[Course Overview](https://app.pluralsight.com/course-player?clipId=1d3d817f-1a03-4be7-b609-8c29ba7e7faa)

[Course Overview](https://app.pluralsight.com/course-player?clipId=1d3d817f-1a03-4be7-b609-8c29ba7e7faa)

[Hello.](https://app.pluralsight.com/course-player?clipId=1d3d817f-1a03-4be7-b609-8c29ba7e7faa&startTime=0) [I'm Peter van Rijn.](https://app.pluralsight.com/course-player?clipId=1d3d817f-1a03-4be7-b609-8c29ba7e7faa&startTime=0.9561666666666667) [Welcome to my course on Spring REST.](https://app.pluralsight.com/course-player?clipId=1d3d817f-1a03-4be7-b609-8c29ba7e7faa&startTime=4.780833333333334) [I do have more than 20 years' experience with Java](https://app.pluralsight.com/course-player?clipId=1d3d817f-1a03-4be7-b609-8c29ba7e7faa&startTime=11.474) [in many roles as a programmer, architect,](https://app.pluralsight.com/course-player?clipId=1d3d817f-1a03-4be7-b609-8c29ba7e7faa&startTime=14.179714285714294) [consultant, teacher, and coach.](https://app.pluralsight.com/course-player?clipId=1d3d817f-1a03-4be7-b609-8c29ba7e7faa&startTime=17.242) [Spring is by far the most popular Java enterprise framework,](https://app.pluralsight.com/course-player?clipId=1d3d817f-1a03-4be7-b609-8c29ba7e7faa&startTime=19.882) [and REST is the ubiquitous way of sending data over to it.](https://app.pluralsight.com/course-player?clipId=1d3d817f-1a03-4be7-b609-8c29ba7e7faa&startTime=23.111) [Especially Spring boot has many libraries to make Spring REST easy](https://app.pluralsight.com/course-player?clipId=1d3d817f-1a03-4be7-b609-8c29ba7e7faa&startTime=26.997) [and fun not only for sending plain JSON messages,](https://app.pluralsight.com/course-player?clipId=1d3d817f-1a03-4be7-b609-8c29ba7e7faa&startTime=30.219) [but also for sending error messages and messages with hyperlinks.](https://app.pluralsight.com/course-player?clipId=1d3d817f-1a03-4be7-b609-8c29ba7e7faa&startTime=33.5856923076923) [And to make your code robust there are many testing libraries available.](https://app.pluralsight.com/course-player?clipId=1d3d817f-1a03-4be7-b609-8c29ba7e7faa&startTime=38.803) [All this will be covered in this course in theory](https://app.pluralsight.com/course-player?clipId=1d3d817f-1a03-4be7-b609-8c29ba7e7faa&startTime=43.17) [and illustrated with many demos.](https://app.pluralsight.com/course-player?clipId=1d3d817f-1a03-4be7-b609-8c29ba7e7faa&startTime=47.438) [By the end of this course you will have a solid foundation to](https://app.pluralsight.com/course-player?clipId=1d3d817f-1a03-4be7-b609-8c29ba7e7faa&startTime=49.572) [write professional Spring REST applications.](https://app.pluralsight.com/course-player?clipId=1d3d817f-1a03-4be7-b609-8c29ba7e7faa&startTime=52.56664285714288) [Before beginning this course you should have some](https://app.pluralsight.com/course-player?clipId=1d3d817f-1a03-4be7-b609-8c29ba7e7faa&startTime=54.95949999999999) [basic knowledge on Java programming.](https://app.pluralsight.com/course-player?clipId=1d3d817f-1a03-4be7-b609-8c29ba7e7faa&startTime=58.86644444444444) [From here you can continue your learning with courses on Spring,](https://app.pluralsight.com/course-player?clipId=1d3d817f-1a03-4be7-b609-8c29ba7e7faa&startTime=61.047) [especially Spring 5,](https://app.pluralsight.com/course-player?clipId=1d3d817f-1a03-4be7-b609-8c29ba7e7faa&startTime=63.887) [and Angular and React in combination with the Spring backend. I hope you'll join me on the journey to learn Spring REST at Pluralsight.](https://app.pluralsight.com/course-player?clipId=1d3d817f-1a03-4be7-b609-8c29ba7e7faa&startTime=65.3593076923077)

[Introduction](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c)

[Introduction and Getting Started](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c)

[This module is the Introduction combined with the Getting Started.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=0) [In the Introduction we talk about what is REST,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=6.84) [what is Spring, the Getting Started is a table of some basic REST messages,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=10.90125) [and in the end I will explain what's happening on the \_\_\_\_\_.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=15.427) [What is REST?](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=21.563) [This is the description from Wikipedia.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=22.920666666666666) [REST stands for Representational State Transfer.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=25.636) [It's an architectural style based on HTTP.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=29.855) [What does that mean?](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=33.709) [We'll look at an example.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=36.09033333333335) [This is a client and a server.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=39.067) [Data is transferred from the client to the server.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=42.07787499999998) [Client is a Web Form running in a browser, and the server is a database.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=45.949) [The Web Form has some data represented by this icon,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=51.202) [and the data is encoded to a JSON format.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=54.66557142857144) [JSON is sent to the server via HTTP,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=58.66484615384614) [and on the server the JSON is decoded to a Java object,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=62.847) [which is sent to the database.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=66.59420000000006) [Where REST works then, the data is the state.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=69.45174999999999) [JSON is the representational state, and HTTP is the transfer.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=75.214) [REST in summary.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=81.124) [REST is a client server architecture, based on HTTP,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=83.03349999999999) [and it uses a Request - Response Model, for example,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=88.762) [the browser sends a request, and the server gives a response.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=91.80815384615389) [Resources are located via a URL, and we are using the HTTP methods GET,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=95.853) [POST, PUT, and DELETE.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=102.30760000000001) [What do these methods mean?](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=104.358) [A GET will retrieve data from the server,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=106.46338461538463) [a POST will send data to the server, a PUT will modify an already existing data,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=109.832) [and a DELETE will delete data.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=116.2775) [The next question, what is Spring?](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=121.703) [Again, we'll use Wikipedia.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=124.74200000000002) [Spring is an application framework and inversion of control](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=126.768) [container and running on the Java platform.