

# N Things You May Not Know About the Router

Deborah Kurata

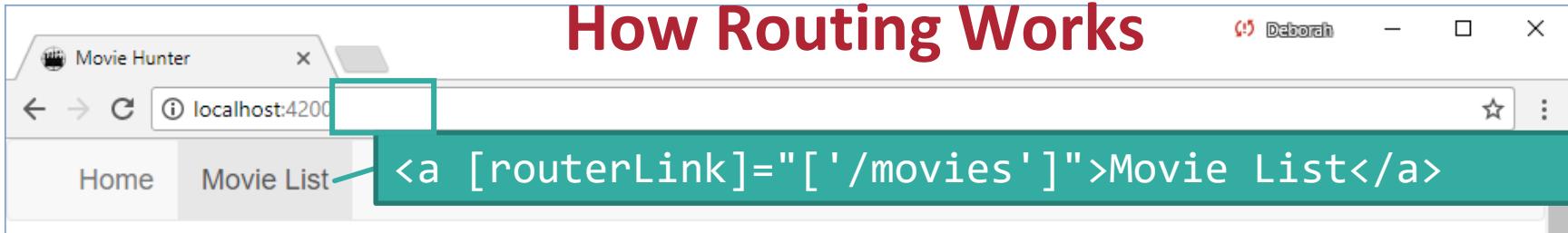
Developer | Author | Consultant | **MVP** | **GDE**  
@deborahkurata



# Deborah Kurata

- Developer
- Angular Documentation Team Member (**angular.io**)
- **Pluralsight** Author
  - Angular Getting Started
  - Angular Reactive Forms
  - Angular Routing
  - Angular Component Communication
- Microsoft Most Valuable Professional (**MVP**)
- Google Developer Expert (**GDE**)
- **@DeborahKurata**

# How Routing Works



```
{ path: 'movies', component: MovieListComponent }
```

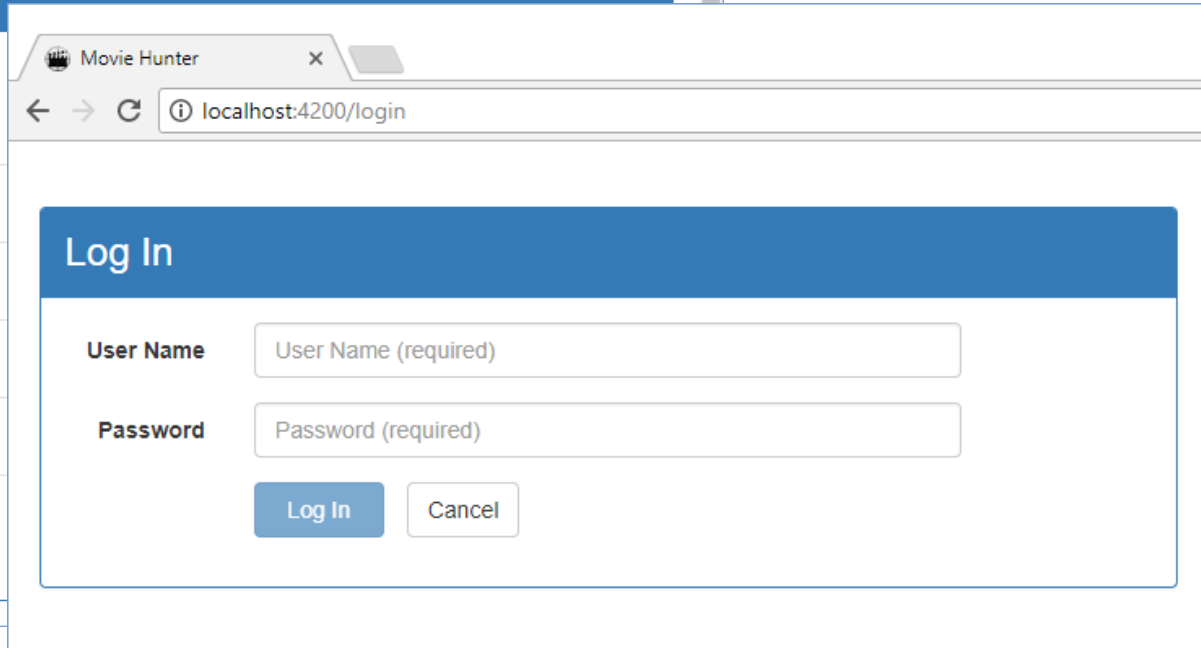
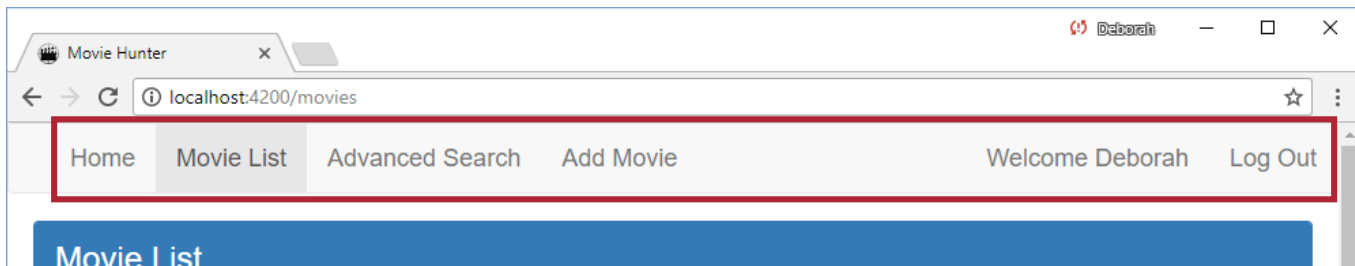
```
<router-outlet></router-outlet>
```

