[Course Overview](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m0&clip=0&mode=live" \t "psplayer)

[Course Overview](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m0&clip=0&mode=live)

[Hello. My name is Deborah Kurata. Welcome to my course, Angular Routing, from Pluralsight.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m0&clip=0&mode=live&start=0) [This course is for Angular version 2 and above. There is more to routing in Angular than just moving the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m0&clip=0&mode=live&start=12.5) [user between multiple views of an application. After covering the basics, this course details how to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m0&clip=0&mode=live&start=17.5) [configure child routes to build a tabbed edit form and secondary routes to display multiple routable panels.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m0&clip=0&mode=live&start=24.5) [Sometimes you need to pass a little data to your routes with parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m0&clip=0&mode=live&start=32.5) [You'll see how to use required, optional, and query parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m0&clip=0&mode=live&start=36.5) [To make your app more visually appealing, you'll walk through how to style and animate routes, and react to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m0&clip=0&mode=live&start=40.5) [route events to display a spinner while a route is loading. You'll learn how to pre-fetch data required for](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m0&clip=0&mode=live&start=46.5) [a route and guard routes to add permissions or prevent the user from leaving a route without confirmation.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m0&clip=0&mode=live&start=52.5) [And you'll discover how to improve your application startup time by asynchronously loading your routes using](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m0&clip=0&mode=live&start=60.5) [lazy loading. By the end of this course, you will have the knowledge you need to leverage more sophisticated](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m0&clip=0&mode=live&start=67.5) [routing features and support more real-world routing scenarios. I hope you'll join me on this journey](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m0&clip=0&mode=live&start=73.5) [through Angular Routing from Pluralsight.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m0&clip=0&mode=live&start=79.5)

[Introduction](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=0&mode=live)

[Introduction](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=0&mode=live)

[Ah, routing, we often just use you to move the user between multiple views of an application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=0&mode=live&start=0) [But you can do so much more. Welcome to Angular Routing from Pluralsight.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=0&mode=live&start=8.5) [My name is Deborah Kurata, and this course moves quickly beyond the basics to cover more sophisticated](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=0&mode=live&start=13.5) [routing features to support more real-world routing scenarios.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=0&mode=live&start=19.5) [With routing, we have the tools we need to set up navigation through our application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=0&mode=live&start=24.5) [We can define multiple routes, pass data to routes, reload data for our views, group our routes,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=0&mode=live&start=29.5) [guard our routes, add styling and animation, and even improve performance by asynchronously loading our routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=0&mode=live&start=35.5) [This introduction module explores the sample application, details how routing works, suggests several tips](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=0&mode=live&start=43.5) [for getting the most from this course, and outlines the topics we'll cover in the remainder of this course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=0&mode=live&start=51.5) [Let's get started.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=0&mode=live&start=57.5)

[Sample Application: Demo](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live)

[Welcome to Acme Product Management. As its name implies, this application manages our company's current list](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=2.339) [of products. You may remember this application from such courses as Angular 2: Getting Started and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=8.339) [Angular 2: Reactive Forms. The root app component provides the application shell.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=15.339) [It has a title, a navigation bar, and an outlet where the router places the views.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=21.339) [When I launch the application, the router activated a default route to display our Welcome page in that outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=27.339) [And the router highlights the selection in the menu so the user can keep a sense of where they are in the application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=34.339) [Let me refresh so you can see the route's animation. Cool! If I click on the Product List,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=41.339) [the route guard knows I have not logged in so navigates to the Log In component's view.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=48.339) [After logging in, I'm taken to the Product List as I had requested.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=55.339) [I'll click Show Messages to show a secondary or auxiliary route. With secondary routes, we define multiple](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=59.339) [outlets and display routed views here or here. Our little message log is not very exciting,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=65.339) [but imagine using this feature to display a changing toolbox or a detail view or to build a more complex UI](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=73.339) [such as a dashboard. Notice how the URL defines the content of the primary outlet and the secondary outlet](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=81.339) [called popup in parentheses. Let's see the display of the Product List again.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=88.339) [I'll click on Home to navigate back to the Welcome page. As I click Product List, watch the little bit of](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=94.339) [delay between displaying the page header](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=101.339) [and displaying the Product List. This occurs because we display the page and then go get the data.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=104.339) [This delay will be more pronounced for users with a slower connection or more data.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=110.339) [Now watch as I click on the product name and navigate to the Product Detail page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=115.339) [We see a wait cursor, and then the page displays in its entirety.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=120.339) [There is no awkward partial page display.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=125.5) [This demonstrates the routing prefetch feature where we get the data for a component before displaying](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=128.339) [its template. That prevents the partial display of our pages. We can navigate back to the Product List page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=135.339) [Now I'll filter the list and click to show the product images. I'll again click on a product to view its](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=141.339) [detail and navigate back to the Product List page. It remembered its settings.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=149.339) [It retained the Product List page state using query parameters on the URL.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=155.339) [Clicking on the Edit button, we navigate to the Product Edit page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=160.339) [It also uses prefetching so it displays a wait cursor then the entire page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=164.339) [Notice the tabs here. In many real-world applications, we have more data that can fit nicely on a simple form.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=170.339) [A common technique for breaking the data onto multiple forms is using tabs.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=178.339) [I only show two tabs here, but you can imagine any number of tabs.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=182.339) [The tabs are implemented using child routes. The parent route displays the outer portion of the edit page,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=187.339) [and the child routes display the tab details. And notice the styling here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=193.339) [It provides an indication of the active tab. This page also demonstrates validation across multiple components.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=199.339) [If I have a validation error here and navigate to the other page,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=207.339) [the Save button is still disabled. Instead of forcing the user to look through each tab to find the error,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=211.339) [an icon appears on the tab containing the error. If I try to navigate away, a route guard asks for a confirmation.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=219.339) [This prevents the user from leaving the page accidentally and losing their changes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=225.339) [We'll implement all of these features and more as we navigate through this course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=232.339) [Now let's look at the architecture of this sample application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=1&mode=live&start=237.339)

[Sample Application: Architecture](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=2&mode=live)

[When building an Angular application, we break the features and user interface pieces into components.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=2&mode=live&start=0) [For the APM sample application, we have a main application component called App component, and a component](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=2&mode=live&start=8.447) [for each primary page of the application. The Welcome page has a Welcome component.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=2&mode=live&start=13.447) [The dashed line here represents route navigation. The user can navigate from the menu defined in the App](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=2&mode=live&start=20.447) [component to the Welcome page and the Welcome component. The Log In page has a Log In component.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=2&mode=live&start=26.447) [The Product List page has a Product List component. The Detail page has its own component.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=2&mode=live&start=33.447) [And we route to the Detail page from the Product List. We can navigate to the Edit page from the Add](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=2&mode=live&start=38.447) [Product menu option defined in the App component or from the Edit buttons on the Product List and Product](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=2&mode=live&start=43.447) [Detail pages. Our Product Edit component tabs use child routes to navigate to the Edit Info component and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=2&mode=live&start=50.447) [Edit Tags component. An auxiliary or secondary route displays the Message component side by side with the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=2&mode=live&start=58) [primary routed component. And we route to a Page Not Found component if a navigation request does not match](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=2&mode=live&start=64.447) [any of our routes. Though we could not see it in the demo, we'll lazy load the entire set of product](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=2&mode=live&start=71.447) [features to improve our application startup performance. Now that we have our sample application components](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=2&mode=live&start=77.447) [defined, let's see how we organize them into Angular modules.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=2&mode=live&start=82.447) [For the APM sample application, I divided the application pieces into multiple modules--App Module for the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=2&mode=live&start=88.447) [basic application and startup pieces, the Product Module is for the product feature pieces, User Module for](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=2&mode=live&start=94.447) [the pieces that manage the login and user authentication, Message Module for the message feature pieces](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=2&mode=live&start=102.447) [such as the display of our message log, and Shared Module for pieces sharable across our feature modules.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=2&mode=live&start=110.447) [That's the basics of the sample application. In the demo, we saw many routing features in action.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=2&mode=live&start=117.447) [But how does routing work?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=2&mode=live&start=124.447)

[How Routing Works](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live)

[Here is a simple example to walk through the basic routing process.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live&start=0.5) [We tie a route to a link button or menu option using a built-in routing directive called routerLink.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live&start=6.112) [When the user clicks on the Product List option, for example, the Angular router navigates to the products route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live&start=14.112) [The browser's location URL changes to include this route segment, and we see products appear in the address bar.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live&start=21.112) [We can think of this URL as a serialization of the router state. Each navigation operation results in a URL](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live&start=29.112) [change, and any URL change kicks off the router. The router looks in the route configuration for a route](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live&start=36.112) [definition with a path matching the URL segment. The route definition includes the component to activate,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live&start=45.112) [in this case, the ProductListComponent. The Angular router then loads the component's template in the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live&start=51.112) [primary outlet. We specify the location of that outlet using the built-in routing directive called router-outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live&start=58.112) [And the Product List appears in that location. Now that we've seen simple routing from the user's](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live&start=66.112) [perspective, let's see how routing works from the router's perspective.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live&start=73.112) [The router's job begins when it sees a change to the URL in the address bar.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live&start=78.112) [The URL changes when the user clicks on an element that includes the routerLink directive, when the user](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live&start=83.112) [directly updates the address bar, or when we navigate in code. The router then looks through the array of](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live&start=88.112) [configured routes checking the path of each route against the URL segments.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live&start=95.112) [If the router is not able to match the URL, the navigation fails.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live&start=100.112) [Otherwise, the process continues with the route associated with the matching path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live&start=105.112) [If the matching route defines a redirect, the redirect is immediately processed, and the revised URL is matched.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live&start=110.112) [The router then checks any route guards to ensure the navigation to this new route is permitted.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live&start=118.112) [If the route guards allow access, the router resolves and prefetches any data required by the route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live&start=124.112) [Once any required data is retrieved, the router activates all components associated with the route,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live&start=130.112) [instantiating them if needed and places their templates into their defined router outlets.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live&start=137.112) [Then the router waits for the next URL change. We'll cover all of the concepts mentioned here throughout this course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live&start=143.5) [Now let's take a moment and look at some tips for getting the most from this course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=3&mode=live&start=152.5)

[Get the Most from This Course](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live)

[First, the Prerequisites. To get the most from this course, it is important that you minimally know the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=2.184) [basics of Angular version 2 or higher. This means understanding Angular modules, components, templates, and services.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=8.184) [If you don't have the requisite knowledge, consider taking one of the introductory Angular version 2 courses](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=16.184) [such as Angular 2: Getting Started or Angular 2: First Look. You do not need extensive routing experience.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=22.184) [We'll cover what you need in this course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=30.184) [Another way to get the most from this course is to join the discussion.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=33.184) [Thoughts, comments, or questions as you watch this course? Feel free to use the Discussion tab.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=38.184) [You can find the link to the discussion on the Pluralsight page for this course, or follow me on Twitter.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=43.184) [It would be great to hear about your experiences with routing.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=49.184) [There is also a blog post specifically for this course at the URL shown here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=53.184) [This post identifies common issues along with their solutions. If you have problems with the code for the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=58.184) [course, check here first. There may already be a solution provided.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=65.184) [When building web applications regardless of the technologies we use, there are often lots of steps and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=70.184) [places where things can go wrong. That's when a good checklist can come in.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=76.184) [I'll present checklists at the end of many of the modules and will use them as a brief review of what was](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=81.184) [covered in that module. Feel free to jump ahead to the checklist if you have any problems when coding](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=87.184) [along with the demoes. And consider referencing these checklists as you start implementing routing in your](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=93.184) [own Angular applications.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=100.184) [Coding along through the demoes is another great way to get the most from this course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=102.184) [Though not required, it is often helpful to try out the presented techniques.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=107.184) [To get you started, I've set up an public GitHub repository specifically for this course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=112.184) [It is called Angular-Routing, and you can find it at this URL. The starter files for the sample application are here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=117.184) [You can use these files as a starting point if you want to code along with the demoes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=125.184) [If you'd prefer to follow along with the completed sample application code, you can find that here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=131.184) [This is the application we saw in the demo at the beginning of this module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=137.184) [If you are new to GitHub, simply click this button to download all of this code as a zip file.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=141.184) [Because this is a course on routing, I've created the majority of the components and templates as part of the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=147.184) [starter files but added no routing. The hands-on activities then focus on setting up the routing features.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=153.184) [By the end of this course, you'll have a completed sample application with extensive routing features.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=162.184) [Now let's finish up this introductory module with a look at the outline for the remainder of this course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=4&mode=live&start=169.184)

[Course Outline](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=5&mode=live)

[In this course, we began at the beginning with routing basics. Next, we see how to route to features defined](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=5&mode=live&start=2.539) [in feature modules. We examine how to configure, populate, and read route parameters, optional parameters,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=5&mode=live&start=9.539) [and query parameters. To prevent partial page display, we'll see how to prefetch the data for our component](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=5&mode=live&start=17.539) [before navigating to that component using route resolvers. We examine how to set up child routes and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=5&mode=live&start=23.539) [implement a tabbed edit page. We'll look at how to group child routes that don't need to share a component](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=5&mode=live&start=31.539) [using a component-less route. Then we switch gears a bit and examine how to improve our route UI with](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=5&mode=live&start=36.539) [styling and animation, and how to watch routing events so we can display a spinner when loading a route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=5&mode=live&start=43.539) [We look at how to configure secondary routes often called auxiliary routes so we can navigate to multiple](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=5&mode=live&start=50.539) [views at once. We protect our routes with route guards. And to improve our startup time, we'll set up lazy](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=5&mode=live&start=56.539) [loading and load one of our modules asynchronously.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=5&mode=live&start=64.539) [By the end of this course, our sample application will have a route hierarchy that demonstrates all of these features.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=5&mode=live&start=68.539) [And the meaning of the items on this illustration will be clear, I promise.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=5&mode=live&start=74.539) [We're covering a lot of awesome yet practical techniques. So let's get started.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m1&clip=5&mode=live&start=79.539)

[Routing Basics](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=0&mode=live)

[Introduction](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=0&mode=live)

[You have to know where you are to get where you're going, so we begin at the beginning with routing basics.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=0&mode=live&start=0) [Welcome back to Angular Routing from Pluralsight. My name is Deborah Kurata, and in this module,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=0&mode=live&start=8) [we walk through the basics of routing.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=0&mode=live&start=15) [As we design and build our application, we provide ways for the user to navigate its pages and features.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=0&mode=live&start=18) [This is the purpose of routing. With routing, we define the paths through the application and the user](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=0&mode=live&start=24) [actions required to access each path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=0&mode=live&start=30) [In this module, we examine how to set up basic routing in an Angular application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=0&mode=live&start=34) [And we look at the differences between HTML5 and hash-based URLs used for routing.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=0&mode=live&start=40) [By the end of this module, we'll have the basic application routes defined including the route to the Welcome](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=0&mode=live&start=47) [page, the default route, and a wildcard route to the 404 Not Found page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=0&mode=live&start=52.5) [Let's get started.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=0&mode=live&start=58)

[Setting up Routing](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=1&mode=live)

[We begin by setting up the sample application. The sample application for this course is called APM for](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=1&mode=live&start=2.246) [Acme Product Management. It manages a list of products that Acme sells.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=1&mode=live&start=9.246) [If you want to work with the code for this application, you can clone the repo or download it as a zip file](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=1&mode=live&start=14.246) [and unzip it. To look at the completed code for this sample application, use the APM Final Files.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=1&mode=live&start=19.246) [If you want to code along with this course, start with the APM Start Files.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=1&mode=live&start=27.246) [Copy the APM Start Files to an APM folder.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=1&mode=live&start=32.246) [I've already copied the APM Start Files to an APM folder and opened that APM folder in VS Code.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=1&mode=live&start=37.246) [But feel free to use your favorite editor. We can use the terminal window provided in VS Code to run npm](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=1&mode=live&start=43.246) [install and install the required packages. As you can see here, I've created the majority of the components](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=1&mode=live&start=50.246) [and templates for the sample application. That way, the hands-on activities can focus on the routing features.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=1&mode=live&start=58.246) [If you begin with these starter files and code along with this course, by the end you'll have a completed](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=1&mode=live&start=65.246) [sample application with extensive routing capabilities. If you'd like to walk through the steps for building](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=1&mode=live&start=71.246) [the basic components and templates for this application, check out the Angular 2: Getting Started course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=1&mode=live&start=77.246) [It looks like the install completed successfully. Before we add routing code to this application,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=1&mode=live&start=84.246) [let's run it using npm start.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=1&mode=live&start=90.246) [Here is our Welcome page, but none of the menu options work. That's because the starter files have no routing.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=1&mode=live&start=94.246) [So we can't navigate to any other page. We'll add all of the routing features we need for this application](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=1&mode=live&start=102.246) [as we navigate through this course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=1&mode=live&start=109.246) [Setting up routing requires that we define a base path, import the Angular router, configure the routes to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=1&mode=live&start=112.246) [define which route path activates which component, identify where to place the activated component's](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=1&mode=live&start=119.246) [template, and activate routes to navigate based on user actions.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=1&mode=live&start=125.5) [Ready to take the first step?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=1&mode=live&start=131.246)

[Defining the Base Path](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live)

[Our first step to setting up routing is to define a base path for the application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=1.733) [A base path is the part of the URL that specifies the application subfolder on the server.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=7.733) [This base path is appended to the server's URL. Here the server is accessed with www.mySite.com,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=13.733) [and the subfolder containing the application is APM. So the base path for the application is APM.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=22.733) [Note that this is not a real website, this is only an example. This base path is used by the router when](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=29.733) [building a URL for navigation. The router uses this base path to compose the URL and locate the component,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=36.733) [template, and module files. And it is used by the browser to prefix relative URLs when downloading and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=43.733) [linking to scripts, images, and stylesheets.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=51.733) [The base path is set at the top of the index.html file using a base element.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=54.733) [For development, we often set the base path to slash, specifying that the routing and downloaded files are](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=60.733) [relative to the application root. This generates a URL such as this.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=67.733) [Using the root base path makes sense during development, but when the application is deployed, it may be](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=72.733) [deployed to a subfolder on the server such as this. The base path must then be set to /APM/ when the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=78.733) [application is deployed to production. How do we deal with this change when moving from dev to production?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=86.733) [When deploying our app to production, if the application resides in a subfolder of the website, we need to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=93.733) [reset the base path from the root defined with a slash to the appropriate subfolder.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=100.733) [One option for making this change is to manually reset the base path in the base element each time we deploy](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=106.733) [the app to production. This option is fraught with peril as it requires that the developer deploying the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=112.733) [app to production remembers to make the change. A better option is to use a task runner such as Gulp to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=117.733) [automatically update the base path as part of a production build. Another good option is to use the Angular CLI.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=126.733) [The Angular CLI is a set of command line tools for building, testing, and deploying Angular applications.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=133.733) [It has a command line switch that modifies the base element when building the application for production.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=140.733) [ng is the Angular CLI, build is the command, and the -- defines command options.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=147.733) [Here we set the base-href option to /APM/. This CLI allows for abbreviated versions of these options like this.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=154.733) [Now to prepare for routing, let's add the base path to our sample application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=164.733) [I have the APM Starter Files open in VS Code. In the index.html file, we add a base element just after the head tag.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=170.733) [Since the app folder is in the same folder as the index.html file, we set the href for the base tag to slash.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=180.733) [Looking at the application in the browser, it appears as before. Wait a minute!](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=188.733) [We've only done the first step to set up routing in this application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=195.733) [How are we routing to this Welcome page? Well, we aren't. Let's take a look.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=199.733) [Our welcome.component is in the home folder. As part of this component's decorator, I defined a selector.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=206.733) [This allows us to use the component as a nested component. And here in our app.component template,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=213.733) [I use the selector as a directive. This displays our Welcome page as a nested component.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=221.733) [So there is no routing going on here. Not yet. We'll remove the selector and this directive when we have](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=228.733) [routing in place. Notice all of the class attributes here. This app uses the Bootstrap CSS style classes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=235.733) [to make our application look a little nicer.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=243.5) [Now that we have the base path set, let's import the Angular router.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=2&mode=live&start=246.733)

[Importing the Angular Router](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live)

[The Angular router is in an external Angular module called RouterModule.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=2.498) [The RouterModule provides the router service to manage navigation and URL manipulation,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=6.498) [configuration for configuring our routes, and directives for activating and displaying routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=13.498) [The RouterLink directive ties a clickable HTML element to a route path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=19.498) [When a user clicks on the element, the route is activated, and the associated component's template is displayed.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=25.498) [RouterLinkActive associates a style with the active RouterLink. And RouterOutlet defines where to display](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=30.498) [the template. We'll cover each of these directives in detail. Before we can implement routing in an](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=37.498) [application or use these directives, we need to import this RouterModule.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=44) [Since RouterModule is an external module, we add it to one of our Angular modules just like any other external module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=48.498) [We import the RouterModule from the @angular/router library using the import statement.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=56.498) [Then we add it to the imports array of the @NgModule decorator. We can add the RouterModule to the imports](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=62.498) [array of multiple Angular modules. In this example, we add the RouterModule to the root app.module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=68.498) [We'll see later how to add it to multiple feature modules. But since the router service deals with the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=75.498) [globally shared resource, the URL location, there can only be one active router service.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=80.498) [To ensure that there is always only one active router service, even when importing RouterModule multiple](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=88.498) [times, RouterModule provides two methods--](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=96.498) [forRoot and forChild. RouterModule.forRoot declares the router directives, manages our route configuration,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=100.498) [and registers the router service. We use it only once in an application normally when defining the basic](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=108.498) [application routes. forChild declares the router directives and manages our route configuration, but it does](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=116.498) [not register the router service again. We normally use this method in every feature module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=123.498) [By using the forRoot method only once in the application, only one router service is registered.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=130.498) [Every other time we import RouterModule, we use the forChild method.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=137.498) [In this example, we import RouterModule for our basic application routes so we use RouterModule.forRoot.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=143.498) [The forRoot function takes in an array that defines our root configuration.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=151.498) [We'll add our root configuration here shortly. But, first, let's import the RouterModule in our sample application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=156.498) [We define the primary application routes in the root app.module. So that is where we import the RouterModule.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=164.498) [First, we use the import statement to import RouterModule from the router library.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=172.498) [Then we add the RouterModule to the imports array here. Since we are defining the basic application routes,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=177.498) [we'll call forRoot. forRoot takes in an array of route definitions.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=184.498) [We'll add those definitions in the next clip. Since we've imported RouterModule in this Angular module,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=190.498) [the routing directives defined in RouterModule are accessible to any components, directives,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=196.498) [or pipes declared in this Angular module. And since we called the forRoot method, the router service](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=201.498) [provided by RouterModule is registered and available to the entire application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=208.498) [Our root @NgModule is now importing BrowserModule to pull in basic directives such as ngIf and ngFor,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=213.498) [HttpModule to access the HTTP client services and get and save data.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=221.498) [We'll see it in use when we route to our product pages a little later.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=227.498) [InMemoryWebApiModule is a separate service developed and provided as part of the Angular documentation](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=230.498) [quick start files. We use it to simulate calls to a back-end web service to get and save product data](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=238.498) [without the need to set up a back-end web server. For more information on this service and getting and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=244.498) [saving data, check out the Angular 2: Reactive Forms course. In that course, we walk through the create,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=250.498) [read, update, and delete or CRUD operations. We just added RouterModule here, and we import our feature](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=256) [modules--ProductModule, UserModule, and MessageModule. We'll talk more about these features modules in the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=264) [next course module. We are ready for the next step--configuration.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=3&mode=live&start=271)

[Configuring Application Routes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live)

[The router has no routes until we configure them. A route configuration defines a set of components and the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=2.439) [route segment required to activate the component and display the component's template.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=9.439) [Routing is basically transitioning from one route path to another.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=14.439) [Here an empty path or a path of Welcome displays the welcome.component's template.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=19.439) [I've used color coding here to designate routes intended for different outlets.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=24.439) [The turquoise component template are all slated for the primary outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=30.439) [And the green component template is currently earmarked for a secondary outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=35.439) [For each outlet, only one route path can be active at a time. So if the welcome.component's template is](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=39.439) [displayed and the user navigates to the login path, the Log In component is instantiated, and its template](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=46.439) [replaces the welcome.component's template in the primary outlet. If the user navigates to the messages path,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=54.439) [the Message component is instantiated, and its template appears in the secondary outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=61.439) [It has no effect on the primary outlet contents. We'll talk more about secondary routes later in this course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=66.439) [We define each path and its associated component as part of the route configuration with syntax that looks like this.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=75.439) [A route configuration is an array of route objects. Each route object includes a path and other properties.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=83.439) [The path property identifies the URL segment or segments for the route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=90.439) [Any time the address bar URL changes, the router searches this list looking for a match between the path](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=96.439) [property and the current URL segments. In most cases, we also specify the component associated with this route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=103.439) [It is this component's template that is displayed when the route is activated.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=111.439) [This first route definition simply maps a specific path to a specific component.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=116.439) [When the URL segments match welcome, the WelcomeComponent is activated, and its template is displayed in](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=121.439) [the primary outlet. When the application first loads and there are no URL segments, we want to display a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=128.439) [default page. We specify that default using an empty path as shown here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=135.439) [In this example when there are no URL segments, we want the router to redirect to the welcome route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=140.439) [A redirect route requires a pathMatch property to tell the router how to match the URL segments to the path property.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=147.439) [We only want this default route when the entire client-side portion of the path is empty, so we set the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=155.439) [pathMatch to full. But redirects are not limited to use with an empty path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=160.439) [We'll talk more about redirects shortly. The asterisks in the last route definition denote a wildcard path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=167.439) [The router matches this route if the URL segments don't match any prior paths defined in the configuration.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=174.439) [This is useful for displaying a 404 Not Found page or redirecting to another route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=181.439) [A few things to note: There are no leading slashes in our path property here in the configuration.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=187.439) [Paths are case sensitive, so be sure to carefully match the casing when referring to route paths.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=194.439) [The component property is a reference to the component, not a string name, so it is not enclosed in quotes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=201.439) [And since it is a reference, we need an import statement for each referenced component.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=208.439) [A key point to keep in mind is that the order of the routes in this array matters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=213.439) [The router uses a first-match-wins strategy when matching route paths.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=217.439) [This means that more specific paths should always be before less-specific paths such as the wildcard path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=223.439) [We can use a redirect with an empty path to define a default route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=231.439) [But redirects have other uses as well. For example, we can use them if our URLs ever change such as with](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=236.439) [a URL refactoring. Say we had a path of welcome, but marketing decides to reorganize this site.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=243.439) [And now the path is home. Bummer! Now we have to change every place we activate the route to use this new path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=250.439) [Or we could instead configure a redirect, then none of the navigation needs to change, and any saved](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=257.439) [bookmarks using the old path will still work. The redirect will replace the URL segment welcome with home.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=265.439) [But there is one potential gotcha here. Redirects cannot be chained.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=272.439) [The router will do one of these redirects but not both during one navigation.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=277.439) [So if the default route is activated, the route is redirected to the welcome route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=282.439) [When it matches the welcome path, it won't redirect again. Instead, we need to change the default route to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=288.439) [also redirect to home. Redirects can be local or absolute. Local redirects replace a single URL segment](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=295.439) [with a different one such as our examples here. An absolute redirect replaces the entire URL.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=303.439) [Redirects are local unless we prefix the URL with a slash. Ready to try out some route definitions?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=310.439) [Hmm, where did we put our route definitions? Within the array here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=320.439) [In our AppComponent, we only specify the basic application routes such as the route to display our Welcome page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=325.439) [We'll set the path to welcome and specify the WelcomeComponent. When the application first loads,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=331.439) [we want to default to the Welcome page, so we'll specify a default route that redirects to our welcome route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=338.439) [And let's define a wildcard path in case the requested URL segments don't match any prior paths defined in](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=346.439) [the configuration. We'll use this route to activate the PageNotFoundComponent and display a 404 Not Found page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=353.439) [This code then creates three route definitions, configures the router via the RouterModule.forRoot method,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=362.439) [and adds the result to the app.module's imports array. When we first launch the application, there is no](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=369.439) [URL segment specified, so the router will redirect to the default route and display the welcome.component's template.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=375.439) [But display it where? We need to identify where we want the router to display the primary routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=383.439) [If we attempt to view the application in the browser at this point, we'll see errors because we have not](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=390.439) [finished setting up routing.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=4&mode=live&start=396)

