

Northwind Manager

Create an app for browsing the Northwind database, whereby the individual pages should be accessible via routing.

The following routes should exist (on the client):

- **customers**: all customers (CompanyName, ContactName, City, Country)
- **employees**: all employees (Name, City, Country)
- **orders/empOrCust_id**: specific Orders (OrderDate, RequiredDate, ShippedDate, Nr OrderDetails), e.g. orders/employee_1 or orders/customer_ALFKI
- **orders/details/id**: all OrderDetails of a specific Order (Quantity, UnitPrice, ProductName)

Backend

The backend is attached as a zip to this exercise, including all DTOs of the backend.

- The various REST services from the subtasks below and must be programmed in NorthwindController.cs.
- The reading of the data must be programmed in the service class NorthwindService.cs. So the controller shall not access the DbContext directly.

Frontend

Material Design

Style the app with Material Design. To do so add it to the project with **ng add @angular/material**.

The components are well described on the homepage <https://material.angular.io/components/categories>.

A small problem is that you must import every element you want to use from Angular Material explicitly.

Create a module that imports and exports all Material Design modules. This module then shall be used as a shared module.

Use the components for button and table in the pages below.

Main Page

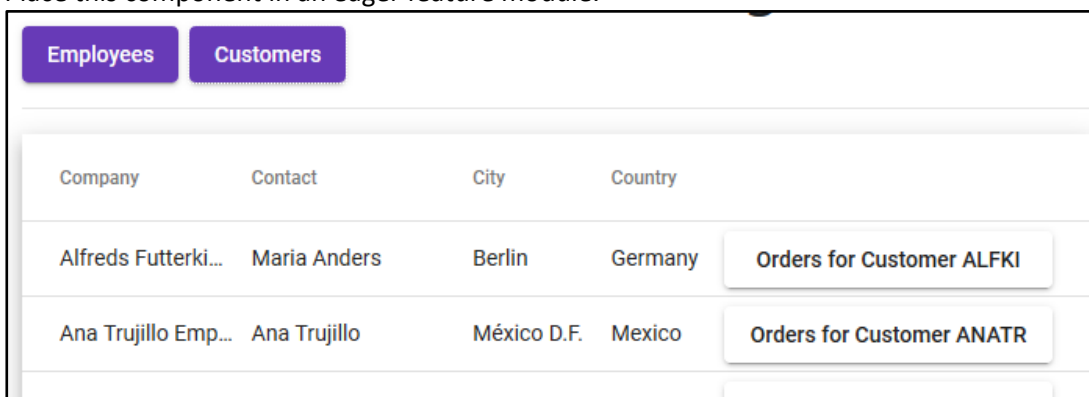
On the main page, besides the <router-outlet> only two buttons for the navigation to Customers and Employees must be noted.



Customers

The most important customer data should be displayed here as a simple table. With a button you can navigate to the orders page for the corresponding customer.

Place this component in an eager feature module.



Pipe

Create a pipe **ellipsisNotation** in a shared module that allows you to abbreviate long strings above a certain length with "...". Use this pipe for the Companyname:

```
<td mat-cell
  *matCellDef="let element"> {{element.companyName | ellipsisNotation:16}} </td>
```

Employees

Same as the Employees page, but for employees data. You can place this in the same eager feature module as the customers page or in a new eager feature module. Use **navigateByUrl** to navigate.

Name	City	Country	
Buchanan Steven	London	UK	Orders for Employee 5
Callahan Laura	Seattle	USA	Orders for Employee 8

Orders

The orders for the customer or employee are displayed here. You can use the button to navigate to the corresponding order details. Place this page in a lazy feature module

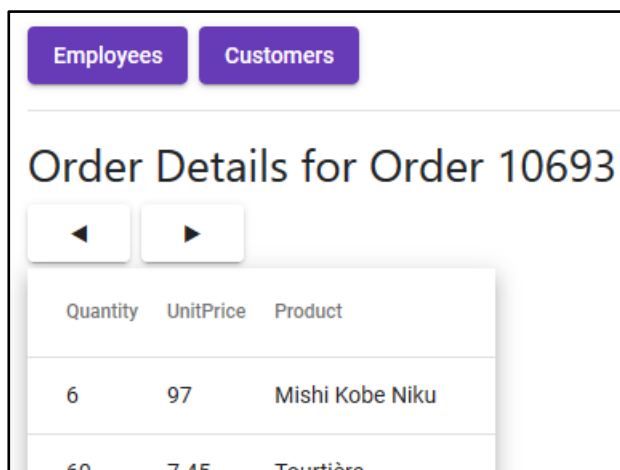
Ordered	Required	Shipped	Nr Details	
1997-10-03	1997-10-31	1997-10-13	1	OrderDetails for Order 10692
1997-10-13	1997-11-24	1997-10-21	2	OrderDetails for Order 10702
1998-03-16	1998-04-27	1998-03-24	2	OrderDetails for Order 10952

The values for the date should be displayed as shown by means of the **date** pipe. To do this, convert the date for the transmission into a value "milliseconds since 1970" (extension method).

```
2 references | 0 exceptions
public DateTime? OrderDate { private get; set; }
0 references | 0 exceptions
public long OrderDateMillis => OrderDate?.ToJavaScriptMilliseconds() ?? 0;
```

Order Details

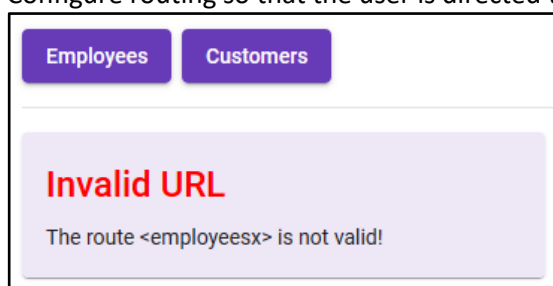
All details of an order are displayed here. Place it in the same module as the orders page.



With the two buttons you can navigate to the order details of the previous or next order, respectively.

Invalid page

Configure routing so that the user is directed to an error page indicating the invalid route.



Sorting

Try to use sorting for one or the other table.

Quantity	UnitPrice	↑Product
6	10	Sir Rodneys Scones
3	18	Chai
10	31	Ikura

Quantity	UnitPrice	Product ↑
3	18	Chai
10	31	Ikura
6	10	Sir Rodneys Scones

See <https://material.angular.io/components/table/overview#sorting> for details.