System Reservasi Ruangan

Index.html

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Selamat Datang di Sistem Reservasi</title>
    <!-- Link Bootstrap CSS -->
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
    <link rel="stylesheet" href="css/style.css">
</head>

<body>
    <div class="container my-5 text-center">
        <!-- Header Utama -->
        <h1 class="display-4 mb-4">Selamat Datang di Sistem Reservasi
Ruangan</h1>
        <p class="lead mb-4">Tempat terbaik untuk mengelola reservasi ruangan
Anda dengan mudah dan cepat.</p>

        <!-- Pilihan Menu Reservasi -->
        <div class="row">
            <div class="col-md-6 mb-4">
                <a href="reservation.html" class="btn btn-lg btn-primary btn-
block">Buat Reservasi Baru</a>
            </div>
            <div class="col-md-6 mb-4">
                <a href="reservations-list.html" class="btn btn-lg btn-primary
btn-block">Lihat Daftar Reservasi</a>
            </div>
        </div>

        <!-- Footer -->
        <div class="mt-4">
            <p class="small">Powered by <strong>Reservasi Ruangan</strong></p>
        </div>
    </div>

    <!-- Link Script JS untuk Bootstrap dan script.js -->
    <script src="https://code.jquery.com/jquery-3.5.1.min.js"></script>
    <script
src="https://cdn.jsdelivr.net/npm/bootstrap@4.5.2/dist/js/bootstrap.bundle.min
.js"></script>
    <script src="js/script.js"></script>
</body>

</html>
```

Reservation.html

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Buat Reservasi Baru</title>
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
    <link rel="stylesheet" href="css/style.css">
</head>

<body>
    <div class="container my-5">
        <h2 class="mb-4">Buat Reservasi Baru</h2>
        <form id="reservation-form">
            <div class="form-group">
                <label for="name">Nama Pemesan</label>
                <input type="text" class="form-control" id="name" required>
            </div>
            <div class="form-group">
                <label for="roomNumber">Nomor Ruangan</label>
                <select class="form-control" id="roomNumber" required>
                    <!-- Daftar Ruangan akan dimasukkan disini oleh JavaScript
-->
                </select>
            </div>
            <div class="form-group">
                <label for="date">Tanggal Reservasi</label>
                <input type="date" class="form-control" id="date" required>
            </div>
            <div class="form-group">
                <label for="startTime">Waktu Mulai</label>
                <input type="time" class="form-control" id="startTime"
required>
```