movie-list.component.ts

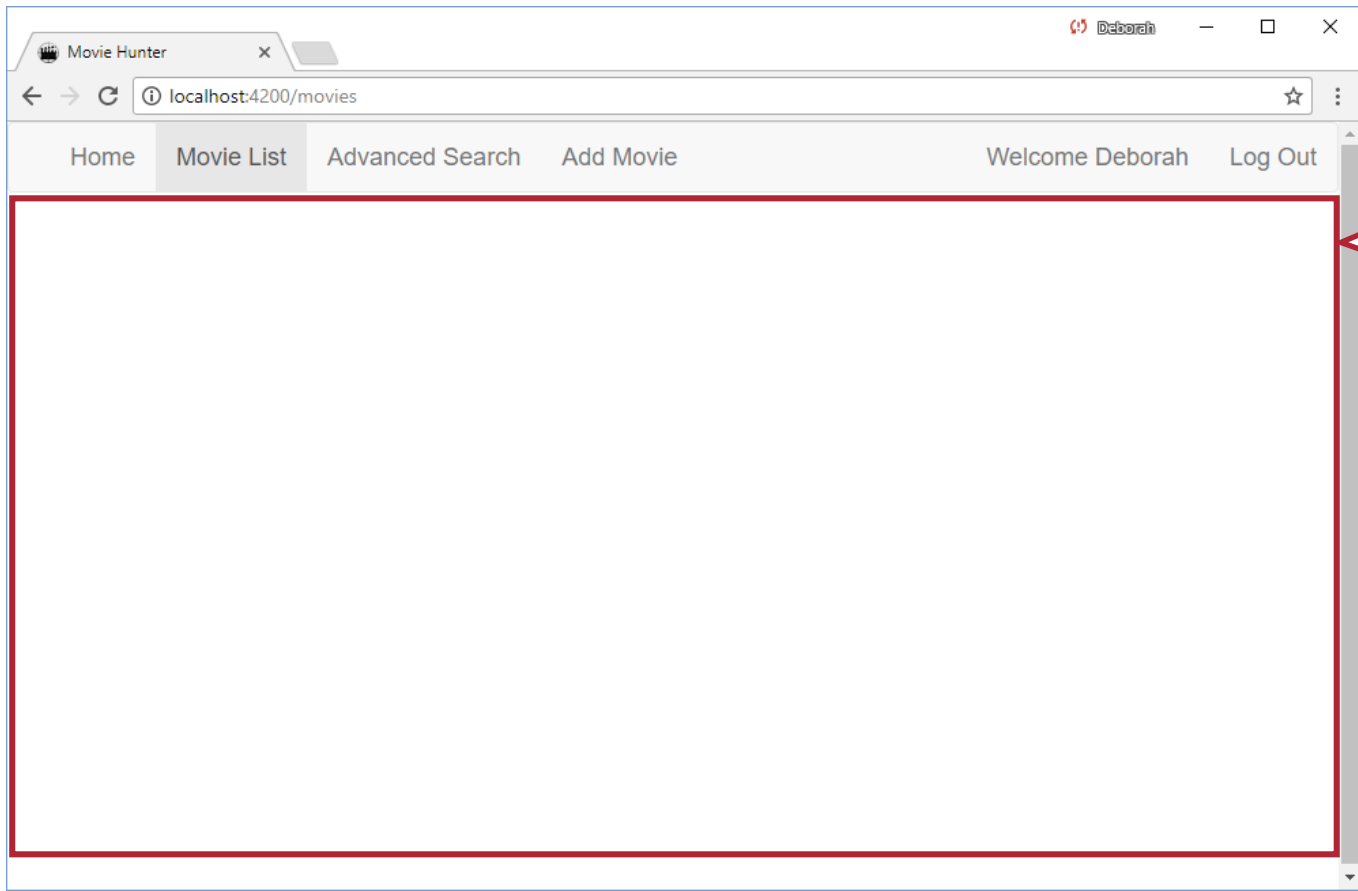
```
import { Component } from '@angular/core';

@Component({
  templateUrl: 'movie-list.component.html'
})
export class MovieListComponent { }
```

**To Menu  
or Not To Menu**



```
<div class="container">
  <nav class="navbar navbar-default">
    <div class="container-fluid">
      <a class="navbar-brand">{{pageTitle}}</a>
      <ul class="nav navbar-nav">
        <li><a [routerLink]="['/welcome']">Home</a></li>
        <li><a [routerLink]="['/movies']">Movie List</a></li>
      </ul>
    </div>
  </nav>
  <div class="container">
    <router-outlet></router-outlet>
  </div>
</div>
```



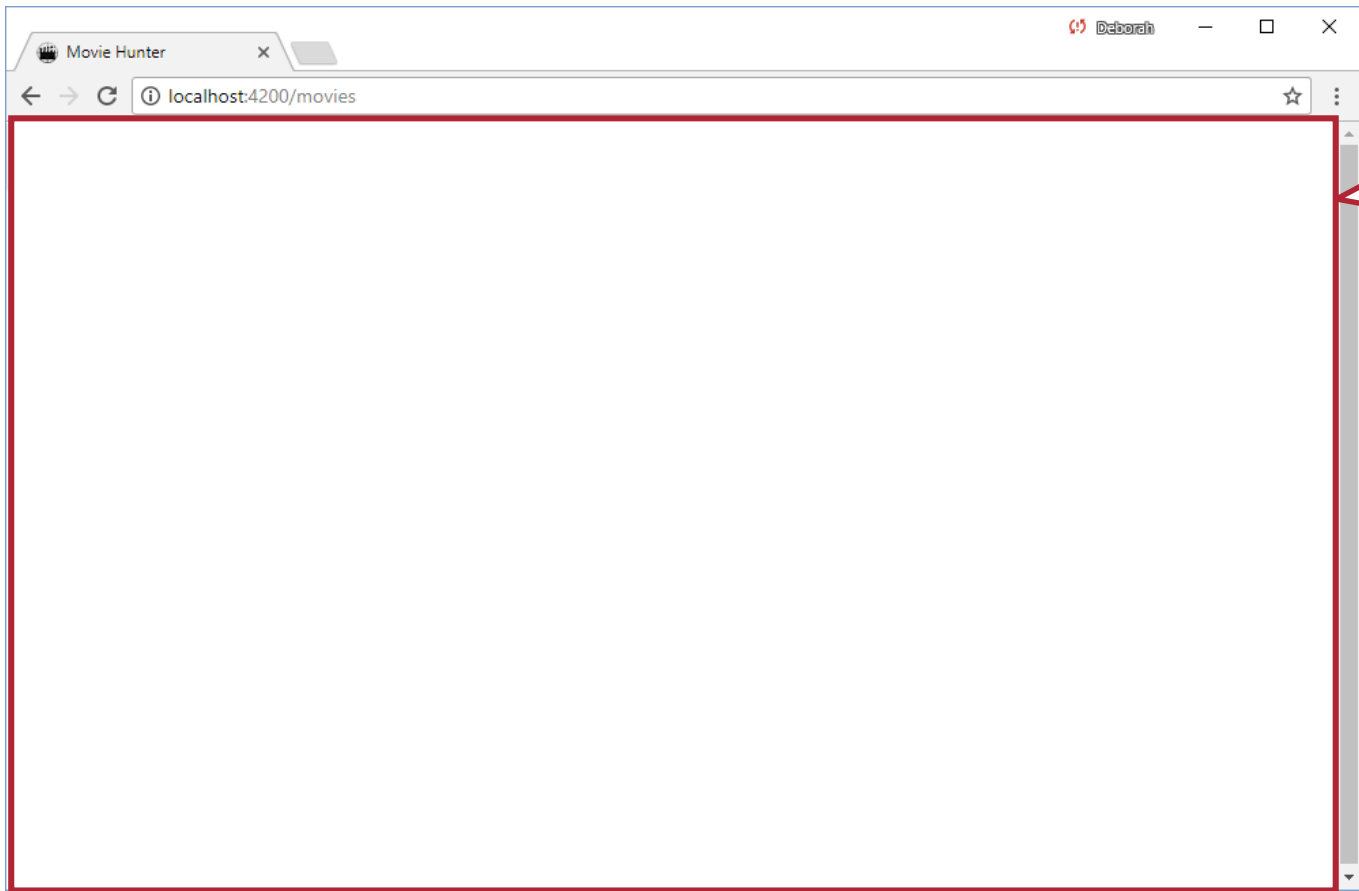
app  
router outlet

```
<div class="container">
  <nav class="navbar navbar-default" *ngIf="displayMenu">
    <div class="container-fluid">
      <a class="navbar-brand">{{pageTitle}}</a>
      <ul class="nav navbar-nav">
        <li><a [routerLink]="['/welcome']">Home</a></li>
        <li><a [routerLink]="['/movies']">Movie List</a></li>
      </ul>
    </div>
  </nav>
  <div class="container">
    <router-outlet></router-outlet>
  </div>
</div>
```



