?

Code-

```
.triangle {

  width: 0;

  height: 0;

  border-left: 50px solid transparent;

  border-right: 50px solid transparent;

  border-bottom: 100px solid red;

}
```

42. What is the clip-path property in CSS?

clip-path is used to clip an element to a specific shape, such as a circle or polygon:

code-

```
div {

  clip-path: circle(50%);

}
```

43. How do you optimize CSS performance?

- Minify CSS files.
- Use shorthand properties.
- Avoid inline styles.
- Reduce unnecessary selectors and nesting.
- Use will-change property sparingly for animations.

44. How do you apply multiple styles in CSS?

Separate multiple class selectors with a space:

Code-

```
.class1.class2 {

  color: red;

}
```

45. What is a CSS reset?

A CSS reset resets browser-specific default styles to a consistent baseline across different browsers. Popular reset stylesheets include Normalize.css.

46. How do you prevent text wrapping in CSS?

Use the white-space: nowrap property to prevent text from wrapping.

47. How do you apply conditional CSS rules?

You can use media queries to apply conditional rules based on device characteristics like width:

Code-

```
@media (max-width: 768px) {

  body {

    background-color: lightgray;

  }

}
```