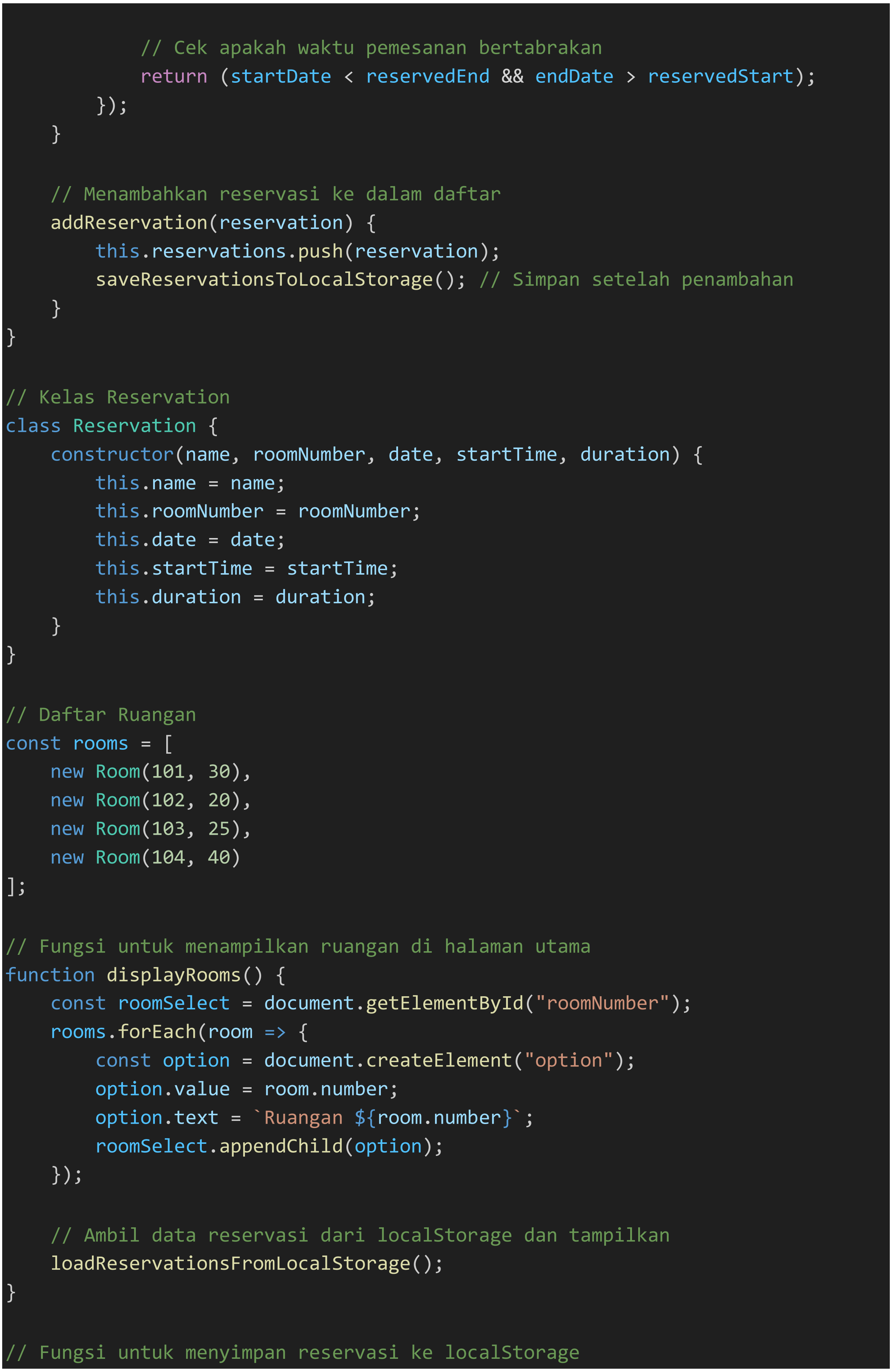
```
            </div>
            <div class="form-group">
                <label for="duration">Durasi (Jam)</label>
                <input type="number" class="form-control" id="duration"
min="1" required>
            </div>
            <button type="submit" class="btn btn-primary">Buat
Reservasi</button>
        </form>
        <a href="reservations-list.html" class="btn btn-primary mt-3">Lihat
Daftar Reservasi</a>
    </div>

    <script src="https://code.jquery.com/jquery-3.5.1.min.js"></script>
    <script
src="https://cdn.jsdelivr.net/npm/bootstrap@4.5.2/dist/js/bootstrap.bundle.min
.js"></script>
    <script src="js/script.js"></script>
</body>

</html>
```

resercations-list.html

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Daftar Reservasi</title>
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
    <link rel="stylesheet" href="css/style.css">
</head>

<body>
    <div class="container my-5">
        <h2 class="mb-4">Daftar Reservasi</h2>
        <table class="table table-bordered" id="reservations-table">
            <thead>
                <tr>
                    <th>#</th>
                    <th>Nama Pemesan</th>
                    <th>Nomor Ruangan</th>
                    <th>Tanggal</th>
                    <th>Waktu Mulai</th>
                    <th>Durasi</th>
                    <th>Aksi</th>
                </tr>
            </thead>
            <tbody id="reservations-table-body">
                <!-- Data Reservasi Akan Ditambahkan Disini -->
            </tbody>
        </table>
        <a href="index.html" class="btn btn-secondary">Kembali ke Halaman
Utama</a>
    </div>

    <script src="https://code.jquery.com/jquery-3.5.1.min.js"></script>
    <script
src="https://cdn.jsdelivr.net/npm/bootstrap@4.5.2/dist/js/bootstrap.bundle.min
.js"></script>
    <script src="js/script.js"></script>
</body>

