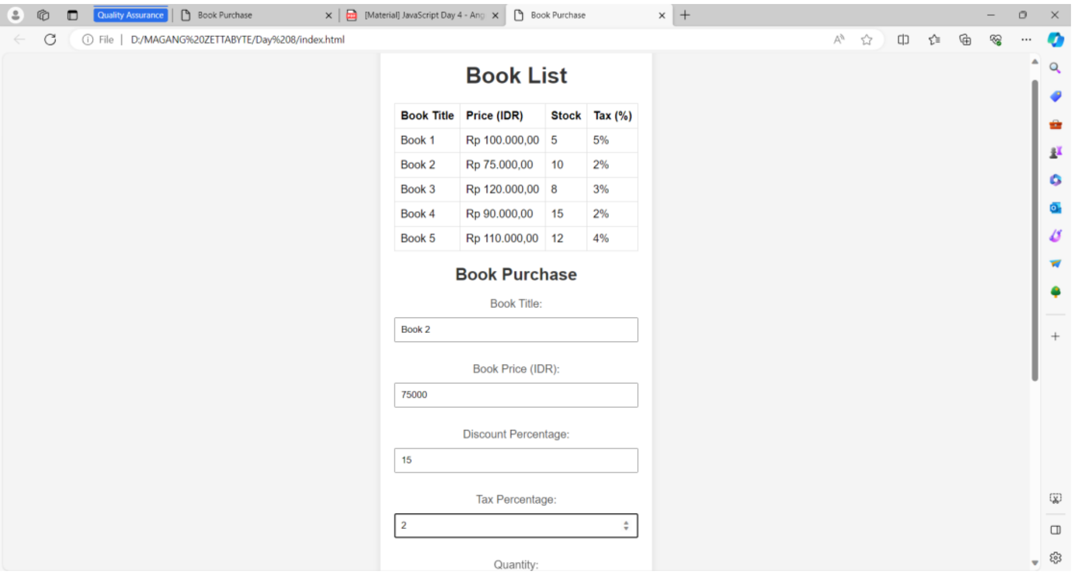
LEARNING

1. Introduction to iterable in JavaScript. Watch course section 8 part 184: [Part 184]()
2. Array function in JavaScript: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array

TASK

1. Update your book purchasing function to add parameter for the total duration of credit (indicating the credit term length in months) and calculate the due date for each month **Starting from the next month when you work on this code using array function in javascript** and display the results as an array of strings.


Quality Assurance
Book Purchase
[Material] JavaScript Day 4 - Ang
Book Purchase
File | D:/MAGANG%20ZETTABYTE/Day%208/index.html

Book List

Book Title	Price (IDR)	Stock	Tax (%)
Book 1	Rp 100.000,00	5	5%
Book 2	Rp 75.000,00	10	2%
Book 3	Rp 120.000,00	8	3%
Book 4	Rp 90.000,00	15	2%
Book 5	Rp 110.000,00	12	4%

Book Purchase

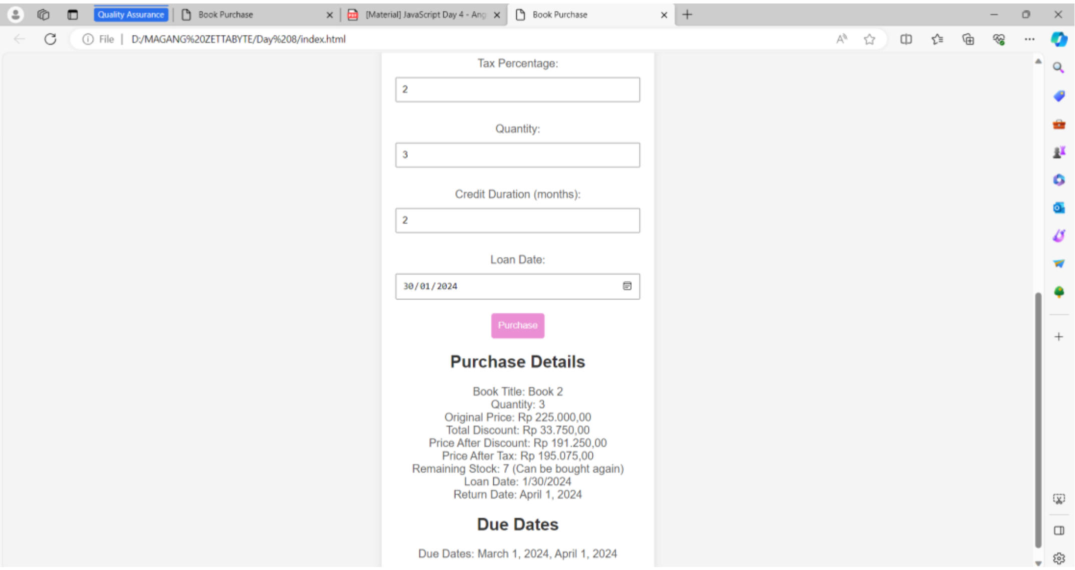
Book Title: Book 2

Book Price (IDR): 75000

Discount Percentage: 15

Tax Percentage: 2

Quantity:


