# Routing Basics

**Deborah Kurata**

CONSULTANT | SPEAKER | AUTHOR | MVP | GDE

@deborahkurata | blogs.msmvps.com/deborahk/
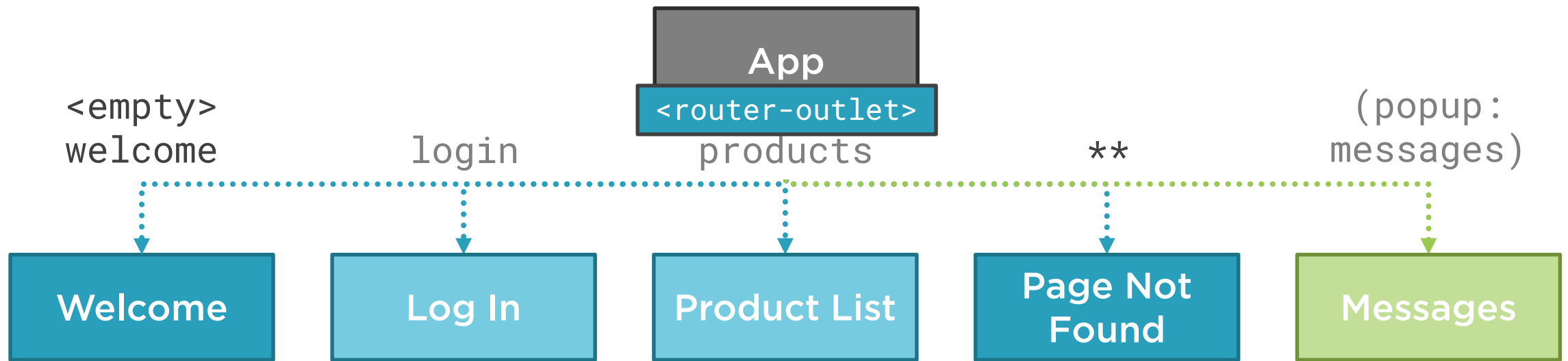
# Module Overview

**Setting up Basic Routing**

**HTML 5 vs. Hash-based Urls**

App

`<router-outlet>`

<empty>
welcome

login

products

**

(popup:
messages)

Welcome

Log In

Product List

Page Not
Found

Messages

·········· Primary Routes

·········· Secondary Routes

·········· Child Routes

# Setting up Routing

- Define base path
- Import Router
- Configure routes
- Place template
- Activate routes

# Defining the Base Path

www.mySite.com/APM/

**Used by the router**

- Building the Url for navigation

**Used by the browser**

- Building the Url for downloading and linking

```
<base href="/">
```

◄ **For development:**
  localhost:4200

```
<base href="/APM/">
```

◄ **For deployment:**
  https://www.mySite.com/APM/

# Resetting the Base Path

**<base href="/">**

(localhost:4200)

**<base href="/APM/">**

(www.mySite.com/APM/)