</html>
```

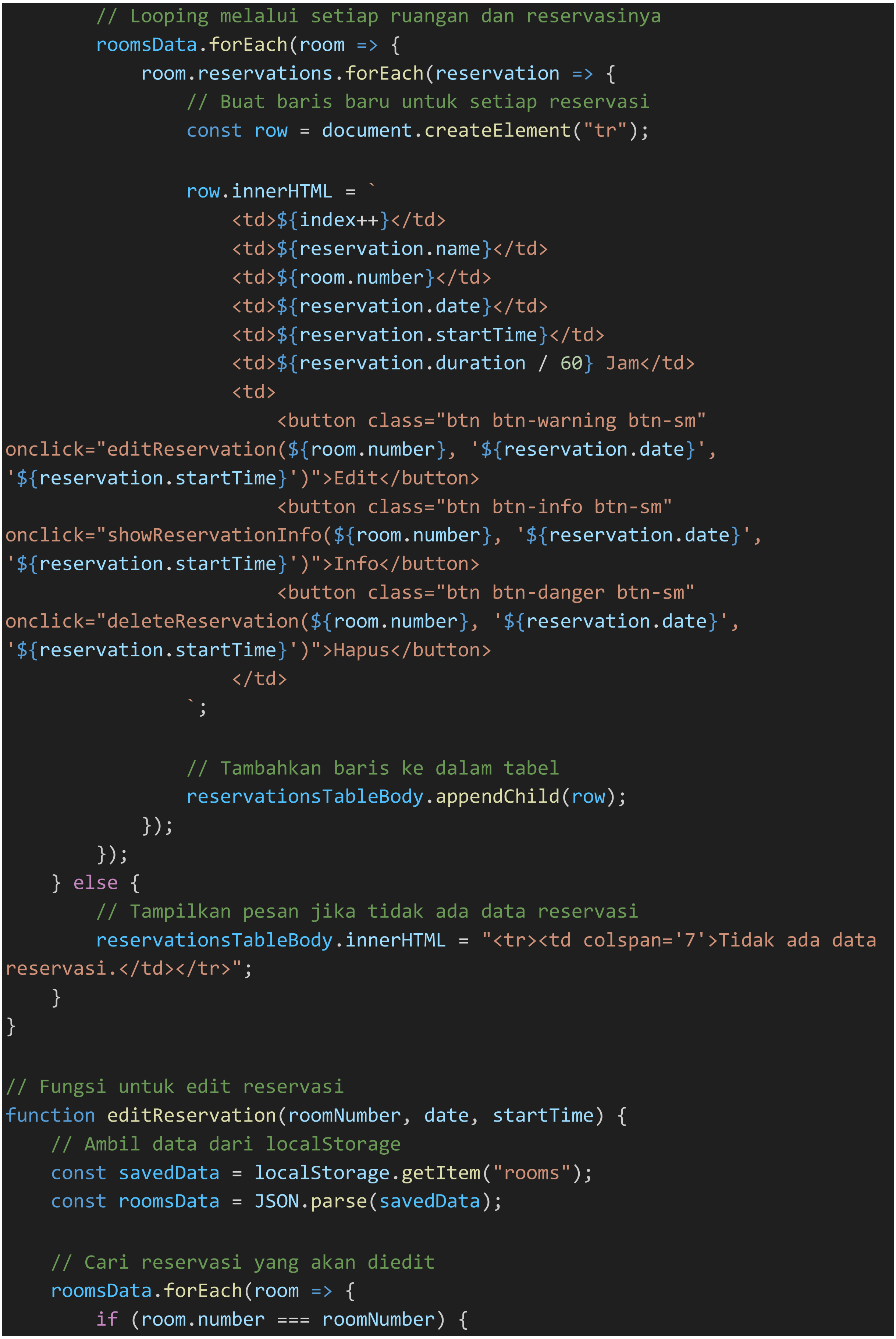
Js/script.js

```
// Kelas Room dan Reservation
class Room {
    constructor(number, capacity) {
        this.number = number;
        this.capacity = capacity;
        this.reservations = []; // Menyimpan daftar reservasi
    }

    // Fungsi untuk mengecek ketersediaan ruangan
    isAvailable(date, startTime, duration) {
        // Ubah waktu menjadi format yang bisa dihitung
        const startDate = new Date(`${date}T${startTime}:00`);
        const endDate = new Date(startDate.getTime() + duration * 60000); //
Durasi dalam menit