Quality Assurance
Book Purchase
[Material] JavaScript Day 4 - Ang
Book Purchase
File | D:/MAGANG%20ZETTABYTE/Day%208/index.html
Purchase

Tax Percentage: 2

Quantity: 3

Credit Duration (months): 2

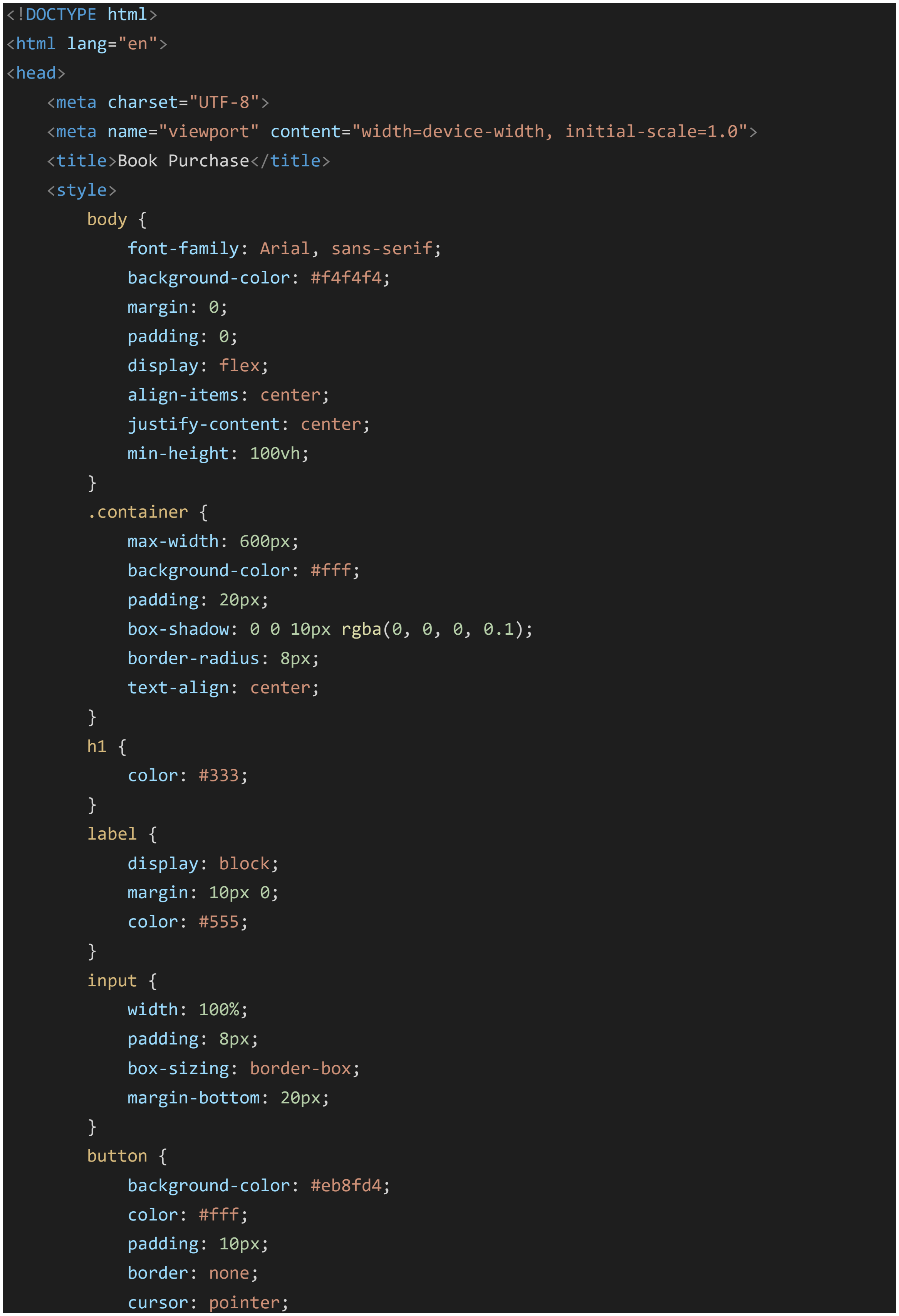
Loan Date: 30/01/2024

Purchase Details

