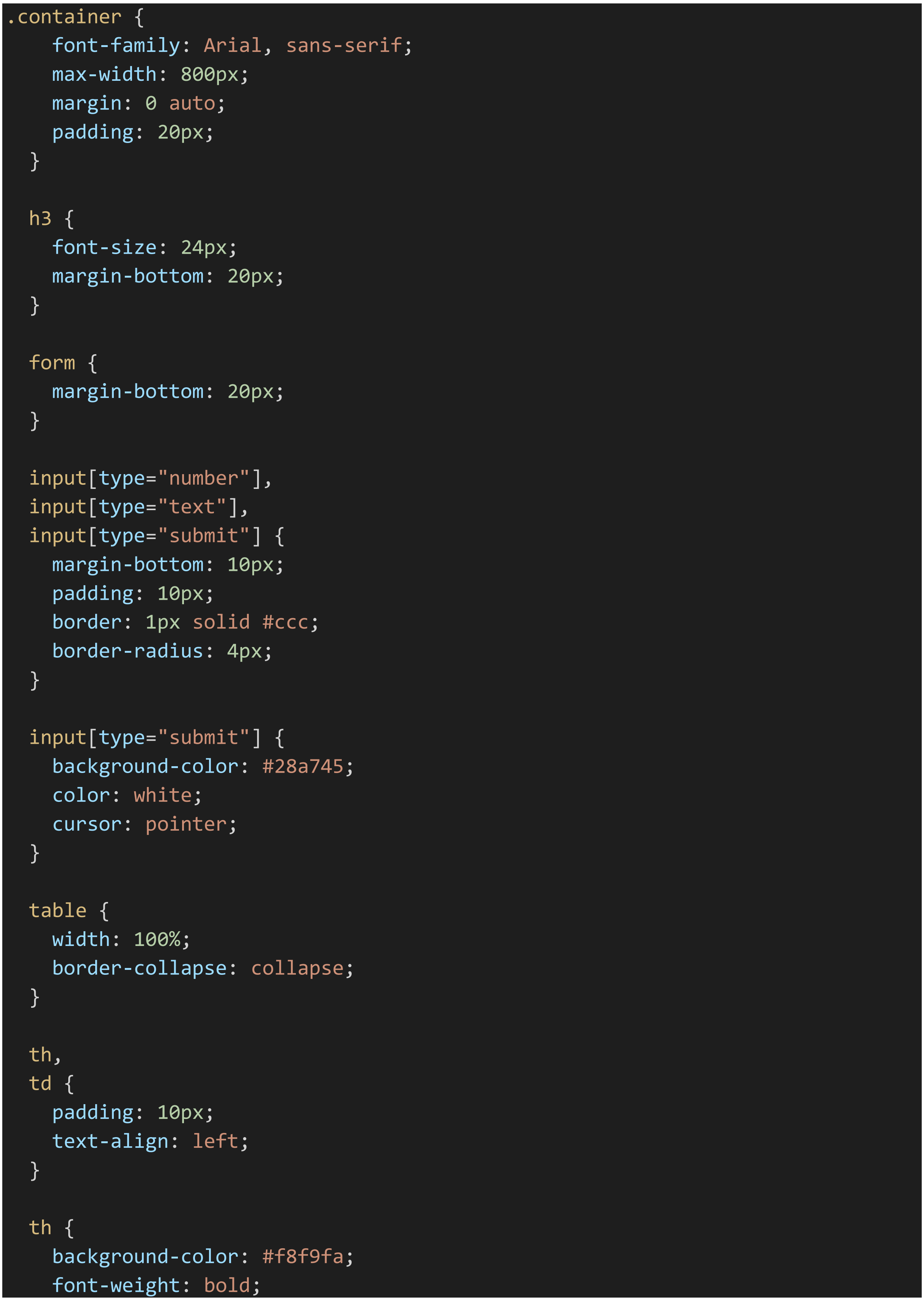
Source code

Employee.component.css



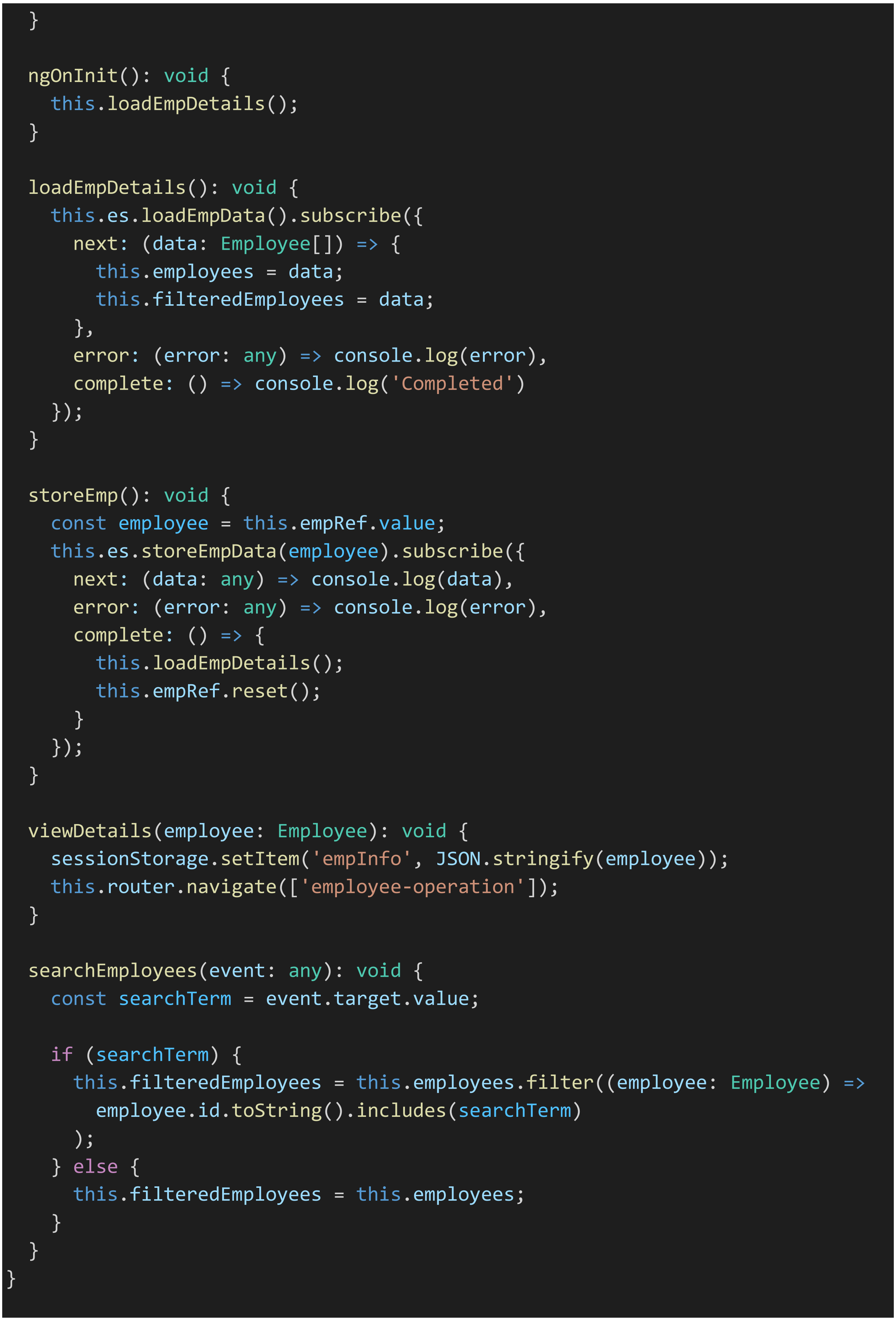
```css
.container {
    font-family: Arial, sans-serif;
    max-width: 800px;
    margin: 0 auto;
    padding: 20px;
  }

  h3 {
    font-size: 24px;
    margin-bottom: 20px;
  }

  form {
    margin-bottom: 20px;
  }

  input[type="number"],
  input[type="text"],
  input[type="submit"] {
    margin-bottom: 10px;
    padding: 10px;
    border: 1px solid #ccc;
    border-radius: 4px;
  }

  input[type="submit"] {
    background-color: #28a745;
    color: white;
    cursor: pointer;
  }

  table {
    width: 100%;
    border-collapse: collapse;
  }

  th,
  td {
    padding: 10px;
    text-align: left;
  }

  th {
    background-color: #f8f9fa;
    font-weight: bold;
  }

  tr:nth-child(even) {
    background-color: #f8f9fa;
  }

  tr:hover {
    background-color: #e9ecef;
  }

  .btn-view-details {
    background-color: #007bff;
    color: white;
    border: none;
    padding: 5px 10px;
    border-radius: 4px;
    cursor: pointer;
  }

  .btn-view-details:hover {
    background-color: #0056b3;
  }
```