app.component.html

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



app  
router outlet

Movie Hunter

localhost:4200/movies

## Log In

**User Name**

**Password**

app  
router outlet

```
<div class="container">
  <nav class="navbar navbar-default">
    <div class="container-fluid">
      <a class="navbar-brand">{{pageTitle}}</a>
      <ul class="nav navbar-nav">
        <li><a [routerLink]="['/welcome']">Home</a></li>
        <li><a [routerLink]="['/movies']">Movie List</a></li>
      </ul>
    </div>
  </nav>
  <div class="container">
    <router-outlet></router-outlet>
  </div>
</div>
```

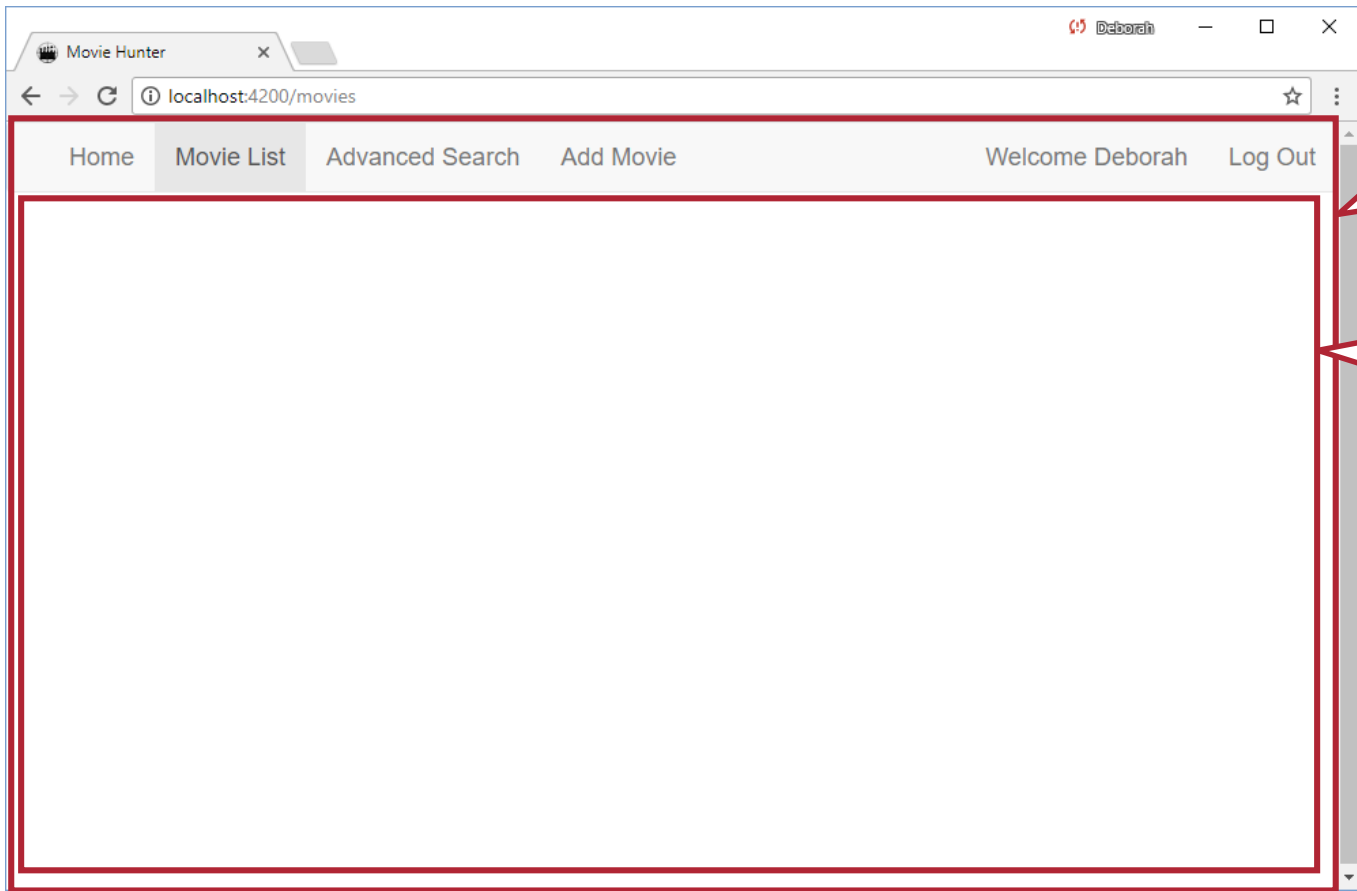
shell.component.html

```
<mh-menu></mh-menu>
```