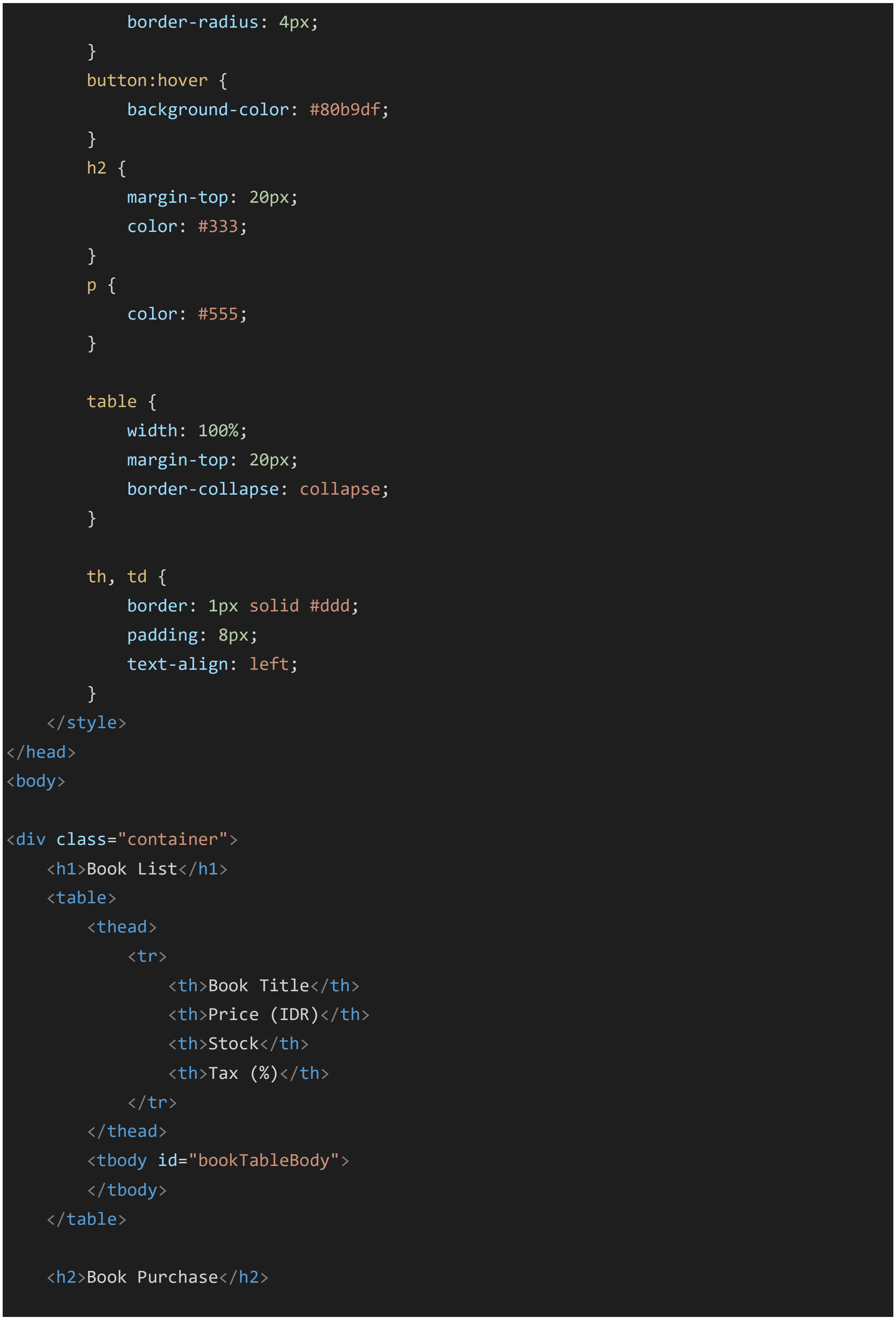
Book Title: Book 2
Quantity: 3
Original Price: Rp 225.000,00
Total Discount: Rp 33.750,00
Price After Discount: Rp 191.250,00
Price After Tax: Rp 195.075,00
Remaining Stock: 7 (Can be bought again)
Loan Date: 1/30/2024
Return Date: April 1, 2024