Employee.component.html

```
<div class="container">
    <h3>Add New Employee</h3>
    <form [formGroup]="empRef" (ngSubmit)="storeEmp()">
      <input type="number" placeholder="Enter Employee ID"
formControlName="id" required>
      <input type="text" placeholder="Enter First Name"
formControlName="first_name">
      <input type="text" placeholder="Enter Last Name"
formControlName="last_name">
      <input type="text" placeholder="Enter Email" formControlName="email">
      <button type="submit" class="btn btn-success">ADD EMPLOYEE</button>
    </form>

    <br>

    <h3>EMPLOYEE DETAILS</h3>

    <div class="search-container">
<input type="text" placeholder="Search by Employee ID"
(input)="searchEmployees($event)">


    </div>
    <table class="table table-bordered">
      <tr>
        <th scope="col">Employee ID</th>
        <th scope="col">First Name</th>
        <th scope="col">Last Name</th>
        <th scope="col">Email</th>
        <th scope="col">View Details</th>
      </tr>
      <tr *ngFor="let employee of filteredEmployees">
        <td>{{employee.id}}</td>
        <td>{{employee.first_name}}</td>
        <td>{{employee.last_name}}</td>
        <td>{{employee.email}}</td>
        <td><button type="button" class="btn btn-primary"
(click)="viewDetails(employee)">VIEW DETAILS</button></td>
      </tr>
    </table>
  </div>
```

