CSS

1. **Write a CSS selector to target all paragraphs (p elements) within a <div> element with the class "content".**

.content p {

/\* Your styles here \*/

}

1. **Style the text inside a <h1> element to have a font size of 24px, a color of red, and a text-align center.**

h1 {

font-size: 24px;

color: red;

text-align: center;

}

1. **Create a CSS rule to set the padding of a <div> element to 20 pixels on all sides.**

h1 {

font-size: 24px;

color: red;

text-align: center;

}

1. **Set a background image for the <body> element with the image "background.jpg" and ensure it repeats horizontally.**

h1 {

font-size: 24px;

color: red;

text-align: center;

}

1. **Position an element with the class "sidebar" to be fixed to the right side of the browser window, 20 pixels from the top.**

div {

padding: 20px;

}

1. **Float two <div> elements with the classes "left-box" and "right-box" so they appear side by side. Clear any floats after them.**

.left-box {

float: left;

}

.right-box {

float: right;

}

/\* To clear the floats \*/

.clearfix::after {

content: "";

display: block;

clear: both;

}

1. **Apply a different text color to a link when the user hovers over it using the :hover pseudo-class.**

a:hover {

color: blue; /\* Replace 'blue' with your desired color \*/

}

1. **Create a media query that changes the font size to 18px when the screen width is 600 pixels or less.**

@media screen and (max-width: 600px) {

/\* Replace 600px with your desired screen width \*/

body {

font-size: 18px;

}

}

1. **Use flexbox to center an element both horizontally and vertically within its parent container.**

.parent-container {

display: flex;

justify-content: center;

align-items: center;

}

1. **Create a CSS transition that smoothly animates the background color change of a button when the user hovers over it.**

button {

background-color: blue; /\* Replace 'blue' with the original background color \*/

transition: background-color 0.5s ease;

}

button:hover {

background-color: red; /\* Replace 'red' with the desired background color on hover \*/

}

JAVA SCRIPT

1. **Declare a variable called age and assign your age to it. Print the value of the variable to the console.**

let age = 25;

console.log(age);

1. **Write a function called calculateArea that takes two parameters (length and width) and returns the area of a rectangle.**

function calculateArea(length, width) {

return length \* width;

}

1. **Write a program that takes a number as input and checks if it's positive, negative, or zero. Print the result to the console.**

let number = 10;

if (number > 0) {

console.log("Positive");

} else if (number < 0) {

console.log("Negative");

} else {

console.log("Zero");

}

1. **Write a program to print the numbers from 1 to 10 using a for loop.**

for (let i = 1; i <= 10; i++) {

console.log(i);

}

1. **Create an array called fruits containing three different fruit names. Print the third element of the array to the console.**

let fruits = ["apple", "banana", "orange"];

console.log(fruits[2]);

1. **Create an object called person with properties name and age. Assign values to these properties and print them to the console.**

let person = {

name: "John",

age: 30,

};

console.log(person.name);

console.log(person.age);

1. **Given a string "Hello, World!", write code to extract the word "World" from the string and print it to the console.**

let str = "Hello, World!";

let word = str.substring(7, 12);

console.log(word);

1. **Create an HTML button with the id "myButton." Using JavaScript, add a click event listener to the button that displays an alert saying "Button clicked!" when clicked.**

<button id="myButton">Click Me</button>

1. **Create an HTML ul element and three li elements. Using JavaScript, append the li elements to the ul element and then append the ul element to the document body.**

<ul id="myList">

<li>Item 1</li>

<li>Item 2</li>

<li>Item 3</li>

</ul>

let ulElement = document.getElementById("myList");

document.body.appendChild(ulElement);

1. **Write a function called divideNumbers that takes two parameters (numerator and denominator) and returns the result of the division. Handle the scenario when the denominator is zero and throw an appropriate error.**

function divideNumbers(numerator, denominator) {

if (denominator === 0) {

throw new Error("Cannot divide by zero!");

}

return numerator / denominator;

}