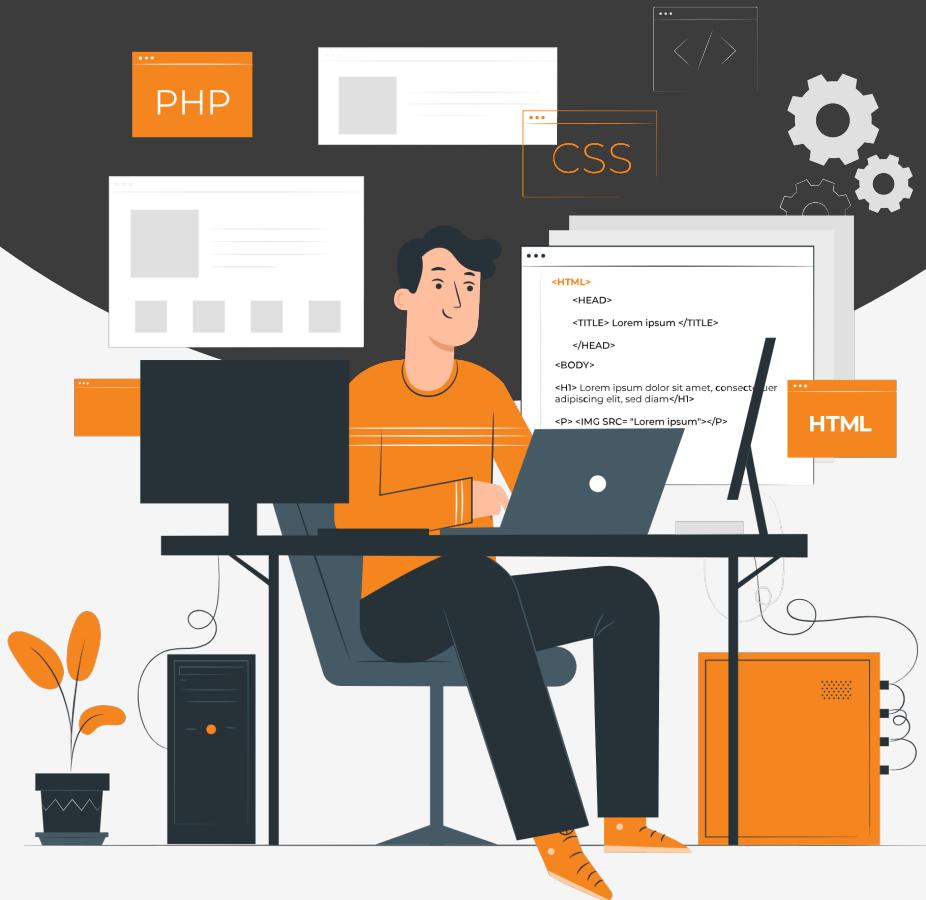


Advanced CSS

Assignment Solution



Task 1:

Problem Statement

Explain the purpose of using the var() function in css. Also you have created two buttons with id named primaryBtn and secondaryBtn which should be given background colors using the var() function. The color code for primaryColor is #00b7ff and secondaryColor is #6c757d.

Expected Code Output:

Primary Button Secondary Button

Activate Windows
Go to Settings to activate Windows.

Solution:

The var() function in CSS is used to reference and use the value of a custom property (also known as a CSS variable) within your stylesheets. CSS variables are defined using the -- prefix followed by a variable name. They allow us to store and reuse values throughout our stylesheet, making it easier to maintain consistency and make global changes.

Purpose of var() Function:

- 1. Reusability:** CSS variables allow us to store values that we can reuse across our stylesheet. The var() function lets us reference these values wherever needed, reducing redundancy and making our code cleaner.
- 2. Consistency:** By using variables, we can ensure consistent values for colors, spacing, fonts, and more throughout our entire design. Changing a variable's value will automatically update all instances where the variable is used.
- 3. Global Changes:** If we need to make a design change, we can update the variable's value once, and the change will be applied to all instances using that variable.

Code Solution

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>CSS Variables</title>
  </head>
```

```
<style>
:root {
    --primaryColor: #00b7ff;
    --secondaryColor: #6c757d;
}
button {
    font-size: 25px;
    border: none;
    padding: 5px 20px;
    color: white;
}
#primaryBtn {
    background-color: var(--primaryColor);
}
#secondaryBtn {
    background-color: var(--secondaryColor);
}
</style>

