Solution of CSS Final Exam

Answer To The Question No:1

In CSS flexbox there are types of axis available and these two types are main axis and cross axis. This two axis are used to describe the two primary axes in a flexbox container. The main axis is defined by the flex-direction property and it is by default work in the row wise direction. The main axis usually moves along the horizontal axis. To align the items present in the main axis, the justify-content property is being used. For main axis flex direction can be four types such as row, row-reverse, column, column-reverse. When the main axis flex direction is set to column, then the main axis runs along the vertical axis. On the other hand the cross axis works perpendicular of main axis. It usually runs alongs the vertical axis. It also requires flex-direction property to set the axis formation. To align the elements along the cross axis align-items property is being used. Example of main axis and cross axis in CSS is given below:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width,
initial-scale=1.0">
  <style>
    .flex-container {
      display: flex;
      flex-direction: row;
      justify-content: space-between;
      align-items: center;
      height: 200px;     }

    .flex-item {
      width: 100px;
      height: 100px;
      background-color: lightblue;
      border: 1px solid darkblue;
    }
  </style>
</head>
<body>
  <div class="flex-container">
    <div class="flex-item">1</div>
    <div class="flex-item">2</div>
    <div class="flex-item">3</div>
  </div>
</body>
</html>
```

Here in this example, the flex-container has three child elements with the class flex-item. The flex-direction property is set to row, so the main axis is horizontal. The justify-content:space between property is used to distribute the items along the main axis with space between them. Also, the align-items: center property is used to center items along the cross axis. So, in this code both main axis and cross axis properties of CSS is being implemented.

Answer To The Question No:2

Yes, multiple CSS background images can be used in a div. Multiple background properties need to specified for that div. The background images are separated by a comma. Here is an example using multiple CSS background images:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width,
initial-scale=1.0">
  <title>Multiple Background Images</title>
  <style>
    .multibackground {
      width: 400px;
      height: 200px;
      background-image: url('image1.jpg'), url('image2.jpg'),
linear-gradient(to right, #ffcc00, #ff9900);
      background-size: 50% 100%, 50% 100%, 100% 100%;
      background-position: 0 0, 50% 0, 0 0;
      background-repeat: no-repeat, no-repeat, repeat;
    }
  </style>
</head>
<body>
  <div class="multibackground"></div>
```

```
</body>
</html>
```

Here, background-image property is used to multiple images and the images are separated by commas. The background-size property is used to define the size of background image. In the above code, the first two images take up to 50% of the width each and the linear gradient takes up to 100%. The background-position property is being used to set the initial position of each of the background images. In the above code, the first image starts from the top-left corner (0,0), the second image starts from the middle of the container horizontally (50% 0) and the linear gradient starts from the top-left corner (0 0). The background-repeat property is being used to specify whether the images should repeat or not. In the given example, the images are set as no-repeat. So, this is how multiple CSS background images can be used in a div.

Answer To The Question No:3

-CSS EXAM HEADER NAV PART (65aaafc12c62f40099930024--amazing-kheer-4c70e6.netlify.app)

Answer To The Question No:4

CSS specificity is a set of rules to determine which styles are implemented on an element when multiple conflicting styles exist. It also helps the browser to decide which style declarations should take precedence. Specificity is important for resolving conflicts when different selectors target the same element with conflicting styles. There are four levels of specificitie and they are Inline Styles, ID Selectors, Class Selectors, Attribute Selectors, and Pseudo-classes and Element Selectors and Pseudo-elements.

- Inline Styles: These styles are applied directly to an HTML element using the style attribute. Inline styles have the highest specificity.
- ID Selectors: Selectors target an element by using ID. These IDs have a higher specificity than class selectors or element selectors.
- Class Selectors, Attribute Selectors, and Pseudo-classes: These selectors have a lower specificity than IDs but are higher than element selectors.
- Element Selectors and Pseudo-elements: These are the least specific selectors and have the lowest priority.

While comparing between two or more selectors, the brower tends to consider the specificity value for each selector. If there is a conflict, the style with higher specificity will be applied. If the specificity values are same then the rule that appears last in the stylesheet takes precedence. So, these are the CSS specificity which is important to understand to develop a proper website.

Answer To The Question No:5

The mixins and partials are concepts usually associated with CSS preprocesseors, such as SaSS, which extends the capabilities of traditional CSS.

Here, mixin is a reusable piece of code that defines a set of styles and can be included in other selectors or rules. Mixins are mainly used to encapsulate and reuse common styles across multiple selectors. They increase the efficiency of developers by resuing the code and maintaining a consistent design. On the other hand, partials are separate Sass or less files that contain snippets of CSS code. They are mainly used to organize and modularize styles. The main purpose of them is to break down large stylesheets into smaller, manageable pieces. Each partial can focus on a specific aspects of the styling such as variables, mixins, specific components and many more to make the codebase more maintainable.