Due Dates

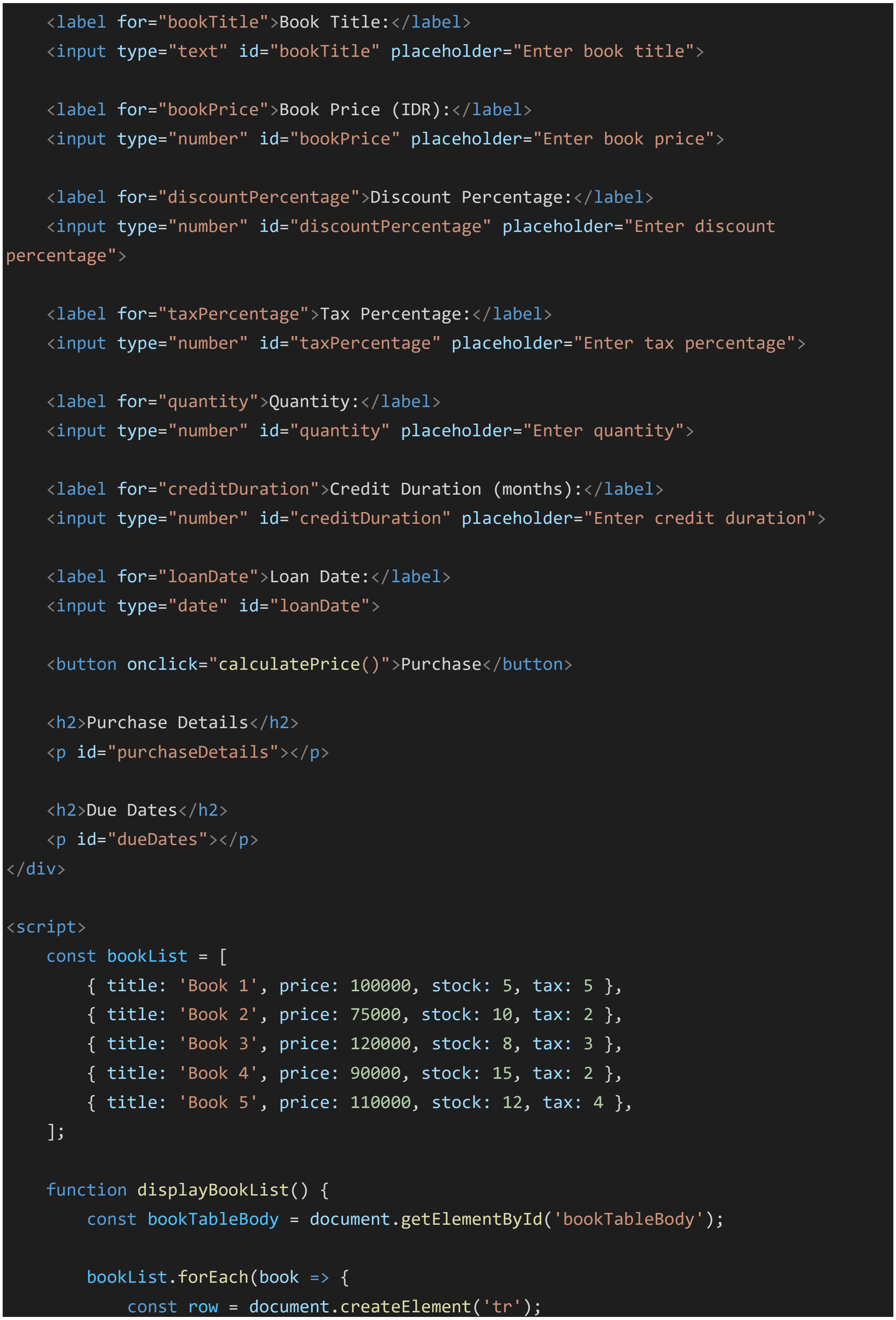
Due Dates: March 1, 2024, April 1, 2024



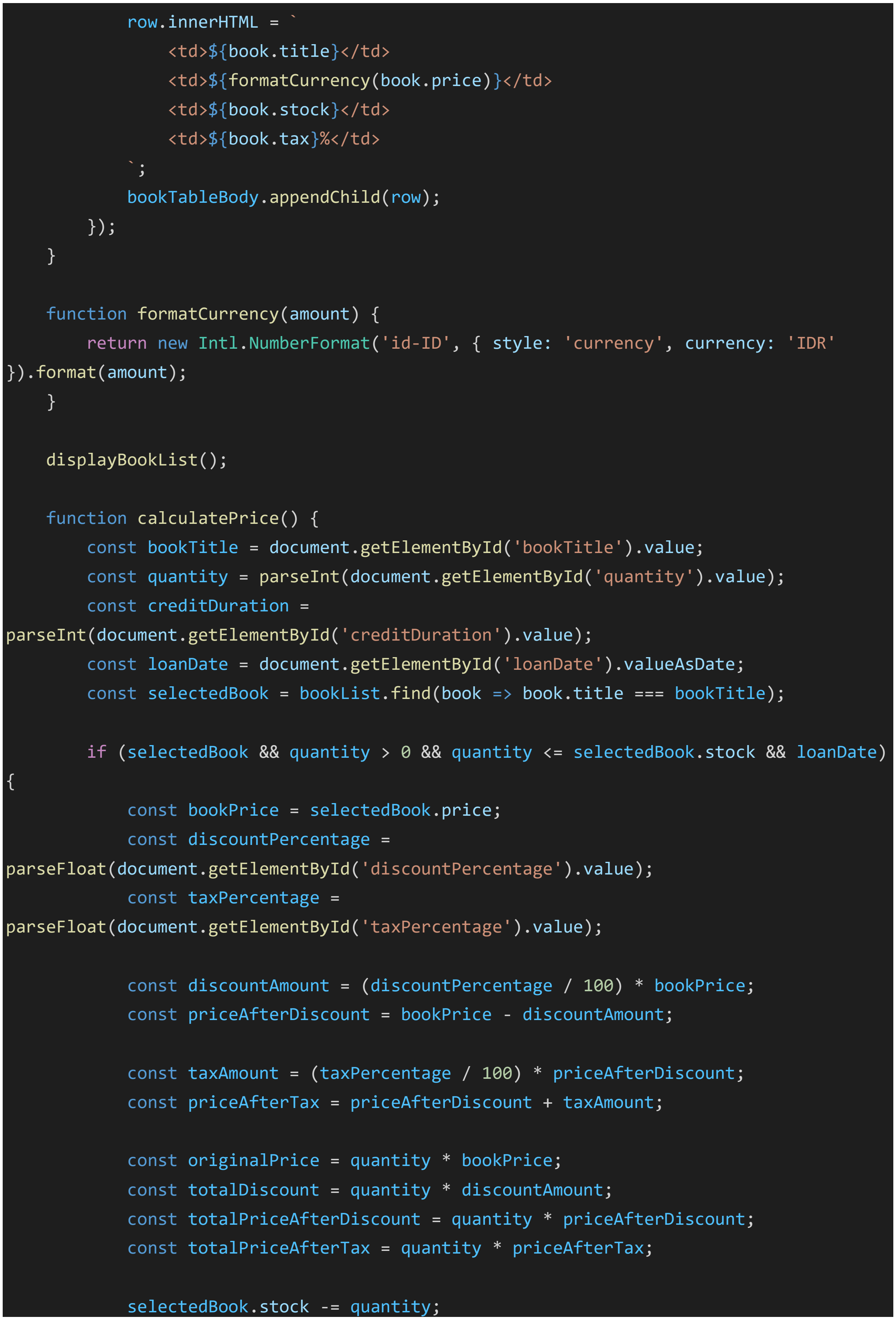
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Book Purchase</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #f4f4f4;
            margin: 0;
            padding: 0;
            display: flex;
            align-items: center;
            justify-content: center;
            min-height: 100vh;
        }
        .container {
            max-width: 600px;