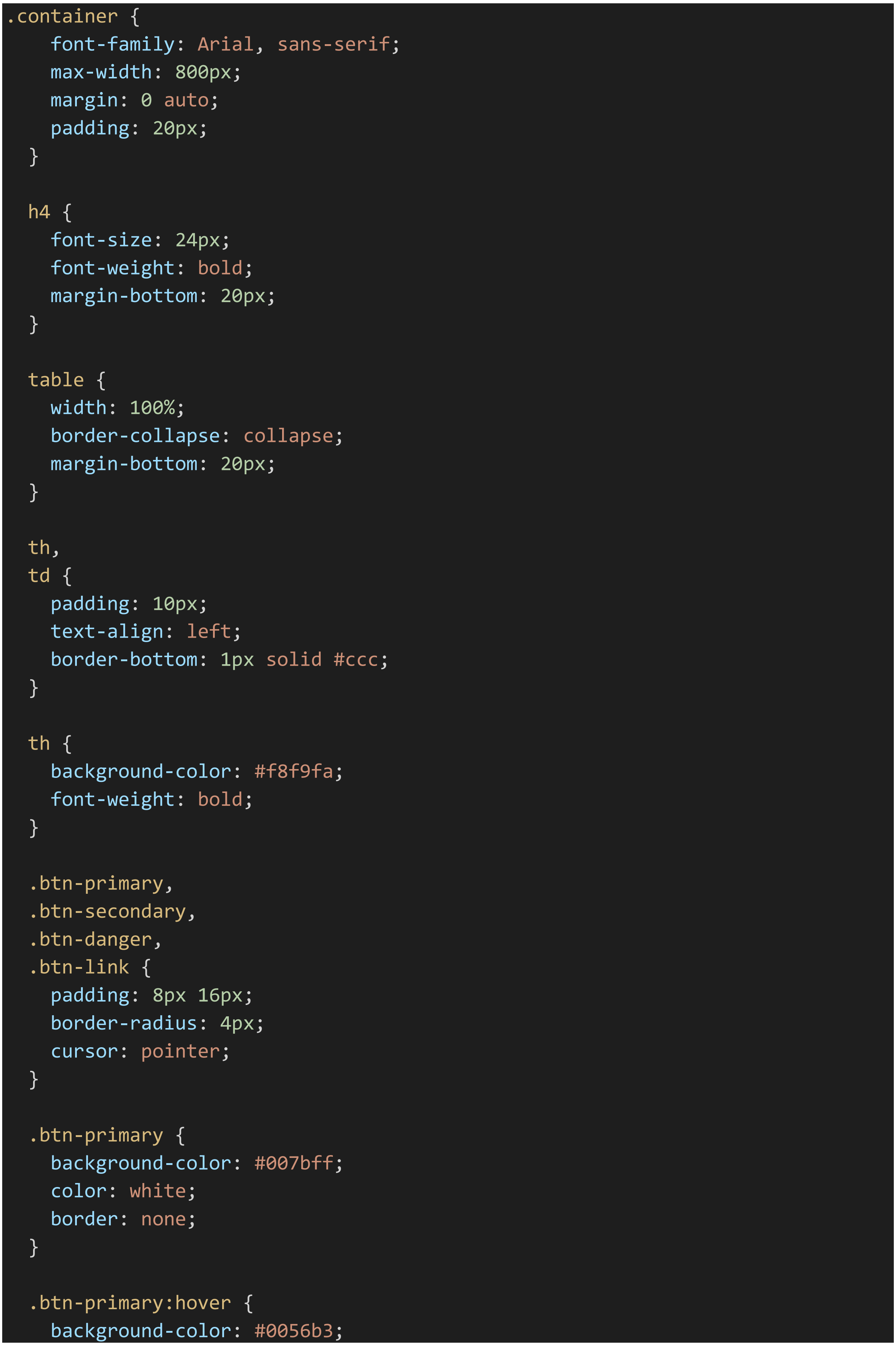
Employee.component.ts

```
import { Component, OnInit } from '@angular/core';
import { Router } from '@angular/router';
import { Employee } from '../employee';
import { EmployeeService } from '../employee.service';
import { FormGroup, FormBuilder, Validators } from '@angular/forms';

@Component({
  selector: 'app-employee',
  templateUrl: './employee.component.html',
  styleUrls: ['./employee.component.css']
})
export class EmployeeComponent implements OnInit {

  empRef: FormGroup;
  employees: Employee[] = [];
  filteredEmployees: Employee[] = [];

  constructor(
    public es: EmployeeService,
    public router: Router,
    private formBuilder: FormBuilder
  ) {
    this.empRef = this.formBuilder.group({
      id: ['', Validators.required],
      first_name: [''],
      last_name: [''],
      email: [''],
    });
```

