

Introduction to task 1.3P

brief

task 1

<!--
Responsive web design is an approach to web design that makes web pages render well on a variety of devices and window or screen sizes. It is crucial because with the increasing variety of devices with different screen sizes, it's important that a website can adapt to these varying formats to provide a user-friendly experience.
-->

2 – Understanding how responsive web design can be implemented using CSS. Instruction:

Responsive web design allows a webpage to adapt to the device's viewport size. One way to implement this is by using CSS @media rule. This rule applies different styles for different device screen widths.

For instance, let's consider a simple webpage with a div. Normally, this div takes up 60% of the screen width. However, on devices with screen width less than 600px, it should take up 100% of the screen width. Here's how we can achieve this with CSS @media rule:

demonstration in my code

```
.container {  
  width: 60%;  
}  
  
@media screen and (max-width: 600px) {  
  .container {  
    width: 100%;  
  }  
}
```

In this code, .container will have a width of 60% by default. But when the screen width is less than 600px, the @media rule gets triggered and .container width is set to 100%.

reflection

Reflecting on the tasks, I appreciate the critical role that both responsive design and JavaScript play in modern web development. Responsive design, executed using CSS media queries, ensures that web content remains accessible and user-friendly across a variety of

devices, screen sizes, and orientations. It was an enlightening exercise to implement these concepts in code and see their effects in real-time.

The tasks involving JavaScript highlighted the language's versatility. By working with different data types and built-in methods, I gained practical experience manipulating strings, numbers, arrays, and dates. These operations form the backbone of many complex functionalities in web applications.

Finally, seeing the JavaScript execution results in the browser console reinforced the connection between the code we write and the tangible outcomes on the webpage. Overall, these tasks offered valuable hands-on experience and a deeper understanding of the fundamental building blocks of web development.