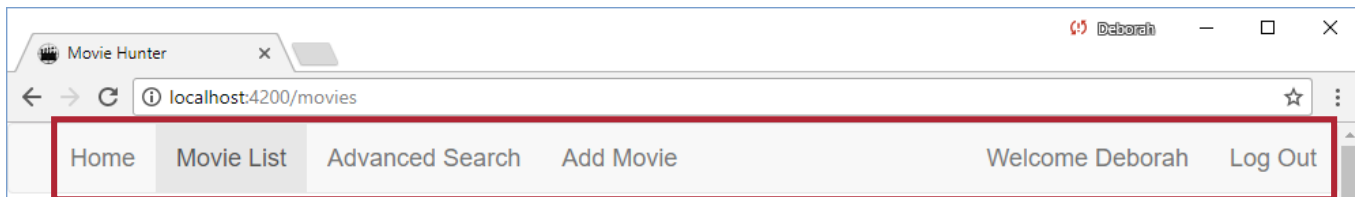
```
<div class='container'>
```

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

```
</div>
```



```
RouterModule.forRoot([
  {
    path: '', component: ShellComponent,
    children: [
      { path: 'welcome', component: WelcomeComponent },
      { path: 'movies', component: MovieListComponent },
      { path: '', redirectTo: 'welcome', pathMatch: 'full' }
    ]
  },
  { path: 'login', component: LoginComponent },
  { path: '**', component: PageNotFoundComponent }
])
```



## Movie List

Filter by:

Show Poster

Title

Director

The Lord of the Rings: The Fellowship of the Ring

Peter Jackson

The Lord of the Rings: The Two Towers

Peter Jackson

The Lord of the Rings: The Return of the King

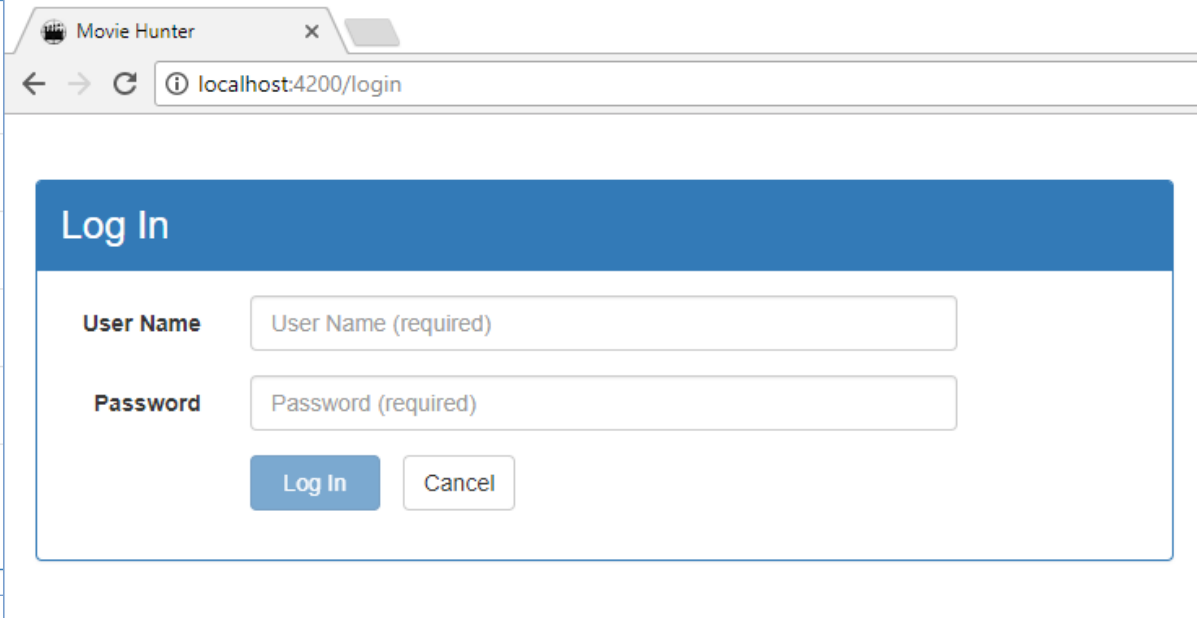
Peter Jackson

The Point

Fred Wolf

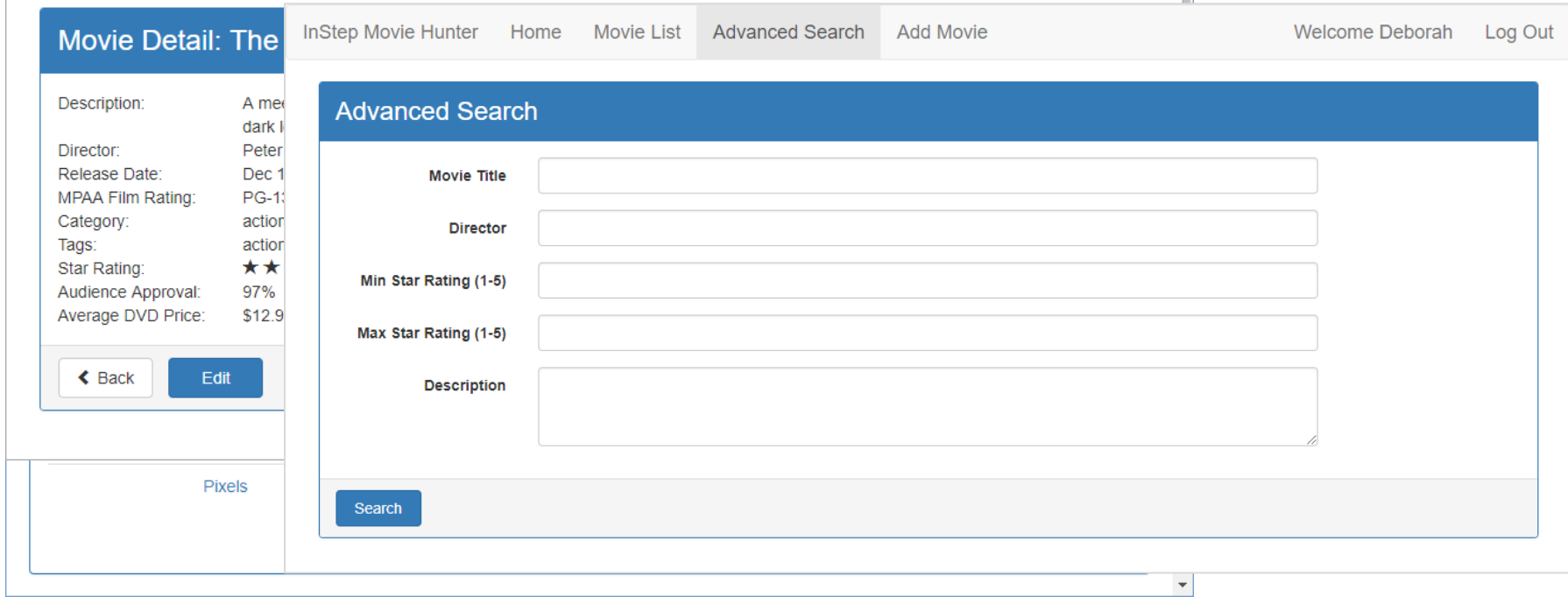
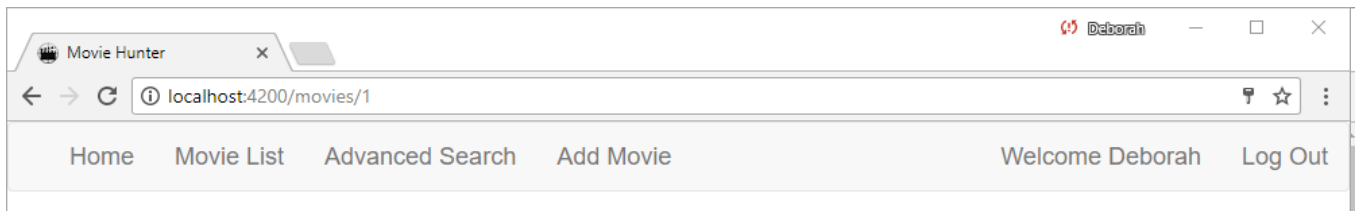
Pixels