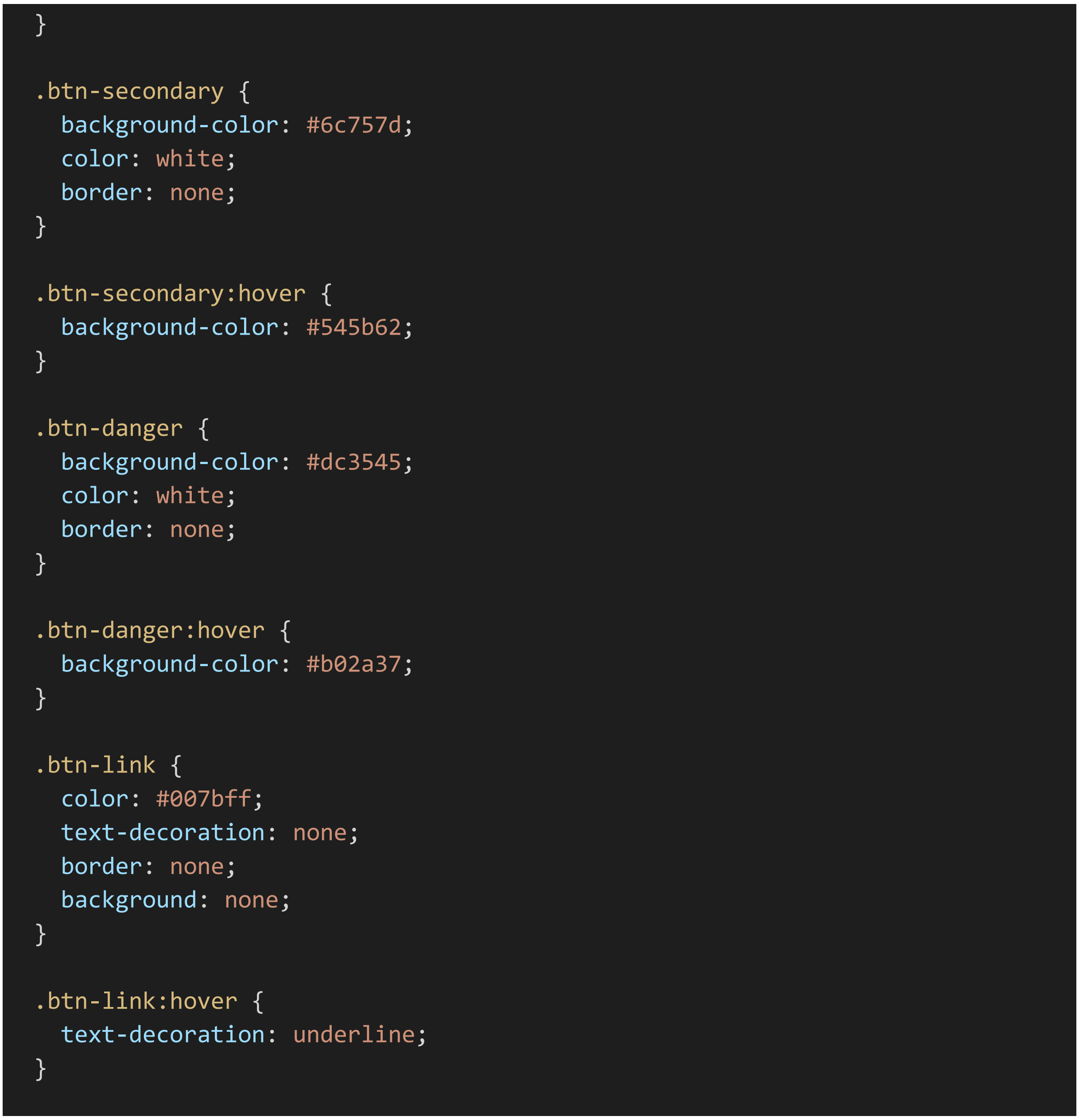


```
  }

  ngOnInit(): void {
    this.loadEmpDetails();
  }

  loadEmpDetails(): void {
    this.es.loadEmpData().subscribe({
      next: (data: Employee[]) => {
        this.employees = data;
        this.filteredEmployees = data;
      },
      error: (error: any) => console.log(error),
      complete: () => console.log('Completed')
    });
  }

  storeEmp(): void {
    const employee = this.empRef.value;
    this.es.storeEmpData(employee).subscribe({
      next: (data: any) => console.log(data),
      error: (error: any) => console.log(error),
      complete: () => {
        this.loadEmpDetails();
        this.empRef.reset();
      }
    });
  }

  viewDetails(employee: Employee): void {
    sessionStorage.setItem('empInfo', JSON.stringify(employee));
    this.router.navigate(['employee-operation']);
  }

  searchEmployees(event: any): void {
    const searchTerm = event.target.value;

    if (searchTerm) {
      this.filteredEmployees = this.employees.filter((employee: Employee) =>
        employee.id.toString().includes(searchTerm)
      );
    } else {
      this.filteredEmployees = this.employees;
    }
  }
}
```