**Manually reset the base path**

**Use the Angular CLI**
- ng build --base-href /APM/

# Importing the Angular Router

RouterModule

**Service**

**Configuration**

**Directives**
- RouterLink
- RouterLinkActive
- RouterOutlet

# Importing the Angular Router

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

@NgModule({
  imports: [
    BrowserModule,
    RouterModule
  ],
  declarations: [
    ...
  ],
  bootstrap: [ AppComponent ]
})
export class AppModule { }
```

# Importing the Angular Router

**RouterModule.forRoot()**

Declares the router directives

Manages our route configuration

Registers the router service

Used once for the application

**RouterModule.forChild()**

Declares the router directives

Manages our route configuration

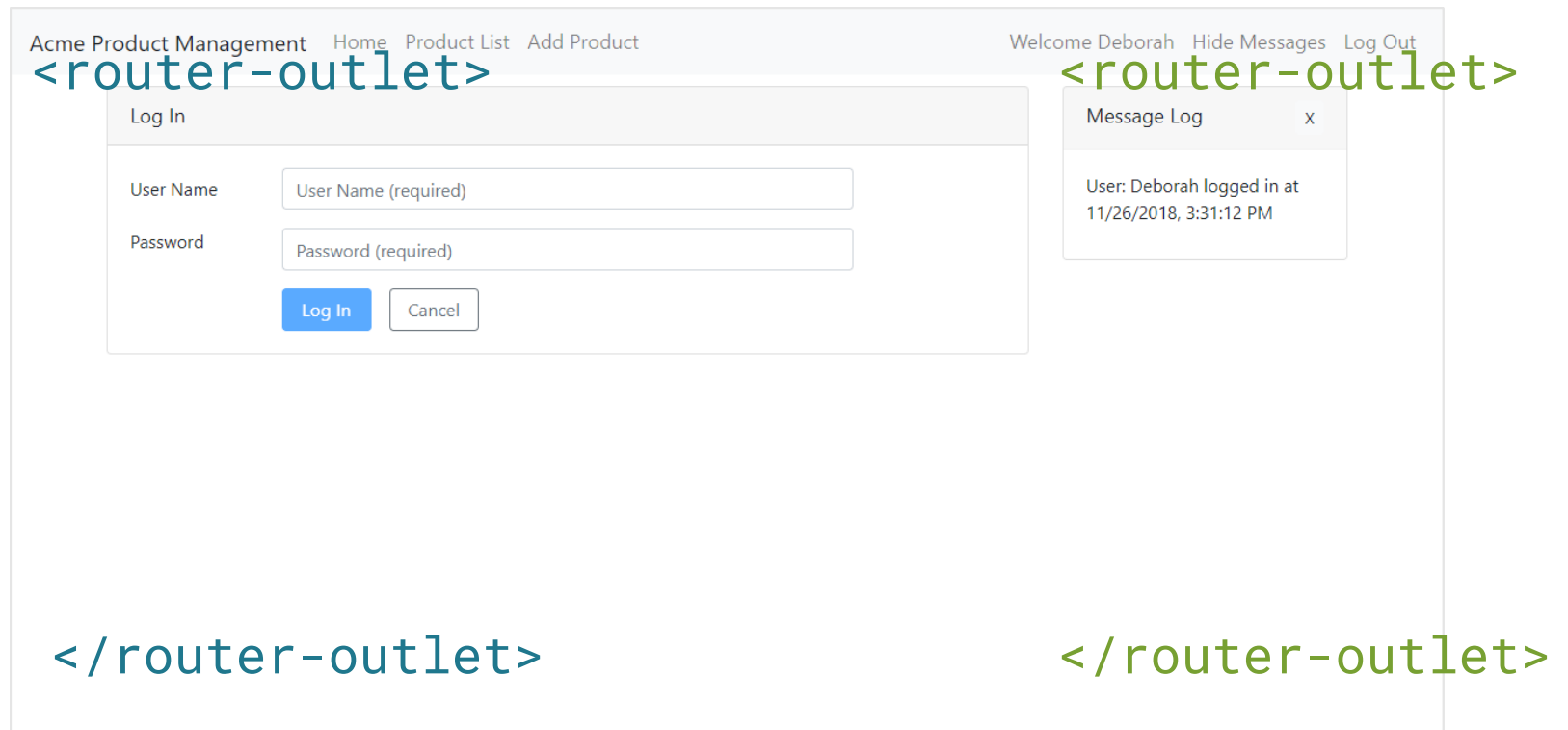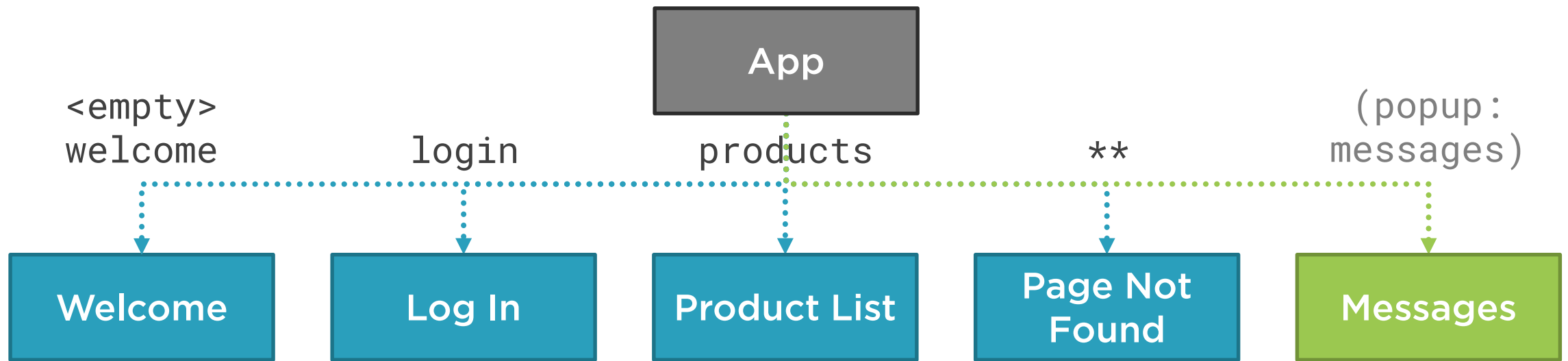Does NOT register the router service

Used in feature modules

# Importing the Angular Router

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

@NgModule({
  imports: [
    BrowserModule,
    RouterModule.forRoot([])
  ],
  declarations: [
    AppComponent
  ],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

# Configuring Routes

