To achieve a "print all" functionality for multiple web pages, you would typically need to use a combination of HTML, CSS, and JavaScript. Here's an example code snippet that demonstrates how you could implement this feature using JavaScript:

HTML (print.html):

html

<!DOCTYPE html>

<html>

<head>

<title>Print All Pages</title>

<script src="script.js"></script>

</head>

<body>

<button onclick="printAllPages()">Print All Pages</button>

</body>

</html>

JavaScript (script.js):

javascript

function printAllPages() {

var pages = ['page1.html', 'page2.html', 'page3.html']; // List of pages to print

// Loop through each page and open it in a new window for printing

for (var i = 0; i < pages.length; i++) {

var newWindow = window.open(pages[i]);

// Wait for the page to finish loading before triggering the print function

newWindow.onload = function() {

newWindow.print();

newWindow.close();

};

}

}

In the above example, the printAllPages() function is called when the "Print All Pages" button is clicked. It opens each page in a new window using window.open(), waits for the page to load using onload, and then triggers the print function using print(). Finally, the window is closed using close().

Please note that this code assumes you have separate HTML files for each page (page1.html, page2.html, etc.) and that you have appropriately styled those pages with CSS for printing.

You can customize the pages array in the JavaScript code with the actual URLs of your pages (featurestab.html and productgroupstable.html) to achieve the desired functionality.

make var say const instead

Certainly! Here's the updated code snippet using const instead of var for the pages array:

javascript

function printAllPages() {

const pages = ['page1.html', 'page2.html', 'page3.html']; // List of pages to print

// Loop through each page and open it in a new window for printing

for (let i = 0; i < pages.length; i++) {

const newWindow = window.open(pages[i]);

// Wait for the page to finish loading before triggering the print function

newWindow.onload = function() {

newWindow.print();

newWindow.close();

};

}

}

By using const, we ensure that the pages array remains constant and cannot be reassigned. The loop counter variable i is updated to use let instead of var to ensure block-level scoping within the loop.

Remember to adjust the URLs in the pages array to match the desired pages (featurestab.html and productgroupstable.html) you want to print.