        // Periksa apakah ada reservasi yang bentrok
        return !this.reservations.some(reservation => {
            const reservedStart = new
Date(`${reservation.date}T${reservation.startTime}:00`);
            const reservedEnd = new Date(reservedStart.getTime() +
reservation.duration * 60000);
```

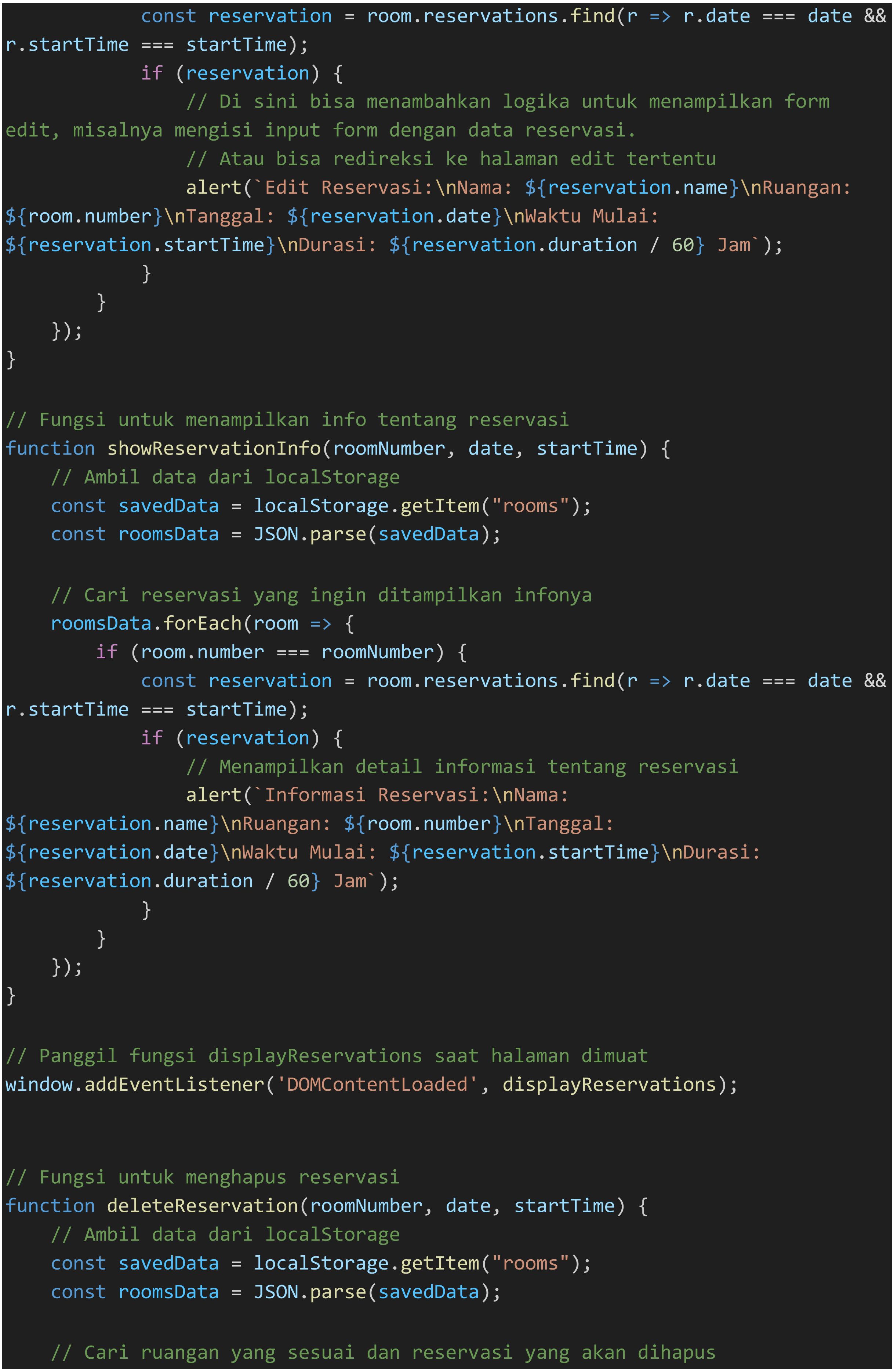


```javascript
            // Cek apakah waktu pemesanan bertabrakan
            return (startDate < reservedEnd && endDate > reservedStart);
        });
    }

    // Menambahkan reservasi ke dalam daftar
    addReservation(reservation) {
        this.reservations.push(reservation);
        saveReservationsToLocalStorage(); // Simpan setelah penambahan
    }
}

// Kelas Reservation
class Reservation {
    constructor(name, roomNumber, date, startTime, duration) {
        this.name = name;
        this.roomNumber = roomNumber;
        this.date = date;
        this.startTime = startTime;
        this.duration = duration;
    }
}

// Daftar Ruangan
const rooms = [
    new Room(101, 30),
    new Room(102, 20),
    new Room(103, 25),
    new Room(104, 40)
];

// Fungsi untuk menampilkan ruangan di halaman utama
function displayRooms() {
    const roomSelect = document.getElementById("roomNumber");
    rooms.forEach(room => {
        const option = document.createElement("option");
        option.value = room.number;
        option.text = `Ruangan ${room.number}`;
        roomSelect.appendChild(option);
    });

    // Ambil data reservasi dari localStorage dan tampilkan
    loadReservationsFromLocalStorage();
}