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=131.06910000000005) [It's an enterprise application framework used by companies to build their](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=135.306) [administration software based on databases and websites.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=139.462625) [It's a large set of libraries, and we are only using a small part of them.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=145.592) [Spring Boot contains the basic Spring libraries;](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=150.607) [Spring MVC for the web framework, and Spring Data for accessing the database.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=154.814) [The Spring basic architecture is a container.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=163.557) [Its main responsibilities are lifecycle management and Dependency Injection.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=166.69) [What does that mean?](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=171.478) [It will instantiate the objects and wire them together,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=173.03766666666667) [so you don't have to use the new keyword,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=176.407) [you're not even allowed to do so.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=178.657) [Containers are built from configuration,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=181.67766666666668) [and in Spring you'll add annotations to the class definition.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=184.516) [The reason to use containers is to make your](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=188.435) [software more modular and extendable.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=193.26221428571415) [This is the architecture of our application.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=195.944) [This is the server, the server talks with the database, and there is a client.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=198.415) [Spring REST is based on Spring MVC, and it stands for Model View Controller,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=204.1841666666666) [and we are also using a service.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=211.9365) [The model will hold the data, the view is the presentation,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=216.575) [the controller handles the requests and the responses.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=221.517) [In the service we will access the database.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=225.904) [The view runs outside the server and could be implementing the Angular,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=228.946) [Android, or iOS.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=233.21136363636364) [We won't do that in this course.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=234.90145454545458) [We will use Postman as a client to send and receive JSON messages.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=238.845) [Let's get started.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=244.321) [With the first example project we will generate a Spring](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=245.74825) [boot demo project from the spring.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=250.39525) [io website,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=252.58675000000002) [and the demo practice imported in IntelliJ will](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=253.31725000000003) [look at the structure in the code, try to run the project.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=256.2013333333333) [This is the Spring Initializer website.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=261.806) [It can generate a Maven Protocol with Java and Spring Boot.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=264.1896363636362) [Maven needs a group and artifact name.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=271.025) [The default are com.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=273.6855833333334) [example, and the artifact is demo.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=275.2059166666668) [In this project we only need one dependency, web.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=278.98564285714275) [I'm pushing the Generate button.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=285.105) [A demo zip is downloaded.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=287.35499999999996) [We unzip the demo, and we go over to IntelliJ.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=289.605) [\_\_\_\_\_ Let's have a look at the structure.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=293.794) [It's a Maven project, it has a source, main, Java structure, and a com.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=302.031) [example.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=309.87225000000007) [demo package, and that contains one Java file, the DemoApplication.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=310.3433333333334) [We start this application with a terminal, a Maven spring-boot:run.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=315.5646666666667) [When spring-boot is started we go over to a browser localhost:8080.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=322.84125) [There's nothing yet.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=332.08759999999995) [The URL is not found, but the code is compiling and the server is running.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=336.092) [The next step.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=342.144) [We will build a web service and retrieve data from the server.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=343.32620000000003) [We will send a request URL,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=348.055) [and we want the server to respond with the next JSON message.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=349.76579999999984) [In the controller for that first we create a control package,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=357.594) [and inside the package we create Java class called Message Controller,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=365.766) [and we have to tell the Spring container that this is a controller,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=373.888) [so it will be instantiated \_\_\_\_\_.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=378.051) [Inside the controller we need the URL mapping.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=381.0417272727272) [This is a GetMapping to the message URL.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=386.70090909090914) [To return a JSON message with the content we need a model class for that,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=395.146) [and a model package, and in the model package we create a Java class message.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=399.7020625) [The message has only one property, content,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=409.198) [and we let IntelliJ generate the getters and setters,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=414.4677272727271) [and two constructors, a constructor for convenience,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=421.3715) [and a default constructor.