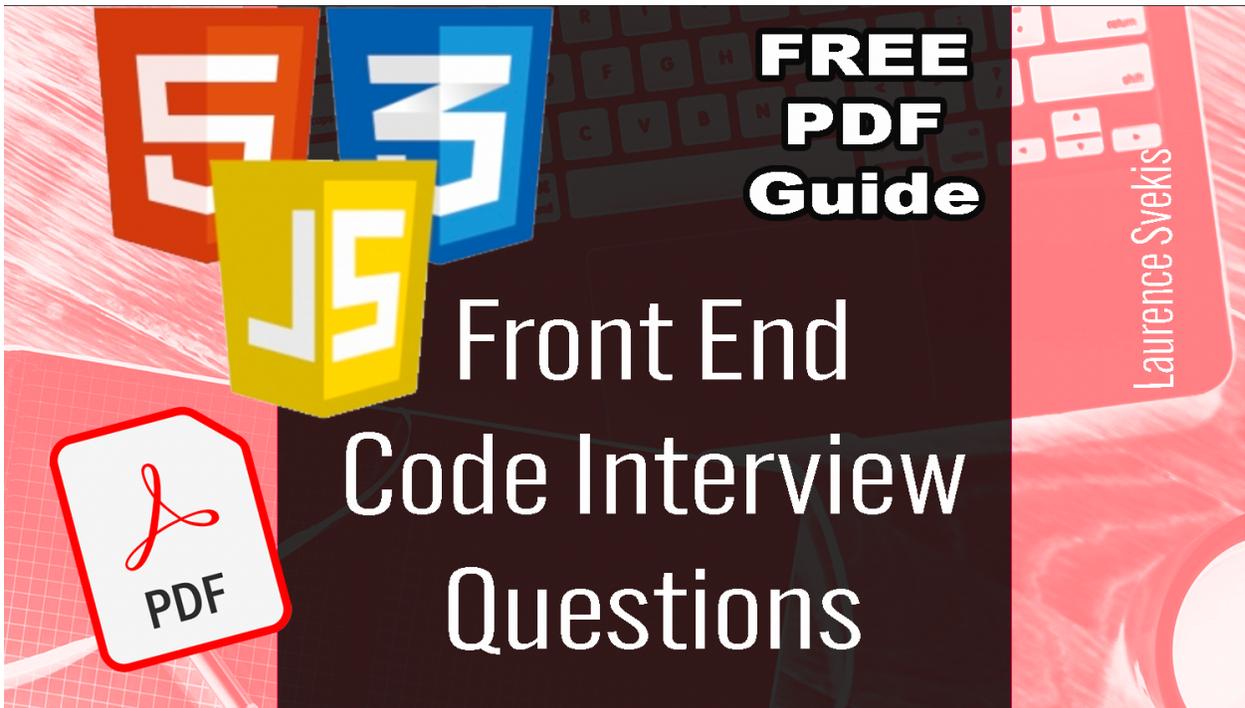


Front-end code interview questions



What is the difference between responsive and adaptive design?	2
Can you explain what is the box model in CSS and how does it work?	3
Can you explain the difference between classes and IDs in CSS?	3
Can you explain what is the CSS float property and how does it work?	5
Can you explain what is the CSS float property and how does it work?	6
Can you explain what is the CSS position property and how does it work?	7
Can you explain what is the CSS z-index property and how does it work?	9
Can you explain what is the CSS overflow property and how does it work?	10
Can you explain what is the CSS clear property and how does it work?	11
Can you explain what is the CSS display property and how does it work?	13
Can you explain the box-sizing property in CSS and how it works?	14
Can you explain the CSS position property and its different values?	16
Can you explain what is a CSS selector and how does it work?	19

Can you explain what is a CSS preprocessor and why would you use it? **19**

Can you explain what is cross-browser compatibility why is it important? **20**

What is the difference between responsive and adaptive design?

Responsive design is a web design approach that makes web pages render well on a variety of devices and window or screen sizes. Adaptive design, on the other hand, uses specific layouts for specific devices and screen sizes. While responsive design changes dynamically based on the viewport size, adaptive design has a set of fixed layouts for specific viewport sizes.

Example of responsive design using CSS media queries:

```
@media screen and (max-width: 640px) {  
  body {  
    font-size: 1.5em;  
  }  
}
```

Can you explain what is the box model in CSS and how does it work?

The box model in CSS refers to the way the browser calculates the size of an HTML element and its content. It consists of four parts: the content, padding, border, and margin. The content is the actual content of the element, padding is the area surrounding the content and affecting the element's size, the border is a decorative line around the content and padding, and the margin is the transparent area outside the border.

Example of CSS box model:

```
.element {  
  width: 300px;  
  padding: 20px;  
  border: 10px solid black;  
  margin: 20px;  
}
```

Can you explain the difference between classes and IDs in CSS?