In conclusion, it can be said that both mixins and partials contribute to a more modular and maintainable CSS codebase which are widely used in projects that leverage CSS preprocessors like Sass and many more.

Answer To The Question No:6

-CSS TIMELINE
(65ae7cb1ba900409a5b4f899--aquamarine-gumption-bd42dd.netlify.app)

Answer To The Question No:7

-Welcome to CSS Glass Form
(65ae671662cf67007d893ab9--amazing-churros-25aa22.netlify.app)

Answer To The Question No:8

In CSS pseudo-selectors are used to select and style the elements based on the state or position within a document, without any additional classes or IDs. Pseudo-selectors also allow styles to be applied on specific elements in certain situations. Now, five of the most important pseduo-selectors are given below:

- :hover: The hover pseudo-selector is used to select and style an element whenever the user hovers over it with mouse pointer. It is mainly used to create interactive and visually pleasing effects such as changing the color or appearance of a button when the user hovers over it.

- :active: The active pseudo-selector is applied to an element when it is being activated or clicked. It is often used to provide visual feedback to the user when they are actively interacting with an element.
- :focus: The focus pseudo-selector is generally used to select and style an element whenever it has received focus, typically through keyboard navigation or user interaction. It is commonly used for improving accessibility and providing a clear indication of the focused element.
- :nth-child(): The nth-child pseudo-selector is usually used to select and style elements based on their position within a parent container. A certain formula can be specified to target a specific children which is useful for styling odd or even elements.
- :not(): The not() pseudo-selector is used to select elements that do not match a given selector. It is very much useful when styling is implemented based on certain criteria but exclude others.

So, these are the five most important pseudo-selector that is almost implemented in all the projects.

Answer To The Question No:9

In CSS media queries allow to apply styles based on certain conditions such as the characteristics of the device such as its width, height or its orientation. Media queries are mainly used to implement responsive web design. In media queries there is a concept called mobile first approach which refers to the design and development of mobile version of a website first and then gradually enhance it for larger screens. This approach is based on the assumption that is easier to scale up from a mobile design to larger screens then it is to scale down from a desktop to smaller screens.

Now, to decide between Min-width or max-width for usage in Mobile First Approach we need to first overlook the terms which are given below:

In mobile first approach the first approach generally starts with min-width as for mobile first approach we assume that the default styles are for small screens. By using min-width it allows us to progressively enhance the styles as the screen size increases. So, first start with the basic mobile styles and then additional styles for larger screens can be added. On the other hand, if the max-width is used for mobile first approach then a lot of styles will be overridden for larger screens which will lead to a less efficient and more complex stylesheet.

So, for Mobile First Approach design min-width will be more helpful interms of efficiency and complexity.

Answer To The Question No:10

Question: What is Bootstrap and how does it streamline front-end development?

Bootstrap is a popular and widely used open-source front-end framework developed by twitter. As bootstrap is developed by twitter that's why it is also called as twitter Blueprint. It is a collection of pre-designed components based on HTML, CSS and JavaScript. It aims to simplify and accelerate the process of building responsive and visually pleasing web pages. Some of the key features of bootstrap is given below:

- Grid System: Bootstrap is mainly known for its responsive grid system. In it bootstrap includes a responsive, mobile first grid system that makes it easy to create flexible and fluid layouts. The grid system consists of 12-column layout, allowing developers to design responsive pages for various screen sizes.
- Pre-styled Components: Bootstrap offers a wide variety of pre-styled components such as navigation bars, buttons, forms, alerts and many more. These components are easily customizable and after customization these components can be integrated into a project.
- CSS and JavaScript Utilites: Bootstrap provides a set of utilities classes for quick styling and layout adjustments.
- Responsive Typography: Bootstrap ensures consistent and responsive typography across several different devices with its typography styles. It also includes classes for headings, paragraphs, and other text elements that automatically adjust based on the screen size.
- Browser Compatibility: Bootstrap is comapatiable with modern browsers and it provides a consistent experience across various platforms.

So, bootstrap eases the use and extensive documentation make it a go to choice for developers. Developers use it to create responsive and visually pleasing websites without starting from scratch. It is also modular and customizable which allows developers to use only the components of their needs. It also promotes efficiency in front-end development as well.

Solution of Final Project

-Welcome to BOOKMARK (65ae57a131747e28933598f1--thriving-kringle-00620b.netlify.app)