            background-color: #fff;
            padding: 20px;
            box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
            border-radius: 8px;
            text-align: center;
        }
        h1 {
            color: #333;
        }
        label {
            display: block;
            margin: 10px 0;
            color: #555;
        }
        input {
            width: 100%;
            padding: 8px;
            box-sizing: border-box;
            margin-bottom: 20px;
        }
        button {
            background-color: #eb8fd4;
            color: #fff;
            padding: 10px;
            border: none;
            cursor: pointer;
```



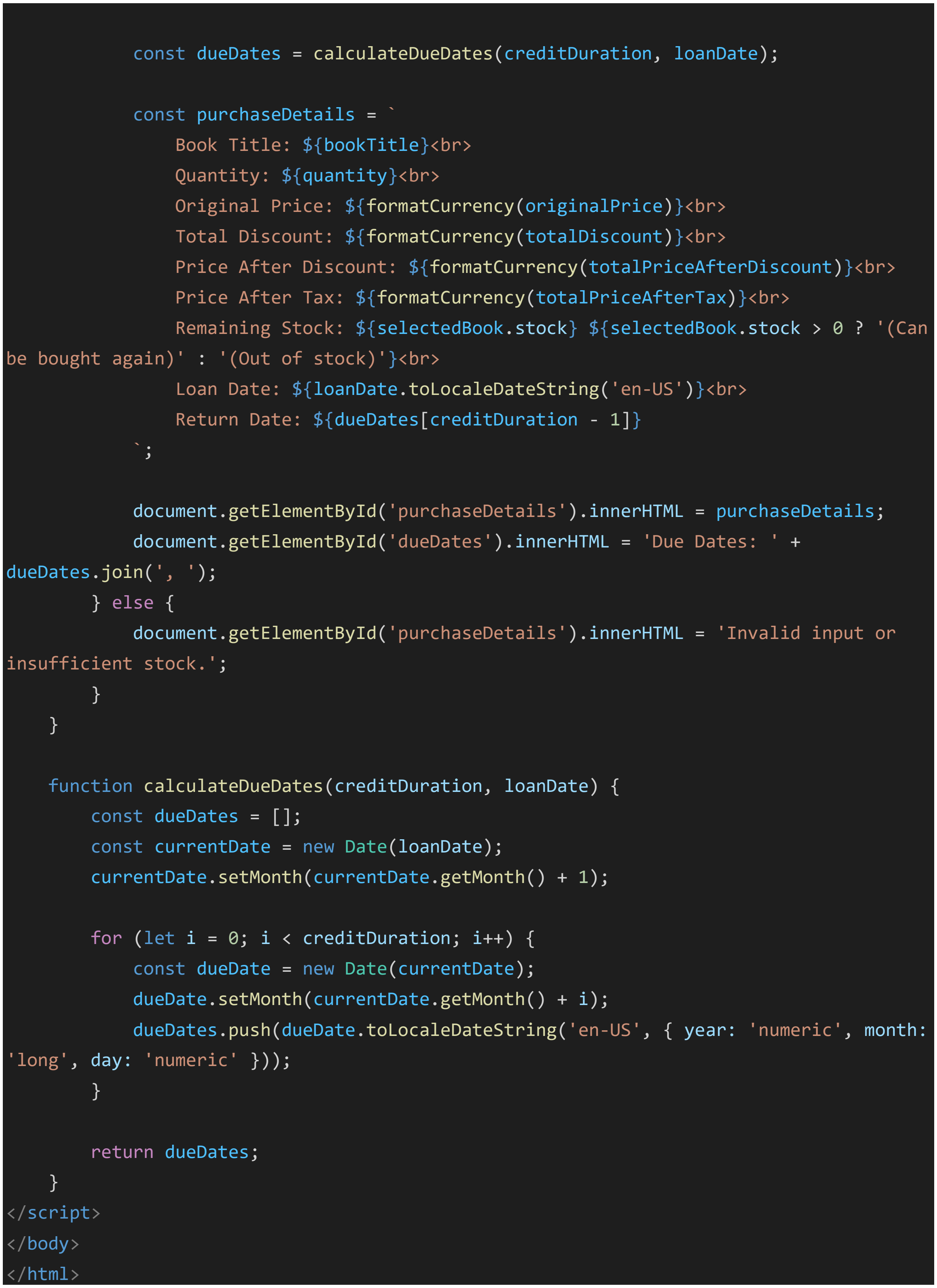
```
            border-radius: 4px;
        }
        button:hover {
            background-color: #80b9df;
        }
        h2 {
            margin-top: 20px;
            color: #333;
        }
        p {
            color: #555;
        }