Employee-operation.component.css

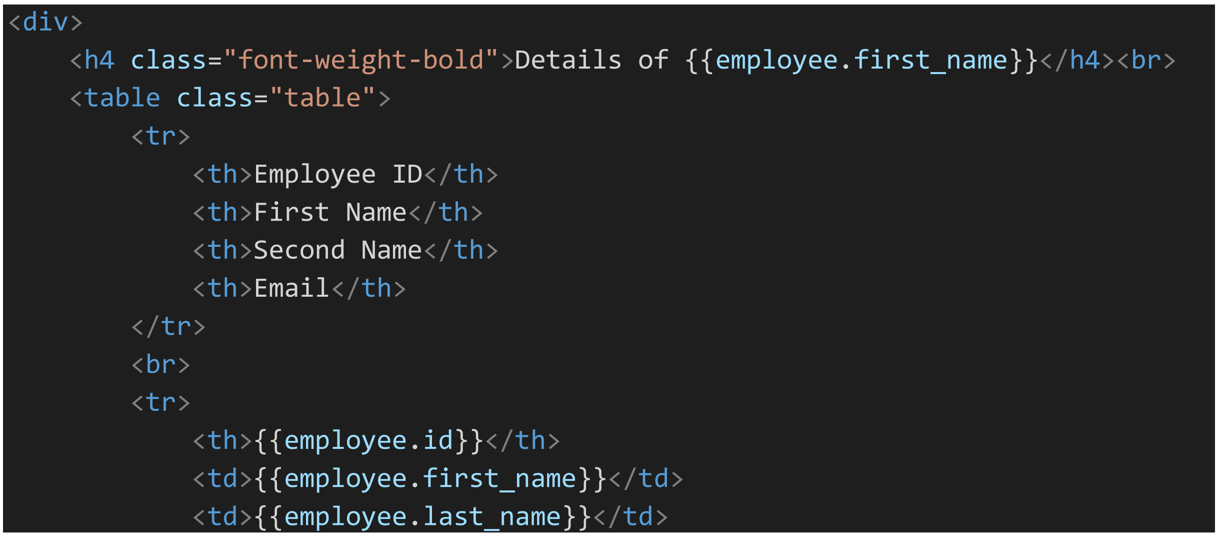


```css
.container {
    font-family: Arial, sans-serif;
    max-width: 800px;
    margin: 0 auto;
    padding: 20px;
  }

  h4 {
    font-size: 24px;
    font-weight: bold;
    margin-bottom: 20px;
  }

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

  th,
  td {
    padding: 10px;
    text-align: left;
    border-bottom: 1px solid #ccc;
  }

  th {
    background-color: #f8f9fa;
    font-weight: bold;
  }

  .btn-primary,
  .btn-secondary,
  .btn-danger,
  .btn-link {
    padding: 8px 16px;
    border-radius: 4px;
    cursor: pointer;
  }

  .btn-primary {
    background-color: #007bff;
    color: white;
    border: none;
  }

  .btn-primary:hover {
    background-color: #0056b3;
```



```
  }

  .btn-secondary {
    background-color: #6c757d;
    color: white;
    border: none;
  }

  .btn-secondary:hover {
    background-color: #545b62;
  }

  .btn-danger {
    background-color: #dc3545;
    color: white;
    border: none;
  }

  .btn-danger:hover {
    background-color: #b02a37;
  }

  .btn-link {
    color: #007bff;
    text-decoration: none;
    border: none;
    background: none;
  }

  .btn-link:hover {
    text-decoration: underline;
  }
```

Employee-operation.component.html



```
<div>
    <h4 class="font-weight-bold">Details of {{employee.first_name}}</h4><br>
    <table class="table">
        <tr>
            <th>Employee ID</th>
            <th>First Name</th>
            <th>Second Name</th>
            <th>Email</th>
        </tr>
        <br>
        <tr>
            <th>{{employee.id}}</th>
            <td>{{employee.first_name}}</td>
            <td>{{employee.last_name}}</td>
            <td>{{employee.email}}</td>
        </tr>
    </table>
    <hr><br>
    <input type="button" class="btn btn-primary" value="View All Employee" (click)="ViewAll()"/>&nbsp;
    <input type="button" class="btn btn-secondary" value="Update Employee" (click)="updateEmployee(employee)"/>&nbsp;
    <input type="button" class="btn btn-danger" value="Delete Employee" (click)="deleteEmployee(employee.id)"/>&nbsp;<br>
    <input type="button" class="btn btn-link" value="Back To Home" (click)="ViewAll()"/>&nbsp;
</div>
```

