

Chapter 1 : Learning Event Scheduled: Angular 2 In Action | Keyhole Software

Angular in Action teaches you everything you need to build production-ready Angular blog.quintoapp.comghly practical and packed with tricks and tips, this hands-on tutorial is perfect for web devs ready to build web applications that can handle whatever you throw at them.

In this tutorial, we cover how to use Http to make requests and how to handle the responses. Check out the repo for the tutorial to see the code. This is Part 3 in our Angular 2 series. Be sure to check out Part 1 on Pipes and Part 2 on Models. Incidentally, the sample we use to test out Http will also let us see the basics of adding authentication to an Angular 2 app. Differences between Angular 1. Using observable streams gives us the benefit of greater flexibility when it comes to handling the responses coming from HTTP requests. For example, we have the potential of tapping into useful RxJS operators like `retry` so that a failed HTTP request is automatically re-sent, which is useful for cases where users have poor or intermittent network communication. With these we get providers such as `RequestOptions` and `ResponseOptions`, which allow us to modify requests and responses by extending the base class for each. To get started, clone the repo and run `npm install`, then `gulp play`. We also need to get our random quote server running. We also need to set up our app component by importing what we need and providing a template. The form calls an `authenticate` method on submit. If we were expecting a JSON response, we would call `res`. After this, we need to subscribe to it so that we can observe values that are returned. The first function is the next case, or what happens when the HTTP call is successful. The second function is the error case, and in our example we are logging any error messages to the console by calling our `logError` method. Finally, the third function defines what happens once the call is complete, and here we are simply logging out the things are finished. The `authenticate` method accepts data that is passed down from the view when the form is submitted. Then we simply pass this as a header in the options object when we make the POST request. From here, the returned observable is handled in the same way that we saw in the GET request, but this time the response is JSON, so we use `res`. We also call the `saveJwt` method and pass it the token that is received to save it in local storage. We can check this in the browser to make sure our call to the protected endpoint works. What Can We Do with Observables? The fact of the matter is that we can do much more with observables than promises. With observables, we have a whole bunch of operators to pull from, which let us customize our streams in nearly any way we want. We can get a taste for how they work by plugging in some simple operators. Now only quotes with a length greater than characters will come through. If we click the "Get Random Quote" button a few times, we see it at work. We can also delay our subscription if we want. Here we wait two seconds before printing the quote to the screen. While that might not be the most useful, what we will really benefit from once Http is solidified are operators like `retry`. If we were expecting network connectivity issues, we could define a number of times to retry the request. Another benefit of observables is that they can be cancelled. This can be useful for situations where HTTP requests are going on for a long time, perhaps to an unresponsive server somewhere. This means that you can have a solid identity infrastructure , including single sign-on , user management, support for social identity providers Facebook, Github, Twitter, etc. We implemented a tight integration with Angular 1. You can read the documentation here or you can checkout the SDK on Github. We also saw how we can customize the headers we send with the `Headers` class. As with Angular 2 in general, Http is always going through changes. The Angular team keeps a very good backlog of Http issues and feature implementations , which is a good place to keep updated on how Http progresses. Authentication that just works.

Chapter 2 : AngularJS in Action - PDF Free Download - Fox eBook

Angular 2 in Action has 25 repositories available. Follow their code on GitHub.

Chapter 3 : angular - How to submit form to server in Angular2? - Stack Overflow

DOWNLOAD PDF ANGULAR 2 IN ACTION

Solving real life challenges with Angular 2. A short look at the development of a cutting edge medical imaging application @Algotec.

Chapter 4 : Using Angular \$Http and Angular 2 Http - Angular 2 Series Part 3

Download angular 2 in action or read angular 2 in action online books in PDF, EPUB and Mobi Format. Click Download or Read Online button to get angular 2 in action book now. This site is like a library, Use search box in the widget to get ebook that you want.

Chapter 5 : Universal Angular 2 In Action

Covers Angular 2 and above. Angular in Action teaches you modern application development with Angular. You'll begin with the basics of Angular and the technologies and techniques you'll use throughout the book, such as key features of TypeScript, newer ES6 syntax and concepts, and details about the tools needed to write professional Angular applications.

Chapter 6 : ruby on rails - Angular 2 and ActionCable integration - Stack Overflow

The switch from Angular 1 and adoption of Angular 2 has been a source of anxiety for members of the enterprise. That enterprise concern is what we seek to alleviate in this talk. In this educational Breakfast Boost, Keyhole Principal Consultants will present a live demonstration of Angular 2 in action.

Chapter 7 : Manning | AngularJS in Action

Export PHP/HTML table to PDF, CSV, PNG, XML, JSON, SQL, TXT, MS Excel and MS Word - English - Duration: Back 2 Tech 28, views.

Chapter 8 : Angular In Action - Gnome on the run

Angular 2 In Action, a free public Keyhole Software educational event has been scheduled for Thursday, February 16th. Get more info & reserve your free tickets.

Chapter 9 : Angular in Action

How to submit form to server in Angular2? Angular 2 form not sending action request when submitting. 0. Angular 2 navigate to a different page on login. 0.