Chris Columbus

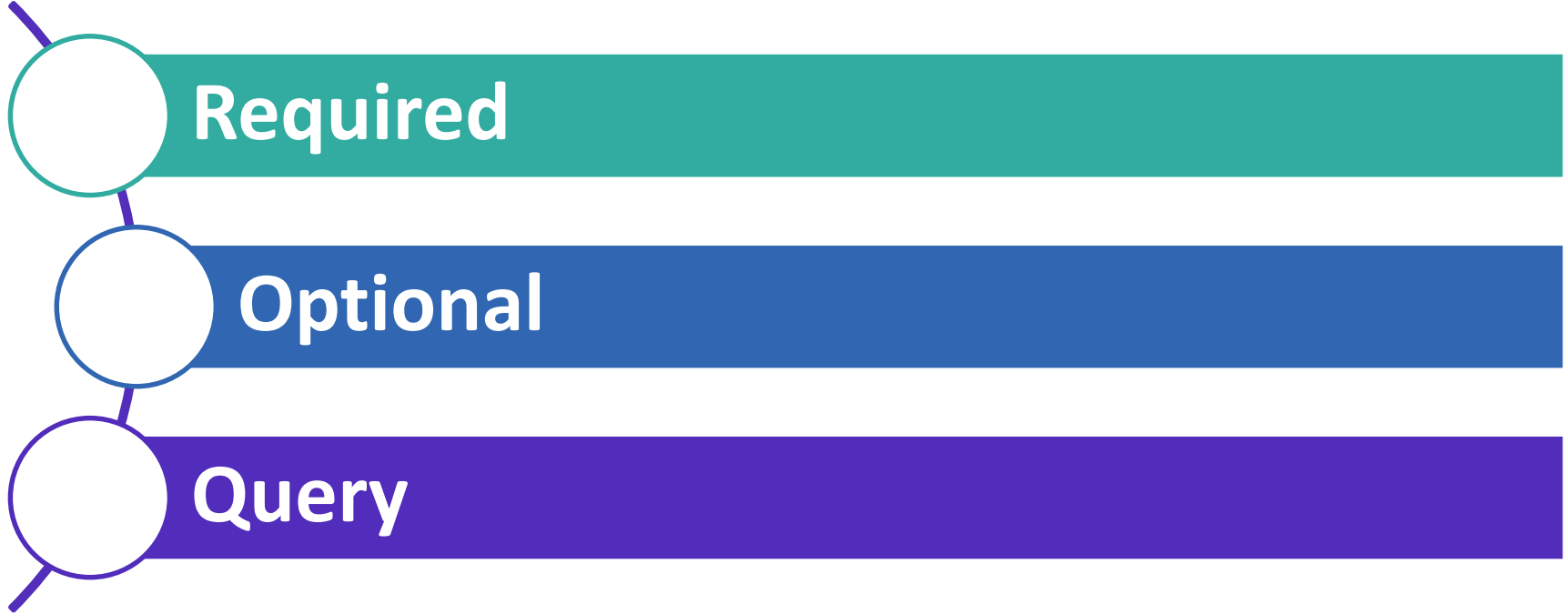




**Data, Data, Data**



# Routing Parameters



# Required Parameters

Configure

```
{ path: 'movies/:id', component: MovieDetailComponent }
```

Activate

```
<a [routerLink]="['/movies', movie.id]">...</a>
```

```
this.router.navigate(['/movies', this.movie.id]);
```

Resulting Url

```
http://localhost:4200/movies/5
```

Read

```
this.route.snapshot.paramMap.get('id');
```

# Optional Parameters

Configure

```
{ path: 'movies', component: MovieListComponent }
```

Activate

```
<a [routerLink]="['/movies',{title:ring,si: true}]">...</a>
```

```
this.router.navigate(['/movies',{title: ring, si:true}]);
```

Resulting Url

```
http://localhost:4200/movies;title=ring;si=true
```

Read

```
this.route.snapshot.paramMap.get('title');
```

# Query Parameters

Configure

```
{ path: 'movies', component: MovieListComponent }
```

Activate

```
<a [routerLink]="['/movies']"  
  [queryParams]= "{title: ring, si: true}">...</a>
```

```
this.router.navigate(['movies'],  
                      {queryParams: {title: 'ring', si:true}});
```

Resulting Url

```
http://localhost:4200/movies?title=ring&si=true
```

Read

```
this.route.snapshot.queryParamMap.get('title');
```

Movie Hunter

localhost:4200/movies/1

HomeMovie ListAdvanced SearchAdd MovieWelcome DeborahLog Out

## Movie Detail: The

Description: A me  
dark l  
Director: Peter  
Release Date: Dec 1  
MPAA Film Rating: PG-13  
Category: action  
Tags: action  
Star Rating: ★ ★  
Audience Approval: 97%  
Average DVD Price: \$12.9

[← Back](#)[Edit](#)[InStep Movie Hunter](#)[Home](#)[Movie List](#)[Advanced Search](#)[Add Movie](#)[Welcome Deborah](#)[Log Out](#)

## Advanced Search

Movie Title

Director

Min Star Rating (1-5)

Max Star Rating (1-5)

Description

[Search](#)

# **Preventing Partial Page Display**



Movie Hunter

Deborah

localhost:4200/movies/1

🔑 ☆ ⋮

Home

Movie List

Advanced Search

Add Movie

Welcome Deborah

Log Out

## Movie Detail: The Lord of the Rings: The Fellowship of the Ring

Description:

Director:

Release Date:

MPAA Film Rating:

Category:

Tags:

Star Rating:

Audience Approval:

Average DVD Price:

⬅ Back

Edit

# Build a Resolver Service

movie-resolver.service.ts

```
@Injectable()
export class MovieResolver implements Resolve<IMovie> {

  constructor(private movieService: MovieService) { }

  resolve(route: ActivatedRouteSnapshot,
          state: RouterStateSnapshot): Observable<IMovie> {
    const id = route.paramMap.get('id');
    return this.movieService.getMovie(+id);
  }
}
```

# Register the Resolver

app.module.ts

```
providers: [  
  MovieService,  
  MovieResolver  
]
```

# Add Resolver to Route

app.module.ts

```
{  
  path: 'movies/:id',  
  resolve: { movie: MovieResolver },  
  component: MovieDetailComponent  
},
```

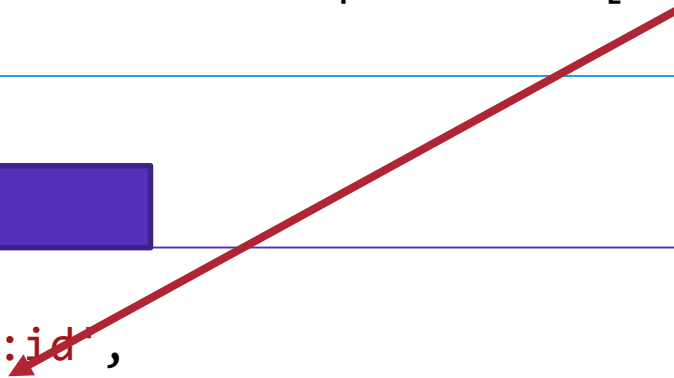
# Read Resolver Data

movie-detail.component.ts

```
ngOnInit(): void {  
  this.movie = this.route.snapshot.data['movie'];  
}
```

app.module.ts

```
{  
  path: 'movies/:id',  
  resolve: { movie: MovieResolver },  
  component: MovieDetailComponent  
},
```



Movie Hunter

Deborah

localhost:4200/movies/1

🔑 ☆ ⋮

Home

Movie List

Advanced Search

Add Movie

Welcome Deborah

Log Out

## Movie Detail: The Lord of the Rings: The Fellowship of the Ring

Description:	A meek hobbit of the Shire and eight companions set out on a journey to Mount Doom to destroy the One Ring and the dark lord Sauron.
Director:	Peter Jackson
Release Date:	Dec 19, 2001
MPAA Film Rating:	PG-13
Category:	action
Tags:	action,adventure,hobbits
Star Rating:	★★★★★
Audience Approval:	97%
Average DVD Price:	\$12.95

⬅ Back

Edit

**I'm Waiting ...**

## Movie List

Filter by:

[Show Poster](#)

Title	Director	Release Date	MPAA Rating	DVD Price	5 Star Rating	Audience Approval	
<a href="#">The Lord of the Rings: The Fellowship of the Ring</a>	Peter Jackson	Dec 19, 2001	PG 13	\$12.95	★★★★★	97%	<a href="#">Edit</a>
<a href="#">The Lord of the Rings: The Two Towers</a>	Peter Jackson	Dec 18, 2002	PG 13	\$14.95	★★★★★	94%	<a href="#">Edit</a>
<a href="#">The Lord of the Rings: The Return of the King</a>	Peter Jackson	Dec 17, 2003	PG 13	\$15.95	★★★★★	99%	<a href="#">Edit</a>
<a href="#">The Point</a>	Fred Wolf	Feb 2, 1971	NR	\$9.95	★★★★★	93%	<a href="#">Edit</a>
<a href="#">Pixels</a>	Chris Columbus	Jul 24, 2015	PG 13	\$5.59	★★★	17%	<a href="#">Edit</a>



# Subscribe to Router Events

app.component.ts

```
constructor(private router: Router) {  
  router.events.subscribe((routerEvent: Event) => {  
    this.checkRouterEvent(routerEvent);  
  });  
}
```

# Check Router Events

app.component.ts

```
checkRouterEvent(routerEvent: Event): void {  
  if (routerEvent instanceof NavigationStart) {  
    this.loading = true;  
  }  
  
  if (routerEvent instanceof NavigationEnd ||  
      routerEvent instanceof NavigationCancel ||  
      routerEvent instanceof NavigationError) {  
    this.loading = false;  
  }  
}
```

# Display the Spinner

app.component.html

```
<span class="glyphicon glyphicon-refresh glyphicon-spin spinner"  
  *ngIf="loading"></span>  
<router-outlet></router-outlet>
```

## Movie List

Filter by:

[Show Poster](#)

Title	Director	Release Date	MPAA Rating	DVD Price	5 Star Rating	Audience Approval	
<a href="#">The Lord of the Rings: The Fellowship of the Ring</a>	Peter Jackson	Dec 19, 2001	PG 13	\$12.95	★★★★★	97%	<a href="#">Edit</a>
<a href="#">The Lord of the Rings: The Two Towers</a>	Peter Jackson	Dec 18, 2002	PG 13	\$14.95	★★★★★	94%	<a href="#">Edit</a>
<a href="#">The Lord of the Rings: The Return of the King</a>	Peter Jackson	Dec 17, 2003	PG 13	\$15.95	★★★★★	99%	<a href="#">Edit</a>
<a href="#">The Point</a>	Fred Wolf	Feb 2, 1971	NR	\$9.95	★★★★★	93%	<a href="#">Edit</a>
<a href="#">Pixels</a>	Chris Columbus	Jul 24, 2015	PG 13	\$5.59	★★★	17%	<a href="#">Edit</a>

**What's going on?**

Console

Elements

Sources

Network

Performance

Memory

Application

Security

Audits

top

Filter

Default levels ▼

▼ Router Event: NavigationStart

NavigationStart(id: 5, url: '/movies/3')

▶ NavigationStart {id: 5, url: "/movies/3"}

▼ Router Event: RoutesRecognized

RoutesRecognized(id: 5, url: '/movies/3', urlAfterRedirects: '/movies/3', state: Route(url:'', path:'') { path:'id' } } )

▶ RoutesRecognized {id: 5, url: "/movies/3", urlAfterRedirects: "/movies/3", state: RouterStateSnapshot}

▼ Router Event: GuardsCheckStart

GuardsCheckStart(id: 5, url: '/movies/3', urlAfterRedirects: '/movies/3', state: Route(url:'', path:'') { path:'id' } } )

▶ GuardsCheckStart {id: 5, url: "/movies/3", urlAfterRedirects: UrlTree, state: RouterStateSnapshot}