[Placing the Template](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live)

[When a route is activated, the associated component's template is displayed in an outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=2.359) [We identify where we want that routed template to appear using the router-outlet directive.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=7.359) [In this example, we place the router-outlet under the menu. The routed component's template then appears](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=13.359) [in this location. Any one outlet can only display one routed component's template at a time.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=18.359) [If the user clicks Home, the welcome route is activated, and its template is displayed in this primary outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=26.359) [If the user then clicks Product List, the products route is activated, and its template replaces the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=33.359) [welcome.component template in this primary outlet. However, we can have multiple router outlets,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=40.359) [each displaying a different routed component's template. We'll see more on multiple router outlets later](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=46.359) [in this course. Let's add the router outlet to our sample application and complete our first route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=52.359) [The nav element lays out our menu. We want our templates to appear in the container below the menu.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=60.359) [So let's delete this selector here and instead use the router-outlet directive.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=66.359) [The templates for our routed component will then appear here. While we're at it, let's also remove the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=72.359) [selector from the welcome.component in the Home folder. A component only needs a selector if it will be](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=78.359) [used as a nested component. Since we are now routing to the welcome.component, instead of using it as a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=84.359) [nested component, we no longer need this selector. Let's try it out in the browser.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=90.359) [Our Welcome page comes up as before, so how can we tell if our routing is working?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=96.359) [Look at the URL. Notice in the address bar that our URL now shows the welcome route path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=102.359) [Yay! What if you're coding along and don't see the welcome page?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=109.359) [The first thing to do is open the developer tools and view the console.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=113.359) [Are there any errors displayed? Often the first or second error listed is relatively clear on what could be wrong.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=117.359) [The second thing to try is to recheck each step of the process. Check out the checklists presented at the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=125.359) [end of this module and verify each step. Another option is to download the code from the Pluralsight page](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=131.359) [for this course. It has the sample code as it should be at the end of each module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=137.359) [Compare your code to the exercise files for module 2. Repeat these steps any time throughout this course if](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=143.359) [you don't see the expected result. Now if we delete the path and access our app with no routing path,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=150.359) [the default route we specify kicks in, and the Welcome page again displays.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=157.359) [If we put in an invalid path, our wildcard route kicks in, and the Page Not Found displays.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=162.359) [In a real application, you may want something nicer here. Click Home, and we see that our menu is not yet working.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=169.359) [Let's hook that up next.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=5&mode=live&start=177.359)

[Activating Routes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live)

[With routing, the user can navigate through the application in several ways.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=1.488) [The user can click a menu option, link, image, or button that activates or navigates to a route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=7.488) [The user can type the associated URL segment in the address bar after the application URL or use a bookmark](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=12.488) [to that URL. Or the user can click the browser's forward or back buttons.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=19.488) [The router handles any changes to the URL so the last two techniques will just work.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=24.488) [For the first technique, we must activate the route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=30.488) [we use the routerLink directive to associate visual elements with route paths.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=33.488) [The routerLink is an attribute directive, so we add it to a clickable element such as a button image or](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=38.488) [the anchor tag as shown here. When the user clicks the element, the route with the path matching the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=45.488) [defined URL segments is activated. We bind the routerLink directive to a template expression that returns](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=49.488) [a link parameters array. Since this is an array, we need square brackets.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=56.488) [The first element of this array is the root or parent URL segment.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=63.488) [Be sure to include a slash before the segment. And we want this URL segment to match one of the paths from](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=67.488) [our route configuration so be sure to exactly match the path casing.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=73.488) [Additional elements can be added to this array to specify route parameters or additional route segments.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=79.488) [We'll look more at route parameters later in this course. Since these particular arrays only contain simple](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=86.488) [string elements, we could use the shortcut syntax and assign the routerLink to a simple string.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=92.488) [We can then use a one-time binding and don't specify the square brackets around the routerLink directive.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=98.488) [When the user clicks on an element with a routerLink directive, the router uses the link parameters array to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=104.488) [compose the appropriate URL, including any provided parameters, and locates the associated route configuration.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=110.488) [Now let's add the routerLink directive to our sample application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=119.488) [I already have a menu set up for this application in the app.component's template.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=124.488) [Of these menu options, we've only defined a route configure for the Welcome page, so let's add the routerLink](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=129.488) [director to the Home anchor tag and bind it to that welcome route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=136.488) [We'll add more routing as we progress through this course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=141.488) [Let's view it in the browser. Click Home,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=145.488) [and we are routed to the Welcome page. It works! Now our app.component template acts as a shell with a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=149.488) [router outlet where the router can display our views. We have the basics of routing in place.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=155.488) [Before moving on, let's take a moment to look at the difference between HTML5 style URLs and hash-based URLs](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=162) [when using routing.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=6&mode=live&start=170)

[Using HTML 5 or Hash-based URLs](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live)

[When the user clicks on an element with a RouterLink, the Angular router updates the browser's address bar](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=1.954) [and history with one or more URL segments for that route. The URL segments are managed locally on the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=7.954) [client by the router. When changing only the URL segments, the browser should not send this URL to the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=14.954) [server, nor should it reload the page. Rather, the router locates the route with a path that matches the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=20.954) [URL segment, activates the associated component, and displays that component's template.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=26.954) [And if the application has multiple outlets, the router can activate multiple components and display their](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=32.954) [templates, one in each outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=38.954) [There are two ways that the Angular router can compose local URLs for routing--HTML5 style and hash-based.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=41.954) [Let's consider the HTML5 style first. The browser's document object model or DOM provides access to the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=50.954) [browser's history through a history object. That history object exposes useful methods that allow us to move](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=58.954) [back and forth through the user's browser history and, starting with HTML5, manipulate the contents of the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=64.954) [history stack. The Angular router uses these history features, specifically the pushState, to update the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=70.954) [browser's history stack and display URLs in the address bar without triggering a server request.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=78.954) [This allows the router to make in-app URLs look like anything it wants.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=85.954) [Since the pushState was introduced in HTML5, these are often called HTML5 style URLs.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=90.954) [Browsers that don't support HTML5 send page requests to the server whenever the URL changes unless the change](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=97.954) [to the URL occurs after a hash symbol. The part of a URL after the hash symbol is called a URL fragment.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=106.954) [A URL fragment is processed on the client. So when hash-based URLs are used, the routers composes the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=113.954) [in-app URLs with a hash prefix. The browser then processes the URL fragment as an in-app URL.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=121.954) [HTML5 style URLs are the default used by the router. They are often preferred for routing because they](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=128.954) [generate a more natural looking URL and they preserve the option to do server-side rendering later.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=134.954) [For more information on server-side rendering, check out Angular Universal.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=141.954) [But what happens after the app is deployed to production and a user uses a bookmark or enters an HTML5 style URL?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=146.954) [The full URL is passed to the server, and the server will try to parse and process each segment of the URL.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=154.954) [The server won't understand the in-app portion of the URL and will throw a 404 Not Found error.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=160.954) [To prevent this error when using the HTML5 style URLs, configure your web server to perform URL rewriting.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=167.954) [With URL rewriting, you configure your server to rewrite any in-app URL segments it receives to index.html](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=175.954) [so the app is properly loaded. The app then processes the original URL and routes to the desired page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=184.954) [How the URL rewriting is done depends on your web server. See the documentation for your web server on how](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=192.954) [to configure URL rewriting. To use hash-based URLs instead, set useHash to true as part of the route configuration.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=198.954) [The key benefit of hash-based URLs is that they don't require URL rewriting because the server ignores](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=207.954) [everything after the hash. In this example, the server locates the default index.html file in the app folder](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=214.954) [and returns it. The app then processes the original URL and routes to the desired page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=221.954) [To use hash-style URLs, we pass an object into the forRoot method with any extra options we want to set.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=229.954) [In this case, we set useHash to true. Consider using HTML5 style browser URLs where possible, but know you](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=235.954) [can use hash-based URLs if needed. Let's finish off this module with some checklists we can use as we](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=243.954) [implement basic routing in our applications.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=7&mode=live&start=250.954)

[Checklist and Summary](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live)

[Checklists are a great way to recheck our understanding and our work.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=1.606) [To set up routing, define the base path using a base element in the index.html file.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=6.606) [Add RouterModule to an Angular module's imports array. Use the forRoot method for application routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=12.606) [And be sure to use it only one time in the application. Use the forChild method for feature routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=19.606) [When configuring routes, each route definition requires a path, which identifies the URL segments for the route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=26.606) [These are the segments that appear in the address bar. These URL segments allow the user to bookmark and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=34.606) [navigate back to a specific component's view, a process known as deep linking.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=41.606) [So far, all of our path properties have defined a single URL segment.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=46.606) [We'll see additional segments when we get to route parameters. When defining a path, be sure the path has](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=51.606) [no leading slash. Use an empty path for a default route and two asterisks for a wildcard route,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=57.606) [which is matched if no prior path matches. And casing matters. Ensure the casing of the path in the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=64.606) [route configuration matches everywhere the route path is specified.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=71.606) [Most route definitions also include a component. The component is a reference to the component itself.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=76) [It is not the string name, and it is not enclosed in quotes. Each reference component must be imported](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=82.606) [with the import statement. And remember that order matters! The router will pick the first route with a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=87.606) [path that matches the URL segments.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=95.606) [Use the RouterOutlet directive to identify where to display the routed component's template.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=99.606) [The primary RouterOutlet is most often specified in the App component's template.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=104.606) [When a route is activated, the routed component's template is displayed at the location of the RouterOutlet directive.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=110.606) [Once we have the routes configured, we can activate the routes based on a user action.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=117.606) [Add the RouterLink directive as an attribute to any clickable element in a component's template.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=122.606) [We can use them in menu options, toolbars, buttons, links, images, and so on.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=128.606) [Enclose the RouterLink in square brackets unless you are specifically using the shortcut syntax.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=133.606) [Bind the RouterLink directive to a link parameters array. The first element of the link parameters array](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=139.606) [is the root or parent URL segment. All other elements are values for the route parameters or additional](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=145.606) [URL segments.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=152.606) [This module was all about the basics of routing. We examined how to set up routing including defining the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=154.606) [base path, importing RouterModule using forRoot or forChild, configuring application routes,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=161.606) [placing the routed component's template using the RouterOutlet directive, and activating routes using the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=168.606) [RouterLink directive. Lastly, we looked at the differences between HTML5 style URLs and hash-based URLs](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=173.606) [when routing.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=181.606) [So far in our sample application, we've added a route to our Welcome component, a default route that](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=183.606) [redirects to the welcome route, and a wildcard route to the Page Not Found component.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=189.606) [Up next, let's see how to configure routes and feature modules so we can route to our product features and login.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m2&clip=8&mode=live&start=195.606)

[Routing to Features](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=0&mode=live)

[Introduction](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=0&mode=live)

[Most non-trivial applications have their features refactored into separate Angular feature modules.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=0&mode=live&start=0) [Welcome back to Angular Routing from Pluralsight. My name is Deborah Kurata, and in this module, we look at](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=0&mode=live&start=9) [how to set up routing to features defined in feature modules.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=0&mode=live&start=16) [What is a feature module? A feature module is an Angular module created with the express purpose of](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=0&mode=live&start=21) [organizing the components for a specific application feature area such as products or users.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=0&mode=live&start=28) [Using feature modules allows us to keep our code more organized, and it makes it possible to lazy load all](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=0&mode=live&start=35) [of the routes for a particular feature area. We'll look at lazy loading later in this course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=0&mode=live&start=42) [Every Angular application has at least one Angular module by convention called AppModule.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=0&mode=live&start=47) [But as an application gets more features, using a single Angular module can get a little unwieldy.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=0&mode=live&start=54) [We have no separation of responsibilities. Here we are mixing our basic application features with our](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=0&mode=live&start=60) [product features, our login and user features, and our message features.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=0&mode=live&start=67) [Using feature modules allows us to consolidate the components for a specific application feature area into](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=0&mode=live&start=72) [their own Angular module. Here we have a Product Module for the product features such as the Product List,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=0&mode=live&start=79) [Product Detail, and Product Edit. We have a User Module for the login, and we could add other components](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=0&mode=live&start=85) [here for User Management and Reset Password. And we can add more feature modules as our application grows.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=0&mode=live&start=92) [In this module, we look at how to set up routing differently when working with feature modules.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=0&mode=live&start=100) [We discuss route path naming strategies. We walk through how to activate a route with code instead of using](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=0&mode=live&start=106) [the RouterLink directive. We look at how to access routes from feature modules and how to refactor our basic](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=0&mode=live&start=113) [application routes into their own routing module. By the end of this course module, we'll have configured](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=0&mode=live&start=120) [basic routing for our Log In and Product List features. Let's get started.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=0&mode=live&start=127)

[Setting up for Feature Routing](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live)

[Setting up for feature routing uses a subset of the steps we used to set up our basic application routing.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=2.196) [We don't need to redefine a base path because it's already defined for the entire application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=9.196) [We do need to import the router, configure the routes, and activate the routes based on user actions.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=15.196) [But we don't need to identify where to place the activated components template because we can use the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=21.196) [primary RouterOutlet we've already defined.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=27.196) [As we saw in the last module, the Angular router is provided in an external module called RouterModule.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=31.196) [The RouterModule provides the router service to manage navigation and URL manipulation,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=38.196) [configuration for configuring our routes, and directives for activating and displaying routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=44.196) [Before we can implement routing in any feature module, we need to again import this RouterModule.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=50.196) [And since we have already imported RouterModule in our root AppModule, we need to ensure that we don't](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=58.196) [re-register the router service.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=64.196) [We import the RouterModule without re-registering the router service by using RouterModule:forChild.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=67.196) [We then pass our feature route configuration in to the array here. We've already covered how to configure](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=75.196) [routes in the last course module, and that syntax is the same for both application and feature routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=82.196) [So let's jump right into a demo.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=89.196) [Here is our product feature module. This module is currently importing SharedModule, an Angular module](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=92.196) [I created to pull in modules that can be shared across all of the feature modules.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=99.196) [This product feature module manages all of the product components and a filter pipe as shown here in the declarations.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=105.196) [And it provides a product data service to encapsulate the communication with the back-end server to get and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=113.196) [post data. To add routing to this feature module, let's start by importing the router and configuring the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=119.196) [Product List route. We first add an import statement for the RouterModule from the router library.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=126.196) [Then we add the RouterModule to the imports array here. Since we are defining feature routes, we'll use forChild.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=133.196) [This method takes in an array of route definitions. Recall how to configure a route?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=141.196) [We want to define a route with a path named products that activates the ProductListComponent.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=147.196) [That's it. Next, we need to define a user action to activate this route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=153.196) [We already have a menu option defined for the Product List here in the app.component template.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=159.196) [Recall how to activate a route? If not, you can cheat because it's right here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=164.196) [We do the same thing for the Product List route. We add the routerLink directive and assign it to a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=170.196) [link parameters array. The first element is our root URL segment, which is products.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=175.196) [To ensure that the router matches the appropriate route definition, confirm that the casing exactly matches](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=182.196) [the path in the route configuration. So we've imported RouterModule, configured the route, and activated the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=189.196) [route from a user action. Are we ready to try it out in the browser?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=196.196) [Of course we are!](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=202.196) [Our default route redirects to the welcome route, so that comes up first.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=203.196) [Click on Product List, and there it is. Excellent!](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=208.196) [And notice the address bar.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=214.196) [The URL includes the root URL segment, products. Click on Home, and it displays Welcome.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=215.196) [Going back to the Product List, we can use the Filter by.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=222.196) [We can hide and show images, but our detail links and our Edit buttons don't yet work.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=228.196) [We'll implement those in the next module of this course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=233.196) [So now we have a route path called products. How do we come up with these path names?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=238.196) [Before we move on, let's take a moment to talk about route path naming strategies.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=1&mode=live&start=244.196)

[Route Path Naming Strategies](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=2&mode=live)

[For clarity and to provide flexibility down the road, it is important to give some consideration to route](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=2&mode=live&start=2.386) [path names, especially when working with related sets of features.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=2&mode=live&start=8.386) [This is particularly important when we get to grouping routes and lazy loading later in this course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=2&mode=live&start=13.386) [We could just give each route a logical name. For example, the Product List route path is products.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=2&mode=live&start=19.386) [Our route to the Product Detail page could be product or productDetail/:id where :id is a route parameter](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=2&mode=live&start=26.386) [that represents the Id of the product to display in the detail page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=2&mode=live&start=35.386) [We'll talk more about route parameters in the next course module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=2&mode=live&start=38.386) [A route to the Product Edit page could be productEdit/:id. Are these good route path names?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=2&mode=live&start=42.386) [They could be okay, especially for a small app that won't use grouping or lazy loading.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=2&mode=live&start=50.386) [But notice that even though these features are related, their route path names are not.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=2&mode=live&start=56.386) [To make the routes easier to manage and potentially group or lazy load, consider giving each related](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=2&mode=live&start=61.386) [feature route a similar root path name. So we could keep the Product List route as products.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=2&mode=live&start=68.386) [But then name the Product Detail route as products/:id. Using products, plural, instead of product,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=2&mode=live&start=75) [singular, gives us a shared root path of products. The Product Edit could similarly be named products/:id/edit.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=2&mode=live&start=85.386) [Using common root path names clearly expresses the relationship between these features.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=2&mode=live&start=94) [As we configure the Product Detail and Product Edit routes later in this course, we'll use the root paths](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=2&mode=live&start=100.386) [with the common root path names shown here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=2&mode=live&start=107.386)

[Activating a Route with Code](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live)

[We've seen how to activate a route using the RouterLink directive, but what if we need to execute some code](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=2.816) [before we perform the routing? For example, when the user clicks the Log Out option, we want to log the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=8.816) [user out before navigating back to the Welcome page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=16) [To route with code, we use Angular's router service. We import Router from @angular/router.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=20.816) [We define a dependency on the router service using a constructor parameter.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=26.816) [Angular's dependency injector then injects the Router service instance into this component.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=31.816) [We use this router service instance to activate a route. Here we define a logOut method that we can call](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=36.816) [from the template based on a user action. The code first does some processing to log out the user.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=44.816) [Then it calls the navigate method of the router service and passes in the same link parameters array we use](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=50.816) [when binding the RouterLink directive. In this example, we navigate to the welcome route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=57.816) [So after the user logs out, they are returned to the Welcome page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=62.816) [Actually, behind the scenes, the RouterLink directive calls router.navigate with the provided link](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=68.816) [parameters array so everything that we can pass to router.navigate, we can assign to the RouterLink directive](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=74.816) [and vice versa.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=82) [When the link parameters array contains static strings, we can use a shortcut syntax just as we did with RouterLink.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=84) [We simply pass in a string instead of an array. When a route is activated using the navigate method or](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=92.816) [RouterLink, the URL is reformulated based on the current URL segments and the defined link parameters array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=98.816) [Some parts of the current URL such as secondary route information is retained when navigating.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=107.816) [If we don't want this behavior, there is another navigate method we can use, navigateByUrl.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=114.816) [The navigateByUrl method specifies a complete URL path clearing any other passed segments.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=120.816) [Let's consider this difference with an illustration. Say our URL looks like this.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=127.816) [The products route is activated, and the Product List component's template is displayed in the primary outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=134.816) [The information in parentheses is the serialization of a secondary route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=140.816) [So the messages route is activated, and the Message component's template is displayed in a secondary outlet](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=145.816) [named popup. We'll talk more about secondary routes later in this course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=151.816) [If we use the navigate method, the root URL segment is changed to welcome, but the secondary outlet](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=156.816) [information is unchanged. The Message component's template remains in the secondary outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=163.816) [If we use the navigate by URL method, the entire set of URL parameters is replaced with a defined path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=168.816) [No secondary route is specified in this example so the secondary outlet is empty.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=176.816) [You may find that you won't use navigateByUrl very often, but it's nice to know that the option is available.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=182.816) [Now let's implement the login and logout features in our sample application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=3&mode=live&start=190.816)

[Activating a Route with Code: Demo](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live)

[In this demo, we activate a route with code to implement our login and logout features.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=1.807) [If you are coding along, here is your first assignment. You've already seen how to import RouterModule](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=9.807) [and configure routes in the product feature module. Now do the same in the user feature module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=16.807) [Define a route with a path of login that activates the login component.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=23.807) [Pause the video now if you want to give it a try. Are you ready to see my solution?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=28.807) [In the user feature module that's in the user folder, I imported the RouterModule here and added it to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=35.807) [the imports array here. I used the forChild method since this is a feature module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=43.807) [And I configured the login path specifying the login component. Is this similar to what you came up with?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=48.807) [We can't try this out yet because we need to first activate the route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=56.807) [So your next assignment is to activate the login route in the HTML.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=61.807) [Here in our app.component template is a login menu option. Activate our new login route when the user](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=66.807) [selects this option. And if you've forgotten how, here's a hint. Pause the video and give it a try.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=72.807) [Are you ready for my solution? I added a routerLink directive here similar to that used for the Home](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=80.807) [and Product List menu options. Remember that the path strings are case sensitive, so be sure to match the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=87.807) [URL segment casing used here against the path entered in the route configuration.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=94.807) [Now let's check it out in the browser.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=101.807) [Click the Log In option, and there is our Log In form. There is no validation here so you can type in any](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=104.807) [username and password. When they are both entered, the Log In button is enabled.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=111.807) [Click the button, and you are logged in. You can see you're logged in because the menu options change here](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=117.807) [on the right. But this is not very user friendly. After a login, we should redirect the user to another](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=122.807) [page like the Product List page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=129.807) [Here in the login.component file is where we perform the login. After the user enters their username and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=134.807) [password and clicks to log in, this code logs them in. We then want to take the user to the Product List](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=139.807) [page so we need to activate the route with code as we saw on the slides.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=146.807) [First, we import the router from the router library. Then we add a dependency on the router service using](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=152.807) [the constructor. Lastly, we use the router's navigate method. We then specify the desired route path](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=159.807) [as the first element of the link parameters array. We want to route to the Product List so we specify](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=166.807) [products here. For clarity, I'm not using the shortcut syntax. But feel free if you want to try it out.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=173.807) [Let's check it out in the browser.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=180.807) [Enter a username and password, click the Log In button,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=182.807) [and we are redirected to the Product List page. It works!](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=186.807) [For extra practice, let's modify the login.component's template to activate the welcome route if the user](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=191.807) [clicks the Cancel button on the login form. Add the routerLink to the Cancel button here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=198.807) [To complete the Logout feature, we need to again activate a route with code to navigate to the Welcome page](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=204.807) [after the user has logged out. In the app.component template is our logout menu option.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=211.807) [This uses event binding and calls a method in the component class. The logOut method currently logs the user](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=217.807) [out and displays a message to the console. As an assignment, modify this code to activate the welcome route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=225.807) [After a logout, the user should be returned to the Welcome page. Pause the video now if you want to give it a try.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=231.807) [Are you ready for my solution? First, I imported the router from the router library.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=239.807) [Then I added a dependency on the router service using the constructor.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=246.807) [Lastly, I used the navigateByUrl method and specified the desired route path, in this case welcome.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=250.807) [Note that we pass in a simple string, not a link parameters array when using the navigateByUrl.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=258.807) [I used the navigateByUrl here instead of the navigate method to ensure that every existing parameter or](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=265.807) [secondary route is removed when the user logs out. Is that what your code looks like?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=272.807) [Now let's try it out in the browser.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=278.807) [Click Log In and enter any username and password.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=282) [Logging in causes the Log Out option to appear.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=286) [Logging out now takes us back to the Welcome page. Success!](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=4&mode=live&start=289)

[Accessing Feature Routes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live)

[Now that we've added navigation to the Product List, Log In, and Log Out options, you may be asking yourself,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=2.02) [How is the menu from our app.component accessing all of these product and user routes from the feature modules?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=9.02) [The answer is here in the app.module. In the imports array, we import RouterModule and the feature modules.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=16.02) [The router merges the application routes explicitly defined here with the routes from the product feature](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=25.02) [module and user feature module. We have not yet defined any routes for the message module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=31.02) [We'll look at that module in detail later in this course. The application then has access to all of the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=37.02) [application and feature routes. Hmm, but didn't we say that the router uses a first-match-win strategy](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=44.02) [when matching routes? Doesn't this wildcard route match everything else before we get to the feature routes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=52.02) [defined in the feature modules? Let's consider this further.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=59.02) [In looking at our route configuration in the app.module, it seems that our list of route paths would be](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=65.02) [ordered like this. First, the basic application route path is welcome, the empty path providing a default](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=72.02) [when no path is defined, and the wildcard path matching any path that does not match the paths above it.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=79.02) [Then the products path from the product feature module, then the login path from the user feature module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=87.02) [But if this were the case, the router would never find the feature routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=93.02) [The wildcard would match everything but welcome and an empty path and redirect to the Not Found page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=98.02) [So it looks like we shouldn't be able to get to our products and login routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=105.02) [But looks can be deceiving. Here is the actual order of the route paths from the route configuration defined](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=109.02) [in the app.module. How is this the order that is generated from this configuration?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=117.02) [Any route definitions that are explicitly configured in a module such as these are processed last after any](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=124.02) [imported modules. So in this configuration, the product feature module routes are defined first,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=132.02) [then the user feature module routes, then lastly these explicitly defined routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=139.02) [That's why our current route configuration works including the products and login feature routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=145.02) [Keep this ordering rule in mind as you define your app in feature routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=151.02) [To simplify our root app.module, we can move these explicitly defined routes into their own routing module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=155.5) [Let's do that next.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=5&mode=live&start=163.02)

[Defining a Routing Module](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live)

[Let's start with the Why. Why would we want to separate out routes into their own routing module?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=2.256) [Well, it's the same basic reasons we don't put our socks in the same drawer as our office supplies.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=9.256) [Separating our route configuration into its own module provides better organization.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=16.256) [Socks aren't strewn amongst the paperclips and toner cartridges and notebooks.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=22.256) [The routes are then easier to find and review or edit as needed. Just like our socks are easier to find if](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=26.256) [they are in their own clearly defined drawer. And it provides a separation of concerns.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=32.256) [When we inventory our office supplies, we don't have a bunch of socks in our way.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=39.256) [So let's refactor our root app.module and move our application routes into their own routing module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=44.256) [We start by creating a new file in the app folder called app-routing.module.ts.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=52.256) [We'll move the app.module to the side so we can see it and close the Explorer.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=59.256) [Since this is an Angular module, we create it like any other Angular module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=65.256) [We export a class, add the NgModule decorator, and import what we need.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=70.256) [This module is for our application routes so we import the RouterModule, and we need to add it to the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=75.256) [imports array. Let's just copy the route configuration from app.module and paste it here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=83.256) [Looks like we need import statements for the WelcomeComponent and PageNotFoundComponent.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=89.256) [We can copy them from the app.module and paste them here. Now in the app.module, add the import statement](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=94.256) [for the new AppRoutingModule. Then replace the explicit configuration with the AppRoutingModule and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=101.256) [remove the import statement for the RouterModule since it is no longer needed here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=109.256) [Because we removed the RouterModule from the app.module, we need to export RouterModule here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=115.256) [By exporting RouterModule, when the app.module imports AppRoutingModule, the components declared an](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=122.256) [app.module will have access to the router directives. Will our routing still work?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=128.256) [What do you think? Let's try it out in the browser.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=135.256) [The Welcome page comes up, but if we click on Product List](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=139.256) [or Log In, we are instead routed to the Not Found page. I bet you know why.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=142.256) [Now that our application routes are in an external module, the route paths are processed in the order](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=150.256) [they are specified here so the wildcard path matches every path that is not the welcome or empty path](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=157.256) [preventing navigation to any other path below it. We need to reorder our imports array to ensure that](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=165.256) [our wildcard route is last in the list of route paths. Let's fix that now.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=172.256) [We'll move the AppRoutingModule to the end of our imports array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=179.256) [Let's recheck it in the browser.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=184.256) [And it's working again. The bottom line here is that route path order matters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=187.256) [There is another common syntax you may see when working with routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=193.256) [Some developers prefer to define the route configuration array as a constant and then pass that constant into](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=199.256) [the forRoot or forChild method, like this. Using a constant simplifies the NgModule decorator.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=205.256) [Feel free to use this syntax as desired. Now let's finish up this module with some checklists we can use as](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=214.256) [we route to features defined in feature modules.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=6&mode=live&start=220.256)