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=429.3739999999999) [This is our class.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=435.041) [Back to the controller.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=438.65100000000007) [Create a send function that returns the message, new Message with the content.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=442.261) [This is not working yet.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=459.617) [We have to tell Spring that we are returning a response body,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=460.9730714285714) [and Spring will convert our message object into a](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=464.97649999999993) [JSON message \_\_\_\_\_ spring-boot, and back to the browser.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=469.664) [Localhost /message, yes, it's working.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=482.409) [The second service, send data to the server.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=489.812) [This is a POST method on the same URL, and we will use Postman for that.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=493.681) [Before we go further we can simplify the program code a bit.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=502.517) [When we make the controller a REST controller,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=506.853) [then the ResponseBody annotation can be removed.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=509.9206363636362) [Next, we'll add a PostMapping to the same URL.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=514.9205384615385) [Return the Message object,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=523.582) [and we want the Message object to be received and be called a method echo.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=525.0556666666668) [And on the input we have to add annotation RequestBody.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=537.648) [We cannot use the browser anymore.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=545.688) [We use Postman to send POST messages.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=547.8872307692305) [Postman can be downloaded from this website.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=550.453) [This is Postman.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=554.101) [Let's first retry the GET on the URL localhost:8080/message.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=556.6284999999999) [Now we'll create a Postman, and we have to put the JSON data inside the body.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=564.211) [It's the same JSON structure, so we copy it form the previous message.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=573.189) [I'll change the text.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=581.089) [It stopped working because we have to tell the main type is the JSON type.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=583.8320769230769) [Try again.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=595.615090909091) [Yes, it's working.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=597.4854545454547) [Explanation.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=600.291) [What is happening under water?](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=601.0138333333334) [We will look at Spring Boot and how the container is working,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=604.628) [look at the locking information on the startup of the application,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=608.354) [and we'll look at the workflow of the RESTful web service.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=612.307) [The application starts on the main method inside a demo application class.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=617.074) [The work is done by the SpringBootApplication annotation.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=622.167) [When we look inside we see a ComponentScan.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=626.838) [It will scan all the classes in the project looking for annotations.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=631.289) [It's fine to control our annotation inside the message controller,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=636.272) [and then the Spring container will instantiate the](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=640.774) [MessageController and wire it to the web context.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=643.7882222222221) [Next we'll look at the start up of the application.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=647.8) [In the logging we'll see on the last sentence the DemoApplication is started,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=651.543) [and Tomcat is started on port 8080.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=656.387375) [By searching for the /message text we'll find that they have two mappings,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=661.997) [one to a GET and one to a POST.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=666.98975) [The GET is mapped onto the send method,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=669.6694999999995) [and the POST is mapped onto the echo method.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=672.5483636363637) [Now we'll talk about the workflow of the RESTful web service.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=677.464) [The main entry point on Spring MVC is the dispatcher servlet.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=680.733) [There's only one dispatcher servlet, and it handles all the incoming requests.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=684.39) [We request our router to the controller or more controllers as](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=689.229) [long as the URL mapping are not conflicting.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=692.6347692307697) [Let's look at the workflow.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=695.703) [The client is sending a JSON request.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=697.9921666666669) [The dispatcher servlet then forwards it to the controller,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=701.197) [and it recognizes the annotation request body,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=706.432) [and then the dispatcher servlet will try to map the JSON data to a Java object.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=708.4231818181818) [When the controller wants to send back the data to the](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=715.14) [client controller returns the Java object.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=717.9257499999998) [Because the method annotation with ResponseBody the servlet will](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=721.3862222222223) [confer the Java properties with a JSON string,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=725.7060909090911) [which is sent back to the client.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=729.2216666666667) [In summary,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=734.159) [REST stands for Representational State Transfer that means](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=735.3056) [data converted to JSON and sent via HTTP.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=739.892) [Spring is an application framework with a lot of libraries,](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=743.2339999999997) [and we are only using Spring Boot and Spring MVC and the](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=746.7857692307691) [architecture of Spring MVC is Model View Controller. As a demo we have built two REST web services, a GET, and a POST.](https://app.pluralsight.com/course-player?clipId=a56510d0-5441-4129-b78d-f34ff600087c&startTime=752.1542307692309)