```
[
 { path: 'welcome', component: WelcomeComponent },
 { path: '', redirectTo: 'welcome', pathMatch: 'full' },
 { path: '**', component: PageNotFoundComponent }
]
```

# Redirects

```
[
 { path: 'home', component: WelcomeComponent },
 { path: 'welcome', redirectTo: 'home', pathMatch: 'full' },
 { path: '', redirectTo: 'home', pathMatch: 'full' },
 { path: '**', component: PageNotFoundComponent }
]
```

# Placing the Templates

```
...

<ul>
    <li><a [routerLink]="['/welcome']">Home</a></li>
    <li><a [routerLink]="['/products']">Product List</a></li>
</ul>
<router-outlet></router-outlet>
```

# Navigating the Application Routes

**Menu option, link, image or button that activates a route**

**Typing the Url in the address bar / bookmark**

**The browser's forward or back buttons**

# Activating a Route

```
...

<ul>
    <li><a [routerLink]="['/welcome']">Home</a></li>
    <li><a [routerLink]="['/products']">Product List</a></li>
</ul>
<router-outlet></router-outlet>
```
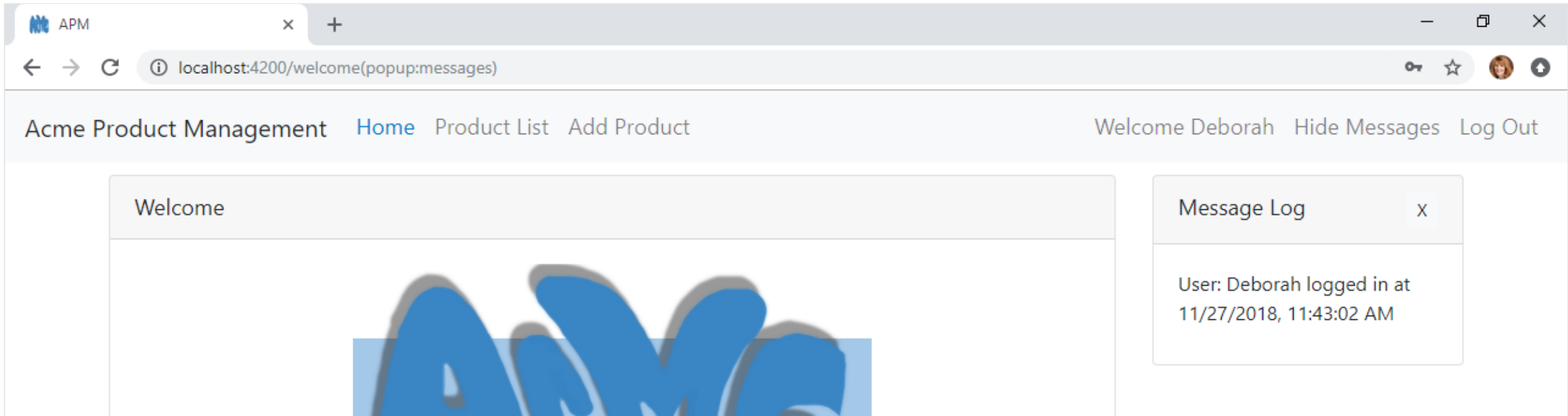
```
...