[Checklists and Summary](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live)

[Routing to features in feature modules uses a subset of the steps we used to set up our basic application routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=0) [Import the RouterModule in the feature module. Be sure to use RouterModule.forChild so we don't re-register](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=9.713) [the router service. Configure the routes using the same syntax as with our basic application routes,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=17.713) [and pay close attention to the order of the routes especially when the routes are merged from application](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=25.713) [and feature modules.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=31.713) [When naming routes, consider using a common root path name for related feature routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=33.713) [This will make it easier to group routes or use lazy loading. We'll see more on grouping and lazy loading](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=39.713) [later in this course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=46.713) [To activate a route with code, first use an import statement to import the router from the router library.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=49.713) [Then add a dependency on the router service using a constructor parameter.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=55.713) [The Angular injector then provides an instance of the router service.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=60.713) [Use the router service and call the navigate method. And specify the same link parameters array used with](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=65.713) [a RouterLink directive. The first element in the array is the root URL segment.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=71.713) [All other elements are values for the route parameters or additional URL segments.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=76.713) [Or use the router service navigateByUrl method and pass in a string with a full URL path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=83.713) [Consider separating the routes out of the root app.module into their own routing modules.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=90.713) [This keeps the module code more organized, makes the routes easier to find and edit, and provides for](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=96.713) [separation of concerns. Simply build a new Angular module and move the routes to that module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=101.713) [Just take care to watch the order of the feature and routing modules in the AppModule imports array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=108.713) [The routes contained in each module are processed first. All explicitly defined routes are processed last](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=114.713) [regardless of where they are in the imports array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=121.713) [This module was all about routing to features and feature modules.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=125.713) [We saw how to set up routing differently when working with feature modules.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=130.713) [We discussed route path naming strategies and the benefits of giving feature routes a similar route path name.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=134.713) [We walked through how to activate a route with code using the router's navigate or navigateByUrl methods.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=141.713) [We looked at how to access routes from feature modules by defining them in the AppModule paying special close](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=149.713) [attention to the route order. And we refactored our basic application routes into their own routing module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=156.713) [We've now added routes for our Log In and Product List components.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=163.713) [Up next, let's see how to configure routes with parameters so we can route to our Product Detail and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=168) [Product Edit components.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m3&clip=7&mode=live&start=175)

[Route Parameters](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=0&mode=live)

[Introduction](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=0&mode=live)

[Sometimes we need to just pass a little data to our routes. Welcome back to Angular Routing from Pluralsight.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=0&mode=live&start=0) [My name is Deborah Kurata, and in this module, we pass data to our routes with route parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=0&mode=live&start=10) [One component may have data that another component needs. We can swiftly pass that data as we route from](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=0&mode=live&start=17) [one component to the next using route parameters. Parameters are best suited for little bits of data](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=0&mode=live&start=24) [just like a runner would not want to pass a kettle bell, we don't want to use parameters to pass large](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=0&mode=live&start=32) [amounts of data. Instead, we pass simple Id's or keywords.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=0&mode=live&start=37) [As an example, a Product List component displays a list of products, each with a unique product Id.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=0&mode=live&start=42) [To navigate to the Product Detail component, the Product List component must pass the Id of the product to display.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=0&mode=live&start=49) [We pass data as part of the route using route parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=0&mode=live&start=56) [In this module, we start with required parameters. We look at how to configure a route with required](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=0&mode=live&start=61) [parameters, how to populate the required parameters, and how to read the parameters from the route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=0&mode=live&start=68) [We then look at optional parameters and query parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=0&mode=live&start=74) [By the end of this module, we'll have configured the parameters and routed to the Product Detail and Product](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=0&mode=live&start=79) [Edit routes. And we'll improve the Product List route to retain its filter by and image display settings](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=0&mode=live&start=85) [as query parameters. Let's get started.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=0&mode=live&start=92)

[Configuring a Parameterized Route](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live)

[Before we can pass required data on a route, we need to define placeholders for that data as part of the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=2.255) [route configuration. The first route here is our simple route to the ProductListComponent.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=9.255) [The second route uses a similar root path name, products, and adds a placeholder for the data that must be](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=14.255) [passed to that route. The colon here denotes that this is a placeholder for a required parameter.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=22.255) [We name this particular placeholder id because it defines the location in the URL for the Id of the product](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=27.255) [to display in the Product Detail page. But we could name it anything.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=35.255) [The third route uses a similar placeholder because the Product Edit page also needs an Id parameter.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=40.255) [And we add a simple text route segment to distinguish this path from the detail path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=47.255) [So we can see that our path expressions contain two types of URL segments--constant segments that are simple](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=53.255) [text and variable segments that are placeholders for data passed to the route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=60.255) [Each variable segment is identified with a colon.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=66.255) [We can add any number of placeholders to a route definition path as long as each placeholder within a path](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=70.255) [is defined with a unique name. This first route is a simple route to an OrderListComponent.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=76.255) [The second route uses a placeholder for the Id to pass to the OrderDetailComponent.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=82.255) [The third route displays details for one item of an order so it needs a placeholder for the order Id and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=87.255) [for the order item Id. The second placeholder cannot also be named Id, so we named it itemId.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=96.255) [We can continue to add more constant and variable segments to a route definition as needed.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=102.255) [Let's configure some route parameters in our sample application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=108.255) [In our sample application, we need a route parameter to pass the desired product Id to the Product Detail](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=113.255) [and Product Edit routes. We configure the product features routes in the product.module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=120.255) [Here is our current route to the ProductListComponent. Let's add routes for the ProductDetail and ProductEditComponents.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=125.255) [As we discussed earlier in this course, we want to name all of our product feature routes with the same root](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=133.255) [path name, products. So our Product Detail route path is products, and we add /:id for the Id placeholder.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=138.255) [When the user navigates to this route, we want to display the Product Detail page, so we specify the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=148.255) [ProductDetailComponent. For the Product Edit route, we'll use the same root path name and then Id placeholder.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=154.255) [Then we add /edit to make the path unique from the detail route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=162.255) [For this route, we want to display the Product Edit page so we specify the ProductEditComponent.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=167.255) [We can't try this out yet because we don't have user actions tied to our new routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=173.255) [Let's do that next.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=1&mode=live&start=178.255)

[Populating Route Parameters](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live)

[Once we have our routes configured, we define the user actions that activate each route and populate that](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=2.398) [route's parameters. From our Product List page, we activate the Product Detail route when the user clicks](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=8.398) [on the name of a product. To display the details for the selected product, we'll pass the product Id on](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=14.398) [that route as a route parameter. We activate the Product Edit route when the user clicks on the Edit button](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=21.398) [in the product's row. To display the selected product for edit, we'll again pass the product Id on that](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=28.398) [route as a route parameter. We also want to activate the Product Edit route when the user clicks on the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=34.398) [Add Product menu option. In that case, we display the edit page for entry of a new product.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=40.398) [What then do we pass for the product Id route parameter? We can't leave it blank or it won't match our configuration.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=47.398) [We have to pass something. We could select a number that represents a newly created product,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=54.398) [something that would not be a valid existing product Id such as 0 or -1.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=61.398) [The selected number would then indicate a request to add a new product.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=67.398) [For our sample application, we'll specify that an Id of 0 means we are working with a newly created product.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=72.398) [How then do we populate the route parameters?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=79.398) [As we've seen previously in this course, we activate a route on a clickable element at the HTML by binding](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=83.398) [the routerLink directive on that element to a link parameters array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=90.398) [Or we can activate a routing code using the router's navigate method passing in a link parameters array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=94.398) [We populate the route parameters as part of that link parameters array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=101.398) [The link parameters array contains the root URL segment, any route parameters, and any additional constant](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=106.398) [URL segments. In the first example, we route to the ProductDetailComponent when the user clicks on the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=114.398) [productName link. We specify products as the root URL segment and populate the Id placeholder with the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=119.398) [product's Id. In the second example, we route to the ProductEditComponent when the user clicks the Edit button.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=125.398) [We specify products as the root URL segment and populate the Id placeholder with the product Id.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=133.398) [Then we add the edit URL segment to distinguish this route from the Product Detail route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=139.398) [The third example routes to the Product Edit page when the user clicks the Add Product menu option.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=145.398) [This route uses 0 for the product Id parameter indicating a request to add a new product.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=151.398) [Since these array elements are all literal values, no variables, we could instead specify all of the URL](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=158.398) [segments as a simple string using the shortcut syntax. These two router links generate the same URL.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=165.398) [This last example uses the router's navigate method passing in a similar link parameters array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=173.398) [It activates the Product Detail route for the defined product Id.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=179.398) [Let's jump back to the sample application and activate some routes with route parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=2&mode=live&start=183.398)

[Populating Route Parameters: Demo](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live)

[When the user clicks on the product name here on the Product List page, we want to route to the Product](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=2.158) [Detail page for that specific product. That's our first task.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=8.158) [To accomplish that task, we modify the product-list template. Here is the table data element for the productName.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=13.158) [I already have it in an anchor element. We add the routerLink directive to this element.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=20.158) [We bind the routerLink directive to a link parameters array. The first element of that array is the root](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=26.158) [URL segment, which is products. To populate the Id placeholder, we pass the product.id as the second](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=33.158) [element of the array. When the user clicks on this link, the router builds the appropriate URL based on the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=39.158) [contents of this array and navigates to the Product Detail page. Let's check it out in the browser.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=46.158) [Click on a product to view its detail. The URL is correctly created with the appropriate Id, but we don't](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=52.158) [see the product detail. Well, that's because we are not yet reading the parameter value from the URL](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=61.158) [and getting the appropriate product to display. Before we do that, let's hook up the Product Edit routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=66.158) [In the product-list template, the Edit button is here. Again we add the routerLink directive and bind it to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=72.158) [a link parameters array. The first element of that array is again the root URL segment, which is products.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=80.158) [To populate the Id placeholder, we again pass the product.id in as a second element of the array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=87.158) [And we set the edit URL segment as the third element of the array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=93.158) [That distinguishes this route from the Product Detail route. When the user clicks the Edit button,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=98.158) [the router will build the appropriate URL based on the contents of this array and navigate to the Product Edit page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=103.158) [Before we give it a try, let's hook up the Add Product menu option as well.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=111.158) [The application menu is defined in the app.component template. We add the routerLink directive to the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=115.158) [Add Product menu option and bind it to a link parameters array. Here we populate the Id placeholder with 0](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=121.158) [to indicate that we want to display the edit page initialized for a new product.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=129.158) [Let's check it out in the browser.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=133.158) [On the Product List page, click Edit for one of the products.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=136.158) [We see the correct URL, and the page header shows us we are on the right page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=140.158) [But we don't see the edit form. That's again because we are not yet reading the parameter value from the URL](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=145.158) [and getting the appropriate product to display for editing. Now trying selecting the Add Product menu option.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=152.158) [We see the URL with the parameter of 0 but still no edit form. Sounds like our next step is to read the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=159.158) [parameter values from the URL so we can correctly display these pages.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=165.158) [Before we move on, though, I want to call out something that just happened here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=170.158) [To maximize performance and minimize changes to the browser's document object model or DOM, the router reuses](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=175.158) [a component and its template if only the parameters of the route change.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=182.158) [Let's see that again. Click on Product List,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=187.158) [and our URL changes to products. This activates the ProductListComponent and displays its template in the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=190.158) [primary outlet. Click on Edit,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=196.158) [and our URL changes to products/Id/edit. This activates the ProductEditComponent and displays its template](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=199.158) [or at least as much as it can in the primary outlet. If I then click on Add Product, only the route](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=208.158) [parameter changes so the ProductEditComponent and its template are reused and not reinitialized.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=215.158) [This feature of component reuse is important to understand as we reroute parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=222.158) [Let's do that next.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=3&mode=live&start=228.158)

[Reading Route Parameters: Snapshot](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=4&mode=live)

[As we saw in the last demo, passing parameters as part of a route doesn't help much unless the activated](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=4&mode=live&start=2.876) [component reads and uses those parameters. The router extracts any route parameters from the URL and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=4&mode=live&start=8.876) [supplies them to the component through its ActivatedRoute service.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=4&mode=live&start=15.876) [The ActivatedRoute service is basically a one-stop shop for route information.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=4&mode=live&start=20.876) [It provides access to the set of URL segments, route parameters, query parameters, route data, and more as](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=4&mode=live&start=25.876) [we'll see later in this course. Any component instantiated by the router can inject its ActivatedRoute service.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=4&mode=live&start=32.876) [We inject the ActivatedRoute service into our component just like every other service on the constructor.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=4&mode=live&start=39.876) [Once it's injected, there are two basic ways to use the ActivatedRoute service to read route parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=4&mode=live&start=45.876) [The ActivatedRoute service has a snapshot that provides the initial state of the route including the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=4&mode=live&start=51.876) [initial value of the route parameters. We access the params array of that snapshot to easily retrieve a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=4&mode=live&start=58.876) [parameter value from the route in one line of code using the placeholder name.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=4&mode=live&start=65.876) [The ActivatedRoute service also provides a params observable. The observable keeps watch on the parameters](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=4&mode=live&start=70.876) [and receives a notification every time the parameters change. So we can obtain the initial value of the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=4&mode=live&start=78.876) [parameters from the static params array that is part of the snapshot or get notified of parameter changes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=4&mode=live&start=85.876) [by subscribing to the params observable. For now, let's check out the snapshot approach.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=4&mode=live&start=92.876) [For reference, here is the route configuration for the Product Detail route we configured earlier in this module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=4&mode=live&start=98.876) [And here is the code we just added to activate the Product Detail route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=4&mode=live&start=105.876) [To get the parameter from the URL, the ProductDetailComponent uses the snapshot provided by the ActivatedRoute service.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=4&mode=live&start=109.876) [We import the ActivatedRoute using an import statement, and we want an instance of that service so we define](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=4&mode=live&start=118.876) [it as a dependency in our constructor. We use the instance of the ActivatedRoute service to access the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=4&mode=live&start=124.876) [snapshot and retrieve the desired parameter from the params array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=4&mode=live&start=130.876) [We access the appropriate params array element using the placeholder name.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=4&mode=live&start=135.876) [The name must exactly match including casing. Let's give this a try.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=4&mode=live&start=140.876)

[Reading Route Parameters: Snapshot Demo](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live)

[When we route from the Product List page to the Product Detail page, we pass the Id of the desired product](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=2) [on the route as a route parameter. The ProductDetailComponent needs to read the Id from the route parameter](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=7.976) [and display the data for the selected product. To read the route parameter using the ActivatedRoute](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=15.976) [snapshot, we first import the ActivatedRoute service from the @angular/router library.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=21.976) [We then add the ActivatedRoute as a dependency on the constructor.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=27.976) [When this component is created, the Angular injector injects an instance of the ActivatedRoute service](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=32.976) [into this component. Before we write the code to use this snapshot, we need to think a moment about where](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=38.976) [to put that code. In the slides, we added the code to read the parameter to the constructor.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=44.976) [But in our sample code, once we get that Id, we'll call this local getProduct method to retrieve the product.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=49.976) [We don't normally want to execute asynchronous operations such as getting data from a server in the constructor.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=56.976) [Let's instead use the OnInit lifecycle hook. The OnInit lifecycle hook is executed when the component is initialized.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=64.976) [It's a great place to perform operations such as getting the data needed for the component.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=71.976) [to use the OnInit lifecycle hook, we import the OnInit from @angular/core, so we add it to the import](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=77.976) [statement here. We then add the implements keyword to our class declaration to implement the OnInit interface.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=84.976) [Lastly, we write the ngOnInit method. I'll paste the code, and we can talk through it.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=91.976) [In the ngOnInit method, we retrieve the Id from the snapshot. We use the ActivatedRoute instance,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=97.976) [access the snapshot, and retrieve the Id parameter from the params array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=105.976) [Since the URL is a string, we add a plus sign at the front to cast to a number.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=110.976) [This gives us a numeric product Id. If we hover over the snapshot, we see that it's type is ActivatedRouteSnapshot.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=115.976) [We'll use this type directly a little later in this course. We then call the local getProduct method and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=124.976) [pass in that product Id. The local getProduct method calls the getProduct method in our product data service](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=132.976) [to get the product using HTTP. Since this course is on routing, not data access, I've provided all of the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=138.976) [data access code as part of the starter files. To see a walkthrough of the data access code and Angular's](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=146.976) [HTTP features, check out the Angular 2: Getting Started or Angular 2: Reactive Forms course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=152.976) [Now let's check the result in the browser. On the Product List page, click on a product.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=161.976) [And there is our Product Detail page. Yay! The URL shows our root URL segment and our route parameter.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=166.976) [The ProductDetailComponent accessed the ActivatedRoute service snapshot to read that parameter and used it](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=174.976) [to retrieve the requested product for display. What about these two buttons here?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=181.976) [Use what you've learned previously in this course to implement the Back and Edit buttons on this page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=186.976) [The Back button should activate the product's route and navigate back to the Product List page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=191.976) [The Edit button should activate the Product Edit route passing the Id of the product to edit.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=197.976) [Pause the video now if you want to give it a try. Are you ready to see my solution?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=202.976) [In the product-detail template, I added the routerLink directive to the Back button here and defined the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=208.976) [link parameters array to activate the products route. And I added routerLink to the Edit button specifying](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=214.976) [the link parameters array to activate the product edit route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=221.976) [Trying it out in the browser, the Back button returns to the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=224.976) [Product List page. Viewing the Product Detail again](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=229.976) [and clicking the Edit button,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=233.976) [we see only the partial display of the Product Edit page. To see the Product Edit page, we need to read](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=235.976) [the product Id from the route in the ProductEditComponent using the ActivatedRoute service just as we did](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=241.976) [in the ProductDetailComponent. Want to give it a try? If so, pause the video here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=248.976) [Ready for my solution? I imported OnInit and ActivatedRoute. I implemented the OnInit interface.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=254.976) [I defined the ActivatedRoute service as a dependency on the constructor.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=263.976) [And I added the exact same ngOnInit method code. Is that similar to what you did?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=267.976) [Let's check it out in the browser.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=274.976) [Click Home to start fresh. Select Add Product.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=275.976) [And it works! Our edit page displays for entry of a new product. Yay!](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=280.976) [Now let's try editing an existing product. On the Product List page,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=285.976) [click on an Edit button,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=289.976) [and there is our Product Edit page. Cool! It displays the current values of our product and allows the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=291.976) [user to update the values. But what if we click on Add Product now?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=298.976) [The route parameters change in the URL, but our edit form does not change for entry of a new product.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=303.976) [If we look again at the code, we see why. We retrieve the Id from the ActivatedRoute service snapshot in](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=310.976) [the ngOnInit method when the component is initialized. But as we discussed earlier in this module,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=317.976) [if only the parameters of the URL change, the component is not initialized again.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=324.976) [So the ngOnInit method is not executed again, and we don't get a new initialized product.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=330.976) [How do we handle a change in parameters? Instead of reading the route parameters from a snapshot,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=337.976) [we watch for parameter changes using an observable.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=5&mode=live&start=343.976)

[Reading Route Parameters: Observable](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live)

[For reference, here again is the route configuration for the Product Edit route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=2.471) [And here is the code we added to activate the Product Edit route. To display the appropriate product in](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=7.471) [the Product Edit page, the ProductEditComponent needs the parameter from the URL.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=13.471) [But because the route parameters can change without reinitializing the component, we can't just read the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=19.471) [initial state of the route parameter from the ActivatedRoute snapshot.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=25.471) [Instead, we watch for parameter change notifications and re-get the product Id every time the parameter changes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=29.471) [We watch for events using observables. You can think of an observable as an object that provides a stream](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=36.471) [of events or notifications that occur over time. Luckily for us, the ActivatedRoute exposes a route params](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=43.471) [property as an observable. Here we subscribe to that observable to start receiving its notifications.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=50.471) [We pass to the subscribe method a function defining the action to take each time a notification is received.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=57.471) [The first time this route is activated and every time a route parameter changes without activating a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=64.471) [different component, the observable emits a notification and passes in the current set of parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=71.471) [We access the appropriate params array element using the exact placeholder name.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=77.471) [Let's give this a try.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=83.471) [We set up the observable when the component is initialized in the ngOnInit method.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=85.471) [In ngOnInit, we'll delete the lines that read the Id from the route snapshot.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=91.471) [Instead, we'll access the ActivatedRoute service params observable.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=97.471) [I'll paste the code, and we can talk through it. Hovering over params, we see that it does indeed return](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=101.471) [an observable. We subscribe to this observable so we are notified when the parameters change.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=107.471) [The notification provides the current set of parameters, which we'll hold in a params variable.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=113.471) [We use the arrow syntax to define the code that should execute each time we receive a notification.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=120.471) [Here we retrieve the Id parameter from the passed-in params array. Since the URL's a string, we add a plus](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=125.471) [sign at the front to cast to a number. Then we call the local getProduct method passing in that Id.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=133.471) [This local getProduct method calls the product data service getProduct method to get the product data for](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=139.471) [display in the edit page. Since this call is within our subscribe, this call will be re-executed each time](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=145.471) [the parameters change. Before we give this a try, let's take a peek at the product data service getProduct method.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=153.471) [Right-click on the method and select Go to Definition.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=160.471) [That takes us to the product data service. Here we see special code to handle the Id of 0.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=164.471) [If we pass in an Id of 0 to add a new product, this code returns an initializeProduct.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=171.471) [Otherwise, it uses HTTP and gets the requested product from a back-end server.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=177.471) [Now let's check out the result in the browser.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=183.471) [Select Product List.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=185.471) [Click the Edit button.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=187.471) [And we see the edit page populated with the existing product data.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=189.471) [The user can then edit this information as desired. Click Add Product, and the edit form is initialized](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=193.471) [for entry of a new product. It works! To finish up the features here, hook up the Cancel button to route](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=200.471) [back to the Product List page and add code for the Save and Delete buttons to route back to the Product List](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=207.471) [page after the Save or Delete is complete. Pause the video now if you want to give it a try.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=213.471) [Ready for my solution? The code to activate the route from the Cancel button is in the product-edit template.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=218.471) [It now routes to the products route and displays the Product List page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=225.471) [Notice that the Save and Delete buttons call methods to save or delete the product.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=230.471) [Those methods handle the operation, saving or deleting. Each method then calls onSaveComplete to complete](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=234.471) [the operation. I added the navigation there. This code uses the router service, which must be imported](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=241.471) [here, added as a dependency here, and then used here. Let's ensure the buttons work.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=250.471) [From the Product List, select Edit,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=259.471) [Cancel,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=262.471) [and we return to the Product List page. Select Edit again.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=263.471) [This time let's update the product name and save.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=267.471) [The change is saved, and we are returned to the Product List page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=271.471) [We can see the change here. Select Edit one more time,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=275.471) [click Delete, confirm the delete,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=278.471) [and we are returned to the Product List page. The deleted product no longer appears.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=282.471) [It all works! Note that we are using an in-memory data service.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=288.471) [So if you stop this server and restart with npm start or refresh the page,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=293.471) [the product data is returned to its original values.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=299.471) [Let's review when to read route parameters using a snapshot versus when to use an observable.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=304.471) [Use the snapshot if you only need to read the initial value of the parameters and those parameters won't](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=310.471) [change once the component is activated. The code can simply read a parameter from the activated route](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=315.471) [service snapshot params array. If you expect the parameters to change without navigating to another](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=322.471) [component, you can watch for parameter changes using an observable.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=328.471) [Subscribe to the ActivatedRoute service params observable, and each time the parameters change,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=332.471) [read the parameters from the provided params array. Note that when subscribing to an observable in a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=338.471) [component, you almost always add code to unsubscribe when the component is destroyed.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=343.471) [There are a few exceptional observables where this is not necessary including the activated route](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=349.471) [observable used here. The router destroys the routed component when it is no longer needed, and the injected](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=356.471) [ActivatedRoute service and its associated observables are destroyed with it.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=362.471) [So no need to unsubscribe. Angular 4 introduced another ActivatedRoute interface called paramMap.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=367.471) [When using the snapshot, we call the paramMap.get method to get a parameter value.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=376.471) [Since this is a method, we use parentheses and place in the placeholder name.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=381.471) [Or if the parameter has multiple values, we use getAll to get the values as an array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=387.471) [When using the observable approach, we subscribe to the paramMap observable, then call the get or getAll](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=392.471) [method passing in the placeholder name. Feel free to use this syntax instead if you are using Angular](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=400) [version 4 or higher. But what if we want optional route parameters? How do we handle those?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=6&mode=live&start=405)

[Defining Optional Route Parameters](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live)

[Say you have a search page for the user to select search criteria such as a product name, product code,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=2.089) [or an availability date. You want to pass these criteria to a list page to display a list of products that](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=8.089) [match the defined search criteria. To accomplish this, we could use required route parameters as we've done](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=14.089) [previously in this module. We'd add a placeholder in the path for each criterion and populate these](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=21.089) [placeholders when activating the route. The resulting URL would look something like this.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=27.089) [These odd-looking values are an encoding of the spaces and commas in the dates.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=33.089) [The ProductListComponent would then read the parameters and display the products that match the selections.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=38.089) [There are several issues with using required parameters in this scenario.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=44.089) [Complex URLs such as this complicate the pattern matching required to translate the URL segments into a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=48.089) [path in the route configuration. And every time we add more search criteria, we need to change the route](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=55.089) [configuration and everywhere that route is used. Another issue is lack of any naming.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=62.089) [And looking at the parameters in the URL, it's difficult to tell what each of these parameters means](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=68.089) [unless we look at the route configuration. We could add constant route segments in between each of these](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=74.089) [identifying what each parameter is, but that makes the path even longer, more complex, and more difficult](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=80.089) [to get right. And sometimes some of these parameters, such as the date range, may not be set by the user](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=87.089) [based on their particular search. So what do we pass when a particular parameter is not specified?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=93.089) [Instead of using required route parameters, a better option for this scenario is to use optional route parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=100.089) [Optional route parameters make it easier to pass optional or more complex information as part of the route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=108.089) [This optional information is not part of the configuration. This means that it is not involved in matching](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=114.089) [route paths for navigation. And if we add more optional parameters over time, we don't affect the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=122.089) [application routing. Optional parameters are defined in the link parameters array as a set of key and value pairs.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=128.089) [Notice the syntax here. We define the optional parameters within curly braces in one array element.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=136.089) [We specify a key and a value for each optional parameter. We can easily add more optional parameters without](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=143.089) [affecting the route configuration. We can easily tell which parameter is which because each has a unique key.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=150.089) [And if a parameter is not set, we can leave off its key and value.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=158.089) [Note that any optional parameters must always be after any required route parameters in the link parameters array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=164.089) [The resulting URL looks like this. Notice the syntax. The optional parameters are listed with their keys](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=171.089) [and values separated by semicolons. Again, the spaces and commas are encoded.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=179.089) [Optional parameters are the ideal choice when conveying arbitrary or complex information during navigation.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=184.089) [Optional parameters are route specific and scoped to the particular URL segment so there is no risk of](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=191.089) [key name collisions with other URL segments. And like required parameters, they are not retained when](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=198.089) [navigating to other components. How do we read these parameters in our components?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=205.089) [We read optional parameters the same way we read normal parameters--using the ActivatedRoute service snapshot](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=210.089) [or an observable. In our example search page, we route to the ProductListComponent and pass along all of the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=217.089) [user defined search criteria as optional parameters. Just like reading required parameters, we use the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=224.089) [snapshot provided by the ActivatedRoute service. We import the ActivatedRoute using an import statement](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=232.089) [and define it as a dependency in our constructor. Where we need the values, we retrieve the parameters from](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=238.089) [the snapshot by their key name. I have not implemented a search page in the sample application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=244.089) [Feel free to add one and try out optional parameters at your leisure.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=250.089) [Now we've covered required parameters and optional parameters, but there is one more kind of route](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=254.089) [parameter--query parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=7&mode=live&start=259.089)

[Defining Query Parameters](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live)