[Building the Friends Web Services](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7)

[The CRUD Operations](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7)

[In this module we are building a project and we will use](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=0) [the standard product phases; analysis, what are we going to build; design,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=7.487999999999999) [how are we going to build it; the setup,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=11.905846153846152) [the installation and configuration of the database;](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=15.663) [and a build consists of two phases.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=19.294599999999992) [In the first we build the architecture,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=22.353545454545454) [and in the second we build the REST API in a number of small iterations.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=24.983) [This is the product description.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=30.658) [The Wired Brain Coffee Company wants a loyalty program,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=33.158) [and they will call it Wired Brain Friends.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=36.386750000000006) [It consists of a client app running a mobile input and the](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=39.063) [central server holding the friends database.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=43.83859999999999) [The client's apps are built by other teams,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=47.162) [and we are responsible for the REST API talking to the database.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=49.26538461538464) [We start with the analysis.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=56.04) [The REST API should be able to register a friend,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=57.68607142857143) [find one or more friends, change a friend,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=61.066833333333335) [and delete a friend, and they are built on the basic CRUD operations,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=64.40950000000001) [CREATE, READ, UPDATE, and DELETE.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=70.34209090909089) [The design phase.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=73.764) [Spring REST follows the Model View Controller pattern,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=74.92290000000001) [so they have to divide the architecture in three packages; the model package,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=78.03390909090909) [holding the data object, it contains the friend class; the View is Postman,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=83.48354545454546) [we won't build that, the Controller Package controls the FriendController,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=89.38888888888887) [and inside the FriendController the REST API is implemented and](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=93.69600000000003) [the server package that talks to the database.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=97.79989999999997) [This is the architecture.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=102.424) [The model is Friend with three properties, id, firstName, and lastName.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=103.97963636363639) [The Friend Controller implements the REST API and is a REST controller.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=108.87666666666667) [The Friend Service is a CRUDRepository for accessing the database.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=114.58108333333328) [The database is MySQL on MariaDB, and the client is Postman.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=120.759) [Before we start programming we have to do some installation.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=126.984) [First, we have to install the database,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=130.973) [next we will generate the product at Spring Initializer,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=133.19549999999998) [and the database has to be configured.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=136.9782727272727) [Installing the database.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=140.82) [We'll use a MySQL-like database, MySQL itself or MariaDB.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=142.337625) [They are both based on the same source code.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=146.67992307692307) [You can download MySQL at the MySQL website and MariaDB at the MariaDB website.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=150.759) [After installation we can test if it's running by creating a new](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=157.239) [Friend database on the MySQL command line.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=161.005) [Open a terminal and start your mysql client.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=165.706) [Mysql -u root, that's the username, -p, the password.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=171.943) [MariaDB has no password.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=178.618) [Create database friends.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=183.32714285714286) [We're ready.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=186.859) [Over to the Spring Initializer website.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=188.345) [The Group is the company name in reversed order, so it will be com.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=190.54299999999998) [wiredbrain, and the artifact is the name of the product, friends in this case.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=195.89300000000006) [We'll have the following dependencies, Web,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=202.848) [which is Spring MVC, jpa or Java Persistence API for the database,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=206.73419999999993) [MySQL driver, so JPA can talk to MySQL.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=215.19299999999993) [Generate Project, and it will be downloaded at the frame.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=219.07753846153852) [zip.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=224.83724999999998) [Here the project is opened in IntelliJ.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=225.817) [The product is called friends, and we'll look at the structure.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=228.02772727272722) [We see that the package names are called com.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=231.72000000000006) [wiredbrain.