Summary and Analysis Report

Summary: The following is a list of thirty-five JavaScript and CSS questions and answers. The questions encompass a range of topics, including JavaScript variables, hoisting, scoping, inheritance, and operators as well as CSS positioning, variables, and specificity. The answers are concise and provide fundamental explanations.

Sentiments: negative review

classifications: Tech

paraphrase: Here is your paraphrased text:

1. Meta tags provide information about a webpage. Description provides more detailed information about the webpage.
2. The http-equiv tag is used to refresh the page. When the refresh tag is provided with a URL and a time, it will refresh and direct it to open to that URL.
3. The character encoding is a numeric representation of characters.
4. Semantic HTML uses tags such as nav, header, and footer to communicate more clearly to the user and browser.

Here are some JavaScript definitions and explanations:

- Specificity refers to ID-100, class-10, tag-1, and *-0.
- In CSS, var is a root property and property is a CSS variable.

In JavaScript:

- Type coercion is the process of converting values to numbers, such as strings, booleans, etc.
- Hoisting is the action of accessing a function or variable before it is initialized.
- The this keyword refers to the global window object in a browser.

In JavaScript, global scope means that any variable or function created in global space attaches to the window object and can be accessed by using window.varName or varName.

"Undefined" refers to a variable that has been declared but not given a value, while "not defined" refers to a variable that does not exist. JavaScript is a loosely typed language, meaning that types can be changed after creating a var.

Scope refers to the area where specific functions or variables can be accessed.

A constructor is a JavaScript function that creates an instance of an object typically called a class. The super keyword is used to access properties on an object's prototype, or to invoke a superclass's constructor.

Prototype inheritance means that objects inherit properties and methods from a prototype.

To horizontally center a block element, such as a div, use margin: auto;. When a border radius of 50% is applied to a div, it creates a circle.

A hoisting in JavaScript allows you to use functions and variables before they're declared.

A closure is a function that is nested in another function, also known as a function with lexical scope that is bundled and referred to as a closure.

Here are some basic CSS definitions and explanations:

- Padding refers to the space between an element's content and its border. The padding-top, padding-right, padding-bottom, padding-left are specific padding properties.

Margin is the space around an element's border, while padding is the space between the element's border and its content.

In JavaScript, the this keyword refers to an object. Which object depends on how this is being invoked. In an object method, this refers to the object. Alone, this refers to the global object. In a function, this refers to the global object. In a function in strict mode, this is undefined. In an event, this refers to the element that received the event. Methods such as call(), apply(), and bind() can refer this to any object.

A higher-order function is a function that either receives a function as an argument or returns a function as output. For example, Array.prototype.map, Array.prototype.filter, and Array.prototype.reduce are higher-order functions built into the language.

JavaScript is a synchronous language, meaning that tasks are executed in order. Synchronous tasks must be completed before moving on to the next task. Asynchronous tasks are executed in any order or at once.

Promises are used to handle asynchronous operations in JavaScript. They are used to avoid unmanageable code that can result from dealing with multiple asynchronous operations using callbacks and lead to callback hell. A promise has four states: fulfilled, rejected, pending, and settled. A promise can be created using the Promise constructor.

Currying is the process of transforming a function that takes multiple arguments into a function that takes a single argument and returns another function if any arguments are still needed.