[Here again is our Product List page. If we enter a filter string, the products are filtered by the specified string.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=2.556) [And if we click Show Image, the product images appear. But if we then click to view a product's detail,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=10.556) [then use the Back button](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=17.556) [to return to the Product List page, our selections are gone.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=19.556) [It would be a much nicer user experience if we retained the user settings when navigating to the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=24.556) [ProductDetailComponent and return those settings when the user navigates back to the ProductListComponent.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=30.556) [To define parameters that work across multiple routes to the Product Detail page and back again, for example,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=36.556) [we use query parameters. Just like optional parameters, we use query parameters to pass optimal or complex information.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=43.556) [Unlike optional parameters, they can be retained across routing paths.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=52.556) [And notice the syntax here. They look like classic query parameters with a question mark at the beginning](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=56.556) [and an ampersand separator.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=63.556) [Like optional parameters, query parameters are not part of the route configuration and are not involved with](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=66.556) [matching route paths. We populate required and optional parameters by adding them to the link parameters array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=73.556) [Not so with query parameters. We pass them separately. When populating query parameters in the HTML,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=80.556) [we use the queryParams directive and bind it to a set of key and value pairs, one for each query parameter.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=88.556) [When routing in code, we add a second argument to the navigate method outside of the link parameters array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=96.556) [That argument is a set of options defined with key and value pairs.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=102.556) [To populate the query parameters, we define a key of queryParams and set its value to an object defining](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=107.556) [the query parameters. Unlike optional parameters, query parameters are not scoped to the route so care must](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=115.556) [be taken to ensure there are no key name collisions. Let's try it out in our sample application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=122.556) [Our task is to retain the user settings on the Product List page so that when the user navigates to the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=130.556) [Product Detail page and returns, the settings are reapplied. So here in the product-list template,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=136.556) [we add the queryParams directive and define a set of key and value pairs.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=142.556) [We'll call the first key filterBy and set its value to the current listFilter property.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=148.556) [The listFilter property is bound to the filterBy input element and will contain the user's current filter](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=154.556) [selection. We'll call the second key showImage and set its value to the current showImage property.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=160.556) [The showImage property tracks whether or not the user selected to show the product image.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=168.556) [We can define any number of query parameters here, but two will do for our purposes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=173.556) [Let's check out the result in the browser.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=179.556) [Enter a filter and click Show Image.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=182.556) [Then click on a product name to view the product details. The URL now includes what looks like a classic](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=185.556) [set of query parameters with a question mark before the first query parameter and an ampersand separating](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=192.556) [the parameters. Click Back,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=198.556) [and the parameters are lost. Didn't we just say that query parameters can be retained across paths?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=201.556) [Well, yes, yes they can, but not by default.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=208.556) [By default, the router resets the query parameters during navigation.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=214.556) [However, we can elect to preserve the current query parameters using the preserveQueryParams directive in](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=219.556) [the HTML or the preserveQueryParams key when navigating in code.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=225.556) [In Angular version 4, preserveQueryParams was deprecated and replaced with queryParamsHandling,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=232.556) [which provides more flexibility. We can set queryParamsHandling to preserve to preserve the current query](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=239.556) [parameters or merge to merge the query parameters with any existing query parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=246.556) [In the upcoming demo, I'll use preserveQueryParams. If you are using Angular version 4 or higher,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=252.556) [you can try out the queryParamsHandling instead.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=259.556) [Where do we specify this preserveQueryParams? Well in our example, we want the parameters retained when we](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=265.556) [navigate back from the ProductDetailComponent so that's where we'll add it.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=271.556) [Here is the Back button in the product.detail template. We specify the preserveQueryParams directive and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=277.556) [set it to true. Now when the user clicks the Back button, any query parameters passed to the Product Detail](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=282.556) [page are retained and passed back to the Product List page. Let's see how that changes things.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=289.556) [Again, enter a filter and click Show Image. Click on a product](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=296.556) [name to view the product details. The URL has our query parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=300.556) [Click Back, and the parameters are still here. Yay! But the page does not yet read these parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=305.556) [Let's do that next.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=8&mode=live&start=312.556)

[Reading Query Parameters](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=9&mode=live)

[Reading query parameters is much like reading required or optional parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=9&mode=live&start=2.266) [We use the ActivatedRoute service and the snapshot or observable approach.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=9&mode=live&start=7.266) [For reference, here is the definition of our query parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=9&mode=live&start=12.266) [To read the query parameters, we begin by importing the ActivatedRoute service using an import statement.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=9&mode=live&start=16.266) [And we want an instance of that service so we define it as a dependency on our constructor.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=9&mode=live&start=22.266) [Here we use the snapshot approach and access the queryParams array instead of the params array as we do](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=9&mode=live&start=27.266) [with the required and optional parameters. That's really the only difference when reading query parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=9&mode=live&start=34.266) [Let's give it a try.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=9&mode=live&start=40.266) [We want the ProductListComponent to read the query parameters passed back on the route from the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=9&mode=live&start=43.266) [ProductDetailComponent and set the filter and showImage properties based on those parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=9&mode=live&start=48.266) [We first import the ActivatedRoute service from the @angular/router library.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=9&mode=live&start=54.266) [We then define the ActivatedRoute as a dependency on the constructor.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=9&mode=live&start=60.266) [The OnInit interface is already implemented. We add code here to read the query parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=9&mode=live&start=64.266) [We'll set the listFilter property to the query parameter value using the ActivatedRoute service snapshot.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=9&mode=live&start=71.266) [We'll pull the values from the queryParams array using the query parameter key, which is filterBy in our example.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=9&mode=live&start=78.266) [We add an orEmpty string to handle the case when the filterBy is undefined such as when this page is first displayed.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=9&mode=live&start=84.266) [We set the showImage property to true if the showImage query parameter is true.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=9&mode=live&start=93.266) [But recall that our parameters are always strings, so we need to check for a string value of true.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=9&mode=live&start=99.266) [That's it. Let's check out the result in the browser. Once again,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=9&mode=live&start=106.266) [enter a filter and click Show Image. Click on a product name](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=9&mode=live&start=111.266) [to view the product details. The URL has our query parameters. Click Back,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=9&mode=live&start=114.266) [and the Product List displays with our prior settings. Success!](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=9&mode=live&start=120.266) [Let's finish up this module with some checklists you can use as you define required, optional, and query parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=9&mode=live&start=125.266)

[Checklists and Summary](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live)

[Required parameters pass needed data on a route. Use a required parameter whenever the route component](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=2.857) [must have the parameter value. For example, a detail route must have an Id or it won't know which item to display.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=8.857) [Configure required parameters using a placeholder denoted with a colon and a logical placeholder name.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=17.857) [Populate required parameters by specifying them as elements in the link parameters array either assigned](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=23.857) [to the routerLink directive or passed into the router's navigate method.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=29.857) [Read the parameters from the route using the ActivatedRoute service snapshot or its params observable.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=34.857) [The array element key is the placeholder name.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=42.857) [Use an optional parameter when the value is truly optional or if it is complex or requires multiple values.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=46.857) [For example, an advanced search component passes any number of search criteria as optional parameters to a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=53.857) [list component so it can filter the data in the list.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=59.857) [Optional parameters are not configured as part of the route configuration.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=63.857) [Populate optional parameters using a set of key and value pairs in one element of the link parameters array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=68.857) [They must be specified after all required parameters in the array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=75.857) [Just like required parameters, read optional parameters from the route using the ActivatedRoute service](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=80.857) [snapshot or its params observable. The array element key is the key from the key and value pair.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=86.857) [Be sure to match the casing of the keys.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=94.857) [Use a query parameter to pass optional or complex information to a route when you want that information](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=97.857) [optionally retained across routes. For example, a list component provides user selections such as a filterBy string.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=103.857) [The list component passes those user selections to the detail component so it can pass them back when](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=111.857) [returning to the list component.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=117.857) [Query parameters are not configured as part of the route configuration.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=119.857) [Populate query parameters using the queryParams directive and specifying a set of key and value pairs or by](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=124.857) [passing an object to the router's navigate method and setting the query params to the set of key and value pairs.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=131.857) [Read the parameters from the route using the ActivatedRoute service snapshot queryParams array or its](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=138.857) [queryParams observable. The array element key is the key from the key and value pair.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=145.857) [Be sure to match the casing of the keys.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=151.857) [This module was all about route parameters. We began with required parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=154.857) [We looked at how to configure, populate, and read required parameters using the ActivatedRoute service](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=160.857) [snapshot or its observables. Next we looked at how to populate and read optional parameters and query parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=166.857) [Now that we've examined how to pass data to our routes, we have our Product Detail and Product Edit routes in place.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=175.857) [And we improved the Product List route to retain its filter and image display settings as query parameters](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=182.857) [when navigating back from the Product Detail route. Next up, let's see how to prefetch our application data](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=189.857) [using resolve.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m4&clip=10&mode=live&start=196.5)

[Prefetching Data Using Route Resolvers](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live)

[Introduction](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live)

[For a more user friendly experience, we can retrieve the data required for display in a component's template](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live&start=0) [before routing to that component. Welcome back to Angular Routing from Pluralsight.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live&start=9) [My name is Deborah Kurata, and in this module, we prefetch data for our component's using route resolvers.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live&start=15) [When a user navigates to a component with a template that requires data, the page may first appear mostly](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live&start=24) [blank or with only the static text making a less-than-ideal user experience.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live&start=32) [This happens due to a delay in downloading the data required for the page especially if the user has a slower](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live&start=39) [connection or there is lots of data. We can prevent this partial page display by downloading the data for](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live&start=46) [a component before routing to that component with route resolvers.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live&start=54) [There are several benefits to prefetching the data for our components.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live&start=60) [Prefetching prevents display of a blank or partial page while waiting for data to be retrieved.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live&start=64) [Prefetching allows us to reuse more code. In most applications, we are already reusing the data retrieval](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live&start=71) [code because we encapsulate it in a data access service. By prefetching the data for a set of routes,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live&start=80) [we can also reuse the code that calls the data access service. We'll see that later in this module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live&start=87) [And we can handle data retrieval errors before routing to the component.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live&start=94) [For example, there is no point in navigating to a detail page if the Id of the item to display is not found.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live&start=100) [We can instead navigate to an error page or stay on the list page for selection of a different item.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live&start=108) [In this module, we start by listing the numerous ways we can provide data with a route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live&start=116) [Then we outline the steps for using a route resolver to prefetch data.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live&start=122) [We discover how to create a route resolver and examine the syntax for adding the resolver to a route configuration.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live&start=127) [We walk through how to obtain the resolver data in the routed component using a snapshot and using an observable.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live&start=135) [We aren't adding any new routes in this module. Rather, we'll modify the Product Detail and Product Edit](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live&start=144) [routes to prefetch their product data. We could also prefetch the data for the Product List route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live&start=151) [But we'll leave it as is for comparison and just focus on these two.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live&start=159) [Let's get started.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=0&mode=live&start=164)

[Providing Data with a Route](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live)

[There are often times we need to provide data with a route. There are multiple ways we can achieve this task.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=2.679) [The best way in any particular scenario depends on the amount of data, the scope of data sharing,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=9.679) [and how that data is used.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=16.679) [Route parameters are great when a route requires a small amount of data in order to display its content](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=19.5) [such as an Id to display a detail or edit page. We saw how to work with route parameters in the last module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=27.679) [Optional route parameters are useful when we need to pass optional, complex, or multifaceted data from one](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=34.679) [route to another. In the last module, we saw how to use optional parameters to pass search criteria to a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=42.679) [list page that filters by that criteria. Query parameters allow us to obtain small amounts of data between routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=49.679) [A search or list page can retain its selections when navigating to its detail page and back again,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=58.679) [as we saw in the last module. A route has a data property we can use to pass a fixed object to a route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=65.679) [We'll talk more about that in a moment. The router can call a resolver service that can prefetch dynamic](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=72.679) [data for a component. So instead of routing to a component and displaying a partial page as it gets its](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=80.679) [data, the resolver can get the data first and then route to the component.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=87.679) [The component template then has the data it needs to display fully.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=94.679) [We'll see how to build and use resolvers in this module. Another option for sharing data is to use an Angular service.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=100.679) [A service is independent from any route so we can use it to share data with any components at any level of](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=107.679) [the route hierarchy. We'll see how to use a service to share data later in this course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=115.679) [For now, let's take a quick look at the route's data property. We'll spend the rest of this module on route resolvers.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=122.679) [A route definition has a data property. We use it to provide any arbitrary data to a route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=130.679) [We pass it an object specifying a set of key and value pairs where the key is a logical name for the data](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=138.679) [and the value is the data itself. The data defined in the data property cannot change throughout the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=145.679) [lifetime of the application. So we use it for static data such as the pageTitle here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=153.679) [To read the data property, we use the same ActivatedRoute service we worked with in the last module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=160.679) [We then use it's snapshot to access the data array using the key to retrieve a specific data element.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=168.679) [Note that we would not use an observable here because by definition the data property is static and won't change.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=175.679) [So there is no need to watch for changes. We'll use this data property later in this course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=183.679) [For now let's focus on building a route resolver to prefetch dynamic data.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=1&mode=live&start=189.679)

[Using a Route Resolver](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live)

[Here is a common flow for a detail page without a router resolver.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=1.902) [On a list page, the user clicks on an item and activates the detail component's route passing the Id of the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=6.902) [item to retrieve. The component is activated, and its template displays but does not yet have the data it needs.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=14.902) [So it is only partially populated. A component's ngOnInit method executes and requests the item from the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=23.902) [back-end web server via HTTP. While the app is waiting for the data, the user sees a partially completed page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=32.902) [When that data arrives from the server, the remainder of the page is populated.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=40.902) [With the route resolver service, the flow is a bit different, but it starts the same.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=46.902) [On a list page, the user clicks on an item and activates the detail component's route passing the Id of the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=52.902) [item to retrieve. The router executes the route resolver, which requests the item from the back-end web](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=59.902) [server via HTTP. When the data arrives from the server, then a component is activated, and the component's](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=66.902) [template displays fully populated using the data obtained by the resolver.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=75.902) [The key difference between the two approaches is who gets the data and when.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=80.902) [Without a route resolver, the component class gets the data after it's initialized.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=85.902) [With a route resolver, the resolver service gets the data so the template is not displayed until it has](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=91.902) [the data it needs. This provides a much cleaner visual appearance and, hence, a much nicer user experience.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=98.902) [Prefetching data for a route using a route resolver requires three steps.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=107.902) [First, we build and register a route resolver. A route resolver is a custom Angular service that retrieves](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=113.902) [the data needed for our component. Next, we add the resolve property to the route configuration and assign](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=121.902) [it to the route resolver. The router executes the resolver service and retrieves the data before activating](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=128.902) [the associated component. Since the route now has the data that our component needs, we modify the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=136.902) [associated component to read its data from the ActivatedRoute service similar to how we retrieve route parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=143.902) [We read that data using a snapshot or an observable.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=151.902) [We use a snapshot to read the data from the ActivatedRoute service if we don't expect the resolver to get](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=156.902) [different data while staying on the current page. For example, when displaying the Product Detail route,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=163.902) [the resolver will get the data for the specific product to display and won't get data for another product](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=170.902) [until the user first navigates back to the list page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=177.902) [We use an observable to read the data from the ActivatedRoute service if it possible for the resolver to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=182.902) [get different data while staying on the current page. For example, when displaying the Product Edit page](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=189.902) [for a specific product, the user could select the Add Product option.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=195.902) [This will cause the route resolver to get an initialized product, thereby changing the resolver data.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=201.902) [In this case, we use an observable to read the data from the Activated Route service.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=208.902) [Now that we know the basic process, let's jump into the first step--creating a route resolver.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=2&mode=live&start=212.902)

[Building a Route Resolver Service](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live)

[A route resolver is often created as an Angular service similar to how we create any other Angular service.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=2.035) [We export a class, decorate it with the injectable decorator, and import what we need to make this service](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=10.035) [behave as a route resolver. We implement the Resolve interface. Notice that this interface is generic.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=16.035) [That is to say that it uses a generic parameter to define the type of data it retrieves.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=25.035) [And since we want the resolver to retrieve products, we import our product interface.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=31.035) [When we implement the Resolve interface, we much define a resolve method.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=37.035) [That method takes in an ActivatedRouteSnapshot and a RouterStateSnapshot.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=42.035) [The ActivatedRouteSnapshot contains information about the currently activated route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=47.035) [We used this snapshot earlier to retrieve route and query parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=53.035) [The RouterStateSnapshot represents the state of the application's router at a moment in time.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=58.035) [The RouterStateSnapshot is a tree of ActivatedRoute snapshots. The Resolve method can return an observable,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=65.035) [a promise, or just data. In this example, we return an Observable of Product using the IProduct interface.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=74.035) [Code in the Resolve method gets the desired data and returns it to the route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=83.035) [When the route is activated, the router calls this Resolve method and waits for the observable to complete](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=88.035) [before activating the associated component. Let's give this a try.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=95.035) [We begin by creating a new file for our route resolver service. Since it will retrieve product data,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=102.035) [we'll add it to the products folder. We'll call it product-resolver.service.ts.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=108.035) [I'll start the code by pasting the import statements that we know we're going to need.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=116.035) [We export a class we'll call ProductResolver. Since this is a service, we'll decorate it with the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=122.035) [injectable decorator. We want this service to be a route resolver so we implement Resolve.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=128.035) [We set the Resolve generic parameter to our Product interface requesting that this resolver return a single product.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=135.035) [We see a syntax error here because we have not yet implemented the method that the Resolve interface requires.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=143.035) [I'll paste it in. This method takes in the ActivatedRouteSnapshot and RouterStateSnapshot and returns an](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=151.035) [Observable of IProduct to return a single product. The only thing left to do here is to actually retrieve](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=159.035) [and return a product. We are adding this route resolver to prefetch data for the Product Detail and Product](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=167.035) [Edit routes. So the first thing we need is to retrieve the route parameter from the route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=175.035) [We use the ActivatedRouteSnapshot just like we did in the last module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=180.035) [We access its params array and request the Id parameter. We add a plus sign at the beginning to convert](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=186.035) [the return string to a numeric Id. Then we use the existing product data service, which uses HTTP to get](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=193.035) [the product data. We first need to import this service, then we ask the injector to inject the instance of](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=201.035) [that service into this service using the constructor. Now we can modify the return statement to call our](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=208.035) [productService passing in the Id of the product to get. Since this resolver is a service, we also need to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=215.035) [register it in an Angular module. We'll do that in the next clip.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=224.035) [This resolver service can be used any time we want to prefetch a single product where the Id for that](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=229.035) [product is provided as an Id parameter on the route. Since both our ProductDetailComponent route and our](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=235.035) [ProductEditComponent route specify an Id parameter, they can both use this resolver.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=242.035) [One of the stated benefits of using a resolver is to handle data retrieval errors before routing to the component.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=249.035) [So let's add some exception handling here. Often exception handling is not shown in demo applications](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=257.035) [because that makes the code more complex. Feel free to skip typing in all of this error handling and use the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=265.035) [route resolver service as it is now. I'll paste in the changes, and we can talk through them.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=272.035) [When thinking about exception handling for this code, the first thing that could go wrong is an invalid](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=278.035) [product Id. We check whether the Id read from the route parameter is actually a number.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=283.035) [If not, we log the issue and navigate back to the Product List page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=290.035) [We then return an observable that is null. Notice that our logging here is just to the console.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=295.035) [In a real application, we'd have a more formal logging mechanism.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=302.035) [Notice that we don't use the plus cast operator here. That way we can log the original parameter value.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=307.035) [We then use the plus cast operator here to ensure we are passing in a numeric Id.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=315.035) [Then instead of simply returning the result of the getProduct method, we call the observable map operator](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=321.035) [so we can access the return data before passing it on. This allows us to check whether we did indeed](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=327.035) [retrieve a product. If so, we return the product. If not, we log, navigate to the Product List page,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=335.035) [and return null. The map operator returns the value as an observable, so we do not need Observable.of here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=343.035) [Lastly, we add a catch operator to catch any other retrieval errors.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=352.035) [We again log, navigate to the Product List page, and return Observable.of(null).](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=358.035) [This code looks much more complex but handles our retrieval errors so our routed components don't have to--](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=363.035) [Now let's see how we associate this resolver with a route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=3&mode=live&start=372.035)

[Adding a Resolver to a Route Configuration](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=4&mode=live)

[To associate a route resolver with a route, we add it to the route configuration that we define within an](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=4&mode=live&start=2.934) [Angular module. This ensures that the data for the route is retrieved using the resolver before the routed](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=4&mode=live&start=8.934) [component is activated. We use the route configuration's resolve property to specify the list of resolvers.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=4&mode=live&start=16.934) [Here we define a set of key and value pairs where the key is a logical name for the data and the value is](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=4&mode=live&start=24.934) [a reference to the resolver that returns that data. Since this is a reference and not a string, we need an](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=4&mode=live&start=31.934) [import statement for this resolver in the module. In this example, we name the resolver data product and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=4&mode=live&start=38.934) [define a reference to the ProductResolver service. We add the resolve to both the Product Detail route and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=4&mode=live&start=46.934) [to the Product Edit route. We can add any number of resolvers here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=4&mode=live&start=54.934) [Say, for example, that our ProductEditComponent also wants to prefetch data for a category selection box.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=4&mode=live&start=60.934) [We could build a route resolver for the categories and add a reference to it here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=4&mode=live&start=66.934) [Just ensure that each key for a resolve is unique. But they don't have to be unique across resolves as shown](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=4&mode=live&start=71.934) [with the product key here and here. Let's update our route configuration to reference our new resolver.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=4&mode=live&start=79.934) [The route configuration for our product routes is in the product.module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=4&mode=live&start=89.934) [We'll need a reference to our ProductResolver service, so let's start by importing it.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=4&mode=live&start=94.934) [And since it is a service, we need to register it. We do that by adding it to the providers array as part](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=4&mode=live&start=99.934) [of this module's NgModule decorator. Next, we add the resolve to the route configuration.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=4&mode=live&start=105.934) [For the Product Detail route, we add resolve, then define a key and value pair for each resolver.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=4&mode=live&start=113.934) [We specify product as the key because this resolver provides a single product and reference ProductResolver,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=4&mode=live&start=120.934) [our resolver service. We can reformat this for readability. When the Product Detail route is activated,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=4&mode=live&start=128.934) [this route configuration tells the router to use the resolver service to prefetch the product data.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=4&mode=live&start=135.934) [Let's make this same change to the Product Edit route configuration.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=4&mode=live&start=141.934) [Before we can try this out, we need to modify the components to read this data from the route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=4&mode=live&start=147)

[Reading Resolver Data - Snapshot](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live)

[Once we have the resolver service in place and add it to our route configuration, all we have left to do is](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=2.966) [read the data from the route. If we don't expect the resolver to re-fetch different data unless leaving the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=9.966) [page, we can use the ActivatedRoute snapshot to read the resolver data.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=16.966) [Simply access the data property of the snapshot referencing the desired element using the name of the data we](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=22.966) [defined in the route configuration. We assign the result to our local property.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=29.966) [So this code reads the product data from the route and assigns it to the product property.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=35.966) [If the template binds to this property, all of the data immediately appears when the template is displayed.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=41.966) [No more partially presented pages. Yay!](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=48.966) [Accessing the data array from the ActivatedRoute service gives us a reference to the data instance.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=52.966) [All code that retrieves and works with the same element from the data array shares the same instance.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=60.966) [So any change made to this property in any one component is seen by all components that reference the same property.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=68.966) [This sharing of the data instance is useful when we begin to work with child routes as we'll see later in this course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=77.966) [For now, let's try reading the resolver data from the route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=85.966) [Let's start with the product-detail.component. Previously, we imported the ActivatedRoute and defined it](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=90.966) [as a dependency here in the constructor. We no longer need to read the params array to get the product Id](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=97.966) [nor do we need to get the product by passing that Id to our product data service.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=105.966) [We don't even need our product data service instance. Instead, we set the local product property to the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=110.966) [route.snapshot data element containing the product data. We named our product data product.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=118.966) [Wow, that simplifies our code here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=125.966) [Let's check it out in the browser. Select Product List.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=127.966) [And notice the partial page display. That's because we have not defined a resolve for the list of products.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=132.966) [Click on a product.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=139.966) [Notice the pause while the route gets the data, then the entire Product Detail page appears.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=141.966) [No partial page. It works! Before modifying the Product Edit to also use the resolver data,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=147.966) [let's review what it looks like now.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=155.966) [An empty header displays.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=159.966) [Then a few moments later, the edit page appears. This would also look better using a resolver.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=160.966) [Now we can make similar changes to the product-edit.component to read the product data from the route](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=167.966) [using the snapshot. Note that the product-edit.component does still need the product data service instance](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=173.966) [in order to save or delete the product. Pause the video now if you want to give it a try.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=180.966) [Are you ready for my solution? We no longer need to get the Id from the route parameter so we can delete](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=187.966) [all of the code in the ngOnInit. And we don't need the getProduct method.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=193.966) [The code to get the product is now instead in the ProductResolver service and it does not need to be](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=200.966) [duplicated here. Now in the ngOnInit, we simply call the onProductRetrieved method passing in the product](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=206.966) [from the route snapshot. The onProductRetrieved method then assigns the product to the local product](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=214.966) [variable and sets the appropriate header text.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=221.966) [Let's check it out in the browser.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=225.966) [Select Product List](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=227.966) [and click to Edit a product.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=229.966) [Notice the slight pause while the route gets the data, then the entire Product Edit page appears.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=230.966) [It works! But what if the user clicks on the Add Product menu option when this page is displayed.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=237.966) [The route changes. We can see that here. But the page is not correctly displayed for entry of a new product.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=243.966) [As we saw earlier in this course, the ngOnInit method is not re-executed when the route is changed if we](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=251.966) [stay on the same page. So our code does not know that it needs to re-read the product data from the route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=259.966) [How do we fix this? If you said observables, you are correct.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=5&mode=live&start=267.966)

[Reading Resolver Data - Observable](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live)

[Instead of reading the data from the ActivatedRoute snapshot, we subscribe to the ActivatedRoute data observable.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=2.24) [If the route changes without leaving the page, like we saw in the last demo, the route data is re-read from the route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=10.24) [Let's give this a try.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=18.24) [We want to change the product-edit.component code from using this snapshot approach to using an observable.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=21.24) [So let's start by deleting this snapshot code. We'll enter this.route.data to use the router's data](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=29.24) [property, which is an observable. And we'll call this subscribe method to subscribe to changes to the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=37.24) [prefetched resolver data. We provide a function to the subscribe method.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=45.24) [The resolved data is passed into this function so when we receive a notification, we call onProductRetrieved](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=51.24) [and pass along the product element of that resolved data. One thing to make clear here--we are only](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=58.24) [notified when the resolver re-fetches data. We won't receive a notification if our code changes that data.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=66.24) [For example, if we are on the Product Edit page, and the user selects Add Product, we'll get a notification](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=74.24) [when the resolver fetches the initialized product. If we modify one of the product properties such as](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=81.24) [editing the product name, we won't receive a notification.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=88.24) [Are we ready to check it out in the browser? Click to edit a product](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=92.24) [and Bang! The Product Edit page appears fully populated. Click Add Product](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=96.24) [and Bang! The Product Edit page appears initialized for entry of a new product.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=103.24) [No partially presented page. Using a resolver ensures the user does not see partial pages appear when](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=109.24) [waiting for data retrieval. This makes for a much nicer user experience.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=118.24) [Before moving on, let's sidetrack for a moment. Now that we have our resolver working, let's look at an](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=124.24) [alternative way to create a resolver. Instead of creating a service and registering the service using the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=131.24) [providers array, we could instead define the resolver directly here as a function.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=138.24) [We will be undoing the code changes I'm about to make, so feel free to skip coding along with this part](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=144.24) [of the demo if desired. In the providers array, we define a provider function by specifying an object.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=151.24) [I'll paste in the code, and we can talk through it. We set the provide property to the name of this resolver.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=159.24) [I call it productProvider here. Then we set the useValue property to a function.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=167.24) [We can't call our data access service here, but we can hard code in some data.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=173.24) [If we did want to call our data access service, we could use the useFactory property instead, which allows us](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=179.24) [to pass in injected services. In this example, our function needs no parameters and returns the desired data.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=186.24) [Note that we could access the ActivatedRouteSnapshot and RouterStateSnapshop parameters here if we needed them.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=195.24) [Next, we change the resolve in the route configuration to use this provider.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=203.24) [Since this is a string and not a reference, ensure it is enclosed in quotes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=209.24) [Angular detects whether the route resolver is a function or a class.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=214.24) [If it is a function, it uses the function defined in the useValue or useFactory property.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=220.24) [If it is a class, it calls the resolve method in the class. Personally, I like building the resolver as a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=225.24) [service and keeping this logic out of our Angular module. But this technique does work, and you will see it.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=233.24) [I'll undo this change, and we'll instead use our resolver service.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=241.24) [Now let's finish off this module with some checklists you can use as you prefetch data using route resolvers.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=6&mode=live&start=246.24)