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=234.91833333333338) [friends, the name we gave in the Spring Initializer Website.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=235.28416666666672) [Before we go further we have to repair the pom.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=238.93754545454541) [xml file.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=242.675) [In Java 10 some enterprise libraries are not available anymore,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=243.49900000000002) [so we have to add them ourselves.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=247.61166666666665) [It is the jaxb-api library,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=250.47) [and we search on the internet in the Maven record stores.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=252.3492307692307) [We take the version 2.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=257.66599999999994) [3.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=261.51599999999985) [0, copy the dependencies, and paste them into the pom.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=262.28599999999983) [And now we're up-to-date with Java 10 and higher.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=269.898) [Next, we'll need to add the database configuration to the application.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=273.525) [properties file located under source, main, resources.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=277.370625) [First the database URL called friends, then the username,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=283.367) [the password, but you don't have a password,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=289.852) [then the MySQL driver, MySQL dialect,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=293.7199999999999) [then hibernate will create the tables for us,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=296.7772) [and next we want to see the SQL statements in the logging.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=301.27) [Now everything is ready for a test drive.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=305.293) [Let's start the server and see if it's compiling.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=307.4966923076921) [The application is started.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=312.85400000000004) [Next, we'll build the architecture.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=316.834) [We need three Java classes, the model.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=318.8595) [Friend, the service, and the FriendController.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=322.33425) [This is how the Friend class looks like,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=326.682) [the test three password, id, the firstname,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=328.81999999999994) [and the lastName, and the class is annotated with Entity,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=331.265) [and id gets an id and a GeneratedValue.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=335.9763333333333) [With this information jpi/hibernate can generate](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=340.643) [all the necessary SQL statements.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=344.87245454545456) [Next, we'll look at the FriendService.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=348.397) [The FriendService is an interface, and it's inherited from CRUDRepository,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=350.3715999999999) [and now all the CRUD functions are available.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=354.49119999999994) [And then the FriendController annotated as the](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=359.763) [RestController and Autowired with FriendService.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=363.9614444444444) [This is Dependency Injection, so the controller can talk to the service.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=366.74928571428575) [Over to IntelliJ.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=371.96) [First we create the model, controller, and service packages.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=373.3383333333333) [(Clicking) In the model package the Friend class is added with three fields,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=377.352) [an id, it's an integer, a String firstName,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=395.994) [and the String lastName.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=406.1400000000002) [Generate getters and setters and add annotation,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=411.213) [Entity, and Id, and GeneratedValue.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=422.32199999999995) [Okay, this is ready.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=429.5288) [Over to the FriendService.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=434.4203999999999) [This is an interface and it extends the CRUDRepository at](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=439.312) [Friend and the Id that's an integer.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=452.350181818182) [This is ready too.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=457.74119999999994) [Next, we create a REST controller.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=460.466) [It's a FriendController.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=468.51733333333334) [Annotated this RestController and Autowired to the FriendService,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=472.543) [restart the server, over to the database.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=482.312) [Use friends and show the tables.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=485.581) [There's a Friend table generated, and what's inside an id,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=488.39118181818174) [a first\_name, and a last\_name, and the first\_name as snake case,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=493.8025) [and in Java code that was camel cased, so that's the difference.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=498.8252307692309) [Let's insert two records ourself.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=505.49000000000007) [See what's inside with the Select.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=509.876) [Okay.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=513.4905714285715) [The next phase is the build phase.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=514.093) [We will implement the REST API one by one starting with the POST,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=515.9921538461538) [and the GET, and the PUT, and the DELETE.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=520.5503076923075) [This is the POST.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=524.3056153846153) [The PostMapping on the Friend URL mapped onto the create function.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=525.963769230769) [It receives a RequestBody, and it calls save on the service.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=531.072111111111) [Next is a GetMapping mapped onto the read function,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=537.868) [and it calls findAll, and it returns a list of friends,](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=540.83575) [and a PutMapping, same URL, again, a RequestBody.