// Fungsi untuk menyimpan reservasi ke localStorage
```

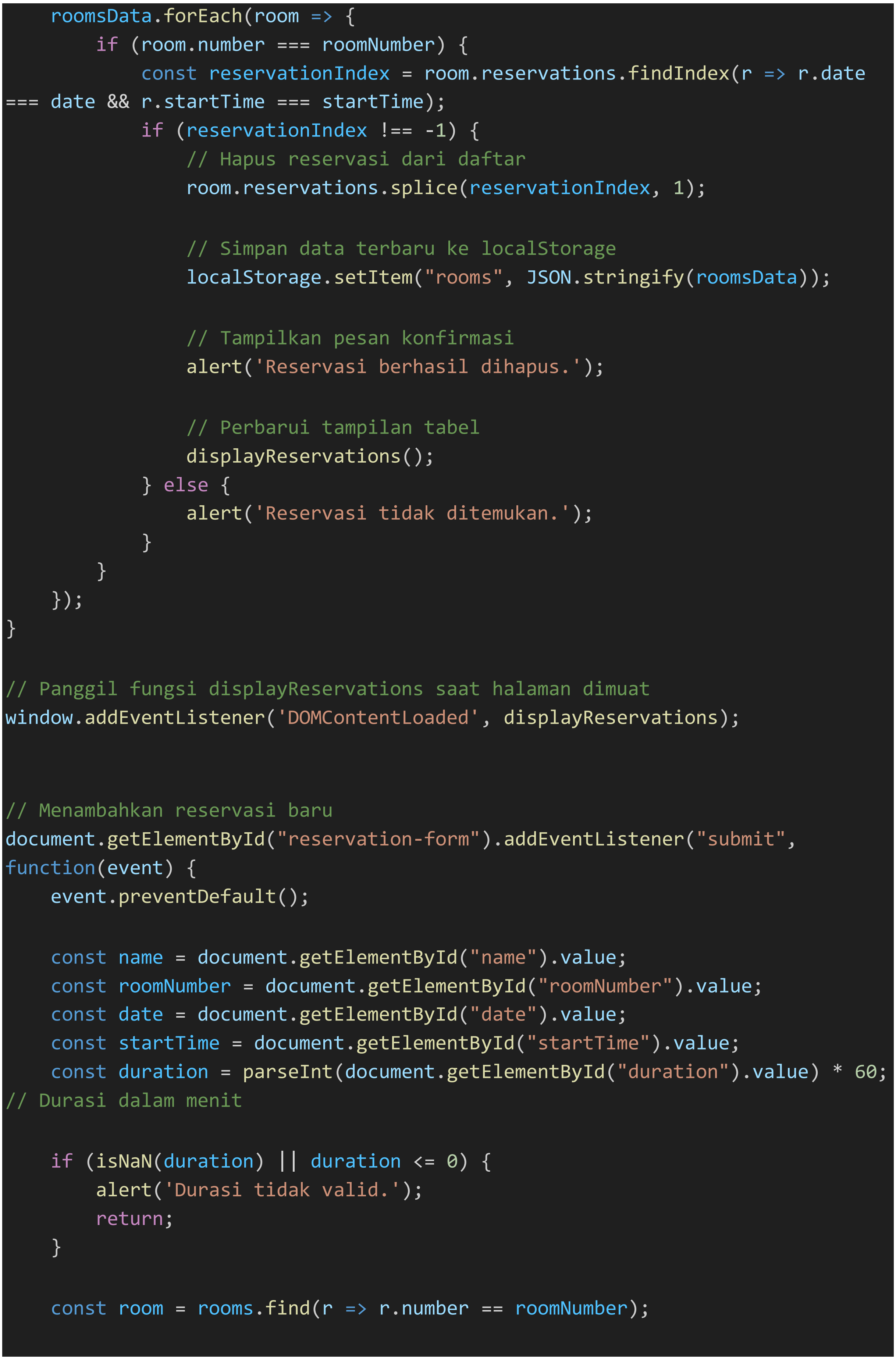
```
function saveReservationsToLocalStorage() {
    const reservationsData = rooms.map(room => ({
        number: room.number,
        reservations: room.reservations
    }));
    localStorage.setItem("rooms", JSON.stringify(reservationsData)); //
Menyimpan data ruangan dan reservasi
}