[Checklists and Summary](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=7&mode=live)

[To build a route resolver, create an Angular service. Then implement the Resolve generic interface](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=7&mode=live&start=2.367) [specifying the type of data to retrieve as the generic parameter.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=7&mode=live&start=10.367) [And add a resolve method that retrieves the desired data. Since this is a service, be sure to register it](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=7&mode=live&start=14.367) [in an Angular module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=7&mode=live&start=22.367) [Configure the route resolver in an Angular module using the resolve property.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=7&mode=live&start=25.367) [Give each type of prefetch data a logical name and specify a reference to its associated route resolver service.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=7&mode=live&start=30.367) [In the component, read the data from the route. Use the ActivatedRoute snapshot if the route never changes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=7&mode=live&start=38.367) [while on the page. Use the data observable if the route could change while on the page such as our Product](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=7&mode=live&start=46.367) [Edit example, or the user can navigate to the Add operation from the edit page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=7&mode=live&start=53.367) [In this module, we began with a look at several techniques for providing data to a route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=7&mode=live&start=60.367) [We examined how to use a route resolver. We then built a route resolver as an Angular service.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=7&mode=live&start=67.367) [We added the resolver to two of our route configurations and saw how to read the resolver data using the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=7&mode=live&start=73.367) [ActivatedRoute snapshot and using the ActivatedRoute data observable to watch for changes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=7&mode=live&start=80.367) [Our Product Detail and Product Edit routes now prefetch the product data before routing to these components.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=7&mode=live&start=88.367) [You can use the techniques shown in this module to prefetch data for other routes such as the Product List route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=7&mode=live&start=95.367) [Next up, we look at child routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m5&clip=7&mode=live&start=103.367)

[Child Routes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=0&mode=live)

[Introduction](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=0&mode=live)

[We can define routes that are displayed within other routes. Welcome back to Angular Routing from Pluralsight.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=0&mode=live&start=0) [My name is Deborah Kurata, and this module is all about child routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=0&mode=live&start=10) [Using child routes, we define a route hierarchy to better organize, encapsulate, and navigate through our application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=0&mode=live&start=16) [Plus it makes it easier to lazy load routes improving the startup performance of the application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=0&mode=live&start=26) [Here we have two primary routes--welcome and products. The products route is a component list route](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=0&mode=live&start=33) [meaning it does not activate a component. Rather it simply acts as a parent route allowing us to group](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=0&mode=live&start=41) [the product routes as child routes. This products route has three children.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=0&mode=live&start=47) [Further down the hierarchy, the Product Edit route has two children--Edit Info and Edit Tags.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=0&mode=live&start=54) [Our focus in this module is on the basics of child routes. So we'll add the two Product Edit child routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=0&mode=live&start=61) [We'll look at grouping and component list routes in the next module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=0&mode=live&start=69) [In this module, we begin with an overview of child routes, what they are, how they work, and when to use them.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=0&mode=live&start=76) [We then look at how to configure child routes, place the child view, and activate child routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=0&mode=live&start=84) [We examine how to obtain data for child routes and how to validate data across child routes when splitting](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=0&mode=live&start=91) [an edit operation across child views.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=0&mode=live&start=99) [In our sample application, we'll change the ProductEditComponent to display a tabbed container.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=0&mode=live&start=103) [Then we'll add a child route for each tab, one for editing basic product information and the other for](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=0&mode=live&start=110) [editing product search tags. Let's get started.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=0&mode=live&start=117)

[Using Child Routes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live)

[Before we look at how to use child routes, it's important to remember how our primary routes work.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=2.957) [This snippet lays out a navigation menu and other elements that make up the outermost container of an application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=9.957) [Recall from earlier in this course that we used the router-outlet directive to define where to place the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=16.957) [routed component's template for display.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=23.957) [When the user views the application in the browser and routes to a component, that component's template is](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=26.957) [displayed within the outlet. Of course the router-outlet directive doesn't appear in the UI.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=33.957) [It is shown here for instructional purposes only. If the user clicks one of the Edit buttons,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=40.957) [the product-edit.component template is displayed in place of the Product List within this same outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=46.957) [But what if we have too many visual elements to display nicely on one page?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=53.957) [One common technique for displaying lots of data elements on a page is to organize related information on tabs.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=59.957) [How then do we display the appropriate template for each tab? Define the Product Edit page as a second](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=67.957) [container with its own outlet. The header tabs and other shared UI elements are defined in the containing](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=74.957) [component, and the specific data elements for each tab are defined in their own components.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=82.957) [When the user clicks a tab, we route to that tab's component using child routing and display the result](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=89.957) [in the inner container's outlet. This is a key purpose of child routes, to define routes that are displayed](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=97.957) [within other routes, or more technically accurate, to display routed component templates within other](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=105.957) [routed component templates.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=114) [Tabs are a great way to show off child routes, but they are not the only use of this technique.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=117.957) [We can use child routes to build a master/detail style page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=123.957) [We can take advantage of child routes any time we want to embed routed component templates within an existing](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=128.957) [routed component template. If we want to truly encapsulate all of the logic for each of our feature modules](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=135.957) [such as a product module or customer module, we could use child routes to display all of the feature's routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=142.957) [We'll see how later in this course. And child routes are required for lazy loading.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=151.957) [We'll talk more about lazy loading later in this course as well.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=157.957) [Now let's go configure some child routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=1&mode=live&start=162.5)

[Configuring Child Routes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live)

[Currently, our sample application only defines primary routes. The routed component templates appear in](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=1.716) [the App component's template primary outlet. We want to modify our ProductEditComponent to display a set](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=9.716) [of tabs and use child routing to route to each tab's component. The child route component templates appear](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=17.716) [in the parent's outlet. Let's see how to configure these child routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=24.716) [As we've seen, routes are configured as part of router module in an Angular module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=31.716) [Child routes are defined within a children array associated with a parent route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=37.716) [Here we add child routes to the Product Edit route. Within the child array, we define each route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=43.716) [First, we set the path. Child routes extend the path of the parent route so we don't repeat the parent's path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=50.716) [This first path is an empty default path. When navigating to the Product Edit parent route without a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=58.716) [specified child route, this path redirects to the info route and displays it within the parent's outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=66.716) [We covered redirect in detail earlier in this course. We define the info route by specifying info as the path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=74.716) [The full path to this component is then products/:id/edit/info. We then define the component associated with](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=82.716) [this path. When this route is activated, the ProductEditInfoComponent is activated, and it's template is](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=93.716) [displayed within its parent's router outlet. The tags path is defined similarly.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=101.716) [Let's give it a try.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=108.716) [In our sample application, we want to change our ProductEditComponent template to a container with tabs](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=111.716) [and use child routes to route to the content for each tab. The product routes are in the product.module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=118.716) [We add the routes for each tab as children of the edit route. So we define the children array as part of](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=126.716) [the Product Edit route configuration. I'll paste the code and the import statements.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=133.716) [And we can talk through it. For each child route, we first define the path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=140.716) [This first route is for an empty path. This defines the default path to display if no child path has been specified.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=146.716) [Here is the info path. We specify the component to activate when this route is activated.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=154.716) [We repeat a similar entry for the tags path. Now we've just referenced two new components, so we need to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=161.716) [add them to the declarations array. I included the code for both of these components and their associated](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=168.716) [templates as part of the starter files. That allows us to keep focused on the routing and not on building](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=176.716) [components and templates. We can't yet try out these changes because we have nowhere to display these child views.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=183.716) [Let's do that next.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=2&mode=live&start=191.716)

[Placing the Child View](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live)

[To display a template for a child route within the parent's template, the parent's template must contain a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=2.111) [router-outlet directive. This router-outlet directive identifies the location where the child's template](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=10.111) [is displayed within the parent's template. The outlet for display of the child routes looks basically the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=17.111) [same as the syntax for the primary outlet. Notice that we have no name or other specifier on the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=24.111) [router-outlet directive. The router knows whether to display a template in the primary or a child outlet](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=31.111) [based on the route hierarchy defined in the route configuration.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=39.111) [Let's give this a try.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=44.111) [Our product-edit.component's template currently displays a form for editing a product.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=47.111) [We want to display a set of tabs instead. Let's delete the form and replace it with two tabs.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=53.111) [We could of course add any number of tabs here, but two are enough for our purposes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=61.111) [Then we'll add the router-outlet directive where we want the child route component templates to appear.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=66.111) [Even though we aren't yet activating any of our child routes, our empty path should kick in.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=72.111) [So let's see how it looks in the browser. Navigate to the Product List,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=79.111) [click on the Edit button for one of our products,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=83.111) [and here it is. The router displays the primary routes such as our Product Edit route in the App component's](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=87.111) [outlet and displays the child routes such as our Product Edit Info route in the ProductEditComponent's outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=93.111) [We have child routes working. Yay! But notice the data here. Regardless of the product we select for edit,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=103.111) [we will always get the same test data. That's because we are not yet retrieving the data in the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=111.111) [ProductEditInfoComponent. Currently, the component has hard-coded data.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=117.111) [We'll fix that shortly. Notice also that the tabs don't work. We'll fix that as well.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=123.111) [First, let's add a set of buttons for the user to select to save, cancel, or delete.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=130.111) [I'll paste the code for the buttons, and we can talk through it.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=137.111) [We want the buttons to appear under the routed child component. The Save button calls the saveProduct method](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=141.111) [to save the product. The Cancel button uses a routerLink directive as we saw earlier in this course to route](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=147.111) [back to the Product List page. And the Delete button calls the deleteProduct method to delete the product.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=155.111) [Viewing the result in the browser, we see the row of buttons below the tabs.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=163.111) [Lookin' good! But our tabs still don't work. Let's look at how to activate child routes and hook up our tabs.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=3&mode=live&start=169.111)

[Activating Child Routes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live)

[There are two ways to activate a child route, with an absolute path or with a relative path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=1.98) [An absolute path begins with a slash and requires definition of each segment of the URL path as part of the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=9.98) [link parameters array. The router matches an absolute path starting from the top of the route configuration.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=17.98) [Using an absolute path ties the route activation to the current route hierarchy, so if any levels are later](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=24.98) [added or changed, each of these paths also needs to be changed.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=32.98) [Often a better option is to use a relative path. A relative path does not begin with a slash and is relative](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=37.98) [to the current URL segment. If the ancestor route segments ever change, this path is unaffected.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=44.98) [We can also activate child routes in code similar to activating a primary route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=52.98) [Each route segment is defined within the link parameters array. We can also use relative routing,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=58.98) [but the navigate method requires an additional parameter. We pass in object with a relativeTo property](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=65.98) [set to this.route where route here is the ActivatedRoute. This provides the router with the current route](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=73.98) [so it can append this relative path. Be sure to leave off the slash here if the first element in the link](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=82.98) [parameters array starts with a slash, the navigation is absolute regardless of whether we specify the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=88.98) [relativeTo property. Let's give this a try.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=96.98) [The visual elements for our tabs are defined in the product-edit.component template.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=100.98) [When the user clicks the Info tab, the router should navigate to the info path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=106.98) [So we specify a routerLink directive here and assign it to the info path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=111.98) [We use relative routing so changes to the ancestor routing segments won't affect this route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=117.98) [Notice that we are using the link parameters array syntax. Since we are assigning to a simple string,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=123.98) [we could instead use the shortcut syntax we learned earlier in this course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=129.98) [We add a similar routerLink directive to this Search Tags tab and route to the tags path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=135.98) [That's all we need to do.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=141.98) [Let's check it out in the browser.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=143.98) [Display the Product List,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=145.98) [click on an Edit button,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=148.98) [and the Edit Product page appears. Click the Search Tags tab,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=149.98) [and the Tags page appears. We can add a single search tag or multiple tags separated by commas.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=153.98) [Click the Add button, and the tags are added to the list. Click any existing search tag to delete it.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=161.98) [Now click on the Basic Information tab, and we are returned to the info page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=168.98) [Cool! But our changes are not retained. And we are still seeing test data.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=174) [So let's look at how to obtain the actual data for our child routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=4&mode=live&start=181.98)

[Obtaining Data for Child Routes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live)

[There are several ways to obtain data for a child route. We could use our product data service to get the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=2.189) [data for this tab. But as we saw in the last module, getting the data in the component itself can cause the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=9.189) [user interface to partially appear while it is waiting for the data.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=16.189) [Another option is to define a route resolver and retrieve the data before routing to the child route's component.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=21.189) [This is a great option if each child route requires different data.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=28.189) [If the child route's work with the same data such as in our product.edit example Product Edit example,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=33.189) [we can instead use a route resolver on the parent route. Even though we have multiple tabs, the display data](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=38.189) [is all part of the same product. So we get the product when routing to the parent route and read it in each](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=46.189) [child route's component. Notice that we then specify the route.parent here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=53.189) [these last two approaches use the ActivatedRoute snapshot for reading route data.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=59.189) [As we saw in the last module, we could instead use the ActivatedRoute's data observable to watch for changes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=65.189) [to the route resolver's data. Since our edit pages all work with the same set of data, let's use the parent](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=72.189) [route's resolver. Recall that this syntax provides a reference to the product data instance so our parent](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=79.189) [route and each child route with a reference will share that instance.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=88.189) [Let's give it a try.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=93.189) [We start with the Info tab component and delete the hard-coded data here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=96.189) [We'll instead define the product type as IProduct. Next, we add code to the ngOnInit method to retrieve](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=102.189) [the data from the parent's route. We could use the snapshot method to retrieve the product data,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=108.189) [but since we are working with the edit page, we know from experience that we'll want to subscribe to the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=115.189) [data observable instead. That way the code is notified if the user selects the Add Products option,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=121.189) [and we need to redisplay for entry of a new product. We'll use this.route, which is the ActivatedRoute,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=127.189) [.parent.data, and subscribe to the parent route's data. We provide a function to the subscribe method.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=135.189) [The resolved data is passed into this function, and we use the arrow syntax to define the code that is](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=143.189) [executed whenever the router re-fetches data. Here we set our local product property to the product element](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=150.189) [of the resolved data. Where is this product data coming from? We defined it in the product.module here](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=157.189) [in the last course module. We set up a resolver to prefetch product data and added the resolver to our](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=165.189) [Product Edit parent route definition, so the child routes can read this resolver data from the parent route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=172.189) [Viewing the app in the browser, we now see that the Info tab displays the correct data.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=180.189) [Wahoo!](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=185.5) [Want to try making this same change to the Search Tags tab component?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=187) [If so, pause the video here. Are you read to see my solution? I deleted the hard-coded product data](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=191.189) [and set the product type to IProduct. I then added code to the ngOnInit method to subscribe to the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=198.189) [ActivatedRoute parent's data observable. In the passed-in function, I set the local product property to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=205.189) [the product data from the route. This is now sharing the same product instance as the parent component and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=212.189) [the other child route. This is important because our parent component still contains the code for saving](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=220.189) [changes to the product. Any changes made to the product instance in any child component is reflected in](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=225.189) [the parent component's product instance. Let's check that out in the browser.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=233.189) [We see that the Info tab displays the correct data. Click the Search Tags tab and it too displays the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=238.189) [appropriate product data. We can now edit any of the product data.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=245.189) [Click Save to save the changes,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=249.189) [and we are returned to the Product List page. Here we see that the changes were indeed saved.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=250.189) [Note that we using an in-memory Web API to save these changes so the next time we restart the application](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=258.189) [with npm start or refresh the page,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=265.189) [the data will be reset to its original values. But we do have a little validation problem.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=268.189) [Let's try and edit again. If we clear the Product Code, the validation lets us know that the Product code is required.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=274.189) [But if we now pick the Add Product menu option, the form values are cleared but the validation errors are not.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=282.189) [Let's make one more change to fix this.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=290.189) [The template for each of our tab components define a form for data entry.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=294.189) [In the product-edit-info.component template, the form is called productForm.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=298.189) [In the Component class, we use the ViewChild decorator to obtain a reference to the template's form.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=304.189) [We can then use that reference to reset the form every time we get new data.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=311.189) [Resetting the form clears the form state including any validation errors.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=316.189) [We want this code within the subscribe function. That way our forms data is reset if the user selects to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=321.189) [add a product. First, we ensure that we successfully obtain a reference to the form.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=328.189) [If so, we call reset. This resets the validation state on the form.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=334.189) [Let's see if this improves the validation experience.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=339.189) [Click to edit a product,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=343.189) [clear the Product Code, and the validation lets us know that the Product code is required.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=346.189) [Pick the Add Product menu option. The form values are cleared, and the validation errors are reset.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=351.189) [Great! We could make the same change to the Product Edit Tags component, but it really isn't necessary.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=358.189) [Every time the user selects Add Product, the application always defaults to the Basic Information tab.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=364.189) [The Search Tags tab component template is then unloaded so none of the validation settings are retained.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=371.189) [There is one other validation issue we should fix. If we again edit a product](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=378.189) [and clear the Product Code, we see the validation errors. But we can still click Save to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=383.189) [save the data. Yikes!](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=390.189) [We should disable this Save button if the form is not valid. But the button is on the parent form,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=392.189) [which doesn't know the state of the child form validation. As it is, this could be confusing for the users](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=398.189) [and potentially cause issues in our database. How do we fix this?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=5&mode=live&start=405.189)

[Validating Across Child Routes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live)

[Recall our current route hierarchy? When the user clicks the Info tab, the product-edit-info.component](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=2.256) [template is displayed in the outlet. When the user clicks the Search Tags tab,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=9.256) [the product-edit-info.component template is completely removed from the outlet and replaced with the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=14.256) [product-edit-tags.component template. This means that the Product Edit Info component and its form is](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=20.256) [unloaded along with any of its validation settings. And when the user clicks the Info tab,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=26.256) [the product-edit-tags.component template is completely removed from the outlet and replaced with a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=33.256) [product-edit-info.component template. That means that the Product Edit Tags component and its form is](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=39.256) [unloaded along with any of its validation settings. If only one form and, hence, one form's validation is](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=45.256) [accessible at any time, how do we validate across child routes?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=53.256) [Let's think through some possible solutions. What if we define the form in the parent component and put the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=60.256) [form's input elements in the appropriate child components? Then the parent component can track the state](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=67.256) [of the form. Nah, that doesn't work. An Angular form does not recognize input elements defined within a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=74.256) [router-outlet, so if you are using template-driven forms, the input elements never appear in the forms model.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=81.256) [What if we define a form in each child component? That's what we have now and saw that this option doesn't](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=89.256) [work either. Only one form is active at a time, so only its validation is known since validation is form-based.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=97.256) [We could forget about child routes and instead put the form and all of the tab elements into the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=105.256) [product-edit.component template. Then hide and show the different tab elements as needed.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=111.256) [While this would work, it would be more difficult to build and maintain, especially as more tabs are added.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=117.256) [What if we define a form in each child component like we have now but then manually perform the validation](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=123.256) [against the data instead of relying on the form? That might work.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=131.256) [Let's give it a try.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=136.256) [Since the Save button is on the parent product-edit.component, let's add the validation there.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=140.256) [First, we need some type of data structure to hold the validation state of each tab.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=146.256) [Since we could have any number of tabs, let's define a property called dataIsValid and set its type to be a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=152.256) [set of key and value pairs where the key is the tab path name and the value is true for valid and false for invalid.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=159.256) [Then let's add a validate method to perform all of our validation.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=167.256) [We'll add it here at the bottom. I'll paste the code, then we can walk through it.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=172.256) [We'll use the product data, not the form input elements, to perform our manual validation.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=177.256) [Recall that the ActivatedRoute service data property gives us a reference to the product instance,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=184.256) [so any changes to the product instance made on any of the tabs is reflected in the parent component's](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=191.256) [instance as well. First, we clear the validation data structure.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=196.256) [This ensures that each time we validate, we start with a clean structure.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=201.256) [Then we add if logic to check our validation rules for the info tab elements.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=205.256) [Each of the validation rules defined in the HTML for this tab are repeated here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=211.256) [Yeah, I know, we are repeating our validation. And if someone later adds validation to the HTML,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=217.256) [that someone will need to remember to add it here as well. But if we want field-level validation as the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=223.256) [user tabs through and correct validation of all of the tabs, then at least at this point in time,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=230.256) [we need validation in both places. If all of the validation rules pass, we set the value in our data](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=236.256) [structure for this tab to true. Otherwise, we set it to false. We repeat similar code for each tab.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=244.256) [Let's also add an easy way to determine if the form is valid by adding an isValid method.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=252.256) [I'll paste the code, and we can talk through it. This method takes in the path of the tab to check.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=258.256) [It then performs the validation. If checking a specific tab, it returns a result from the validation](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=264.256) [data structure for that tab. Otherwise, it checks every entry in the data structure and returns true only](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=270.256) [if the validation of all tabs is true. Now we can ensure the entire product is valid before saving.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=278.256) [In the saveProduct method, we'll call isValid passing in null to check all tabs.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=286.256) [And now we can disable our Save button if the data is not valid. In the HTML, we bind to the disabled](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=292.256) [property, disabling the button if the form is not valid. Let's check it out](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=300.256) [in the browser. Select to edit a product.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=305.256) [Now if we clear the Product Code, we see a validation message, and the Save button is disabled.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=309.256) [When we select the Search Tags tab, the Save button remains disabled.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=315.256) [Our technique works, though it does require duplicated validation logic.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=320.256) [We still have a small user interface issue. If there were a lot of tabs, how would the user know which](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=325.256) [tab contained the validation error that is causing the Save button to be disabled?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=332.256) [We could add an error icon to each tab that has a validation error.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=336.256) [We'll do that later in this course. For now, let's finish off this module with some checklists you can](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=341.256) [use as you define child routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=6&mode=live&start=347.256)

[Checklists and Summary](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=7&mode=live)

[When configuring a child route, add a children array to the parent route, define the child routes within](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=7&mode=live&start=2.223) [that array, and remember that the child paths extend the parent route so only the child's route segment is](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=7&mode=live&start=9.223) [specified here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=7&mode=live&start=16.223) [Place the child view by defining a RouterOutlet directive within the parent component template.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=7&mode=live&start=19.223) [Each child route will then appear in this location. Every parent route component should have a RouterOutlet](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=7&mode=live&start=26.223) [in its template.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=7&mode=live&start=32.223) [We can activate a child route using an absolute path. Be sure to start the path with a slash and define](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=7&mode=live&start=35.223) [each URL segment. A better approach is often to activate a child route using a relative path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=7&mode=live&start=42.223) [Then there is no starting slash, and only the child's URL segment is specified.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=7&mode=live&start=49.223) [Note that when activating the route in code using relative routing, the relativeTo property must be set](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=7&mode=live&start=55.223) [to the ActivatedRoute.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=7&mode=live&start=62.223) [Read data for the child route using the ActivatedRoute snapshot or by subscribing to the ActivatedRoute's](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=7&mode=live&start=64.223) [data observable. When reading the data for a child route that was resolved in the parent route,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=7&mode=live&start=72.223) [be sure to specify the ActivatedRoute parent instead.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=7&mode=live&start=78.223) [In this module, we examined what child routes are, how they work, and when to use them.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=7&mode=live&start=83.223) [We configured several child routes, placed the child view in the parent's template using a RouterOutlet](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=7&mode=live&start=89.223) [directive, and activated the child routes. We looked at how to obtain data for the child route from the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=7&mode=live&start=96.223) [child route itself or from its parent's route. And since our specific scenario was a multi-tabbed edit page,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=7&mode=live&start=103.223) [we walked through how to validate across child routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=7&mode=live&start=110.223) [Our ProductEditComponent is now an edit shell with a the tab bar buttons and an outlet for the child tab components.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=7&mode=live&start=114.223) [And we defined a child route for each tab displaying a child component template in that outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=7&mode=live&start=122.223) [Up next, let's look at how to group child routes under a component-less parent route and discern what that even means.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m6&clip=7&mode=live&start=129.223)

[Grouping and Component-less Routes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=0&mode=live)

[Introduction](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=0&mode=live)

[We may want to organize routes such as feature routes under a single parent route without defining another outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=0&mode=live&start=0) [Welcome back to Angular Routing from Pluralsight. My name is Deborah Kurata, and in this module,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=0&mode=live&start=10.5) [we group child routes under a component-less parent route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=0&mode=live&start=16) [As we saw in the last module, we define child routes to display routed component templates within other](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=0&mode=live&start=21) [routed component templates. In our sample application, we display the templates from the tab components](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=0&mode=live&start=28) [within the ProductEditComponent outlet. But there may be times that we want to organize our routes under](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=0&mode=live&start=35) [a single parent route without defining another outlet. Here we group the product routes under a parent](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=0&mode=live&start=42) [that has no component and, hence, no outlet, the child component templates then appear in the next](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=0&mode=live&start=49) [higher-level outlet. For the color coding here, these are technically child routes and could be denoted with](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=0&mode=live&start=58) [purple as with our other child routes. But since they appear within the primary router-outlet,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=0&mode=live&start=64) [I opted to make them turquoise to match that router-outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=0&mode=live&start=70) [Why? Why should we group routes under a component-less route? Grouping helps us better organize our routes,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=0&mode=live&start=75) [especially as an application gets larger. Grouping allows us to share resolvers and other guards.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=0&mode=live&start=83) [We'll see how to use grouping to share a guard later in this course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=0&mode=live&start=90) [And by grouping all of the routes for a feature area under a single parent, we can lazy load the routes for](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=0&mode=live&start=94) [that feature area. We'll see how to use lazy loading later in this course as well.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=0&mode=live&start=101) [In this module, we group routes under a single parent route and define that parent as a component-less route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=0&mode=live&start=107) [We are making a big change to our route hierarchy in this module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=0&mode=live&start=116) [Instead of defining our product routes as the same level of the route hierarchy as the Welcome, Log In,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=0&mode=live&start=121) [and other routes, we group our product routes under a component-less parent route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=0&mode=live&start=127) [Let's get started.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=0&mode=live&start=134)

[Grouping Routes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=1&mode=live)

[Currently, the configuration for our product routes looks like this.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=1&mode=live&start=1.748) [We could instead group our routes such that the other product routes are children of the products route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=1&mode=live&start=6.748) [Since child routes extend the path of the parent route, we specify relative paths making our paths shorter](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=1&mode=live&start=12.748) [and more durable as paths change over time. But, spoiler alert, these grouped routes don't work as they are shown.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=1&mode=live&start=20.748) [We'll see why in a moment. For now, let's try grouping our routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=1&mode=live&start=28.748) [Our product routes are configured in the product.module. I've reformatted them a bit to fit better on the page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=1&mode=live&start=34.748) [We want to group the Product Detail and Product Edit routes as children of the products route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=1&mode=live&start=40.748) [We start by adding a children property to the products route. Then we move the other product routes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=1&mode=live&start=47.748) [within the children array and reformat to line everything up. Now that the Product Detail and Product Edit](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=1&mode=live&start=55.748) [routes are child routes, we can use relative routing. Just delete the products/ from the path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=1&mode=live&start=61.748) [Don't forget to remove the slash to ensure the route recognizes these as relative routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=1&mode=live&start=70) [Let's check it out in the browser.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=1&mode=live&start=76.748) [Click Product List](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=1&mode=live&start=78.748) [and select a product name. We are not routed to the Product Detail page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=1&mode=live&start=80.748) [Click an Edit button, and we are not routed to the Product Edit page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=1&mode=live&start=85.748) [We broke it!](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=1&mode=live&start=91.748)

[Component-less Routes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live)