<ul>
    <li><a routerLink="/welcome">Home</a></li>
    <li><a routerLink="/products">Product List</a></li>
</ul>
<router-outlet></router-outlet>
```

# Browser Url Styles

# Browser Url Styles

**HTML 5 Style (.../welcome)**

Leverages HTML 5 history pushState

Default

More natural Url

Requires Url rewriting

**Hash-based (.../#/welcome)**

Leverages Url fragments

**Browser**

www.mysite.com/APM/welcome

404!

**Server**

APM/welcome

- APM
  - index.html

# Browser Url Styles

**HTML 5 Style (…/welcome)**

Leverages HTML 5 history pushState

Default

More natural Url

Requires Url rewriting

**Hash-based (…/#/welcome)**

Leverages Url fragments

Must be set with { useHash: true }

**Browser**

www.mysite.com/APM/welcome →

← index.html

**Server**

APM/index.html →

- APM
  - index.html

# Browser Url Styles

**HTML 5 Style (…/welcome)** | **Hash-based (…/#/welcome)**

Leverages HTML 5 history pushState | Leverages Url fragments

Default | Must be set with `{ useHash: true }`

More natural Url | Does not require Url rewriting

Requires Url rewriting

**Browser**

www.mysite.com/APM/#/welcome ⟶

index.html ⟵

**Server**

APM/ ⟶

- APM
  - index.html

# Using Hash-based Urls

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

@NgModule({
  imports: [
    BrowserModule,
    RouterModule.forRoot([...], { useHash: true })
  ],
  declarations: [
    ...
  ],
  bootstrap: [ AppComponent ]
})
export class AppModule { }
```

# Basics Checklist: Setting Up

**Define the base path**

```
<base href="/">
```

**Import RouterModule**

- Use RouterModule.forRoot() for app routes (one time only!)

- Use RouterModule.forChild() for features

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

# Basics Checklist: Configuring Routes

**path: Url segment(s) for the route**

- No leading slash
- '' for default route; '**' for wildcard route
- Casing matters

**component**

- Not string name; not enclosed in quotes
- Component must be imported

**Order matters!**

```
{ path: 'welcome', component: WelcomeComponent },

{ path: '', redirectTo: 'welcome', pathMatch: 'full' },

{ path: '**', component: PageNotFoundComponent }
```

# Basics Checklist: Placing the Template

**Add the RouterOutlet directive**

- Identifies where to display the routed component's template

- Primary RouterOutlet normally specified in the App component's template

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

# Basics Checklist: Activating a Route

**Add the RouterLink directive as an attribute**

- Clickable element
- Enclose in square brackets

**Bind to a link parameters array**

- First element is the root Url segment
- All other elements are route parameters or additional Url segments

```html
<a [routerLink]="['/welcome']">Home</a>
```
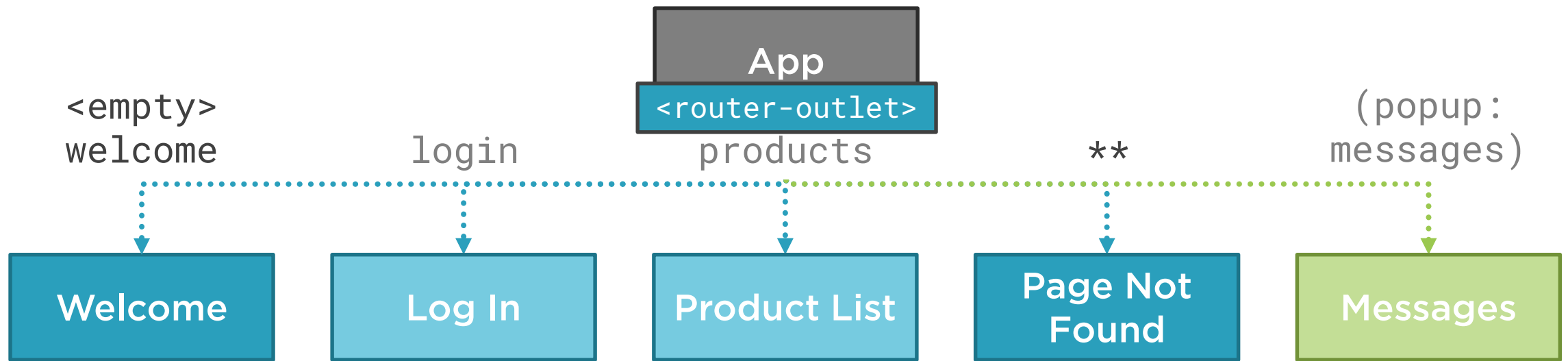
# Summary

**Setting up Routing**

- Defining the Base Path
- Importing RouterModule
- Configuring Application Routes
- Placing the Template (RouterOutlet)
- Activating Routes (RouterLink)

**HTML 5 vs. Hash-based Urls**