// Fungsi untuk memuat reservasi dari localStorage
function loadReservationsFromLocalStorage() {
    const savedData = localStorage.getItem("rooms");
    if (savedData) {
        const roomsData = JSON.parse(savedData);
        roomsData.forEach(savedRoom => {
            const room = rooms.find(r => r.number === savedRoom.number);
            if (room) {
                room.reservations = savedRoom.reservations.map(reservation =>
new Reservation(
                    reservation.name,
                    reservation.roomNumber,
                    reservation.date,
                    reservation.startTime,
                    reservation.duration
                ));
            }
        });
    }
}


// Fungsi untuk menampilkan data reservasi dari localStorage
function displayReservations() {
    const reservationsTableBody = document.getElementById("reservations-table-
body");
    reservationsTableBody.innerHTML = ""; // Kosongkan tabel untuk memastikan
tidak ada data duplikat

    // Ambil data dari localStorage
    const savedData = localStorage.getItem("rooms");

    // Periksa apakah ada data di localStorage
    if (savedData) {
        // Parse data JSON menjadi objek JavaScript
        const roomsData = JSON.parse(savedData);
        let index = 1;
```



```
        // Looping melalui setiap ruangan dan reservasinya
        roomsData.forEach(room => {
            room.reservations.forEach(reservation => {
                // Buat baris baru untuk setiap reservasi
                const row = document.createElement("tr");

                row.innerHTML = `
                    <td>${index++}</td>
                    <td>${reservation.name}</td>
                    <td>${room.number}</td>
                    <td>${reservation.date}</td>
                    <td>${reservation.startTime}</td>
                    <td>${reservation.duration / 60} Jam</td>
                    <td>
                        <button class="btn btn-warning btn-sm"
onclick="editReservation(${room.number}, '${reservation.date}',
'${reservation.startTime}')">Edit</button>
                        <button class="btn btn-info btn-sm"
onclick="showReservationInfo(${room.number}, '${reservation.date}',
'${reservation.startTime}')">Info</button>
                        <button class="btn btn-danger btn-sm"
onclick="deleteReservation(${room.number}, '${reservation.date}',
'${reservation.startTime}')">Hapus</button>
                    </td>
                `;

                // Tambahkan baris ke dalam tabel
                reservationsTableBody.appendChild(row);
            });
        });
    } else {
        // Tampilkan pesan jika tidak ada data reservasi
        reservationsTableBody.innerHTML = "<tr><td colspan='7'>Tidak ada data
reservasi.</td></tr>";
    }
}

// Fungsi untuk edit reservasi
function editReservation(roomNumber, date, startTime) {
    // Ambil data dari localStorage
    const savedData = localStorage.getItem("rooms");
    const roomsData = JSON.parse(savedData);