[Recall how child routes work. Child routes are displayed in a router-outlet defined within the parent](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=2.248) [component's template so the info-tab.component template appears within this child router-outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=10.248) [To display our new child routes, we'd need a child router-outlet in the parent component's template,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=16.248) [which in this case is the Product List page. We don't want the Product Detail and Product Edit pages](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=23.248) [appearing within the Product List page. How do we get around this?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=29.248) [We use a component-less route. Here is our current grouped route configuration.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=34.248) [We move the Product List route as a child as well. So now the parent route has no component.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=40.248) [Hence, it's called a component-less route. Other than its cool name, this may not sound like a big deal,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=48.248) [but it changes how the router displays these child routes. Since the parent no longer has an associated](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=55.248) [component, the router won't attempt to route its children into the parent's outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=62.248) [Instead, the children are displayed in a higher-level outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=67.248) [Our route hierarchy then looks like this. We've defined a component-less route with three children.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=72.248) [Since the parent route is component-less, the child component templates appear in this primary outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=79.248) [Let's give this a try.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=87.248) [In the product.module, we add another child route definition. We give it a relative path, which in this case](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=90.248) [is the empty path, and route to the ProductListComponent. We then remove the component property from the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=97.248) [parent route. This changes our parent route to a component-less route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=104.248) [And let's reformat it to match the others. Let's see if that fixes our routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=110.248) [Click on Product List,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=116.248) [and the router navigates to the Product List page. Click on a product name,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=117.248) [and we see the Product Detail page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=122.248) [Click Edit, and we see the Product Edit page. Select Add Product, and we see the Product Edit page for](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=124.248) [entry of a new product. We've got it all working again. Yay!](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=132.248) [Now let's finish off this module with some checklists you can use as you group your routes under a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=137.248) [component-less parent route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=2&mode=live&start=143)

[Checklists and Summary](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=3&mode=live)

[To group routes, define routes as children of one parent route. Be sure to specify relative paths for the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=3&mode=live&start=2.335) [child routes. But grouping alone is not sufficient. As we saw in this module, the router wants to put these](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=3&mode=live&start=10.335) [child routes into a router-outlet defined in the parent component's template.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=3&mode=live&start=17.335) [To define a component-less route, add a default path that routes to the desired component.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=3&mode=live&start=23.335) [Remove the component from the parent route making it a component-less route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=3&mode=live&start=29.335) [The child routes are then displayed in a higher-level outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=3&mode=live&start=35.335) [In this module, we grouped our product routes under a single parent route and defined that parent as a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=3&mode=live&start=39.335) [component-less route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=3&mode=live&start=46.335) [We've added a component-less parent route and grouped our product routes under it as children.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=3&mode=live&start=49.335) [We can now share resolvers and other guards and lazy load the product routes as we'll see later in this course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=3&mode=live&start=55.335) [Next up, we'll switch gears a bit and see how to style, animate, and watch our routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m7&clip=3&mode=live&start=63.335)

[Styling, Animating, and Watching Routes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=0&mode=live)

[Introduction](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=0&mode=live)

[When working on client-side code, simply implementing features is often not enough.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=0&mode=live&start=0) [The app must also be user friendly and visually appealing. Welcome back to Angular Routing from Pluralsight.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=0&mode=live&start=8) [My name is Deborah Kurata, and in this module, we explore how to style, animate, and watch our routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=0&mode=live&start=16) [Oh my! We don't have to go over the top with our route styling or animation.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=0&mode=live&start=25) [A little goes a long way.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=0&mode=live&start=30) [In this module, we learn how to style the selected route to provide a visual indication of where the user](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=0&mode=live&start=33) [is in the application. We look at ways to animate our route transitions to add some polish to the display.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=0&mode=live&start=41) [We discover how to watch our route events to better resolve routing issues.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=0&mode=live&start=47) [And we examine how to react to routing events to display a spinner while the route is loading.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=0&mode=live&start=53) [We won't add any new routes in this module. Rather, we'll polish the ones we have.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=0&mode=live&start=60) [Let's get started.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=0&mode=live&start=65)

[Styling the Selected Route](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live)

[To provide context and help users keep track of where they are in the application, we may want to add](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=2.478) [styling classes to visually indicate the active route. This cascades down through each level of the route](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=8.478) [tree so parent and child routes can appear active at the same time.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=15.478) [For example, we may want to highlight the menu option and tab that the user selected.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=21.478) [We style the activated route using the routerLinkActive directive.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=27.478) [We add this directive to the anchor or to its parent element. We assign this directive to a space delimited](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=34.478) [string of style classes. Here we add one style class, the Bootstrap class named active.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=40.478) [The router adds the classes when this route is activated and removes them when the route is inactive.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=47.478) [In this example, the selected tab is highlighted when it is selected.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=53.478) [Let's gives this a try.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=60.478) [Before jumping into the code, let's run the application. Select Product List](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=63.478) [and navigate to the Product Edit page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=69.478) [Click on the Search Tags tab, and we can see that there's a very little visual indication of the selected tab.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=71.478) [Looking closely, the text changes from one shade of blue to another.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=79.478) [But that isn't very easy to see. Let's instead change the tab color so the selected tab is more prominent.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=83.478) [Our tabs are defined in the product-edit.component template. We simply add the routerLinkActive directive](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=92.478) [to the anchor tag and assign it to the name of the desired style class or set of classes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=100.478) [We are using the Bootstrap CSS style classes, so we set the directive to active.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=106.478) [And since we are assigning the directive to a string and not to a property or expression, we don't need the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=112.478) [square braces around the directive. We'll repeat this for each tab.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=119.478) [That's it!](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=123.478) [Looking back at the application in the browser, the tabs are now clearly indicated.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=125.478) [Cool! Let's add a little more style to the tabs. If the user makes an error here and navigates to another](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=129.478) [tab, we see that the Save button is disabled. But if there were lots of tabs, it wouldn't be easy for the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=137.478) [user to determine which tab has the validation error. Let's add an icon to the tab if the tab's data](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=144.478) [contains a validation error.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=151.478) [Recall that our product-edit.component has an isValid method that takes in the tab path and returns true if](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=154.478) [the data in the tab is valid. We'll use this method to determine when to add the validation error icon.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=161.478) [In the template, we add a span tag next to the text of the tab. We'll use the ngClass directive,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=169.478) [which turns on a style class based on an expression. We'll use the Bootstrap CSS glyphicon style to add](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=178.478) [an exclamation point. And we want this to display this icon if isValid is not true for the Info tab.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=184.478) [We'll copy and paste this for the Search Tags tab and change the info text to tags.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=193.478) [So the icon will appear here is the Search Tags tab is not valid.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=200.478) [Let's check it out](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=204.478) [in the browser. If we cause a validation error, we see the icon added to the tab.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=205.478) [If we navigate to another tab, that icon still appears. This lets the user know which tab contains the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=212.478) [validation error. That seems helpful. Now our tabs style indicates which tab is active, and if input](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=219.478) [elements on that tab have validation errors, we indicate that as well.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=225.478) [But what about the menu? If we pick Add Product, there's no visual indication that the menu option was selected.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=230.478) [Let's add the routerLinkActive attribute to highlight the selected menu option.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=239.478) [The application menu is defined in the app.component template. We can make this same change to the main](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=244.478) [menu options adding the routerLinkActive directive to each of the anchor tags.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=252.478) [Let's check that out in the browser. Click Product List and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=256.478) [no style change. Bummer!](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=261.478) [Looking back at the code, this is because we are using a list to define our menu.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=264.478) [Its style is overriding our link style. Let's instead move the routerLinkActive directive to the parent li element.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=269.478) [The routerLinkActive directive works on any ancestor of the element containing the routerLink directive.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=276.478) [So let's remove these from here and add it to the li elements instead.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=285.478) [Viewing the result in the browser, we have styling. Yes! But if we select Add Product, both the Product](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=291.478) [List and Add Product menu options are marked as active. What?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=300.478) [Looking at the routerLink directives for the Product List and the Add Product menu options, we see that the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=305.478) [first path segment of each path matches. That's why both options are marked as active.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=311.478) [To require an exact path match, we add the routerLinkActiveOptions directive on the Product List route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=318.478) [We add the routerLinkActiveOptions directive to the same element as the routerLinkActive directive.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=326.478) [We then assign it to an object with the exact property set to true.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=332.478) [Note that we only need to add the routerLinkActiveOptions directive for the Product List route because that](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=337.478) [is the only one here that could match multiple route paths.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=343.478) [Viewing the application in the browser, we see the one selected menu option marked as active.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=347.478) [Now our users have a better indication of where they are in our application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=354) [Next, let's add a little visual interest with animation.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=1&mode=live&start=358.478)

[Animating Route Transitions](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live)

[Smoothly transitioning from one route to the next adds some polish to an application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=3.169) [We define route transitions with animation.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=8.169) [There are two basic ways we can animate our route transitions--with CSS animation or with Angular animation.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=13.169) [CSS animation is quick and easy. We define some CSS and every route in every component is animated.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=22.169) [However, this is not really animating our routes. It's instead animating specific HTML elements of our application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=31.169) [As we access each route the HTML elements are animated making it look like it's animating our routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=39.169) [Using Angular animation to truly animate our route transitions is a bit more complex.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=47.169) [We need to configure the animation, then apply it to every component we want to animate.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=53.169) [At the time this course was recorded, discussions were underway to add animation features that would make](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=59.169) [it easier to animate routes without modifying every component.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=65.169) [For more information on animating route transitions using the current Angular animation features,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=70.169) [check out the Angular documentation. Since our goal in this course is simply to make our route transitions](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=76.169) [look smoother, not to learn Angular's animation features, let's try out the CSS animation technique.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=83.169) [Here is the application stylesheet I provided with the starter files.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=92.169) [I added two different sets of CSS to animate our route transitions.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=96.169) [The first option fades in every div element within our application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=101.169) [The second option slides in our primary panel. In both examples, the selector name is our application](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=106.169) [selector, pm-app. We then specify div in the first example or div.panel.panel-primary to animate each div](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=114.169) [defined within these CSS classes. Uncomment one or the other to try them.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=124.169) [Let's try out the slide transition. I'll uncomment that one.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=131.169) [Let's see it in action. Our panels now slide in from the left instead of just appearing all at once.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=136.169) [This gives the impression of route animation. Nice! But notice the animation of the Product List page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=143.169) [The header slides in nicely, but the data just pops in. What's that all about?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=150.169) [This occurs because the page first transitions, then waits for its data,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=155.169) [then displays the data when it arrives from the server. This would look nicer if we added a route resolver](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=161.169) [and prefetched the data. Then the entire page of data would slide in.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=167.169) [Another good reason to use route resolvers. Whether you use any animation, CSS-based animation,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=172.169) [or Angular's animation depends on your user interface design, your users, and your application requirements.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=180.169) [Next, let's look at what we can do with routing events.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=2&mode=live&start=188.169)

[Watching Routing Events](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live)

[Any time the user navigates in the application, the router generates routing events that we can use to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=2.369) [monitor, troubleshoot, or perform logic. These routing events include NavigationStart, which is triggered](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=9.369) [when navigation begins, RoutesRecognized, which is triggered when the router has found a valid path in](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=16.369) [the configuration that matches the parsed URL, NavigationEnd, which is triggered when navigation ends](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=22.369) [successfully, NavigationCancel, triggered when navigation is cancelled such as by a routing guard or](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=30.369) [redirect (we'll talk more about routing guards a little later in this course), and, of course,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=36.369) [NavigationError, which is triggered when navigation fails.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=42.369) [We can see these events occur if we enable tracing. To enable route event tracing, add an object as a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=47.369) [second argument after the route definitions in the root route configuration and set the enableTracing](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=55.369) [property to true. Let's try it out.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=63.369) [Our root route configuration is in the app-routing.module. Let's add enableTracing to this configuration.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=68.369) [We add an object as a second argument to the forRoot method after the route definitions and set the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=76.369) [enableTracing property to true.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=83.369) [View the app in the browser and open the developer tools. Refresh the page, and we see each routing event](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=86.369) [as it occurs. Scrolling up, as expected, the NavigationStart event comes first.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=94.369) [Then RoutesRecognized and NavigationEnd. The router assigns a sequential Id number to every navigation,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=101.369) [which we can use to match up a set of events. Now let's navigate to the Product List and clear the console.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=109.369) [Click the Edit button for one of the products, and we see the routing events log to the console.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=116.369) [Scrolling up, let's look closer at the NavigationStart. It has unique identifier and includes the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=121.369) [navigation URL. Next is the RoutesRecognized event. Since it is part of the same navigation operation,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=129.369) [it has the same identifier. Notice the difference here between the URL and urlAfterRedirects after redirects.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=137.369) [The URL has the original navigation URL, but the Product Edit route has a default child route that redirects](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=144.369) [to the info page, so the urlAfterRedirects includes that redirect.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=152.369) [The product information displayed here is from our product data service.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=158.369) [Our last routing event is NavigationEnd. It has the same identifier, URL, and urlAfterRedirects.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=163.369) [Feel free to add and remove the enableTracing option as desired. Keeping it off until you need it keeps](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=170.369) [unneeded information from appearing in the console. Route event tracing can be helpful as we get into more](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=176.369) [complex routing scenarios. There may be times that our routes don't work as expected.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=182.369) [Tracing is one tool in our toolbox for resolving routing issues. But we can also use these events in our](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=188.369) [code and perform an operation reacting to a routing event. Let's look at that next.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=3&mode=live&start=195.369)

[Reacting to Routing Events](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live)

[We can watch for routing events and react to them to display a spinner, log actions, or execute logic such as](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=2.736) [updating a page title. For example, we could display a spinner when the NavigationStart event occurs and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=9.736) [turn off the spinner on NavigationEnd. Any guesses on how we watch for routing events so we can react to them?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=17.736) [If you said by subscribing to an observable, you are correct. The Angular router has an events property](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=25.736) [that is an observable. We can subscribe to this observable to get notification of each routing event in the code.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=32.736) [We provide a function to the subscribe method. The routing event, which is of type Event, is passed into](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=40.736) [this function. In the both of the function, we can check the type of event and react accordingly.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=47.736) [Let's see what we can do.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=52.736) [Have you noticed the delay from the time we activate the Product Detail route](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=56.736) [to the time the Product Detail template is displayed? This is because we are using a resolver that retrieves](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=60.736) [the data during the navigation. Let's let the user know that something is happening during this delay](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=67.736) [by displaying a spinner. We'll turn the spinner on at the start of the navigation and turn it off when the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=73.736) [navigation completes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=79.736) [We want the spinner to appear for every route, so we'll start listening for routing events when our app first loads.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=81.736) [This means we'll add the code to the app.component. First, we add the appropriate router imports.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=88.736) [As we saw in the slide, we need Event to correctly type our routing event.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=95.736) [And we'll need each of the routing events that we want to match. We want to watch NavigationStart and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=99.736) [NavigationEnd, but what if an error occurs? We should watch NavigationError so we can turn off the spinner](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=106.736) [if an error occurs. And we should probably watch NavigationCancel as well.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=114.736) [Now let's define a property that specifies whether or not a route is loading.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=119.736) [We'll call it loading and set it to true. We'll bind to this property to turn our spinner on and off.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=123.736) [The router is already defined as a dependency in the class constructor so the Angular injector will provide](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=131.736) [an instance of the router when the application is loaded. Since we want to start watching routing events](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=138.736) [as soon as the application loads, we subscribe to the router's events in the constructor, router.events.subscribe.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=144.736) [We then define the function to execute each time we receive an event.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=152.736) [The current event is passed into this function. We'll call it routerEvent and define its type as Event.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=157.736) [When a routing event occurs, we'll call a method that checks the routerEvent.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=164.736) [We'll call it checkRouterEvent and pass in the event. Next, we create the checkRouterEvent method.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=168.736) [I'll paste the code, and we can talk through it. In this method, we check the type of navigation event.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=177.736) [If the event is NavigationStart, we set loading to true to turn on our spinner.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=182.736) [When the navigation ends, is cancelled, or has an error, we set loading to false to turn off our spinner.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=189.736) [The only thing left is to add the spinner that appears when the route is loading.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=196.736) [Here in the app.component template at the very top of the file, we'll display the spinner using a Bootstrap glyphicon.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=202.736) [We'll use ngIf to only display the spinner if the route is loading.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=210.736) [I included the style classes for the spinner in the starter files as part of styles.css.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=215.736) [The spinner style class increases the size of the glyph and positions it over the current display in the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=222.736) [center of the page. The glyphicon-spin style class uses CSS animation to spin the glyph.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=227.736) [Looking back in the browser, notice that we don't see the spinner when we display the Home page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=236.736) [That's because it's loading too quickly. We may expect to see this spinner when loading the Product List page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=243.736) [But, no. Any guesses as to why that may be? Let's look at it again.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=248.736) [Notice that the page header immediately appears, then the page waits for the data.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=254.736) [Since the page appears before the data, it again loads too quickly to display the spinner.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=259.736) [Now let's click on a product name to display the Product Detail page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=265.736) [There's our spinner. Yay! Note that if you are not seeing a spinner here, I'll talk about a fix in a moment.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=269.736) [How does this work? Recall that we are using a resolver to ensure that we retrieve the data before](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=277.736) [displaying the page. So the navigation starts, the spinner appears, the navigation waits for the data to be retrieved.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=283.736) [Then the navigation ends, the spinner disappears, and the Product Detail page appears.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=291.736) [This is also the case for the Product Edit page,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=297.736) [so we see the spinner here as well. If you want the spinner to appear while the products are loaded for](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=300.736) [the Product List page, you can add the spinner directly to the Product List page and turn it off when the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=307.736) [product property is populated. Or a better option is to create a resolver for the Product List.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=312.736) [If you are coding along and not seeing the spinner when displaying the Product Detail or Product Edit pages,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=319.736) [we may need to slow down the data retrieval. Recall that our application is using an in-memory Web API](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=325.736) [server to serve up the data. That may be occurring too fast for our spinner to appear.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=332.736) [So here in the product module, add a delay. This allows us to delay the loading of the data better](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=339.736) [simulating how the application will work when using an actual back-end web server.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=346.736) [Try it again and see if the spinner now appears. Let's finish up this module with some checklists we can](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=351.736) [use when we style, animate, and watch our routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=4&mode=live&start=356.736)

[Checklists and Summary](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=5&mode=live)

[Styling the active route helps the user track where they are in the application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=5&mode=live&start=2.112) [Styling the active route using the routerLinkActive directive. Be sure to specify the directive on the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=5&mode=live&start=7.112) [correct element. If the anchor tag is within a list element, the routerLinkActive must reside on that](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=5&mode=live&start=13.112) [parent list element. For an exact path match when determining the active route,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=5&mode=live&start=21.112) [use the routerLinkActiveOptions directive and set exact to true. This can prevent multiple routes from](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=5&mode=live&start=25.112) [appearing as active when using similar root path names.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=5&mode=live&start=34.112) [To animate route transitions](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=5&mode=live&start=38.112) [or at least give the appearance of animating the transitions, you can use CSS animation or Angular animation.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=5&mode=live&start=41.112) [CSS animation is easier because with one set of CSS, you can animate every component in the application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=5&mode=live&start=49.112) [But animating routes using Angular animation may be getting better with future releases.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=5&mode=live&start=57.112) [Enable route tracing any time you want to watch routing events in the console.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=5&mode=live&start=64.112) [This is especially helpful when troubleshooting routing issues. To turn on route tracing, add enableTracing](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=5&mode=live&start=68.112) [to the root router module configuration.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=5&mode=live&start=75.112) [We can react to routing events by subscribing to the router's events observable.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=5&mode=live&start=79.112) [Check the event type as needed and add code to react as desired. This is great for displaying a spinner](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=5&mode=live&start=84.112) [or logging actions.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=5&mode=live&start=92.112) [In this module, we examined how to improve the look of our application routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=5&mode=live&start=95.112) [We styled the selected route and animated route transitions. We watched our routes by enabling route event tracing.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=5&mode=live&start=100.112) [And we reacted to routing events to display a spinner in our example.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=5&mode=live&start=109.112) [We didn't add any routes in this module. But at this point, our application routes are looking pretty good.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=5&mode=live&start=115.112) [There is only one route remaining in this diagram that we have not yet implemented.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=5&mode=live&start=121.112) [Up next, let's look at how to configure and use secondary routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m8&clip=5&mode=live&start=126.112)

[Secondary Routes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=0&mode=live)

[Introduction](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=0&mode=live)

[Multiple routes displayed at the same time and at the same level of the hierarchy are referred to as peer,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=0&mode=live&start=0) [sibling, auxiliary, or secondary routes. Regardless of what we call them, they are useful for building more](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=0&mode=live&start=10) [complex user interfaces. Welcome back to Angular Routing from Pluralsight.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=0&mode=live&start=17) [My name is Deborah Kurata, and this module is about secondary routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=0&mode=live&start=22.5) [Secondary routes make it easy to display multiple panels or panes on the page, each containing different](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=0&mode=live&start=29) [content and supporting independent navigation such as for a dashboard like this.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=0&mode=live&start=36) [If each of these panels displayed a single component, then using nested components is sufficient.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=0&mode=live&start=42) [But imagine that each panel supports independent navigation. For example, click on a chart to show its](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=0&mode=live&start=50) [details or click on a country to drill down. Then each panel needs its own routing supported by one primary](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=0&mode=live&start=56) [and any number of secondary routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=0&mode=live&start=64) [In this module, we begin with an overview of secondary routes, what they are, and when to use them.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=0&mode=live&start=69) [We then look at defining a named RouterOutlet to place the secondary views, configuring secondary routes,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=0&mode=live&start=76) [and activating secondary routes. Lastly, we examine how to clear secondary outlets.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=0&mode=live&start=83) [In our sample application, we'll add a name secondary outlet to define a popup area on the page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=0&mode=live&start=90) [Then we'll add a secondary route to display a message log in that outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=0&mode=live&start=98) [Let's get started.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=0&mode=live&start=102)

[Using Secondary Routes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=1&mode=live)

[Before we look at how to use secondary routes, let's once again review how our primary routes work.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=1&mode=live&start=1.024) [This template lays out a navigation menu and other elements that make up the main page of the application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=1&mode=live&start=8.024) [We use the router-outlet directive to define where to place the routed component's template for display.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=1&mode=live&start=15.024) [This is called the primary outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=1&mode=live&start=21.024) [When the user views the application in the browser and routes to a component, that component's template is](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=1&mode=live&start=24.024) [displayed within the primary router-outlet. If we want to display another routable panel of information](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=1&mode=live&start=31.024) [at that same level of the route hierarchy, we define a second outlet and display a secondary set of routes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=1&mode=live&start=38.024) [in that outlet. A secondary route can have its own child routes, its own route parameters, and its own](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=1&mode=live&start=45.024) [secondary routes. In this example, we display a simple message log, but you can imagine the possibilities.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=1&mode=live&start=53.024) [We can use secondary routes to display a dashboard with multiple panels that each support multiple routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=1&mode=live&start=62.024) [We could build a multi-window application like Outlook or Gmail. We could display a second panel that allows](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=1&mode=live&start=69.024) [the user to take notes or make comments. We could display messages or other information to the user, and so on.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=1&mode=live&start=76.024) [Now let's start by defining the router-outlet for the secondary routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=1&mode=live&start=84.024)

[Defining a Named Router Outlet](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=2&mode=live)

[The activated component templates for our primary routes are displayed in our primary outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=2&mode=live&start=2.124) [And the activated components for our secondary routes are displayed in the secondary outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=2&mode=live&start=8.124) [Hmm. How does the router know where to place which content? We must give each secondary router-outlet a name.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=2&mode=live&start=15.124) [Then we use that name to tell the router to place the secondary content in a specific outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=2&mode=live&start=24.124) [We can define any number of secondary outlets at the same level of the hierarchy.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=2&mode=live&start=30.124) [But each must have a unique name. Let's define a named router outlet for our secondary routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=2&mode=live&start=36.124) [For our sample application, we want to define a panel on the right side of our main page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=2&mode=live&start=45.124) [We can then route any component to that panel using a secondary route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=2&mode=live&start=50.124) [The elements for our main page are in the app.component template.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=2&mode=live&start=55.124) [Let's divide the container into two columns using some Bootstrap CSS style classes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=2&mode=live&start=61.124) [Within the div container, we first define a row, then we make the first column for our primary outlet](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=2&mode=live&start=65.124) [10 units wide and the second column 2 units wide. We'll add the secondary outlet within the right set of columns.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=2&mode=live&start=72.124) [Every secondary outlet needs a name. We'll call this one popup because we'll use it to route to any](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=2&mode=live&start=80.124) [component we want to pop up here on the right. We could add any number of other named router-outlets,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=2&mode=live&start=87.124) [but one is enough for our purposes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=2&mode=live&start=93.124) [We can view the result in the browser, but the only difference we should see is that our primary content](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=2&mode=live&start=97.124) [is a bit narrower. Now let's configure our secondary route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=2&mode=live&start=103.124)

[Configuring Secondary Routes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live)

[Secondary routes are configured similar to primary routes. We define a path for the route and specify a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=1.967) [reference to a component. But for secondary routes, we also define an outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=8.967) [This is where we identify the name of the secondary outlet where this component's template will appear.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=14.967) [So when this secondary route is activated, the message component's template is displayed in the secondary](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=20.967) [outlet named popup. Let's give it a try.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=27.967) [In our sample application, we want to display messages in the popup outlet when the user clicks Show Messages.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=30.967) [I've already created the message component and defined the template inline here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=39.967) [The template displays a set of messages managed by a MessageService.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=44.967) [The MessageService demonstrates how to use a service to share data across components.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=48.967) [I created the MessageService to share the message data with all of the features of the application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=54.967) [Any part of the application can access this service and add a message that appears in the popup outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=60.967) [Since the messages are their own feature, I defined a feature module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=66.967) [It is here that we'll configure the route for the MessageComponent.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=71.967) [We first add an import statement to import RouterModule from the @angular/router library.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=75.967) [Then we add the RouterModule to the list of imports. Remember how we configure a route?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=82.967) [Since this is a feature module, we use the forChild method and pass in an array of route definitions.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=89.967) [Since there is only one component in this feature module, we only need one route definition at this time.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=95.967) [We'll specify a path of messages, reference the component, and specify the outlet to use to display this](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=102.967) [route, in this case, the router-outlet named popup. That's it. But we can't try it until we activate this](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=110.967) [secondary route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=118.967) [Before these most recent changes, our sample application route hierarchy looked something like this.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=120.967) [We have a primary router-outlet defined for or primary routes. Our product child routes appear in that](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=126.967) [outlet as well because their parent route is component-less. We have another router-outlet defined for](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=132.967) [our Product Edit child routes. We didn't have to add a name to the child outlet because the router could](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=139.967) [tell which outlet to use based on the route hierarchy. With secondary routes, the outlet is defined at](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=146.967) [the same level in the route hierarchy as other routes, so we give the secondary outlet a name.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=152.967) [We then configure a secondary route specifying a path, a component, and an outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=159.967) [In our example, we configured a route for the MessageComponent. We could configure any other secondary](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=164.967) [routes to use this same outlet. For example, we could define Product Summary component and display a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=171.967) [summary in the popup as the user clicks on products in the Product List or as the user edits the product.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=178.967) [When routing to a named router-outlet, the URL in the address bar reflects both the primary and secondary](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=184.967) [route paths. The secondary paths appear within parentheses after the primary path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=192.967) [Inside the parentheses is the name of the outlet, a colon, and the secondary route path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=198.967) [This may look a little odd, but the Angular router understands this syntax and navigates appropriately.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=204.967) [So how do we activate a secondary route?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=3&mode=live&start=211.967)

[Activating Secondary Routes: RouterLink](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live)