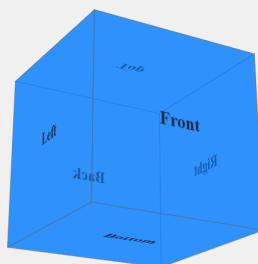
<body>
<button id="primaryBtn">Primary Button</button>
<button id="secondaryBtn">Secondary Button</button>
</body>
</html>
```

Task 2:

Problem Statement

Create a 3D cube using the transform property of CSS.

Expected Output



Activate Windows
Go to Settings to activate Windows.

Solution:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Cube</title>
  </head>

  <style>
    body {
      display: flex;
      justify-content: center;
      align-items: center;
      min-height: 100vh;
      margin: 0;
      background-color: #f0f0f0;
    }

    .scene {
      perspective: 1000px;
    }

    .cube {
      width: 200px;
      height: 200px;
      position: relative;
      transform-style: preserve-3d;
      transform: rotateX(20deg) rotateY(30deg);
    }

    .face {
      position: absolute;
      width: 200px;
      height: 200px;
      background-color: rgba(0, 123, 255, 0.7);
      border: 1px solid #666;
      display: flex;
      justify-content: center;
      align-items: center;
      font-size: 20px;
      font-weight: bold;
      opacity: 0.8;
    }

    .front {
      transform: translateZ(100px);
    }
  </style>
```

```
.back {  
    transform: rotateY(180deg) translateZ(100px);  
}  
.right {  
    transform: rotateY(90deg) translateZ(100px);  
}  
.left {  
    transform: rotateY(-90deg) translateZ(100px);  
}  
.top {  
    transform: rotateX(90deg) translateZ(100px);  
}  
.bottom {  
    transform: rotateX(-90deg) translateZ(100px);  
}  
</style>
```

```
<body>  
    <div class="scene">  
        <div class="cube">  
            <div class="face front">Front</div>  
            <div class="face back">Back</div>  
            <div class="face right">Right</div>  
            <div class="face left">Left</div>  
            <div class="face top">Top</div>  
            <div class="face bottom">Bottom</div>  
        </div>  
    </div>  
</body>  
</html>
```

Task 3:

Problem Statement

Create a simple circular loader which will rotate continuously to look like a loading screen on a website.

Expected Output:



Activate Windows
Go to Settings to activate Windows.

Assignment Solution

As in the image we can see that the blue color is rotating continuously.

Solution:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Circular Loader</title>
  </head>

  <style>
    body {
      display: flex;
      justify-content: center;
      align-items: center;
      min-height: 100vh;
      margin: 0;
      background-color: #f0f0f0;
    }

    .loader {
      border: 4px solid rgba(0, 0, 0, 0.1);
      border-left-color: #007bff;
      border-radius: 50%;
      width: 40px;
      height: 40px;
      animation: spin 1s linear infinite;
    }

    @keyframes spin {
      0% {
        transform: rotate(0deg);
      }
      100% {
        transform: rotate(360deg);
      }
    }
  </style>

  <body>
    <div class="loader"></div>
  </body>
</html>
```

Assignment Solution



Task 4:

Problem Statement

You have to visit the PW Skills website (<https://pwskills.com/>) and have to hide the logo by using the developer tool. This should be done using the css and developer tool only, use of javascript is prohibited.

Expected Outputs

Default Output

The screenshot shows the PW Skills homepage. At the top is a navigation bar with the PW Skills logo, a search bar, and a 'Login / Register' button. Below the navigation bar is a main content area featuring a banner with the text 'Upscaling Made <EASY|> With PW Skills'. To the right of the banner are several circular icons representing different skills or tools. A large image of a woman holding a laptop is also present. At the bottom of the main content area is a red 'Explore Courses' button.

Final Output

The screenshot shows the PW Skills homepage after the logo has been hidden using developer tools. The navigation bar is visible at the top, but the PW Skills logo is no longer present. The main content area, including the banner and skill icons, remains unchanged from the default output.

Solution:

The screenshot shows the PW Skills homepage with developer tools (Chrome DevTools) open. The 'Elements' tab is selected in the toolbar. In the bottom right corner of the DevTools interface, there is a small blue circular icon with a white number '1', indicating a pending change. The 'Styles' panel on the right side of the DevTools shows a CSS rule being applied to an element: 'element.style { display: none; }'. This rule is highlighted with a red border, indicating it is the current selection.

Assignment Solution



Task 5:

Problem Statement

You have to visit the PW Skills website (<https://pwskills.com/>) and have to change the content of the 'Login / Register' button to 'Connect with us' using the developer tool. This should be done using the elements of developer tools only, use of javascript is prohibited.

Expected Outputs

Default Output

The screenshot shows the PW Skills homepage. At the top right, there is a red 'Login / Register' button. Below it, there is a search bar and a navigation menu with links like 'Courses', 'PW Skills Lab', 'Job Portal', 'Experience Portal', 'Become an affiliate', and 'Hall Of fame'. The main content area features a banner with the text 'Upscaling Made <AFFORDAB> With PW Skills'. It includes several images related to technology and learning, such as a person working at a computer, a woman holding a laptop, and various software icons. A blue 'Explore Courses' button is located at the bottom left of the main content area.

Final Output

The screenshot shows the PW Skills homepage after modification. The 'Login / Register' button has been changed to a blue 'Connect with us' button. The rest of the page content, including the banner, images, and footer, remains the same as the default output.

Solution:

The screenshot shows the browser's developer tools with the 'Elements' tab selected. A blue box highlights the 'Connect with us' button in the DOM tree. A tooltip shows the CSS rule for the button: `button.opacity-100.bg-orange-500.text-white.hover\bg-orange-600.text-sm.px-3.py-[10px].medium\text-base.medium\text-base.cursor-pointer.lpx-[10px]>Connect with us</button>`. The right side of the developer tools shows the 'Styles' panel with the CSS rule `button.opacity-100.bg-orange-500.text-white.hover\bg-orange-600.text-sm.px-3.py-[10px].medium\text-base.medium\text-base.cursor-pointer.lpx-[10px]` highlighted. The rule defines the button's appearance with orange background, white text, and rounded corners.