    // Cari reservasi yang akan diedit
    roomsData.forEach(room => {
        if (room.number === roomNumber) {
```



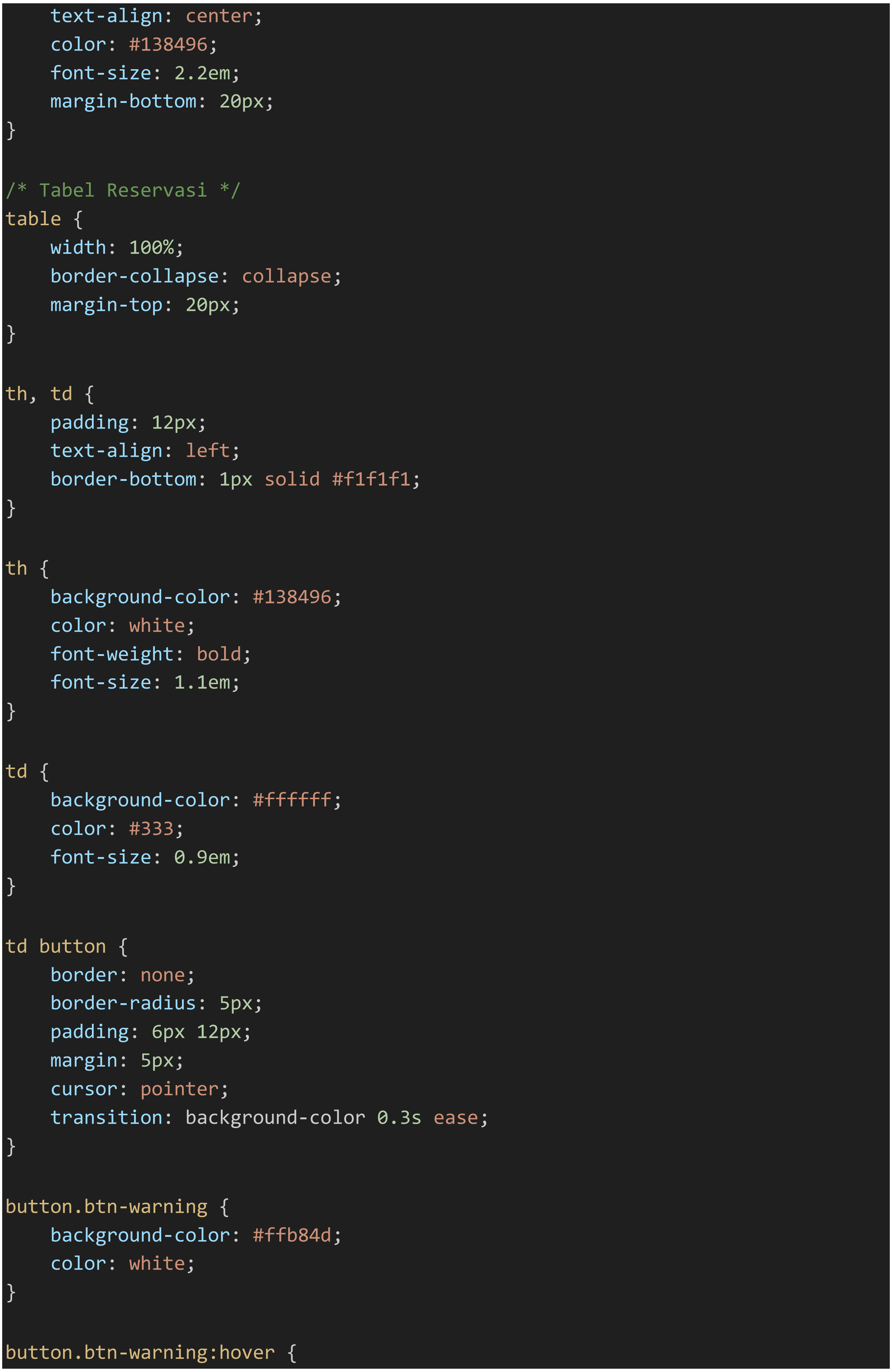
```
            const reservation = room.reservations.find(r => r.date === date &&
r.startTime === startTime);
            if (reservation) {
                // Di sini bisa menambahkan logika untuk menampilkan form
edit, misalnya mengisi input form dengan data reservasi.
                // Atau bisa redireksi ke halaman edit tertentu
                alert(`Edit Reservasi:\nNama: ${reservation.name}\nRuangan:
${room.number}\nTanggal: ${reservation.date}\nWaktu Mulai:
${reservation.startTime}\nDurasi: ${reservation.duration / 60} Jam`);
            }
        }
    });
}

// Fungsi untuk menampilkan info tentang reservasi
function showReservationInfo(roomNumber, date, startTime) {
    // Ambil data dari localStorage
    const savedData = localStorage.getItem("rooms");
    const roomsData = JSON.parse(savedData);

    // Cari reservasi yang ingin ditampilkan infonya
    roomsData.forEach(room => {
        if (room.number === roomNumber) {
            const reservation = room.reservations.find(r => r.date === date &&
r.startTime === startTime);
            if (reservation) {
                // Menampilkan detail informasi tentang reservasi
                alert(`Informasi Reservasi:\nNama:
${reservation.name}\nRuangan: ${room.number}\nTanggal:
${reservation.date}\nWaktu Mulai: ${reservation.startTime}\nDurasi:
${reservation.duration / 60} Jam`);
            }
        }
    });
}

// Panggil fungsi displayReservations saat halaman dimuat
window.addEventListener('DOMContentLoaded', displayReservations);


// Fungsi untuk menghapus reservasi
function deleteReservation(roomNumber, date, startTime) {
    // Ambil data dari localStorage
    const savedData = localStorage.getItem("rooms");
    const roomsData = JSON.parse(savedData);

    // Cari ruangan yang sesuai dan reservasi yang akan dihapus
```



```
    roomsData.forEach(room => {
        if (room.number === roomNumber) {
            const reservationIndex = room.reservations.findIndex(r => r.date
=== date && r.startTime === startTime);
            if (reservationIndex !== -1) {
                // Hapus reservasi dari daftar
                room.reservations.splice(reservationIndex, 1);

                // Simpan data terbaru ke localStorage
                localStorage.setItem("rooms", JSON.stringify(roomsData));

                // Tampilkan pesan konfirmasi
                alert('Reservasi berhasil dihapus.');

                // Perbarui tampilan tabel
                displayReservations();
            } else {
                alert('Reservasi tidak ditemukan.');
            }
        }
    });
}