[As with any other type of route, we activate a secondary route in the HTML using the routerLink directive.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=1.932) [But the syntax is a bit different. We define an objects as the first element in the link parameters array](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=9.932) [and assign the outlets property to a name and value pair where the name is the name of the outlet to use](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=16.932) [and the value is another link parameters array specifying the secondary path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=21.932) [We can route to multiple outlets at once. This code activates the Product Edit route in the primary outlet](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=28.932) [and a Product Summary route in the secondary outlet named popup.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=34.932) [However, at the time of this recording, this syntax was not working.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=40.932) [The router was adding an extra slash between the primary route and the secondary route when building the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=44.932) [URL, and then it could not understand the result. Hopefully this will be fixed in an Angular update.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=49.932) [So let's give this first syntax a try.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=55.932) [We are looking at the app.component template. To activate a secondary route, let's modify the Show Messages](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=60.932) [menu option to display the MessageComponent in the popup outlet. We'll add the routerLink directive and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=67.932) [assign it to a link parameters array. In the first element of that array, we define an object,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=75.932) [specify the outlet's property, and provide a key and value pair. The key is the name of our outlet,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=80.932) [which is popup in our example, and the value is another link parameters array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=87.932) [In the first element of that array, we specify a path of messages.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=93.932) [Let's double-check our syntax here because there are lots of braces.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=99.932) [Link parameters array, object, object, link parameters array, okay.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=103.932) [Viewing the application in the browser, click on the Show Messages option, and the Message Log appears.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=108.932) [Notice the URL in the address bar. It includes the primary route and the secondary route in parentheses.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=114.932) [Click Log In and log in using any name and password.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=122.932) [Then a message is logged to the message list. Notice again the URL in the address bar.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=125.932) [The primary route was changed, but the secondary route stays the same.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=131.932) [Now that the messages appear, we want to change the menu to say Hide Messages.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=136.932) [Clicking the Hide Messages option or the X here should hide these messages.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=142.932) [To modify the menu based on the state of the display, we need a property that tracks when the messages](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=147.932) [are shown and when they're not. We need code to set that property when routing to the secondary route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=154.932) [So we'll need to activate the secondary route with code instead.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=4&mode=live&start=161.932)

[Activating Secondary Routes: In Code](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live)

[In addition to routing to a secondary route in HTML, we can route in code.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=2.965) [The syntax is similar to routing with a routerLink directive. We define an object as the first element of](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=7.965) [the link parameters array and assign the outlets property to a name and value pair where the name is the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=14.965) [name of the outlet and the value is another link parameters array specifying the secondary path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=19.965) [Just like with the routerLink directive, we can route to multiple router outlets at once.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=26.965) [We specify the primary route in the first elements of the link parameters array, and then any secondary routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=32.965) [This code activates the Product Edit route in the primary outlet and the Product Summary route in the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=38.965) [secondary outlet named popup. Also like the routerLink directive, at the time of this recording,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=45.965) [this syntax was not working. The router was adding an extra slash between the primary route and the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=51.965) [secondary routes when building the URL. Hopefully this will be fixed in an update.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=56.965) [As a workaround, we can specify both the primary and secondary routes using the outlets property.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=62.965) [Even though we don't give it a name, the primary outlet has a name called primary.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=69.965) [Here we define key and value pairs for the primary outlet and for the secondary outlet named popup](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=75.965) [specifying a link parameters array for each. Another option is to use the navigateByUrl method and build](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=82.965) [the URL manually. But it is often better to let the router build the URL from the link parameters array](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=89.965) [using the navigate method. Let's try activating the secondary route in code.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=95.965) [For our sample application, we want the Show Messages option to activate the second route in code and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=102.965) [set a property that tracks whether the messages are shown. When the messages are shown, we want to instead](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=108.965) [display the Hide Messages menu option. Selecting the Hide Messages option should clear the secondary route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=115.965) [Let's start by adding a property to the MessageService that tracks whether the messages are displayed.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=122.965) [We'll call it isDisplayed and set its initial value to false. Since this property is defined within a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=128.965) [service, this property is shared and available to any class that injects this service.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=135.965) [Here again is the Show Messages menu option in the app.component template.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=142.965) [Instead of using the routerLink to activate the secondary route, let's call a method on click.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=147.965) [We'll name the method displayMessages. Be sure to add the opening and closing parentheses to indicate that](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=152.965) [this is a method. In the AppComponent class, we'll add that displayMessages method.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=159.965) [It takes no parameters and returns void. In the method, we route to the secondary route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=166.965) [We call this.router.navigate passing in a link parameters array. In the first element of that array,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=171.965) [we define an object, specify the outlet's property, and provide a key and value pair.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=179.965) [The key is the name of our outlet, which is popup in our example, and the value is another link parameters array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=185.965) [In the first element of that array, we specify the path of messages.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=193.965) [Next, we want to set our new isDisplayed property to true. That's in the MessageService.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=198.965) [So we need to import that service, add it to the constructor, and then we can set the property here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=203.965) [While we're here, let's add a hideMessages method that simply sets isDisplayed to false.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=210.965) [We'll add the code to this method to clear the secondary outlet in the next clip.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=216.965) [Going back to the app.component template, we can now use the isDisplayed property to display the correct](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=221.965) [menu option text. We'll add an ngIf directive here to display the showMessages option if isDisplayed is false.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=228.965) [Then copy this option and paste it. We change the text to Hide Messages and display this option if](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=235.965) [isDisplayed is true. When the user clicks this option, we call our new Hide Messages method.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=242.965) [Let's check it out in the browser. Clear the URL to start fresh.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=248.965) [Click Show Messages, and our messages appear in the secondary outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=252.965) [Our menu option changes to Hide Messages. Click Hide Messages, and the menu option changes to Show Messages.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=257.965) [But the secondary router-outlet is not cleared. Let's hook that up next.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=5&mode=live&start=265)

[Clearing Secondary Outlets](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live)

[We may want to clear a secondary outlet and close its content.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=2.095) [We can do that with the routerLink directive. We define an object as the first element in the link](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=7.095) [parameters array. We set the outlets property to a key and value pair where the key is the name of the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=12.095) [outlet, and the value is null. We clear a secondary outlet in code using the navigate method and the same](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=18.095) [link parameters array. We can also use navigateByUrl. Notice here that we simply leave off the outlet](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=27.095) [information so this method clears the secondary outlets and navigates to the login route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=34.095) [Let's try clearing our secondary outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=41.095) [We already have a hideMessages method here in the app.component. In this method, we'll clear the secondary](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=45.095) [outlet by navigating to the outlet with a value of null. We'll call this.router.navigate passing in a link](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=51.095) [parameters array. The first array element is an object with the outlet's property set to a key and value pair.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=60.095) [The key is the name of our secondary outlet, which is popup, and the value is null.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=65.095) [Let's check it out. Clear the URL in the address bar, then click Show Messages.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=72.095) [Log in to add a message to the Message Log. Then click Hide Messages, and the secondary outlet is cleared.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=78.095) [Click Show Messages to show them again. Notice this X here. The user should be able to clear this outlet](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=87.095) [clicking on the X as well.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=93.095) [In the message.component template, we have a button containing an X that should clear the message log from](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=97.095) [the secondary outlet effectively closing it. We could use the routerLink directive here, but this code](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=103.095) [calls a method when the user clicks a button. In the close method, we call this.router.navigate and assign](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=110.095) [the popup secondary outlet to null. Then we set the messageService.isDisplayed property to false to specify](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=118.095) [that the Message Log is no longer displayed. Let's check it out in the browser.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=125.095) [Clear the URL to start fresh.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=131.095) [Then let's log in so we have a message to display.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=133.095) [Click Show Messages, and the Message Log appears in the secondary outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=137.095) [Click Hide Messages, and the Message Log is cleared from the secondary outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=141.095) [Click Show Messages again. This time click the X to close the Message Log.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=146.095) [Looks like we have our secondary route working. Note that if we were only going to show messages here](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=151.095) [in this secondary outlet, we should use a nested component instead.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=158.095) [Secondary outlets only make sense if we are going to display multiple routed components within the outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=162.095) [For example, we could replace the messages with a product summary when editing the product.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=168.095) [Now let's finish up this module with some checklists we can use when working with secondary routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=6&mode=live&start=174.095)

[Checklists and Summary](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=7&mode=live)

[To define secondary routes, add another RouterOutlet directive within a template that contains a RouterOutlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=7&mode=live&start=2.993) [To distinguish the outlets for secondary routes from outlets from primary routes, each secondary RouterOutlet](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=7&mode=live&start=9.993) [must be uniquely named.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=7&mode=live&start=16.993) [When configuring a secondary route, add the outlet property to the route definition and set that outlet](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=7&mode=live&start=19.993) [property to the name of the associated RouterOutlet directive. The activated component's template will then](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=7&mode=live&start=26.993) [appear in that outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=7&mode=live&start=32.993) [Activate a secondary route using a link parameters array with an object as the first element in the array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=7&mode=live&start=35.993) [Set the object's outlets property to a key and value pair. As the key, define the name of the outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=7&mode=live&start=43.993) [As the value, define another link parameters array containing the URL segments for the secondary route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=7&mode=live&start=50.993) [In a template, activate using the routerLink directive. Activate in code using the router's navigate method.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=7&mode=live&start=58.993) [In either case, the link parameters array syntax is the same.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=7&mode=live&start=66.993) [Clear a secondary outlet using a link parameters array with an object as the first element in the array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=7&mode=live&start=70.993) [Set the object's outlets property to a key and value pair. As the key, define the name of the outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=7&mode=live&start=77.993) [As the value, specify null. In the template, clear the secondary outlet using the routerLink directive.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=7&mode=live&start=84.993) [Clear in code using the router's navigate method. In either case, the link parameters array syntax is again the same.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=7&mode=live&start=92.993) [In this module, we began with an overview of secondary routes. We then defined a named RouterOutlet for the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=7&mode=live&start=100.993) [secondary route, configured this secondary route, and activated the secondary route to display it in its](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=7&mode=live&start=108.993) [named outlet. We then look at how to clear a secondary outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=7&mode=live&start=115.993) [We now have our secondary route in place. Yay! Up next, let's talk about how to protect our routes with](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=7&mode=live&start=120.993) [route guards.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m9&clip=7&mode=live&start=128.993)

[Route Guards](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=0&mode=live)

[Introduction](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=0&mode=live)

[We may need to check whether route navigation is permitted for security, authorization,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=0&mode=live&start=0) [or monitoring purposes, or prevent the user from leaving a route without confirmation.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=0&mode=live&start=8) [Welcome back to Angular Routing from Pluralsight. My name is Deborah Kurata, and in this module, we guard our routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=0&mode=live&start=15) [Yeah, I had to do it. I had to go literal here with a route guard.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=0&mode=live&start=24) [Our users aren't getting past this guy unless they meet the specified criteria.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=0&mode=live&start=29) [Did they have appropriate authority to access this route? Have they completed their task before leaving](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=0&mode=live&start=34) [this route? We achieve all of this with route guards.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=0&mode=live&start=39) [The Angular router provides several types of guards including canActivate to guard navigation to a route,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=0&mode=live&start=45) [canActivateChild to guard activation to a child route, canDeactivate to guard navigation away from the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=0&mode=live&start=52) [current route, canLoad to prevent asynchronous routing, and resolve to prefetch data before activating a route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=0&mode=live&start=60) [We covered resolve earlier in this course in the Prefetching Using Route Resolvers module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=0&mode=live&start=70) [And we build a canLoad guard in the next module when we cover asynchronous routing.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=0&mode=live&start=76) [In this module, we begin with an overview of route guards, what they are, and how to build them.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=0&mode=live&start=83) [We then build a canActivate guard and look at how to share data with a guard.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=0&mode=live&start=90) [We discuss the canActivateChild guard and build a canDeactivate guard.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=0&mode=live&start=96) [In our sample application, we guard access to the product routes ensuring that the user is logged in before](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=0&mode=live&start=101) [viewing or updating product information. And now that we've grouped our product routes, defining the guard](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=0&mode=live&start=109) [on the parent guards each of the child routes. And we guard leaving the Product Edit route warning the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=0&mode=live&start=114) [user if they try to leave before saving their changes. Let's get started.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=0&mode=live&start=121)

[Using Route Guards](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live)

[There may be times that we want to limit access to a route. We want routes only accessible to specific users](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=0) [such as an administrator for example. Or we require that the user log in before accessing a specific route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=8.687) [We may want to warn the user before leaving a route, such as asking whether to save before navigating away](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=16.687) [from a edit page with unsaved changes. We may want to retrieve data before accessing a route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=23.687) [Earlier in this course, we prefetched the data for the Product Detail and Product Edit routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=30.687) [For each of these, we use a route guard. The router first executes the canDeactivate guards for the current](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=35.687) [route to determine whether the user can leave that route. If a feature module is loaded asynchronously](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=43.687) [as we discuss in the next module, the canLoad route guard is checked before the module is loaded.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=51.687) [The router then checks the canActivateChild guards and then the canActivate guards.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=57.687) [After all other route guards are checked, then the resolvers are executed.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=63.687) [It makes sense that this is last because we would not want to execute a resolver and retrieve the data for](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=68.687) [a route until we are certain that the user can access that route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=75.687) [If any guard returns false, all pending guards are cancelled, and the requested navigation is cancelled.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=79.687) [Now that we know how guards are processed, let's look at how to build one.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=86.687) [A route guard is often built as an Angular service similar to how we create any other Angular service.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=92.687) [But we can also create it as a function as we saw with the resolver earlier in this course.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=100.687) [To build a guard as a service, we export a class, decorate it with the injectable decorate, and import what we need.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=105.687) [To make this service behave as a route guard, we implement the interface appropriate for the guard type.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=114.687) [We then need to import that guard type from the @angular/router library.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=121.687) [When we implement the interface, we must define the associated method.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=125.687) [In this example, the method is canActivate. For simple cases, this method can return a Boolean value true](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=131.687) [to activate the route and false to cancel the route activation. For more complex cases, we could return](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=137.687) [an observable or a promise from this method. If the returned value is a promise or an observable,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=144.687) [the router will wait for that promise or observable to complete before proceeding with the navigation.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=150.687) [Because the guard class is a service, we need to register the service provider with Angular's injector.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=157.687) [Unlike other services, however, the guard service provider must be provided at the Angular module level,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=164.687) [not in a component. By providing this service at the module level, the router can use these services](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=171.687) [during the navigation process. We register the service by adding it to the providers array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=178.687) [To use the guard on a route, add the guard to the route definition in the configuration.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=184.687) [Simply add a property for the desired type of guard. In this example, it is the canActivate guard, so it is](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=191.687) [the canActivate property. Set that property to an array containing a list of references to the guard services.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=198.687) [In this example, we define only one, but we can add multiple guards on any route at any level of the route hierarchy.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=205.687) [We can also share guards across a set of routes. For example, say we want to guard each of our product routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=214.687) [We could add the guard to each child route, or we could just add the guard to the parent route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=221.687) [Adding a guard to a parent route guards each of its children.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=228.687) [So guarding a route involves three steps--building a guard class as an Angular service, registering the guard](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=232.687) [in an Angular module, and adding the guard to the desired route or routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=240) [Let's try this out and build a canActivate guard.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=1&mode=live&start=245.5)

[CanActivate Guard](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live)

[As its name suggests, we use the canActivate guard any time we want to check criteria before activating a route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=1.597) [It is commonly used to limit access to specific users. It is the primary mechanism for adding permissions](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=10.597) [to the application. We can also use the canActivate guard to ensure prerequisites are met such as requiring](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=17.597) [our user to log in before accessing a route. The canActivate guard is called any time the URL changes and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=23.597) [matches the associated route, even if only the route parameters change.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=32.597) [Let's give it a try.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=37.597) [For our sample application, we want a canActivate guard that ensures the user is logged in before accessing](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=40.597) [any of the product routes. We begin by creating a new file for the guard service.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=47.597) [Checking for a login is related to the user, so we'll add the guard to the user folder.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=53.597) [Select New File and name it auth-guard.service.ts where auth is short for authorization.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=58.597) [I'll paste the boilerplate code for the service, and we can talk through it.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=68.597) [This class is called AuthGuard since this particular guard should be checked before activating a round.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=73.597) [We implement canActivate and create a canActivate method. The canActivate method has two parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=80.597) [The ActivatedRouteSnapshot provides information about the current about-to-be activated route at a particular](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=87.597) [moment in time. We've used the ActivatedRouteSnapshot before in this course to get information on the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=95.597) [current route, such as the route parameters and route data. The second parameter is the RouterStateSnapshot.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=100.597) [It provides access to the entire router state. What do we want this method to do?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=106.597) [Well, if the user is logged in, the route can be activated, so we should return true.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=113.597) [If the user is not logged in, we should return false and instead route the user to the Log In page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=119.597) [So, first, we need to determine if the user is currently logged in.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=125.597) [Luckily for us, the authorization service I provided with the starter files for this course has code to do just that.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=130.597) [It provides a method called isLoggedIn that returns true if there is a current user.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=136.597) [Otherwise, it returns false. The double bangs here effectively coerce the object to a Boolean value.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=142.597) [Back in the guard class, we want to call the isLoggedIn method from the authorization service, so we need](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=149.597) [an instance of that authorization service here. If we want to route the user to the Log In page,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=155.597) [we need the router as well. We define an import statement for the authorization service, add Router to the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=161.597) [router import statement, and specify them both as dependencies in our constructor.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=169.597) [Since there are several steps we want to perform here, we'll build a method.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=174.597) [I'll paste the code, and we can walk through it. The method is called checkLoggedIn and returns a Boolean value.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=179.597) [It calls the authorization service isLoggedIn method and returns true if the user is logged in.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=186.597) [The requested route is then activated. Otherwise, it redirects the navigation to the Log In page and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=192.597) [returns false cancelling the navigation to the requested route. We call this method in the canActivate method.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=198.597) [Because this class is a service, we need to register it with Angular's injector.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=206.597) [Since the auth-guard is part of the user feature, we'll add it to the user.module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=212.597) [We start by importing our guard service, then adding it to the providers array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=217.597) [Now the router can get an instance of this service during the navigation process.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=222.597) [Our last step is to attach the guard to any route that needs it. We want it to guard each of our product](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=228.597) [routes in the product.module. We add the import statement for the guard, then we can attach canActivate](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=233.597) [to the desired routes. We could add the guard to each of the product routes, or we could just add the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=241.597) [guard to the parent route. By adding the guard to the parent route, as we add more product routes over time,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=247.597) [the children are automatically guarded.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=253.597) [Let's try this out in the browser. Ensure we are not logged in,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=257.597) [then click Product List. The Log In page appears for the user to log in.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=261.597) [Click Home to navigate back to the Welcome page and click Add Product.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=266.597) [We are again navigated to the Log In page. Our guard works! If we then log in using any Id and password,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=271.597) [we are taken to the Product List page. But we were trying to get to the Add Product page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=279.597) [Regardless of where the user was trying to go, we always route them to the Product List page after logging in.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=286.597) [That's because here in the login.component, we hard code in navigation to the products route, which takes us](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=292.597) [to the Product List page. Wouldn't it be better if we kept track of where the user was trying to go,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=299.597) [share that information with the route guard, and returned the user there after logging in?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=306.597) [Yes, yes it would be better. So how do we share data with the route guards?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=2&mode=live&start=311.597)

[Sharing Data with a Guard](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live)

[There are several ways to share data across our routes and with our route guards.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=2.064) [We can define route parameters to pass data from a route to its routed component.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=7.064) [We can read route parameters from within our route guards using the ActivatedRouteSnapshot parameter.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=12.064) [This is a great technique if the guard requires the route parameters, for example, if an edit route guard](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=19.064) [wants to confirm the user has access to a specific product based on the product Id provided on the URL.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=25.064) [Our route can prefetch data required by the route component and share that data with its child routes using a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=32.064) [resolver service. However, the resolver is executed after the other route guards.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=39.064) [So the other route guards don't have access to that prefetch data.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=46.064) [Attempting to access prefetch data will always return undefined. So using resolve won't work for sharing](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=51.064) [data with a guard. We could, however, use the route's data property as shown earlier in this course if we](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=58.064) [want to pass static data to a guard. A common way to share data in an Angular application is to define](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=65.064) [properties in a service. Since a service is a singleton, there is only one instance that is shared among](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=71.064) [all code that uses the service. If we define properties in that service to hold data, those properties and,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=78.064) [hence, that data are shared everywhere including our guards. For our specific scenario, we have a route](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=85.064) [guard on the product routes that redirects to the Log In page if the user is not logged in.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=93.064) [We want to retain the user's original requested URL so that after the login, we can route the user based on](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=99.064) [that URL. So let's try the Angular service approach.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=107.064) [We already have an authorization service that was provided with the starter files.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=112.064) [Let's add a property to that service to track our redirect URL.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=117.064) [This property is now accessible to any class that injects this authorization service.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=122.064) [In our route guard service, we are already injecting our authorization service, so we can set this](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=127.064) [redirectUrl before navigating to the login route. What do we set it to?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=134.064) [Well, let's start by assuming we'll pass it into this method, add a URL string parameter, then set the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=140.064) [redirectUrl to this passed-in URL. Now we, of course, see an error here where we are calling this function.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=147.064) [You'd think that one of these two parameters here must have the URL, right?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=153.064) [Let's try the ActivatedRouteSnapshot. Yes, it has a URL, but it is of type UrlSegment array.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=159.064) [This snapshot has separated the URL into its individual segments.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=167.064) [We really just want the URL string and don't want to try to build it from the URL segments.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=172.064) [Let's try the RouterStateSnapshot. Its URL property is the string containing the entire URL.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=177.064) [Great! Next, we change the login method in the login component.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=184.064) [Instead of hardcoding it to route to the products route, we'll use the redirectUrl property from our](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=190.064) [authorization service. This class already injects the service, so we can simply access the property.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=196.064) [I'll paste the code, and we can talk through it. First, we check whether there is a redirectUrl.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=202.064) [There won't be if the user clicks the login menu option before accessing any product routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=209.064) [If there is a redirectUrl, we use the navigateByUrl method to navigate to that URL.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=214.064) [Otherwise, we navigate to the Product List page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=221.064) [Now let's check it out in the browser. Start on the Home page and ensure you are now logged in.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=225.064) [Click Add Product,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=231.064) [and the Log In page displays. Enter any Id and password and click Log In.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=232.064) [The Add Product page appears. Yes! Use a service any time you need to share data across the application,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=238.064) [and it's great for sharing data with our route guards.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=246.064) [From this point onward, it's going to be more of a challenge to try out anything in our application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=250.064) [Every time the browser reloads, it will require that we log in. You may want to consider removing this](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=256.064) [guard as you add features or debug your application. And it's easier to comment out since it's added only](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=262.064) [once to the parent route. Just don't forget to put it back in before deployment.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=269.064) [Next, let's look at the canActivateChild guard.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=3&mode=live&start=275.064)

[CanActivateChild Guard](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=4&mode=live)

[The canActivateChild guard is similar to the canActivate guard except that it is called when a child of the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=4&mode=live&start=2.195) [route is activated and not the route itself. It is commonly used to limit access to child routes and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=4&mode=live&start=9.195) [ensure prerequisites for child routes are met. The canActivateChild guard is called any time the URL changes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=4&mode=live&start=16.195) [and matches the associated child route even if only the route parameters change.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=4&mode=live&start=24.195) [The key difference between the canActivate guard and the canActivateChild guard is that the canActivate guard](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=4&mode=live&start=30.195) [will not re-execute if only the child route is changed. Let's look at some scenarios.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=4&mode=live&start=38.195) [If the user is on the Product List page and clicks the Add Product menu option, only the child route changes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=4&mode=live&start=44.195) [The canActivate guards on the parent route are not executed again, but any canActivateChild guards are.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=4&mode=live&start=52.195) [Same thing for our tabs. Any canActivate guards on the parent Product Edit route are not re-executed when](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=4&mode=live&start=59.195) [changing tabs. But any canActivateChild guards are. Building a canActivateChild guard is very similar to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=4&mode=live&start=67.195) [building a canActivate guard so we won't walk through an example here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=4&mode=live&start=75.195) [Let's move on and look at the canDeactivate guard.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=4&mode=live&start=79.195)

[CanDeactivate Guard](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live)

[The canDeactivate guard checks criteria before leaving a route. It is commonly used to check for unsaved](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=2.362) [changes before leaving an edit page and obtaining a confirmation from the user before leaving an incomplete](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=9.362) [operation and, therefore, potentially losing work. The canDeactivate guard is called any time the URL](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=15.362) [changes and matches a different route even if the only difference is the route parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=22.362) [Note that the canDeactivate only works when navigating within the Angular application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=29.362) [It does not check the guard is the user navigates to an entirely different site or closes the browser.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=34.362) [Let's build a canDeactivate guard.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=41.362) [Let's navigate to the edit page and make some changes. Currently if the user navigates away from this page,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=46.362) [their changes are lost. Bummer! It would be more user friendly to notify them that they have unsaved](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=52.362) [changes and allow them to return to save their data before navigating away.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=58.362) [Let's build a product edit guard to do just that.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=63.362) [We begin by creating a new file for the guard service. This service is specifically for checking unsaved](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=66.362) [changes to product data, so we'll add the guard to the products folder.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=74.362) [We select New File and name it product-guard.service.ts.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=78.362) [I'll paste the start of the code, and we can talk through it. Here we define a ProductEditGuard.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=86.362) [This particular guard should be checked before leaving the Product Edit route, so we implement CanDeactivate.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=92.362) [The CanDeactivate interface is a bit different. It is a generic interface meaning it requires a generic parameter.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=99.362) [The generic parameter identifies the type of component that will use this guard.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=107.362) [We plan to use this guard with a ProductEditComponent's route, so we specify ProductEditComponent as the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=113.362) [generic parameter. Next, we implement the canDeactivate method.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=119.362) [The canDeactivate method passes the component in as its parameter.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=125.362) [The parameter must be the same component type defined in the generic parameter here.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=129.362) [For simple cases, this method can return a Boolean value--true to deactivate the route and continue to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=135.362) [the new route, false to cancel the route deactivation and return to the current page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=140.362) [Now that we have the basic code in place, what do we need this guard to check?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=146.362) [Our ProductEditGuard must determine if there are any unsaved changes to the product data.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=151.362) [If so, it should display a warning and allow the user to decide whether to proceed or return to the edit](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=157.362) [page to save their changes. Since the ProductEditComponent is passed into the canDeactivate method,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=164.362) [we can access any of its properties or methods. Let's take a look at the product-edit.component.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=171.362) [How do we determine here whether the product data has changed. Recall that our edit feature is not just one form.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=177.362) [We have a form for the Info tab and another form for the Tags tab, so we can't just ask the form for its](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=182.362) [dirty state. Instead, we need to implement something ourselves. There are several ways we can track the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=191.362) [dirty state of the product data. For our example, we'll keep a copy of the original product data and check](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=197.362) [the updated product data against the copy to see if the user has made changes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=204.362) [Let's start by defining two properties--currentProduct of type IProduct and originalProduct also of type IProduct.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=209.362) [We don't want any external code to use these properties, so we marked them as private.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=218.362) [The product property here retains the product being edited. We could change this property everywhere to use](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=223.362) [our new currentProduct property. But let's instead just change the productProperty to be a getter and setter.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=229.362) [The getter returns the currentProduct. The setter sets the currentProduct to the passed-in value.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=235.362) [This is the product that the user edits. The setter then uses Object.assign to make a copy and retain the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=242.362) [original values for comparison. Note that it is important that the currentProduct property be the value](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=250.362) [and the originalProduct property be the copy. That's because the parent ProductEditComponent and each](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=256.362) [child edit component shares the currentProduct object instance that we obtained from the resolver we created](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=264.362) [earlier in this course. Next, we need a property that returns whether or not the product data was changed.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=271.362) [We'll define an isDirty property with a getter that returns true if the originalProduct does not match](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=278.362) [the currentProduct. There are numerous ways to compare the two products, but we selected a simple technique](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=285.362) [that works for our simple case. The getter here compares the stringified version of the originalProduct](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=292.362) [with a stringified version of the currentProduct. Be careful if you use this approach to ensure that the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=299.362) [properties of the object are always defined in the same order. Otherwise, even if the data may match,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=304.362) [the stringified objects won't match. A more versatile technique is to walk through each property of both](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=311.362) [products and compare their values. But this will work for our simple example.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=318.362) [Now we can use this isDirty property in our guard.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=324.362) [Going back to our guard, the component is passed into the canDeactivate method so code in the method can](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=328.362) [access the component properties. We'll access our new isDirty property.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=335.362) [I'll paste the code, and we can walk through it. If the component isDirty property is true, the code first](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=340.362) [pulls the productName from the component products property. If the value is null, it sets the name to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=346.362) [New Product. We then use a confirm to notify the user and request a confirmation before navigating away](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=353.362) [from the Product Edit page or navigating to a different product.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=359.362) [In the confirmation message, we use a back tick to define a template literal and display the productName](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=364.362) [within the confirmation message. We return the user's response. If the user clicks OK, the current route is](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=371.362) [deactivated, and the requested route is activated. If the user selects to cancel, the navigation is cancelled.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=378.362) [If the product data is not changed, this method returns true, and the route is deactivated.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=385.362) [Before we can use this guard, we need to register the service provider in an Angular module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=392.362) [Since this guard checks product data, we register the service in the product feature module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=397.362) [We import the guard using an import statement, then register the provider in the list of providers.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=403.362) [The last step is to tie the guard to the appropriate route. We'll add it to the parent Product Edit route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=410.362) [That way, it will guard both child routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=416.362) [Let's check it out in the browser. We log in,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=420.362) [navigate to the edit page, make a change, and select Product List.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=422.362) [We see our confirmation dialog with the name of the product. The user can then click OK to continue to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=427.362) [navigate away or click Cancel to return to the edit page. Cool!](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=434.362) [Let's click OK,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=438.362) [and we lose our changes. Now let's try it again. Click Edit,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=440) [make a change, click Save to save our changes, and we again see the dialog.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=444.362) [That's not right. We see this dialog even if we save because we are not resetting our current product and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=451.362) [original product after the save. Let's do that now to finish this implementation.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=457.362) [In the product-edit.component, let's add a reset method that resets the data after completing an operation.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=464.362) [Here we clear our validation structure, currentProduct, and originalProduct.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=471.362) [We then call this method onSaveComplete.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=476.362) [Let's try it one more time. Log in,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=479.362) [select to edit a product,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=483.362) [make an edit, and select Product List to ensure our confirmation message still works.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=484.362) [Click Cancel, then save changes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=490.362) [We are then returned to the Product List page without a confirmation dialog.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=493.5) [And our change was saved. It works! Let's finish up this module with some checklists we can use when](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=498) [working with route guards.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=5&mode=live&start=504)