        table {
            width: 100%;
            margin-top: 20px;
            border-collapse: collapse;
        }

        th, td {
            border: 1px solid #ddd;
            padding: 8px;
            text-align: left;
        }
    </style>
</head>
<body>

<div class="container">
    <h1>Book List</h1>
    <table>
        <thead>
            <tr>
                <th>Book Title</th>
                <th>Price (IDR)</th>
                <th>Stock</th>
                <th>Tax (%)</th>
            </tr>
        </thead>
        <tbody id="bookTableBody">
        </tbody>
    </table>

    <h2>Book Purchase</h2>
```



```
    <label for="bookTitle">Book Title:</label>
    <input type="text" id="bookTitle" placeholder="Enter book title">

    <label for="bookPrice">Book Price (IDR):</label>
    <input type="number" id="bookPrice" placeholder="Enter book price">

    <label for="discountPercentage">Discount Percentage:</label>
    <input type="number" id="discountPercentage" placeholder="Enter discount
percentage">

    <label for="taxPercentage">Tax Percentage:</label>
    <input type="number" id="taxPercentage" placeholder="Enter tax percentage">

    <label for="quantity">Quantity:</label>
    <input type="number" id="quantity" placeholder="Enter quantity">

    <label for="creditDuration">Credit Duration (months):</label>
    <input type="number" id="creditDuration" placeholder="Enter credit duration">

    <label for="loanDate">Loan Date:</label>
    <input type="date" id="loanDate">

    <button onclick="calculatePrice()">Purchase</button>

    <h2>Purchase Details</h2>
    <p id="purchaseDetails"></p>

    <h2>Due Dates</h2>
    <p id="dueDates"></p>
</div>

<script>
    const bookList = [
        { title: 'Book 1', price: 100000, stock: 5, tax: 5 },
        { title: 'Book 2', price: 75000, stock: 10, tax: 2 },
        { title: 'Book 3', price: 120000, stock: 8, tax: 3 },
        { title: 'Book 4', price: 90000, stock: 15, tax: 2 },
        { title: 'Book 5', price: 110000, stock: 12, tax: 4 },
    ];

    function displayBookList() {
        const bookTableBody = document.getElementById('bookTableBody');

        bookList.forEach(book => {
            const row = document.createElement('tr');
```



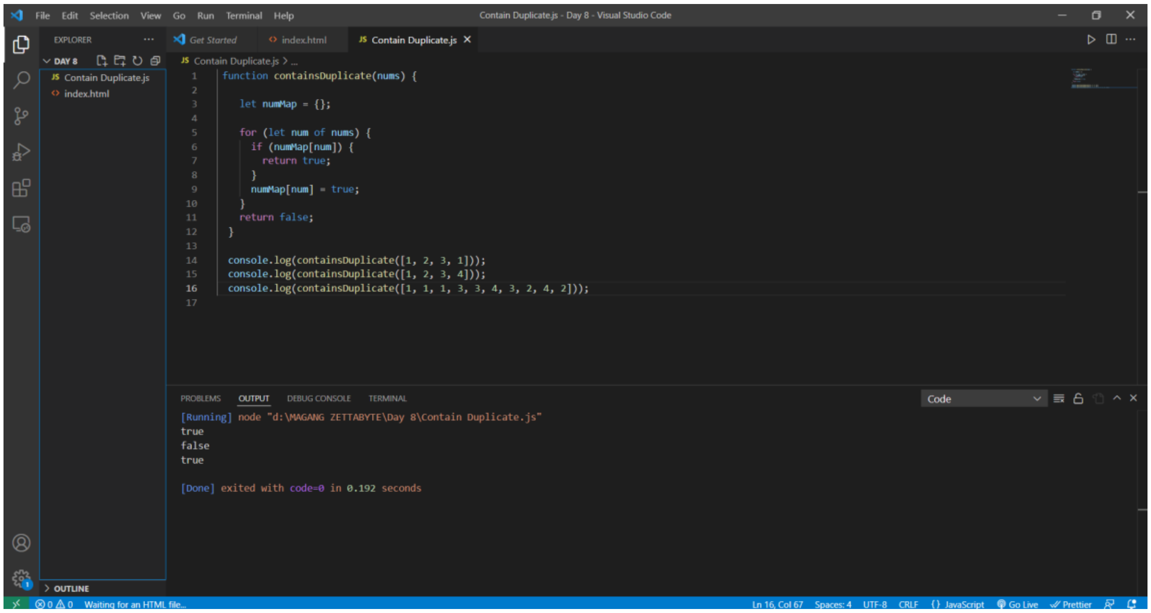
```
            row.innerHTML = `
                <td>${book.title}</td>
                <td>${formatCurrency(book.price)}</td>
                <td>${book.stock}</td>
                <td>${book.tax}%</td>
            `;
            bookTableBody.appendChild(row);
        });
    }

    function formatCurrency(amount) {
        return new Intl.NumberFormat('id-ID', { style: 'currency', currency: 'IDR'
}).format(amount);
    }

    displayBookList();

