To ensure that the PDF generated with jsPDF closely matches the appearance of the web pages featurestab.html and productgroupstable.html, you can use the html2canvas library to capture a screenshot of each page and then add the images to the PDF. Here's an updated code example:

**index.html:**

html

<!DOCTYPE html>

<html>

<head>

<title>Print Pages as PDF</title>

<script src="https://cdnjs.cloudflare.com/ajax/libs/jspdf/1.3.2/jspdf.debug.js"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/html2canvas/0.4.1/html2canvas.min.js"></script>

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

</head>

<body>

<button onclick="printPages()">Print Pages</button>

</body>

</html>

**script.js:**

javascript

function printPages() {

var pdf = new jsPDF(); // Create a new jsPDF instance

// Load the first page

html2canvas(document.getElementById('featurestab'), { scale: 2 })

.then(canvas => {

pdf.addImage(canvas.toDataURL('image/jpeg'), 'JPEG', 10, 10, 190, 0);

addSecondPage();

});

// Load the second page

function addSecondPage() {

html2canvas(document.getElementById('productgroupstable'), { scale: 2 })

.then(canvas => {

pdf.addPage();

pdf.addImage(canvas.toDataURL('image/jpeg'), 'JPEG', 10, 10, 190, 0);

// Save and print the PDF

pdf.save('combined\_document.pdf');

pdf.autoPrint(); // Optionally, automatically trigger the print dialog after saving

});

}

}

In this updated code, we utilize the html2canvas library to capture screenshots of the web pages. We scale the screenshots by a factor of 2 to enhance the quality. Then, we add the screenshots to the PDF using pdf.addImage().

Make sure to add id attributes to the relevant elements in your HTML files: featurestab.html and productgroupstable.html. For example:

html

<!-- In featurestab.html -->

<div id="featurestab">

<!-- Contents of the featurestab page -->

</div>

<!-- In productgroupstable.html -->

<div id="productgroupstable">

<!-- Contents of the productgroupstable page -->

</div>

When you click the "Print Pages" button, it will capture screenshots of the web pages using html2canvas, add the images to the PDF using jsPDF, and save the PDF document. The resulting PDF should closely match the appearance of the web pages.

Please note that due to the nature of capturing screenshots, some complex or interactive elements may not be faithfully reproduced in the PDF.