[Checklists and Summary](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=6&mode=live)

[To build a route guard, build a service, implement the guard type, CanActivate in this example,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=6&mode=live&start=2.333) [and create the associated method. Like every other type of Angular service, register the guard service provider.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=6&mode=live&start=9.333) [Guards must be registered in a module, not a component. Lastly, add the guard to the desired route in the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=6&mode=live&start=17.333) [route configuration.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=6&mode=live&start=24.333) [In this module, we began with an overview of route guards. We build a canActivate guard and saw how to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=6&mode=live&start=26.333) [share data with a guard using a service. We briefly covered canActivateChild guards and built a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=6&mode=live&start=34.333) [canDeactivate guard.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=6&mode=live&start=40.333) [We added an authorization guard to the parent product route to protect all of the children and ensure the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=6&mode=live&start=43.333) [user logged in before accessing any product data. We also added a canDeactivate guard to the Product Edit](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=6&mode=live&start=49.333) [parent route to ask the user for confirmation before navigating away with unsaved changes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=6&mode=live&start=56.333) [Our route hierarchy is now complete. Up next, lazy loading.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m10&clip=6&mode=live&start=63.333)

[Lazy Loading](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live)

[Introduction](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live)

[First impressions matter. We want to minimize the time between the user's request for our application and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=0.5) [the display of its first template. Welcome back to Angular Routing from Pluralsight.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=9.5) [My name is Deborah Kurata, and in this module, we look at asynchronous routing, which loads feature modules](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=15.5) [on request, also known as lazy loading.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=23.5) [When we think of lazy, we often think slow, but like this guy, our routes can be lazy and fast.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=28.5) [We use lazy loading to speed up our application startup time, and we use preloading to quickly launch our](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=36.5) [lazy loaded routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=43.5) [As with any web page, when a user accesses our application, a request is sent from the user's browser to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=46.5) [the application's URL. That URL normally maps to a web server somewhere that serves up the application's](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=53.5) [index.html file in response. The index.html file provides details on the other files such as JavaScript and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=60.5) [CSS files that our application needs. All of those required files are also downloaded.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=71.5) [Angular then compiles the downloaded files and loads the root app component.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=78.5) [After this process is complete, Angular displays our app.component's template.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=84.5) [Depending on the size of the application and the user's download speed, the startup process could take a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=89.5) [noticeable amount of time. Lazy loading speeds up our startup time by splitting the application into](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=96.5) [multiple bundles and loading some of them on demand. So when the index.html file comes down, only some of](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=104.5) [the required files are downloaded. That's less to download and less for Angular to compile before displaying](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=112.5) [that first template improving the startup time of our application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=119.5) [When the user navigates to a route in a lazy loaded module, then the module is downloaded and compiled.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=125.5) [This is especially useful for features that are only accessed occasionally or only by specific users.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=131.5) [The application is not weighed down with the code for these features until they are actually needed.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=138.5) [Note that there are several other techniques we can use to improve the startup time for our application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=144.5) [We can minify the code and consider AOT, the ahead-of-time compiler.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=150.5) [See the Angular 2: Fundamentals module on deployment for more information.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=156.5) [Our focus in this module is lazy loading our routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=160.5) [In this module, we begin with the steps required before we can use lazy loading.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=165.5) [Then we lazy load a feature module and build a canLoad route guard.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=172.5) [We examine how to preload feature areas in the background and then define our own custom module loading strategy.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=178.5) [In our sample application, we'll set up our product routes to route asynchronously by lazy loading our](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=186) [product.module. Let's get started.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=0&mode=live&start=193)

[Preparing for Lazy Loading](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live)

[Before we can lazy load a module and achieve asynchronous routing, the feature area to lazy load must meet a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=2.166) [few requirements. The feature area must be defined in its own feature module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=11.166) [That's because lazy loading loads all of the components declared in one specific Angular module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=16.166) [Another requirement for lazy loading is that the routes are grouped as children of a single parent route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=23.166) [That's because lazy loading is configured on the parent route path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=30.166) [The module for that route path is then loaded asynchronously. Lastly, the feature module must not be](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=35.166) [imported in any other Angular module. That's because if a module is referenced by any other module within](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=43.166) [the application, Angular will download and compile it completely defeating the purpose of lazy loading.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=50.166) [Let's start by viewing our application as it is now in the browser.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=57.166) [Open the developer tools and click on the Network tab. Now let's see what happens when we refresh the application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=62.166) [All of the files required for our application and for Angular to compile and execute our application are](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=70.166) [all downloaded. Let's filter to just our components. Type component in the filter box, and we see only the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=77.166) [component files that were downloaded for our application. Notice that all of our components are downloaded,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=86.166) [including all of the product feature components. If we configure lazy loading of our product.module,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=91.166) [we won't see any of our product feature components downloaded when we first access the application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=98.166) [Looking at the code, our product features are all declared within our product.module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=106.166) [So our first prerequisite is met. Our product paths are all children of a single products parent route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=111.166) [That meets our second prerequisite. However, our app.module contains a reference to our ProductModule.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=120.166) [We need to remove that or the ProductModule will be downloaded with our other application files.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=126.166) [We delete it here and delete the import statement for it here. Now we've met our third and final prerequisite.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=132.166) [If we try to access a product feature at this point,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=140.166) [we'll get our 404 Not Found page. That's because we are no longer loading the ProductModule so our](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=143.166) [application has no information about our product routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=150.166) [Keep these requirements in mind as you lazy load feature areas of your application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=155.166) [With that, let's see how to lazy load a feature module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=1&mode=live&start=161.166)

[Lazy Loading](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live)

[To set up lazy loading, we configure a route to load a feature module asynchronously.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=2) [If we follow the prerequisite requirements for lazy loading as defined in the last clip,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=8) [then implementing lazy loading for a particular feature simply involves configuring this single parent route](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=13.727) [to asynchronously load the feature module. Here is the definition of the products parent route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=20.727) [We load the product feature module asynchronously using loadChildren.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=27) [Note that this is different from setting up child routes with the children property.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=32.727) [The loadChildren property is specifically for lazy loading. The value of this property is a bit different.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=38.727) [We set the loadChildren property to a string containing the feature module file path, a hash symbol,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=44.727) [and the name of the feature module class. The file path here is relative to the location of the index.html file.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=52.727) [The loadChildren property tells the router to fetch the bundle containing this module only when the user](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=61.727) [navigates to a products route. It compiles this module, then merges the route configuration defined in this](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=67.727) [module with the application's route configuration. Lastly, it activates the requested component and displays](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=74.727) [that component's template. Believe it or not, this is all there is to configuring lazy loading.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=82.727) [Let's give it a try.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=89) [We'll configure the route in the app-routing.module. We'll put it after the Welcome route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=92) [It doesn't really matter where we put it as long as it is before the wildcard route path.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=98.727) [Remember, route order matters. We set the path property to products, our root path name.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=104.727) [Then we set loadChildren to the file path of the product feature module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=111.727) [Then we add a hash symbol and our product module class name, which is ProductModule.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=115.727) [That's all we need to do to set up lazy loading.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=122.727) [But if we bring it up in the browser,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=125.727) [we see that it doesn't work. Any guesses why?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=127.727) [Here's a hint. Look at our products route configuration. Notice that our root path name is here as well.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=131.727) [So the configuration for our ProductListComponent is now products/products.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=139.727) [That's not what we want. Now that we have this parent route defined in the app-routing.module, we can remove](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=145.727) [it from the product.module. Notice the route guard. We'll add that to the parent route in a moment.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=152.727) [For now, let's just delete this parent path. And don't forget to delete its closing square brace and curly brace.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=158.727) [I'll reformat to fix the indentation. And we can delete the AuthGuard import statement since we aren't](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=165.727) [referencing it anymore.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=173.727) [Checking it out in the browser,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=174.727) [we again have our list of products. Yay! Did the changes we make really change how our files are downloaded?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=175.727) [Let's see. Open up the developer console again, click Home, then refresh the page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=185.727) [Ensure you still have the filter set to component. Notice that none of our product components are downloaded.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=191.727) [Clear the display and click Product List. Now our product components are downloaded and compiled,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=198.727) [and the Product List route is activated. With asynchronous routing, we can delay downloading features](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=205.727) [until they are requested by the user. This can speed up the initial startup time of the application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=214.727) [Lazy loading is especially useful for features that are not often accessed or are only accessed by specific users.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=220.727) [Before we move on, let's fix our authorization guard. Recall that we just deleted it when we deleted the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=229.727) [extra parent route. So the user is no longer required to log in before viewing the Product List.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=236.727) [If we still want to guard all of our product routes, which we do in this case, we can add the guard back to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=243.727) [the parent route here. In the app-routing.module, add an import statement for the authorization guard.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=249.727) [And add canActivate to the parent route specifying the AuthGuard.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=256.727) [Trying it out again in the browser,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=262.5) [attempting to access the Product List now brings up the Log In page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=265.5) [We have the guard working again. But now that we are using lazy loading, we have another route guard we can](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=269.5) [use--canLoad. How does that one work?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=2&mode=live&start=276)

[CanLoad Guard](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live)

[When working with lazy loaded components, we can use a canLoad guard any time we want to check criteria](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=0) [before loading an asynchronous route. So the module won't even be downloaded unless the guard requirements](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=9.292) [are met. This is great for security because no one except authorized users will even be able to see the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=16.292) [source code because it won't be downloaded. It is commonly used to prevent loading a route if a user can't access it.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=25.292) [Let's give it a try.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=32.5) [Let's look again at the application in the browser.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=36) [Click Home and open the developer tools. Click refresh, and we load all of the basic application components.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=38.292) [Clear the log and click Product List.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=46.292) [Notice that our ProductModule is immediately lazy loaded, and then the Log In page appears.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=49.292) [If the user has no login for this part of the application, we've just loaded the product features unnecessarily.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=57.292) [It would be better if we required the login before even loading the module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=64.292) [That's the purpose of the canLoad guard.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=69.292) [Recall from earlier in this course that we create a guard by building a service.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=73.292) [We already have the authGuard service, which you can find in the user folder.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=78.292) [Let's add the canLoad guard to this existing service. We add CanLoad to the import statement,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=83.292) [implement it, and add its appropriate method. I'll paste the code for that method.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=92.292) [The canLoad method has one parameter, the Route, which we'll add to the import statement.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=97.292) [Unlike the canActivate, the canLoad method cannot access the ActivatedRouteSnapshot or RouterStateSnapshot](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=103.292) [because the module defining the route is not yet loaded. As with the other guards, canLoad can return a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=112.292) [Boolean, observable, or promise. We return a simple Boolean. The code in this guard is similar to the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=119.292) [canActivate guard. We call the local checkLoggedIn method, but since we don't have a RouterStateSnapshot](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=126.292) [to get the URL, we'll instead obtain the URL from the route.path property.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=133.292) [Now that we've defined our guard, we add it to the route. In the app-routing.module, we'll replace the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=139.292) [canActivate guard with the canLoad guard. Let's see how that changed our application flow.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=146.292) [Refresh the page, and our application components are downloaded.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=153.292) [Clear the log and click Product List.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=158) [The Log In page appears, and ProductModule components are not yet loaded.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=161) [So if we cancel the login, the files will not be downloaded. If we do log in using any username and](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=165.292) [password, then our product feature components are downloaded, and the product-list.component template is displayed.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=173.292) [This is all well and good and kind of cool, but lazy loading does have one drawback.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=180.292) [Depending on the user's download speed, it may cause a noticeable delay when navigating to the first](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=187.292) [product route that kicks off the download of the product feature components.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=193.292) [Can we do better?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=3&mode=live&start=198.292)

[Preloading Feature Modules](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live)

[With lazy loading as we have it now, when the user launches our application, the user waits while the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=2.725) [browser downloads the imported modules, their components, their services, and any other declared files](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=8.725) [such as pipes or custom directives. Any lazy loaded module should not be imported so they won't be](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=16.725) [downloaded at this time. After everything is downloaded and compiled, the router completes its initial](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=23.725) [navigation, and the root app component template appears. At some future point, the user navigates to a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=29.725) [feature configured for lazy loading. The user again waits as the associated module is downloaded.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=37.725) [After it is downloaded, the navigation completes, and the routed component's template appears.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=44.725) [If we are loading feature modules asynchronously, why do we have to wait for the user to request it?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=51.725) [Why not preload feature modules behind the scenes while the user is interacting with the application?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=57.725) [With preloading, also known as eager lazy loading, there is a bit less wait time.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=65.725) [When the user launches our application, the user still must wait for the browser to download the imported modules.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=71.725) [After everything is downloaded and compiled, the router completes its initial navigation,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=78.725) [and the root app component's template appears. After the initial navigation, the router checks its](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=84.725) [configuration for any modules that can be preloaded. It then preloads modules behind the scenes based on](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=90.725) [the defined preload strategy. By the time the user navigates to a product feature route, for example,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=97.725) [the ProductModule is already downloaded and ready to go. And the routed component's template immediately appears.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=103.725) [No additional waiting. Preloading makes sense if we know that the module features will most likely be used.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=111.725) [It is less useful for modules that are seldom used or only accessible to specific users, such as an](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=119.725) [application admin feature. For these modules, no preloading may be best.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=125.725) [How modules are preloaded is based on the preload strategy. What does that mean?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=132.725) [The router offers three preloading strategies--no preloading, which is the default, lazy loaded feature](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=139.725) [areas are loaded on demand when the user navigates to a route configured for lazy loading, preload all,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=146.725) [which preloads all of the lazy loaded feature modules, or custom, which allows us to define our own preload](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=153.725) [strategy for finer control over which modules preload when. We've already seen the no preloading strategy](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=161.725) [when we configured lazy loading for our product module. Let's look at preload all next.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=170.725) [To enable preloading, we pass a preloadingStrategy into the forRoot method as a second argument after the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=176.725) [route configuration array. We pass in an object with any extra options we want to set.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=184.725) [For the preload all strategy, specify a type of PreloadAllModules, and we need to add that to the import statement.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=190.725) [This tells the router after the initial application startup to immediately load all lazy loaded routes,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=199.725) [which are any routes with a loadChildren property. Note that this is defined for the forRoot method,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=206.725) [which means this strategy is set for all routes. Let's give this a try.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=4&mode=live&start=212.725)

[Preloading Feature Modules: Demo](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live)

[In the app-routing.module, first, we'll add PreloadAllModules to the import statement.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=2.414) [Then we'll add the ExtraOptions object as the second parameter to our RouterModule.forRoot method.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=8.414) [If you still have enableTracing set here, just add it to the options object.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=15.414) [We set preloadingStrategy to PreloadAllModules.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=20.414) [Let's check it out in the browser. We have our developer tools open so we can view the network traffic,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=25.414) [and we still have the filter set to component so we can focus on our component files.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=31.414) [Click Home and clear the log. Then refresh the browser.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=36.414) [We again see that all of our imported modules are downloaded. We see no product component routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=41.414) [We're off to a good start. But shouldn't the router be preloading our ProductModule by now?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=47.414) [Nope. Something is blocking it.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=54.414) [That something is the canLoad route guard. Surprisingly, the canLoad route guard blocks any preloading.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=58.414) [We just added the canLoad route guard to our products route. Bummer!](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=65.414) [But if we think about this more, it does make sense. The canLoad must execute before the module is downloaded.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=71.414) [When we specify preloading, the module can be downloaded behind the scenes at any point in time.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=78.414) [Imagine if the canLoad didn't block preloading.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=85.414) [The user could launch the application, the Welcome page appears, and the router then decides to preload](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=88.414) [our ProductModule and execute our canLoad guard. They aren't logged in yet, so the canLoad guard routes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=94.414) [to the Log In page. The user was simply enjoying the beautiful artwork on the Welcome page](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=101.414) [and suddenly out of nowhere, the Log In page displays. That could be a bit surprising.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=107.414) [It is because of these types of issues that the canLoad guard simply blocks preloading.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=112.414) [Preloading does work with any of the other route guards.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=118.414) [Going back to the code, if we want preloading yet still want to limit access to our route, we can change](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=122.414) [the canLoad guard back to the canActivate guard. Let's see if we finally](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=130.414) [preload our module. Begin again by clearing the log and refreshing the page.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=136.414) [We see the components and other files from our imported modules. And a moment later, our ProductModule files](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=141.414) [are downloaded. Yay! So without the preloading strategy, none of our lazy loaded modules are preloaded in](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=147.414) [the background. With the preloading strategy, all of our lazy loaded modules are preloaded in the background.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=156.414) [But what if we want finer control of our preloading strategy? We can build our own.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=163.414) [And it's actually easier than you might think.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=5&mode=live&start=169)

[Custom Preloading Strategy](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live)

[So we've seen lazy loading on demand with no preloading, and preload all, which loads all of our lazy loaded](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=2.647) [modules behind the scenes. What if we want something more than an all or nothing approach?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=8.647) [Let's examine how to define our own custom preload strategy. This allows us to only preload some of our](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=14.647) [lazy loaded modules and not others. For example, we may want to preload the most commonly used modules,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=21.647) [then lazy load our admin features or less-commonly used features with no preloading.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=29.647) [Like our route guards, we define a preload strategy by building an Angular service.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=36.647) [We then register the service in an Angular module, and we set the preloading strategy property to our](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=42.647) [preloading strategy service. Our service is then called by the router any time it wants to preload some routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=48.647) [We build a preload strategy service just like any other service. We define a class, specify the injectable](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=57.647) [decorator, and import what we need. Because we want this service to behave like a preload strategy,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=63.647) [we implement PreloadingStrategy. Then we implement its required method, preload.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=70.647) [The preload method has two parameters. The first parameter is the Route, which provides information about](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=76.647) [the current route. The second parameter is the Function that performs the preloading.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=83.647) [In the preload method, we write the code that decides whether or not to call the load function and preload](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=88.647) [the lazy loaded module. Once our preloading strategy service is in place in an Angular module,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=95.647) [we first add the preloading strategy service to the list of providers and then enable the strategy by](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=102.647) [setting the preload strategy routing option to our preload strategy service.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=109.647) [Let's give this a try.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=115.647) [Start by creating a new file. Since this is for an application-wide loading strategy, we'll add it to the](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=118.647) [app folder. We'll call the file selective-strategy.service.ts. I'll paste the code for this service,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=126.647) [and we can talk through it. I named this class SelectiveStrategy.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=135.647) [Since this is a preloading strategy, the class implements PreloadingStrategy.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=141.647) [Here is the preload method. In the body of the function, we decide whether or not to preload a particular module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=145.647) [I'll paste the code, and we can talk through it. For our sample application, we use the route's data](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=153.647) [property to determine whether to preload a route. We discussed the route.data property earlier in this](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=159.647) [course but did not use it in our sample application until now. I called our new data element preload.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=167.647) [We'll assume that the value is set to true if the route should be preloaded.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=174.647) [So here in the method, we check for that element in the route's data property.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=179.647) [If the route data property is set, and the route data property preload element is true, then we return](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=183.647) [the defined load function and the route is preloaded. Otherwise, we return null.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=190.647) [But since this function returns an observable, we return an observable of null.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=197.647) [That means that we need to import the observable of operator as well.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=202.647) [Before we can use this service, we must register it with an Angular module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=207.647) [Since this service is for our application-wide loading strategy, we'll register it in our app-routing.module,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=212.647) [import the service, and add it to the providers array. Now we are ready to use this strategy.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=219.647) [Here is where we are using the PreloadAllModules strategy. We instead use our SelectiveStrategy.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=227.647) [And since we are no longer using PreloadAllModules, we can remove its import statement.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=234.647) [Now we just need to mark which modules we want to preload using the route's data property.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=239.647) [We only have one lazy loaded route, so let's try out our new strategy there.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=245.647) [I'll reformat the path for readability, we'll add our data property to the products route here, and set it to true.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=250.647) [Our SelectiveStrategy reads this data element and preloads if the value is true.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=259.647) [Now let's check it out in the browser. We set our preload property to true for our product routes so they](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=265.647) [should now preload. I have the developer tools open, the Network tab selected, and I'm filtering the log](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=272.647) [to file names containing component. Refresh the browser, and we see the main application routes load.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=279.647) [Then our product routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=285.647) [Now let's turn off preloading by setting the route's data property preload value to false and try again.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=287.647) [Clear the log, refresh the page, and we see that our product routes are not preloaded.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=295.647) [By building our own custom preload strategy, we control which routes are preloaded.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=301.647) [In our sample application, we used a simple preload flag. Using a custom preload strategy is even more](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=307.647) [useful as we add more feature modules to our application. Let's finish up this module with some checklists](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=315.647) [you can use as you lazy load your own modules.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=6&mode=live&start=321.647)

[Checklists and Summary](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=7&mode=live)

[Before we can lazy load a module and achieve asynchronous routing, the feature area must meet these requirements.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=7&mode=live&start=2.347) [The feature area must be defined in a feature module such as the ProductModule in our sample application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=7&mode=live&start=10.347) [The feature area routes must be grouped under a single parent route such as the products route in our sample application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=7&mode=live&start=17.347) [And the feature module must not be imported in any other Angular module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=7&mode=live&start=25.347) [Configure lazy loading by adding the loadChildren property to the parent route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=7&mode=live&start=31.347) [Be sure that this route is defined in a module that is loaded such as the app-routing.module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=7&mode=live&start=36.347) [Set this property to the full path to the module, a hash, and the module's class name.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=7&mode=live&start=43.347) [By default, lazy loaded modules are loaded on demand when the user requests a lazy loaded route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=7&mode=live&start=50.347) [To instead preload all lazy loaded modules, specify the PreloadAllModules loading strategy.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=7&mode=live&start=57.347) [For more fine control over which lazy loaded modules are preloaded, build a custom preloading strategy service.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=7&mode=live&start=64.347) [In this module, we looked at the prerequisites that our feature area must meet before we can lazy load it.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=7&mode=live&start=73.347) [We saw how quick and easy it is to lazy load the routes for a feature module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=7&mode=live&start=80.347) [We build a canLoad guard and examined how to preload our lazy loaded routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=7&mode=live&start=85.347) [We then looked at how to build our own custom preloading strategy service.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=7&mode=live&start=91.347) [We've now set up our product routes to route asynchronously by lazy loading the product feature module.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=7&mode=live&start=97.347) [This improves the startup performance of our application because only the minimum required files are](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=7&mode=live&start=104.347) [initially downloaded and compiled. Only one short module left.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m11&clip=7&mode=live&start=109.347)

[Final Words](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=0&mode=live)

[Introduction](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=0&mode=live)

[As you have seen throughout this course, Angular routing can do so much more than just move the user between](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=0&mode=live&start=0) [multiple views of an application. Welcome back to Angular Routing from Pluralsight.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=0&mode=live&start=9) [My name is Deborah Kurata, and the final words in this course include a recap of our journey and a few](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=0&mode=live&start=15) [pointers to additional information. We've come a long way. Remember when we started?](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=0&mode=live&start=21.5)

[Recapping Our Journey](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live)

[We began at the beginning with routing basics. We configured our first set of routes, an empty default](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=1.964) [route, a welcome route, and wildcard route that displays a Page Not Found message.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=9.5) [Then we examined how to build feature modules and routed to the Log In component in a user feature module](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=15.964) [and the Product List component in a product feature module. We discussed route path naming strategies](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=21.964) [and now see the benefits of using a common root path name such as products in our example.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=28.964) [And we found that route order matters. We can swiftly pass data as we route from one component to the next](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=34.964) [using route parameters. We looked at required, optional, and query parameters.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=41.964) [We used required parameters to define which product to display in the Product Detail and Product Edit](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=47.964) [component templates, and we used query parameters to retain the Product List selections when navigating to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=53.964) [the Product Detail and back again. For a nicer visual experience, we can retrieve the data required for](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=59.964) [display in a component's template before routing to that component using route resolvers.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=67.964) [We learned how to build a resolver service to prefetch the data for our Product Detail and Product Edit components.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=74.964) [No more partial page display as our component waits for its data.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=81.964) [Using child routes, we defined a route hierarchy to better organize, encapsulate, and navigate through our application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=85.964) [We replaced the form in the Product Edit component template with a tabbed Edit page and defined a](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=94.964) [router-outlet for display of the child tab components. As the user clicks a tab, the associated component's](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=99.964) [template is displayed in that router-outlet. And we saw how to share the single instance of the product data](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=106.964) [from the parent's route resolver. Then came the biggest change to our route hierarchy.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=114.964) [We grouped our child routes under a component-less parent route. This helps us better organize our routes](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=120.964) [and share route guards without requiring the parent to provide a router-outlet.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=128.964) [The child component templates are displayed in the next higher level of the hierarchy, in this example,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=133.964) [the primary router-outlet. Then we switched gears a bit and spent some time polishing our routes.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=139.964) [We learned how to style the selected route, animate our route transitions, watch our routes, and react to](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=146.964) [routing events to display a spinner. Secondary routes make it easy to display multiple panels or panes on](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=153.964) [the page, each containing different content and supporting independent navigation.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=160.964) [We discussed a Messages component in our secondary router-outlet named popup.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=166.964) [We could also display other content in this secondary outlet such as a product summary when editing product information.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=171.964) [We used route guards to ensure navigation to a route is permitted for security, authorization,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=178.964) [or monitoring purposes, or to prevent the user from leaving a route without a confirmation.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=185.964) [We added a CanActivate guard to the product's parent route to ensure that the user is logged in before](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=191.964) [accessing any product data. And we added a CanDeactivate guard to the Product Edit parent route.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=198.964) [This guard notifies the user of any unsaved changes before navigating away.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=205.964) [Lastly, we lazy loaded our product routes to improve the startup time of our application.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=210.964) [And we built a custom preloading strategy service so we can control which lazy loaded modules are preloaded when.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=216.964) [Along the way, this course provided a set of checklists containing steps and tips.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=224.964) [Feel free to revisit and reference these checklists as you start incorporating these routing features](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=230.964) [into your own applications.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=1&mode=live&start=237.964)

[Learning More](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=2&mode=live)

[If you'd like more information about Angular components, services, dependency injection, or observables,](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=2&mode=live&start=2.93) [check out one of these courses. Angular 2: Getting Started, Angular 2: First Look, or Angular 2: Fundamentals.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=2&mode=live&start=9.93) [For more information on routing, this book is a comprehensive guide to the Angular router written by its designer.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=2&mode=live&start=18.93)

[Closing](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=3&mode=live)

[Congratulations! You now know how to leverage more sophisticated routing features to support more real-world](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=3&mode=live&start=2.184) [routing scenarios. Yay! Thanks for listening, and I'd love to hear about your experiences with routing.](https://app.pluralsight.com/player?course=angular-routing&author=deborah-kurata&name=angular-routing-m12&clip=3&mode=live&start=10.184)