

Routing

№ урока: 5 Курс: Angular 2 Essential

Средства обучения: Редактор кода
Node.js и npm

Обзор, цель и назначение урока

В этом уроке будут рассмотрены принципы настройки маршрутизации Angular 2 приложения. Вы узнаете, как передавать и получать параметры в компонентах, настраивать специальные Guard объекты для выполнения проверок перед созданием компонента.

Содержание урока

1. Основы маршрутизации
2. Работ с дочерними маршрутами
3. Передача параметров использование опциональных параметров
4. Использование Guard объектов
5. Использование интерфейса Resolve для создания поставщиков данных

Резюме

Шаг 1.

Установить элемент `<base href="/" />` в index.html документ как первый дочерний элемент в элементе head. Данный элемент необходим для того, чтобы установить базовый адрес при использовании относительного пути.

<https://developer.mozilla.org/en/docs/Web/HTML/Element/base>

Шаг 2.

Импорт модуля RouterModule отвечающего за маршрутизацию

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

Шаг 3.

Настройка модуля маршрутизации.

```
@NgModule({
  imports: [
    BrowserModule,
    FormsModule,
    RouterModule.forRoot([
      { path: 'hero/:id', component: HeroDetailComponent },
      { path: 'crisis-center', component: CrisisListComponent },
      {
        path: 'heroes',
        component: HeroListComponent,
        data: {
          title: 'Heroes List'
        }
      },
      { path: '', component: HomeComponent },
      { path: '**', component: PageNotFoundComponent }
    ])
  ],
})
```

Шаг 4.

Определение места на странице, куда будет помещаться представление активного, в данный момент, компонента.

```
<router-outlet></router-outlet>
```

Шаг 5.

Установка ссылок для маршрутизации.

```
<a routerLink="/route-name " routerLinkActive="active">Link Name</a>
```

routerLink – директива для определение адреса, на который будет перенаправлен пользователь.

routerLinkActive – директива для установки класса для той ссылки, которая в данный момент активна.

Передача параметров

Для передачи параметров в маршруте необходимо указать требуемый параметр

```
{  
  path: "phrase/:id",  
  component: PhraseDetailsComponent  
}
```

При таких настройках, параметр с именем id будет доступен в компоненте PhraseDetailsComponent.

Для того, чтобы выполнить переход из одного компонента в другой компонент, необходимо выполнить внедрение Router объекта. С помощью метода navigate можно выполнить перенаправление пользователя на нужный адрес this.router.navigate(["phrase", 10]);

Для получения значений переданных в адресе, необходимо использовать следующий код

```
this.activatedRoute.params.forEach((params: Params) => {  
  let id = +params["id"];  
});
```

Закрепление материала

- Какой модулю нужно импортировать для настройки маршрутизации?
- Какие директивы используются для настройки ссылок для перенаправления пользователя?
- Что такое дочерний маршрут?
- Как передать параметр в компонент и какими способами можно прочесть значения параметра в компоненте?
- Для чего нужны Guard объекты, приведите примеры использования подобных объектов.
- Для чего нужны Resolve объекты, как их использовать.

Самостоятельная деятельность учащегося

Задание 1

Создайте приложение, в котором будет находиться три функциональные области:

Home – отображение приветственного сообщения

Products – работа со список продуктов

Admin – управления продуктами (в данной версии приложения вывести сообщение, о том, что это главная страница админ части приложения)

В app.component создайте меню из трех пунктов и настройте маршрутизацию таки образом, чтобы переход по ссылке направлял пользователя на соответствующий компонент, относящийся к его функциональной области.

Задание 2

В приложении, созданном в первом задании, реализуйте шаблон Master/Details. Используйте сервис из предыдущего урока, для того чтобы получить доступ к списку продуктов (по необходимости модифицируйте сервис). В главном компоненте отображайте только названия продуктов, при клике по продукту, перенаправляйте пользователя на компонент, который будет показывать детали выбранного продукта.

Задание 3

Проведите рефакторинг кода задания 2 и примените дочерние маршруты для отображения деталей выбранного продукта. Детали отображайте под списком всех продуктов в одном представлении.

Рекомендуемые ресурсы

Основы маршрутизации

<https://angular.io/docs/ts/latest/guide/router.html>

Angular 2 Authentication Tutorial

<https://auth0.com/blog/angular-2-authentication/>

Интерфейс Resolve

<https://angular.io/docs/ts/latest/api/router/index/Resolve-interface.html>

Resolving routing data

<http://blog.thoughttram.io/angular/2016/10/10/resolving-route-data-in-angular-2.html>