Employee-operation.component.ts

```
import { Component, OnInit } from '@angular/core';
import { Router } from '@angular/router';
import { EmployeeService } from '../employee.service';

@Component({
  selector: 'app-employee-operation',
  templateUrl: './employee-operation.component.html',
  styleUrls: ['./employee-operation.component.css']
})
export class EmployeeOperationComponent implements OnInit {
  employee:any;
  constructor(public router:Router,public es:EmployeeService) { }

  ngOnInit(): void {
    let obj = sessionStorage.getItem("empInfo");
    if(obj != null){
        this.employee= JSON.parse(obj);
    }
  }

  ViewAll(){
    this.router.navigate(["employees"])
  }

  deleteEmployee(id:any){
    this.es.delete(id).subscribe({
      next:(result:any)=>console.log(result),
      error:(error:any)=>console.log(error),
      complete:()=>this.ViewAll()

    })

  }
  updateEmployee(employee:any){
    sessionStorage.setItem("empUpd",JSON.stringify(employee));
    this.router.navigate(["employee-update"]);
  }


}
```

Login.component.css

```
.container {
    display: flex;
    justify-content: center;
    align-items: center;
    height: 100vh;
}

.card {
    background-color: #f9f9f9;
    box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);
    padding: 40px;
    width: 350px;
    border-radius: 4px;
}

h1 {
    text-align: center;
    color: #333;
    margin-bottom: 30px;
}

.form-group {
    margin-bottom: 20px;
}

label {
    font-weight: bold;
}

.input-group {
    display: flex;
}

.input-group-addon {
    background-color: #e9ecef;
    border-right: none;
    border-top-left-radius: 4px;
    border-bottom-left-radius: 4px;
    padding: 10px;
}

.form-control {
    border-color: #e9ecef;
    border-left: none;
    border-top-right-radius: 4px;
    border-bottom-right-radius: 4px;
    padding: 10px;
}

.btn-primary {
    background-color: #007bff;
    border: none;
    width: 100%;
}

.btn-primary:hover {
    background-color: #0069d9;
}

.message {
    text-align: center;
    margin-top: 20px;
    color: #333;
}

.message a {
    color: #007bff;
    text-decoration: none;
}
```

Login.component.html

```
<div class="container">
    <div class="card">
        <h1>Login</h1>
        <form [formGroup]="loginRef" (ngSubmit)="checkUser()">
            <div class="form-group">
                <label for="email">Email address</label>
                <input formControlName="email" type="email" class="form-
control" id="email" placeholder="Enter email" required>
            </div>
            <div class="form-group">
                <label for="password">Password</label>
                <input formControlName="pass" type="password" class="form-
control" id="password" placeholder="Enter password" required>
            </div>
            <button type="submit" class="btn btn-primary">Login</button>
        </form>
        <p class="message">Don't have an account? <a href="#">Sign Up</a></p>
    </div>
</div>
```

Login.component.ts

```
import { Component, OnInit } from '@angular/core';
import { FormControl, FormGroup } from '@angular/forms';
import { Router } from '@angular/router';


@Component({
  selector: 'app-login',
  templateUrl: './login.component.html',
  styleUrls: ['./login.component.css']
})
export class LoginComponent implements OnInit {

  msg: string = "";
  loginRef = new FormGroup({
    email: new FormControl(),
    pass: new FormControl()
  });

  constructor(public router: Router) { }

  ngOnInit(): void {
  }

  checkUser() {
    let login = this.loginRef.value;
    if (login.email == "admin@gmail.com" && login.pass == "1234") {
      this.router.navigate(["employees"]);
    } else {
      this.msg = "Failure! Try Again!";
    }
  }

}
```