    function calculatePrice() {
        const bookTitle = document.getElementById('bookTitle').value;
        const quantity = parseInt(document.getElementById('quantity').value);
        const creditDuration =
parseInt(document.getElementById('creditDuration').value);
        const loanDate = document.getElementById('loanDate').valueAsDate;
        const selectedBook = bookList.find(book => book.title === bookTitle);

        if (selectedBook && quantity > 0 && quantity <= selectedBook.stock && loanDate)
{
            const bookPrice = selectedBook.price;
            const discountPercentage =
parseFloat(document.getElementById('discountPercentage').value);
            const taxPercentage =
parseFloat(document.getElementById('taxPercentage').value);

            const discountAmount = (discountPercentage / 100) * bookPrice;
            const priceAfterDiscount = bookPrice - discountAmount;

            const taxAmount = (taxPercentage / 100) * priceAfterDiscount;
            const priceAfterTax = priceAfterDiscount + taxAmount;

            const originalPrice = quantity * bookPrice;
            const totalDiscount = quantity * discountAmount;
            const totalPriceAfterDiscount = quantity * priceAfterDiscount;
            const totalPriceAfterTax = quantity * priceAfterTax;

            selectedBook.stock -= quantity;
```



```
            const dueDates = calculateDueDates(creditDuration, loanDate);

            const purchaseDetails = `
                Book Title: ${bookTitle}<br>
                Quantity: ${quantity}<br>
                Original Price: ${formatCurrency(originalPrice)}<br>
                Total Discount: ${formatCurrency(totalDiscount)}<br>
                Price After Discount: ${formatCurrency(totalPriceAfterDiscount)}<br>
                Price After Tax: ${formatCurrency(totalPriceAfterTax)}<br>
                Remaining Stock: ${selectedBook.stock} ${selectedBook.stock > 0 ? '(Can be bought again)' : '(Out of stock)'}<br>
                Loan Date: ${loanDate.toLocaleDateString('en-US')}<br>
                Return Date: ${dueDates[creditDuration - 1]}
            `;

            document.getElementById('purchaseDetails').innerHTML = purchaseDetails;
            document.getElementById('dueDates').innerHTML = 'Due Dates: ' + dueDates.join(', ');
        } else {
            document.getElementById('purchaseDetails').innerHTML = 'Invalid input or insufficient stock.';
        }
    }

    function calculateDueDates(creditDuration, loanDate) {
        const dueDates = [];
        const currentDate = new Date(loanDate);
        currentDate.setMonth(currentDate.getMonth() + 1);

        for (let i = 0; i < creditDuration; i++) {
            const dueDate = new Date(currentDate);
            dueDate.setMonth(currentDate.getMonth() + i);
            dueDates.push(dueDate.toLocaleDateString('en-US', { year: 'numeric', month: 'long', day: 'numeric' }));
        }

        return dueDates;
    }
</script>
</body>
</html>
```

Logic Test

1. Contain Duplicate : https://drive.google.com/file/d/1PgkR0nKsubiRjX71oG-nDefHXHM8zG9F/view



File Edit Selection View Go Run Terminal Help
Contain Duplicate.js - Day 8 - Visual Studio Code
EXPLORER
DAY 8
Contain Duplicate.js
index.html
Get Started
index.html
Contain Duplicate.js
Contain Duplicate.js > ...
OUTLINE
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
Waiting for an HTML file...
Ln 16, Col 67 Spaces: 4 UTF-8 CRLF {} JavaScript Go Live Prettier

```
function containsDuplicate(nums) {

  let numMap = {};

  for (let num of nums) {
    if (numMap[num]) {
      return true;
    }
    numMap[num] = true;
  }
  return false;
 }

 console.log(containsDuplicate([1, 2, 3, 1]));
 console.log(containsDuplicate([1, 2, 3, 4]));
 console.log(containsDuplicate([1, 1, 1, 3, 3, 4, 3, 2, 4, 2]));
```

```
[Running] node "d:\MAGANG ZETTABYTE\Day 8\Contain Duplicate.js"
true
false
true

[Done] exited with code=0 in 0.192 seconds
```

```
function containsDuplicate(nums) {

   let numMap = {};

   for (let num of nums) {
     if (numMap[num]) {
       return true;
     }
     numMap[num] = true;
   }
   return false;
 }

 console.log(containsDuplicate([1, 2, 3, 1]));
 console.log(containsDuplicate([1, 2, 3, 4]));
 console.log(containsDuplicate([1, 1, 1, 3, 3, 4, 3, 2, 4, 2]));
```

Output:
true
False
true