Classes and IDs are both used in CSS to select elements and apply styles to them. The main difference between them is that

an ID must be unique within a web page, while a class can be used multiple times. IDs are generally used to apply styles to a specific element, while classes are used to apply styles to multiple elements.

Example of using classes in CSS:

```
<style>
  .bold-text {
    font-weight: bold;
  }
</style>
```

```
<p class="bold-text">This text is bold.</p>
<p class="bold-text">This text is also bold.</p>
```

Example of using IDs in CSS:

```
<style>
  #special-text {
    color: blue;
  }
</style>
```

```
<p id="special-text">This text is blue.</p>
```

Can you explain what is the CSS float property and how does it work?

The CSS float property is used to specify whether an element should float to the left or right of its containing element. When an element is floated, the other elements around it will wrap around it.

Example of using the CSS float property:

```
<style>
  .left-float {
    float: left;
    width: 50%;
  }
  .right-float {
    float: right;
    width: 50%;
  }
</style>

<div class="left-float">
  <p>I am floated to the left, so the text will wrap
around me on the right.</p>
</div>

<div class="right-float">
```

```
<p>I am floated to the right, so the text will wrap  
around me on the left.</p>  
</div>
```

Can you explain what is the CSS float property and how does it work?

The CSS float property is used to specify whether an element should float to the left or right side of its containing element. When an element is floated, other elements will wrap around it. Floats are commonly used for layout purposes, such as creating multi-column layouts. The float property can be set to left, right, or none.

Example of using the CSS float property:

```
<style>  
  .left-float {  
    float: left;  
    width: 50%;  
  }  
</style>  
  
<div class="left-float">
```

```
<p>I am floated to the left and take up 50% of the
width of my containing element.</p>
</div>
<div>
  <p>I wrap around the floated element.</p>
</div>
```

Can you explain what is the CSS position property and how does it work?

The CSS position property is used to specify the position of an element within the document flow. It can be set to static (default), relative, absolute, or fixed. When set to static, the element follows the normal document flow. When set to relative, the element is positioned relative to its normal position in the document flow. Absolute positioning takes an element out of the document flow and positions it relative to the nearest positioned ancestor. Fixed positioning positions an element relative to the viewport and it will not move when the page is scrolled.

Example of using the CSS position property:

```
<style>
  .relative {
```

```
    position: relative;
    top: 20px;
    left: 20px;
}
.absolute {
    position: absolute;
    bottom: 0;
    right: 0;
}
</style>
```

```
<div class="relative">
```

```
    <p>I am positioned relative to my normal position and
moved 20px down and 20px to the right.</p>
```

```
</div>
```

```
<div class="absolute">
```

```
    <p>I am positioned absolute to my nearest positioned
ancestor (or the body) and am placed at the bottom
right of my containing element.</p>
```

```
</div>
```

Can you explain what is the CSS z-index property and how does it work?

The CSS z-index property is used to specify the stack order of elements that overlap. An element with a higher z-index will be placed in front of an element with a lower z-index. The z-index property only works on elements that have a position value other than static.

Example of using the CSS z-index property:

```
<style>
  .high-z-index {
    position: absolute;
    z-index: 2;
  }
  .low-z-index {
    position: absolute;
    z-index: 1;
  }
</style>

<div class="high-z-index">
  <p>I have a higher z-index, so I appear on top of the
other element.</p>
</div>
```

```
<div class="low-z-index">
  <p>I have a lower z-index, so I appear behind the
other element.</p>
</div>
```

Can you explain what is the CSS overflow property and how does it work?

The CSS overflow property is used to specify what to do when an element's content overflows its bounds. The value of the overflow property can be set to visible (default), hidden, scroll, or auto. When set to visible, the overflowing content will be visible. When set to hidden, the overflowing content will be clipped. When set to scroll, scrollbars will be added to allow the user to scroll to see the overflowed content. When set to auto, scrollbars will be added only when necessary.

Example of using the CSS overflow property:

```
<style>
  .overflow-hidden {
    overflow: hidden;
    height: 100px;
    width: 200px;
  }
```

```
.overflow-scroll {  
  overflow: scroll;  
  height: 100px;  
  width: 200px;  
}  
</style>
```

```
<div class="overflow-hidden">
```

```
  <p>I have overflow set to hidden, so my overflowing  
content will be clipped.</p>
```

```
</div>
```

```
<div class="overflow-scroll">
```

```
  <p>I have overflow set to scroll, so scrollbars will  
be added to allow the user to see my overflowing  
content.</p>
```

```
</div>
```

Can you explain what is the CSS clear property and how does it work?

The CSS clear property is used to specify whether an element should be next to a floating element or not. The clear property takes the value of left, right, both, or none (default). When set to

left, the element will not be next to a floated element on the left. When set to right, the element will not be next to a floated element on the right. When set to both, the element will not be next to floated elements on either side.

