Angular Basics

by Martin Dobrev

Lecture 3:
Routing & Navigation

Lecture Overview

- Routing and navigation - why is it important
- Angular Router Module Basics
- Events and state
- Demo

Routing & Navigation

- Angular Routing resembles the browser navigation model
- Entering a URL navigates to the corresponding component
- Supports optional parameters
- Saves the history of the navigation
  (BACK and FORWARD buttons)

Angular Routing

- Routing is available through a separate module (library)

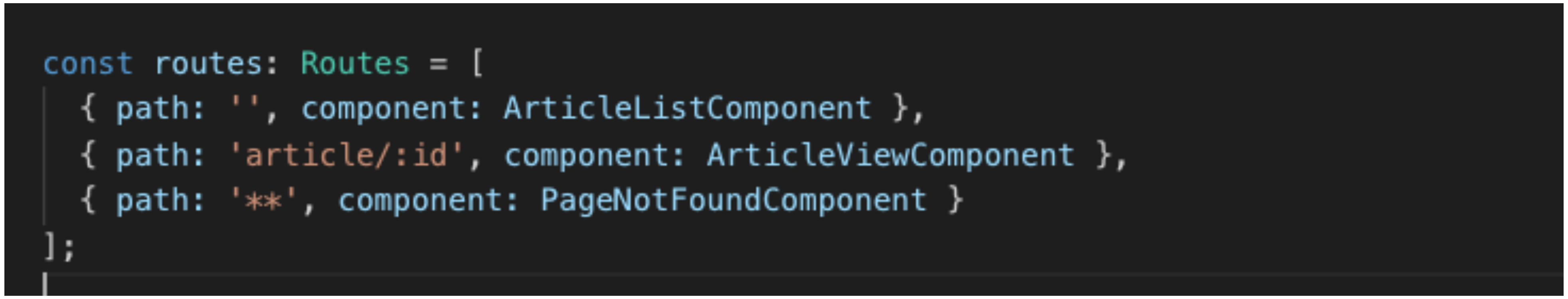
```
import { RouterModule, … } from '@angular/router';
```

- Not a part of Angular core

- Multiple services and directives

Routes definition

- A collection of **Route** objects, defined in a type **Routes**

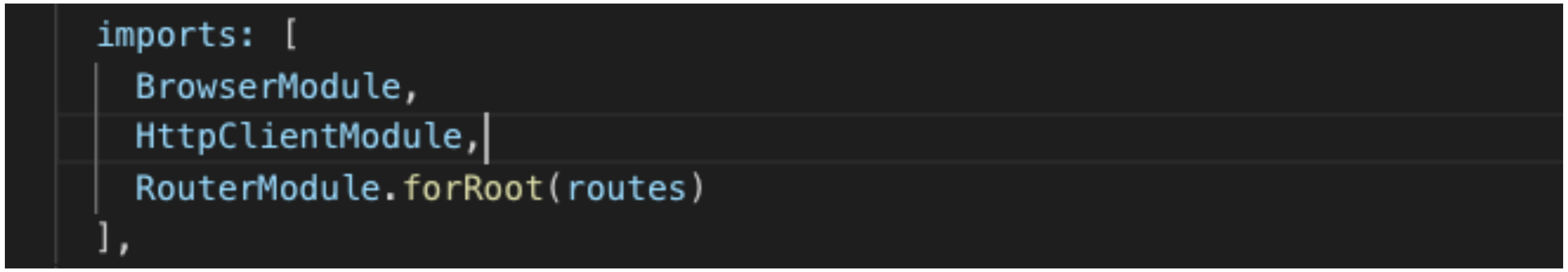


```
const routes: Routes = [
  { path: '', component: ArticleListComponent },
  { path: 'article/:id', component: ArticleViewComponent },
  { path: '**', component: PageNotFoundComponent }
];
```

- Multiple parameters enable advanced functions like redirect, data injection, children definition, etc.