▼ Router Event: GuardsCheckEnd

GuardsCheckEnd(id: 5, url: '/movies/3', urlAfterRedirects: '/movies/3', state: Route(url:'', path:'') { path:'id' } } ), shouldActivate: true)

▶ GuardsCheckEnd {id: 5, url: "/movies/3", urlAfterRedirects: UrlTree, state: RouterStateSnapshot, shouldActivate: true}

▼ Router Event: ResolveStart

ResolveStart(id: 5, url: '/movies/3', urlAfterRedirects: '/movies/3', state: Route(url:'', path:'') { path:'id' } } )

▶ ResolveStart {id: 5, url: "/movies/3", urlAfterRedirects: UrlTree, state: RouterStateSnapshot}

Data: {"id":3,"director":"Peter Jackson","description":"Gandalf and Aragorn lead the World of Men against Ring.", "imageUrl":[http://www.coverwhiz.com/content/The-Lord-Of-The-Rings-The-Return-Of-The-King\\_small.jpg](http://www.coverwhiz.com/content/The-Lord-Of-The-Rings-The-Return-Of-The-King_small.jpg), "price":15.95,"starRating":4.5,"approvalRating":0.9895,"category":"action", "tags":["action","adventure"]}

▼ Router Event: ResolveEnd

ResolveEnd(id: 5, url: '/movies/3', urlAfterRedirects: '/movies/3', state: Route(url:'', path:'') { path:'id' } } )

▶ ResolveEnd {id: 5, url: "/movies/3", urlAfterRedirects: UrlTree, state: RouterStateSnapshot}

▼ Router Event: NavigationEnd

NavigationEnd(id: 5, url: '/movies/3', urlAfterRedirects: '/movies/3')

⋮

Console

```
RouterModule.forRoot([
  {
    path: '', component: ShellComponent,
    children: [
      { path: 'welcome', component: WelcomeComponent },
      { path: 'movies', component: MovieListComponent },
      { path: '', redirectTo: 'welcome', pathMatch: 'full' }
    ]
  },
  { path: 'login', component: LoginComponent },
  { path: '**', component: PageNotFoundComponent }
], { enableTracing: true })
```

**Guards at the Ready**



## Log In

User Name

Password

Log In

InStep Movie Hunter

Home

Welcome Deborah

Log Out

## Edit Movie: The Point

Basic Information

Search Tags

### Basic Movie Information

Movie Title

Director

Star Rating (1-5)

Description

An animated story of an unusual kingdom in which everything and everybody is pointed - except for a young boy named Oblio. Despite his round head, Oblio has many friends. But an evil count, jealous that Oblio is more popular than his own son,

Save

Cancel

Delete

localhost:4200 says:

Navigate away and lose all changes to The Point?

OK

Cancel

# Build a Guard Service

movie-edit-guard.service.ts

```
@Injectable()
export class MovieEditGuard implements
    CanDeactivate<MovieEditComponent> {

    canDeactivate(component: MovieEditComponent): boolean {
        if (component.isDirty) {
            const title = component.movie.title || 'New Movie';
            return confirm(`Lose all changes to ${title}?`);
        }
        return true;
    }
}
```

# Register the Guard

app.module.ts

```
providers: [  
  MovieService,  
  MovieResolver,  
  MovieEditGuard  
]
```

## Add Guard to Route

app.module.ts

```
{  
  path: 'movies/:id/edit',  
  resolve: { movie: MovieResolver },  
  canActivate: [ MovieEditGuard ],  
  component: MovieEditComponent  
}
```

## Log In

User Name

Password

Log In

InStep Movie Hunter

Home

Welcome Deborah

Log Out

## Edit Movie: The Point

Basic Information

Search Tags

### Basic Movie Information

Movie Title

Director

Star Rating (1-5)

Description

An animated story of an unusual kingdom in which everything and everybody is pointed - except for a young boy named Oblio. Despite his round head, Oblio has many friends. But an evil count, jealous that Oblio is more popular than his own son,

Save

Cancel

Delete

localhost:4200 says:

Navigate away and lose all changes to The Point?

OK

Cancel

# Keep Tabs

Movie Hunter

localhost:4200/movies/1/edit/info

Home

Movie List

Advanced Search

Add Movie

Welcome Deborah

Log Out

Edit Movie: The Lord of the Rings: The Fellowship of the Ring

Basic Information

Search Tags

Basic Movie Information

Movie Title

The Lord of the Rings: The Fellowship of the Ring

Director

Peter Jackson

Star Rating (1-5)

4.88

Description

A meek hobbit of the Shire and eight companions set out on a journey to Mount Doom to destroy the One Ring and the dark lord Sauron.

Save

Cancel

Delete

app

shell

edit

```
{
  path: '', component: ShellComponent,
  children: [
    { path: 'welcome', component: WelcomeComponent },
    { path: 'movies', component: MovieListComponent },
    { path: 'movies/:id', component: MovieDetailComponent },
    { path: '', redirectTo: 'welcome', pathMatch: 'full' }
  ]
},
{ path: 'login', component: LoginComponent },
{ path: '**', component: PageNotFoundComponent }
```



```
{
  path: '', component: ShellComponent,
  children: [
    { path: 'welcome', component: WelcomeComponent },
    { path: 'movies', component: MovieListComponent },
    { path: 'movies/:id', component: MovieDetailComponent },
    { path: 'movies/:id/edit', component: MovieEditComponent,
      children: [
        { path: '', redirectTo: 'info', pathMatch: 'full' },
        { path: 'info', component: MovieEditInfoComponent },
        { path: 'tags', component: MovieEditTagsComponent }
      ]
    },
    { path: '', redirectTo: 'welcome', pathMatch: 'full' }
  ]
}, ...
```

# Edit Template

movie-edit.component.html

```
<div class="panel-body">
  <div class="wizard">
    <a [routerLink]="['info']" routerLinkActive="active">
      Basic Information</a>
    <a [routerLink]="['tags']" routerLinkActive="active">
      Search Tags</a>
  </div>

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

Movie Hunter

localhost:4200/movies/1/edit/info

HomeMovie ListAdvanced SearchAdd MovieWelcome DeborahLog Out

Edit Movie: The Lord of the Rings: The Fellowship of the Ring

Basic Information

Search Tags

Basic Movie Information

Movie Title

The Lord of the Rings: The Fellowship of the Ring

Director

Peter Jackson

Star Rating (1-5)

4.88

Description

A meek hobbit of the Shire and eight companions set out on a journey to Mount Doom to destroy the One Ring and the dark lord Sauron.

Save

Cancel

Delete

edit

**Keep Constant**

## app-routing.module.ts

