

# Desenvolvimento Web

Trilha JavaScript com Angular e Node

Instrutor: Júlio Pereira Machado ([julio.machado@pucrs.br](mailto:julio.machado@pucrs.br))



# Angular e Roteamento



# Roteamento

- Angular fornece um roteador para a navegação entre *views*
  - Objeto *Router* é configurado com vários objetos *Route*
  - Define o mapeamento de URLs para os correspondentes componentes responsáveis pelas *views*
- Configuração:
  - Importar *RouterModule* disponível no módulo JavaScript *@angular/router*
  - Configurar dependência a *RouterModule* no módulo que conterá a configuração de rotas
- Importante:
  - Rotas são construídas a partir do elemento `<base href="/">` no arquivo `index.html`

# Roteamento

- Criação:
  - Para adicionar um módulo de rotas pré-configurado ao criar um aplicação via Angular CLI
  - `ng new minhaApp --routing`
  - Para adicionar um módulo de rotas após a criação da aplicação
  - `ng generate module rotas --module app --flat`
- Documentação:
  - <https://angular.io/guide/router>

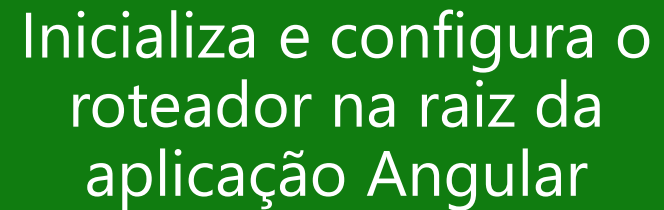
# Roteamento

- Exemplos: módulo para rotas

```
import { NgModule }           from '@angular/core';
import { RouterModule, Routes } from '@angular/router';
...

const routes: Routes = [
  { path: 'heroes', component: HeroesComponent }
];

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



Inicializa e configura o roteador na raiz da aplicação Angular

# Roteamento

- Diferentes tipos de rotas são suportadas
  - Rotas estáticas
  - Rotas estáticas com passagem de dados (somente de leitura) via propriedade *data* da rota
  - Rotas dinâmicas com parâmetros na URL
  - Rotas dinâmicas com parâmetros via query-strings
  - Rotas de redirecionamento
  - Rotas com URLs definidas por “curingas”
- A ordem das rotas é importante
  - Avaliação em ordem
  - Primeira que padrão “coincidir com a rota” é escolhida
  - Definir rotas mais específicas primeiro e rotas mais gerais depois

# Roteamento

- Exemplos: rotas

```
const routes: Routes = [  
  { path: 'dashboard', component: DashboardComponent },  
  { path: 'detail/:id', component: HeroDetailComponent },  
  { path: 'heroes', component: HeroesListComponent, data: { title: 'Heroes' } },  
  { path: '', redirectTo: '/dashboard', pathMatch: 'full' },  
  { path: '**', component: PageNotFoundComponent }  
];
```

# Roteamento

- Serviço **ActivatedRoute** possui informações sobre a rota ativa associada à navegação para a *view* do componente
  - Parâmetros na URL são obtidos via propriedade `paramMap` que retorna um *Observable*
  - Parâmetros via query-string obter via propriedade `queryParams` que retornar um *Observable*
  - Propriedade `snapshot` representa o valor da rota em um determinado ponto no tempo
    - Propriedades *paramMap* e *queryParams* representam um único dicionário ao invés de um fluxo
  - Parâmetros fornecidos em *data* na rota são obtidos via propriedade `data` que retorna um *Observable*



# Roteamento

- Exemplos: rota 'detail/:id'

```
import { ActivatedRoute } from '@angular/router';
...
export class HeroDetailComponent implements OnInit {
  constructor(
    private route: ActivatedRoute,
    private heroService: HeroService,
    private location: Location
  ) {}
  ngOnInit(): void {
    this.route.paramMap.subscribe(...);
  }
}
```

# Roteamento

- Template da *view* que utiliza o roteamento define o local onde o componente da *view* associada à URL irá renderizar o HTML via diretiva **RouterOutlet**

```
<h1>App</h1>  
<router-outlet></router-outlet>
```

# Navegação

- A navegação pode se dar:
  - Por uma ação de click em um link
  - Por uma ação de evento com código associado
  - Por uma URL explícita no navegador
- Diretiva **RouterLink** é usada como atributo em elementos `<a>` para definir a URL da *view* de destino

```
<a routerLink="/heroes">Heroes</a>
```

```
<a [routerLink]="['/hero', hero.id]">  
  <span class="badge">{{ hero.id }}</span>{{ hero.name }}  
</a>
```

# Navegação

- Exemplos: navegação via código

```
import { Router, ActivatedRoute, ParamMap } from '@angular/router';
...
export class HeroDetailComponent implements OnInit {
  constructor(
    private route: ActivatedRoute,
    private heroService: HeroService,
    private router: Router,
  ) {}
  gotoHeroes() {
    this.router.navigate(['/heroes']);
  }
}
```

# Navegação

- Ao efetuar uma navegação entre *views*, o objeto *router* emite diversos eventos através da propriedade **Router.events**