// Panggil fungsi displayReservations saat halaman dimuat
window.addEventListener('DOMContentLoaded', displayReservations);


// Menambahkan reservasi baru
document.getElementById("reservation-form").addEventListener("submit",
function(event) {
    event.preventDefault();

    const name = document.getElementById("name").value;
    const roomNumber = document.getElementById("roomNumber").value;
    const date = document.getElementById("date").value;
    const startTime = document.getElementById("startTime").value;
    const duration = parseInt(document.getElementById("duration").value) * 60;
// Durasi dalam menit

    if (isNaN(duration) || duration <= 0) {
        alert('Durasi tidak valid.');
        return;
    }

    const room = rooms.find(r => r.number == roomNumber);
    if (room && room.isAvailable(date, startTime, duration)) {
        const reservation = new Reservation(name, roomNumber, date, startTime,
duration);
        room.addReservation(reservation);
        alert('Reservasi berhasil dibuat!');
    } else {
        alert('Ruangan tidak tersedia pada waktu tersebut.');
    }
});

// Panggil fungsi displayRooms dan displayReservations saat halaman dimuat
window.onload = function() {
    displayRooms(); // Untuk halaman reservasi baru
    if (document.getElementById("reservations-table-body")) {
        displayReservations(); // Untuk halaman daftar reservasi
    }
};
```

css/style.css

```
/* Reset dasar */
* {
    margin: 0;
    padding: 0;
    box-sizing: border-box;
}

body {
    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
    background-color: #f4f7f6; /* Latar belakang lembut dan tenang */
    color: #333;
    line-height: 1.6;
}

/* Kontainer utama */
.container {
    max-width: 1100px;
    margin: 30px auto;
    background-color: #ffffff;
    padding: 20px;
    border-radius: 10px;
    box-shadow: 0 4px 15px rgba(0, 0, 0, 0.1);
}

/* Judul halaman */
h1 {
```



```css
    text-align: center;
    color: #138496;
    font-size: 2.2em;
    margin-bottom: 20px;
}

/* Tabel Reservasi */
table {
    width: 100%;
    border-collapse: collapse;
    margin-top: 20px;
}

th, td {
    padding: 12px;
    text-align: left;
    border-bottom: 1px solid #f1f1f1;
}

th {
    background-color: #138496;
    color: white;
    font-weight: bold;
    font-size: 1.1em;
}

td {
    background-color: #ffffff;
    color: #333;
    font-size: 0.9em;
}

td button {
    border: none;
    border-radius: 5px;
    padding: 6px 12px;
    margin: 5px;
    cursor: pointer;
    transition: background-color 0.3s ease;
}

button.btn-warning {
    background-color: #ffb84d;
    color: white;
}

button.btn-warning:hover {
    background-color: #e68a00;
}

button.btn-info {
    background-color: #17a2b8;
    color: white;
}

button.btn-info:hover {
    background-color: #138496;
}

button.btn-danger {
    background-color: #dc3545;
    color: white;
}

button.btn-danger:hover {
    background-color: #c82333;
}

/* Form Styling */
form {
    display: flex;
    flex-direction: column;
    gap: 15px;
}

input[type="text"], input[type="date"], input[type="time"], select {
    width: 100%;
    padding: 10px;
    border: 1px solid #ddd;
    border-radius: 8px;
    font-size: 1em;
    transition: all 0.3s ease;
}

input[type="text"]:focus, input[type="date"]:focus, input[type="time"]:focus,
select:focus {
    border-color: #138496;
    outline: none;
}

/* Styling untuk backdrop background */
body {
    background: url('https://images.unsplash.com/photo-1521747116042-d75fae1b4c0b') no-repeat center center fixed;
    background-size: cover;
    font-family: 'Arial', sans-serif;
    color: #333;
}

/* Styling untuk overlay kontainer */
.container {
    background-color: rgba(255, 255, 255, 0.9); /* Membuat latar belakang kontainer sedikit transparan */
    padding: 30px;
}

/* Tabel dengan efek bayangan */
table {
    border-radius: 8px;
    overflow: hidden;
    box-shadow: 0 4px 10px rgba(0, 0, 0, 0.1);
}

/* Responsiveness untuk perangkat mobile */
@media screen and (max-width: 768px) {
    .container {
        width: 95%;
        padding: 15px;
    }

    table {
        font-size: 0.9em;
    }

    h1 {
        font-size: 1.5em;
    }

    button {
        font-size: 0.9em;
        padding: 5px 10px;
    }
}
```