RouterModule

- Add **RouterModule** to the imports



```
imports: [
  BrowserModule,
  HttpClientModule,
  RouterModule.forRoot(routes)
],
```

- Use **forRoot** method to create the module with all the router providers and directives

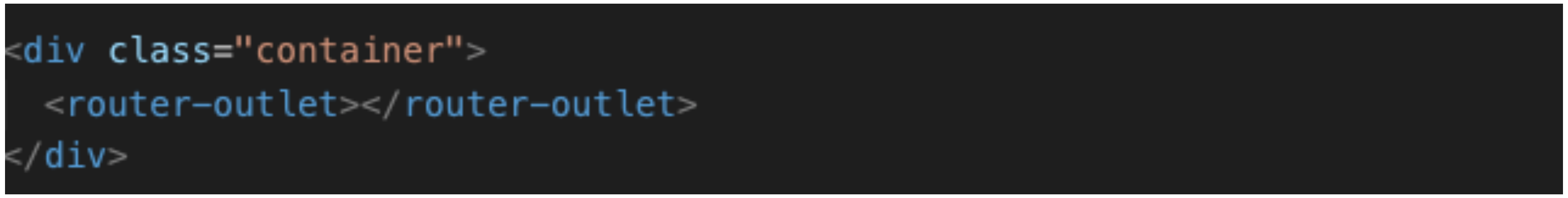
- Pass the defined routes (optional parameters available)

- **forRoot** also creates the Router service

- There is **forChild** method for child modules (future lecture - advanced routing)

Basic Routing Directives

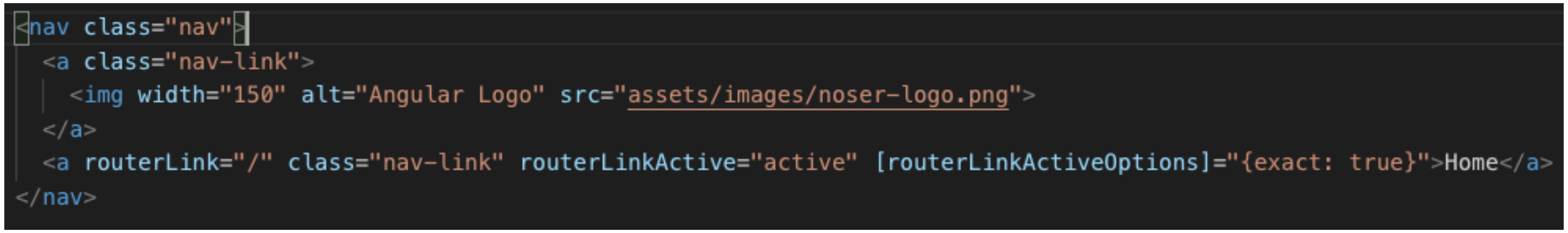
- **RouterOutlet** - placeholder where the router shall display the components



```html
<div class="container">
  <router-outlet></router-outlet>
</div>
```

- **RouterLink** - binds to the on-click element

- **RouterLinkActive** - adds the specified css classes if the current route is active



```html
<nav class="nav">
  <a class="nav-link">
    <img width="150" alt="Angular Logo" src="assets/images/noser-logo.png">
  </a>
  <a routerLink="/" class="nav-link" routerLinkActive="active" [routerLinkActiveOptions]="{exact: true}">Home</a>
</nav>
```

Demo:
Adding Routing to our project

Routing Services

- **Router** - placeholder where the router shall display the components
- **ActivatedRoute** - gives access to the currently active route, you can extract data, parameters etc.

Router Service Events

- **Multiple Navigation Events available**
  - NavigationStart
  - RouteConfigLoadStart
  - …
  - ActivationStart
  - ResolveStart
  - …

See API Documentation for detailed information

Router Events

- The Router service

- Environment is just a TypeScript object that can be used by your application

- Replaced at build time if configured so in your angular.json config

- Demo…

Demo:
Routing Services,
Listening for Events

Homework

- Add available webpages to the main menu (about, contact, etc.)

- See in the REST API
  /api/v1/pages

- All public items can be accessed by GET
  /api/v1/menu

- Separate pages can be loaded by GET
  /api/v1/pages/$page_id

THE END

Danke für die Aufmerksamkeit :)

Resources

- Angular Routing Documentation: https://angular.io/guide/router

- Tutorial Github Repo: https://github.com/martindobrev/nb-angular-tutorial

- Resources will be available on the webpage of the tutorial https://martindobrev.github.io/nb-angular-tutorial/