Updatemployee.html

```
<div>
    <h4 class="font-weight-bold">Update Details of
{{employee.first_name}}</h4><br>
    <form [formGroup]="empuRef" (ngSubmit)="updateEmpDetails()">
        Employee ID
        <input type="number" [(ngModel)]=employee.id formControlName="id"
readonly>
        First Name
        <input type="text" [(ngModel)]=employee.first_name
formControlName="first_name">
        Last Name
        <input type="text" [(ngModel)]=employee.last_name
formControlName="last_name">
        Email
        <input type="text" [(ngModel)]=employee.email
formControlName="email"><br><br>
        <input type="submit" class="btn btn-success" value="UPDATE EMPLOYEE" >
    </form>
    <input type="button" class="btn btn-link" value="Back To Home"
(click)="ViewAll()"/>&nbsp;
</div>
```

Updatemployee.ts

```
import { Component, OnInit } from '@angular/core';
import { FormGroup,FormControl } from "@angular/forms";
import { Router } from '@angular/router';
import { EmployeeService } from '../employee.service';

@Component({
  selector: 'app-update-employee',
  templateUrl: './update-employee.component.html',
  styleUrls: ['./update-employee.component.css']
})
export class UpdateEmployeeComponent implements OnInit {

  constructor(public es:EmployeeService,public router:Router) { }
  employee:any;

  empuRef = new FormGroup({
    id: new FormControl(),
    first_name: new FormControl(),
    last_name: new FormControl(),
    email: new FormControl()
  })

  ngOnInit(): void {
    let obj = sessionStorage.getItem("empUpd");
    if(obj != null){
        this.employee= JSON.parse(obj);
    }
  }

  ViewAll(){
    this.router.navigate(["employees"])
  }

  updateEmpDetails(){
    let employee = this.empuRef.value;
    let id = this.empuRef.value.id;
    this.es.updateEmp(id,employee).subscribe({
      next:(data:any)=>this.employee=data,
      error:(error:any)=>console.log(error),
      complete:()=>this.ViewAll()
    })
  }

}
```

App-routing.module.ts

```
import { NgModule } from '@angular/core';
import { RouterModule, Routes } from '@angular/router';
import { EmployeeOperationComponent } from './employee-operation/employee-operation.component';
import { EmployeeComponent } from './employee/employee.component';
import { LoginComponent } from './login/login.component';
import { UpdateEmployeeComponent } from './update-employee/update-employee.component';

const routes: Routes = [
  { path: '', component: LoginComponent },
  { path: 'employees', component: EmployeeComponent },
  { path: 'employee-operation', component: EmployeeOperationComponent },
  { path: 'employee-update', component: UpdateEmployeeComponent },
];

@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule],
})
export class AppRoutingModule {}
```

app.component.css

```
.container {
    position: absolute;
    text-align: center;
    background-image: linear-gradient(rgb(94, 63, 24), white);
    background-repeat: no-repeat;
    background-repeat: no-repeat;
    background-size: cover;
    min-width: 100%;
    min-height: 100%;
    font-family: system-ui, -apple-system, BlinkMacSystemFont, 'Segoe UI', Roboto, Oxygen, Ubuntu, Cantarell, 'Open Sans', 'Helvetica Neue', sans-serifs;
  }
```

App.component.ts

```
import { Component } from '@angular/core';

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {
  title = 'EVENT-MANAGEMENT';
}
```

App.module.ts

```
import { NgModule } from '@angular/core';
import { BrowserModule } from '@angular/platform-browser';
import { HttpClientModule } from '@angular/common/http';
import { AppRoutingModule } from './app-routing.module';
import { AppComponent } from './app.component';
import { LoginComponent } from './login/login.component';
import { ReactiveFormsModule } from '@angular/forms';
import { FormsModule } from '@angular/forms';
import { EmployeeComponent } from './employee/employee.component';
import { UpdateEmployeeComponent } from './update-employee/update-employee.component';
import { EmployeeOperationComponent } from './employee-operation/employee-operation.component';
import { DashboardComponent } from './dashboard/dashboard.component';

@NgModule({
  declarations: [
    AppComponent,
    LoginComponent,
    EmployeeComponent,
    UpdateEmployeeComponent,
    EmployeeOperationComponent,
    DashboardComponent
  ],
  imports: [
    BrowserModule,
    AppRoutingModule,
    ReactiveFormsModule,
    HttpClientModule,
    FormsModule,
  ],
  providers: [],
  bootstrap: [AppComponent],
})
export class AppModule {}
```