

Angular Hands-on Workshop: Beyond the Basics

Advanced Routing

Route Guards

Route Guards

- Used to check the user's permission to perform various route changes
- Guards are interfaces you implement
 - CanActivate
 - CanActivateChild
 - CanDeactivate
 - Resolve
 - CanLoad
- Guards are services that implement one of the interfaces above
- Guards are applied to a route and automatically apply to all child routes

Lazy Loading

Requirements for Lazy Loading

- Only feature modules may be lazy loaded
- Routes must be grouped under a single parent route in the feature module
- Feature module must not be imported in another Angular module

Pre-loading Routes

- Hybrid approach between eager and lazy loading
- Loads modules asynchronously before they're requested
- Loading strategy is customizable

Advanced Routing

Links to Additional
Information

- Routing and Navigation
 - <https://angular.io/guide/router>