```
@NgModule({
  imports: [
    RouterModule.forRoot([
      {
        path: '', component: ShellComponent,
        children: [
          { path: 'welcome', component: WelcomeComponent },
          { path: 'movies', component: MovieListComponent },
          { path: 'movies/:id', component: MovieDetailComponent },
          {
            path: 'movies/:id/edit', component: MovieEditComponent,
            children: [
              { path: '', redirectTo: 'info', pathMatch: 'full' },
              { path: 'info', component: MovieEditInfoComponent },
              { path: 'tags', component: MovieEditTagsComponent }
            ]
          },
          { path: '', redirectTo: 'welcome', pathMatch: 'full' }
        ]
      },
      { path: 'login', component: LoginComponent },
      { path: '**', component: PageNotFoundComponent }
    ])
  ],
  exports: [RouterModule]
})

export class AppRoutingModule { }
```

## app-routing.module.ts

```
const appRoutes: Routes = [
  {
    path: '', component: ShellComponent,
    children: [
      { path: 'welcome', component: WelcomeComponent },
      { path: 'movies', component: MovieListComponent },
      { path: 'movies/:id', component: MovieDetailComponent },
      { path: 'movies/:id/edit', component: MovieEditComponent,
        children: [
          { path: '', redirectTo: 'info', pathMatch: 'full' },
          { path: 'info', component: MovieEditInfoComponent },
          { path: 'tags', component: MovieEditTagsComponent }
        ]
      },
      { path: '', redirectTo: 'welcome', pathMatch: 'full' }
    ]
  },
  { path: 'login', component: LoginComponent },
  { path: '**', component: PageNotFoundComponent }
];
```

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

**It's a Feature**



# What About Feature Modules?

## app-routing.module.ts

```
const appRoutes: Routes = [
  {
    path: '', component: ShellComponent,
    children: [
      { path: 'welcome', component: WelcomeComponent },
      { path: 'movies', component: MovieListComponent },
      { path: 'movies/:id', component: MovieDetailComponent },
      { path: 'movies/:id/edit', component: MovieEditComponent,
        children: [
          { path: '', redirectTo: 'info', pathMatch: 'full' },
          { path: 'info', component: MovieEditInfoComponent },
          { path: 'tags', component: MovieEditTagsComponent }
        ]
      },
      { path: '', redirectTo: 'welcome', pathMatch: 'full' }
    ]
  },
  { path: 'login', component: LoginComponent },
  { path: '**', component: PageNotFoundComponent }
];
```

## movie.module.ts

```
export const movieRoutes: Routes = [
  { path: '', component: MovieListComponent },
  { path: ':id', component: MovieDetailComponent },
  { path: ':id/edit', component: MovieEditComponent,
    children: [
      { path: '', redirectTo: 'info', pathMatch: 'full' },
      { path: 'info', component: MovieEditInfoComponent },
      { path: 'tags', component: MovieEditTagsComponent }
    ]
  }
];

@NgModule({
  imports: [ ...,
    RouterModule.forChild(movieRoutes) // Won't work
  ], ...
})

export class MovieModule { }
```

# Merged Routes

```
{ path: '', component: MovieListComponent } ,
{ path: ':id', component: MovieDetailComponent },
{ path: ':id/edit', component: MovieEditComponent,
  children: [
    { path: '', redirectTo: 'info', pathMatch: 'full' },
    { path: 'info', component: MovieEditInfoComponent },
    { path: 'tags', component: MovieEditTagsComponent }
  ]
}
{ path: '', component: ShellComponent,
  children: [
    { path: 'welcome', component: WelcomeComponent },
    { path: '', redirectTo: 'welcome', pathMatch: 'full' }
  ]
},
{ path: 'login', component: LoginComponent },
{ path: '**', component: PageNotFoundComponent }
```

# Use the Feature Module Constant

app-routing.module.ts

```
{
  path: '', component: ShellComponent,
  children: [
    { path: 'welcome', component: WelcomeComponent },
    { path: 'movies', children: movieRoutes },
    { path: '', redirectTo: 'welcome', pathMatch: 'full' }
  ]
},
{ path: 'login', component: LoginComponent },
{ path: '**', component: PageNotFoundComponent }
```

## movie.module.ts

```
export const movieRoutes: Routes = [
  { path: '', component: MovieListComponent },
  { path: ':id', component: MovieDetailComponent },
  { path: ':id/edit', component: MovieEditComponent,
    children: [
      { path: '', redirectTo: 'info', pathMatch: 'full' },
      { path: 'info', component: MovieEditInfoComponent },
      { path: 'tags', component: MovieEditTagsComponent }
    ]
  }
];

@NgModule({
  imports: [ ...,
    RouterModule
  ], ...
})

export class MovieModule { }
```

# Lazy Loading



**From: Angular Routing course on Pluralsight**

# Without Lazy Loading: children

app-routing.module.ts

```
{
  path: '', component: ShellComponent,
  children: [
    { path: 'welcome', component: WelcomeComponent },
    {
      path: 'movies',
      children: movieRoutes
    },
    { path: '', redirectTo: 'welcome', pathMatch: 'full' }
  ]
},
{ path: 'login', component: LoginComponent },
{ path: '**', component: PageNotFoundComponent }
```



# With Lazy Loading: loadChildren

app-routing.module.ts

```
{
  path: '', component: ShellComponent,
  children: [
    { path: 'welcome', component: WelcomeComponent },
    {
      path: 'movies',
      loadChildren: './movies/movie.module#MovieModule'
    },
    { path: '', redirectTo: 'welcome', pathMatch: 'full' }
  ]
},
{ path: 'login', component: LoginComponent },
{ path: '**', component: PageNotFoundComponent }
```

## movie.module.ts

```
const movieRoutes: Routes = [
  { path: '', component: MovieListComponent },
  { path: ':id', component: MovieDetailComponent },
  { path: ':id/edit', component: MovieEditComponent,
    children: [
      { path: '', redirectTo: 'info', pathMatch: 'full' },
      { path: 'info', component: MovieEditInfoComponent },
      { path: 'tags', component: MovieEditTagsComponent }
    ]
  }
];

@NgModule({
  imports: [ ...,
    RouterModule.forChild(movieRoutes)
  ], ...
})

export class MovieModule { }
```

## For More Information

**@deborahkurata**

**<https://github.com/DeborahK/MovieHunter-routing>**

**[bit.ly/Angular-GettingStarted](https://bit.ly/Angular-GettingStarted)**

**[bit.ly/Angular-routing](https://bit.ly/Angular-routing)**

**[bit.ly/Angular-Communication](https://bit.ly/Angular-Communication)**

**Thank you!**