Example of using the CSS clear property:

```
<style>
  .left-float {
    float: left;
    width: 50%;
  }
  .clear-both {
    clear: both;
  }
</style>

<div class="left-float">
  <p>I am floated to the left, so the text will wrap
around me on the right.</p>
</div>

<div class="clear-both">
  <p>I have clear set to both, so I will not be next to
any floated elements.</p>
</div>
```

Can you explain what is the CSS display property and how does it work?

The CSS display property is used to specify how an element should be displayed. The value of the display property can be set to block, inline, inline-block, flex, grid, none, or inherit. Block elements create a new block formatting context and take up the full width of their parent container. Inline elements only take up as much width as necessary and do not create a new block formatting context. Inline-block elements are similar to inline elements, but they can have a specified width and height. Flex and grid elements are used for creating flexible and grid-based layouts. None will hide the element and it will not take up any space.

Example of using the CSS display property:

```
<style>
  .display-block {
    display: block;
    width: 50%;
    height: 50px;
    background-color: lightblue;
  }
  .display-inline {
```

```
    display: inline;
    background-color: lightgreen;
}
.display-none {
    display: none;
}
</style>
```

```
<div class="display-block">
  <p>I am a block element and I take up the full width
of my parent container.</p>
</div>
<span class="display-inline">I am an inline element and
only take up as much width as necessary.</span>
<p class="display-none">I have display set to none, so
I will not be visible.</p>
```

Can you explain the box-sizing property in CSS and how it works?

The box-sizing property in CSS is used to specify how the size of an element is calculated. By default, the size of an element is calculated based on the width, height, padding, and border of the element. However, with the box-sizing property set to

border-box, the size of an element is calculated based on the width and height of the element, including any padding and border, and not just the content area of the element.

Example of using the CSS box-sizing property:

```
<style>
  .content-box {
    width: 300px;
    height: 100px;
    padding: 20px;
    border: 10px solid black;
  }
  .border-box {
    width: 300px;
    height: 100px;
    padding: 20px;
    border: 10px solid black;
    box-sizing: border-box;
  }
</style>
```

```
<div class="content-box">
```

```
  <p>I have box-sizing set to content-box (default), so
  my size is calculated based on my width, height,
  padding, and border.</p>
```

Laurence Svekis <https://basescripts.com/>

```
</div>
<div class="border-box">
  <p>I have box-sizing set to border-box, so my size is
calculated based on my width and height only, including
my padding and border.</p>
</div>
```

Can you explain the CSS position property and its different values?

The CSS position property is used to specify the position of an element. The different values of the position property are static (default), relative, absolute, and fixed. Static elements are positioned according to the normal flow of the document. Relative elements are positioned relative to their normal position in the document flow. Absolute elements are positioned relative to the nearest positioned ancestor, if there is one, or to the initial containing block if there is no positioned ancestor. Fixed elements are positioned relative to the viewport and do not move when the page is scrolled.

Example of using the CSS position property:

```
<style>
  .static {
```

```
width: 100px;
height: 100px;
background-color: lightblue;
}

.relative {
  position: relative;
  width: 100px;
  height: 100px;
  background-color: lightgreen;
  top: 20px;
  left: 20px;
}

.absolute {
  position: absolute;
  width: 100px;
  height: 100px;
  background-color: lightcoral;
  top: 20px;
  right: 20px;
}

.fixed {
  position: fixed;
  width: 100px;
  height: 100px;
```

```
background-color: lightyellow;
bottom: 20px;
right: 20px;
}
</style>
```

```
<div class="static"><p>I am a static element positioned
according to the normal flow of the document.</p>
```

```
</div>
```

```
<div class="relative">
```

```
<p>I am a relative element positioned relative to my
normal position in the document flow, with top and left
values set to 20px.</p>
```

```
</div>
```

```
<div class="absolute">
```

```
<p>I am an absolute element positioned relative to
the nearest positioned ancestor, with top and right
values set to 20px.</p>
```

```
</div>
```

```
<div class="fixed">
```

```
<p>I am a fixed element positioned relative to the
viewport, with bottom and right values set to 20px.</p>
```

```
</div>
```

Can you explain what is a CSS selector and how does it work?

A CSS selector is a pattern used to select the HTML elements to which a set of styles will be applied. Selectors can be based on element type, class, ID, attribute, and more. Selectors can be combined to target specific elements and apply styles to them. The styles are then applied to the selected elements based on the specificity of the selector.

Can you explain what is a CSS preprocessor and why would you use it?

A CSS preprocessor is a scripting language that extends the capabilities of CSS and compiles it into regular CSS. The most commonly used CSS preprocessors are Sass, Less, and Stylus. CSS preprocessors allow developers to use variables, functions, and mixins to make writing and maintaining CSS easier and more efficient.

Can you explain what is cross-browser compatibility and why is it important?

Cross-browser compatibility refers to the ability of a website or web application to function and display correctly across different browsers. It is important because different browsers have different rendering engines and may display the same HTML, CSS, and JavaScript code differently. Ensuring cross-browser compatibility helps ensure