**LAZY LOADING A MODULE**

1. Let’s create a module and lazy load it. It will have:
   1. HomeComponent - The main home page. Path: /
   2. CatalogComponent - Product catalog. Path: /catalog
   3. A feature module called AccountModule that will have these components:
      1. AccountHomeComponent - The main account management page. Path: /account
      2. OrderHistoryComponent - Shows order history. Path: /account/orders
      3. AddressbookComponent - User's shipping addresses. Path: /account/address
   4. We will lazy-load the AccountModule, only when user navigates to one of its components.
2. First, you might want to create a new Angular application.
3. Inside, create the *main/root module* components listed above.
4. Create a module called account, *with routing.*

**ng g module account --routing**

1. Create the three module components. Hint:

**ng g component account/account-home**

1. Make sure the three module components got added, in account.module.ts
2. Set up routing for the components in AccountModule, in src/app/account/account-routing.module.ts. Inside, import the component class names. Hint:

**import { AccountHomeComponent } from './account-home/account- home.component';**

1. Note that feature modules use forChild. Add components to the module routing’s routes table, with paths relative to the feature module. In a later step, we will map AccountModule to /account.

**{path: "", component: AccountHomeComponent},**

**{path: "orders", component: OrderHistoryComponent},**

1. Inside the main routing, src/app/app-routing.module.ts:
   1. Import the class names for the components that belong directly to the main module.

**import { HomeComponent } from './home/home.component';**

**import { CatalogComponent } from './catalog/catalog.component';**

* 1. In the routes table add the two components like this.

const routes: Routes = [

**{path: "", component: HomeComponent},**

**{path: "catalog", component: CatalogComponent},**

];

* 1. Add a route for the feature module AccountModule.

const routes: Routes = [

{path: "", component: HomeComponent},

{path: "catalog", component: CatalogComponent},

**{**

**path: "account",**

**loadChildren: () => import('./account/account.module').then( m => m.AccountModule)**

**}**

];

1. Add all links in src/app/app.component.html.

**<a href [routerLink]="['/']">Home</a>&nbsp;**

**<a href [routerLink]="['/catalog']">Shop</a>&nbsp;**

**<a href [routerLink]="['/account']">My Account</a>&nbsp;**

**<a href [routerLink]="['/account/orders']">My Orders</a>&nbsp;**

**<a href [routerLink]="['/account/address']">Address Book</a>**

**<router-outlet></router-outlet>**

1. Create a production version to see what happens with code-splitting!

ng build