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=545.0591818181822) [This one also calls the save function, but now the save acts as an update.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=548.6874545454548) [And the last one is DeleteMapping the Friend.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=557.619) [Now the URL is extended with a PathVariable and the PathVariable](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=560.959) [is used to deleteById on the friendService.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=566.349) [Add mappings to the controller, Post, Get, Put, and Delete, restart the server.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=570.2236000000003) [Over to Postman.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=579.672) [Let's do a GET on localhost:8080, and returns the list from the database.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=581.6327500000001) [Next, we will POST.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=590.8524285714285) [We add in the Body raw, Text JSON, Gordon Moore, and that's now an id 3.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=594.189857142857) [Let's do a GET.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=610.095) [Check this.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=613.1084545454544) [Now we've got three elements.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=614.6151818181816) [Let's do a PUT.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=618.382) [We will upgrade the second to the body, copy it here.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=620.8340000000002) [That will be Linus Torvalds.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=628.4068) [Linus.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=632.8547999999998) [Torvalds.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=633.7443999999998) [See if it's working.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=634.634) [Yeah.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=637.9354545454544) [That's okay.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=638.760818181818) [And the GET again, the second one should have been changed.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=640.4115454545451) [Yep, that's right, and the last is the DELETE.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=646.5809000000003) [Let's delete the third one.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=654.540909090909) [Now it's gone. So everything is working.](https://app.pluralsight.com/course-player?clipId=ca363d08-a846-4e2b-91c8-86b7b9c14ba7&startTime=660.15)

[Search Functions](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949)

[Now that the basic CRUD functionality is ready,](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=0.676) [we want to add some extra features.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=3.593866666666668) [Some search functions, findBy something,](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=6.147) [and we want to add some extra data types to our friend class](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=7.782833333333334) [and see how they are behaving in JSON.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=12.40271428571429) [Let's start with Search.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=15.509) [We will implement a find by id, and a find by first and last name.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=17.33627272727273) [Later on we'll extend this one to a find by first or last name.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=23.804857142857152) [This is the findById.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=29.56) [It's a GetMapping to the Friend URL with a PathVariable,](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=30.968) [and it maps onto the friendService findById,](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=34.748375) [which is already available.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=39.035) [The find by first and last name is not available in the FriendService,](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=40.52946153846153) [but it's easy to edit.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=45.49599999999999) [You can edit with a findBy and then the property name,](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=47.501) [FirstName, And the second property name,](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=50.811) [LastName, and Spring data will generate a working method for us.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=53.77079999999999) [The URL, again, is a Get, and we will use a request parameter now,](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=58.909) [and because request parameter not part of the URL we have to](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=64.607) [resolve this conflict by creating a new URL,](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=67.5331538461538) [friend/search.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=72.10649999999998) [The parameters are mapped to Java strings and can be sent to the service.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=72.725) [Over to IntelliJ where we look at the FriendService.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=77.656) [We see that there already is a findById available,](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=80.95) [and in the controller we add findById,](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=84.86200000000005) [then we go to the find by first and last name.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=88.14920000000001) [We have to edit to the service, and it's a find by first and last name,](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=93.603) [end it with a semicolon because Spring data will generate code for us.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=98.57592857142858) [In the controller add a findByQuery with two request parameters first and last,](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=105.166) [and here we do the call to the friendService return friendService,](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=111.2436153846154) [find by first and last name, and pass the parameters firstName and lastName.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=116.49109090909091) [Over to Postman.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=126.961) [Let's do a GET first.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=129.02200000000002) [Two elements.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=132.45700000000008) [Let's find the second one as a friend/2.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=133.831) [Now do a friend/search with the first and last,](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=140.35161538461543) [and this is also working, and we make a mistake it's also okay.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=146.30042857142857) [It would be nice if we could search on first and last](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=152.0214) [name and also on first or last name, but why not try and see what's happening.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=156.171) [Only the first, and here we get an error,](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=162.094) [last is not present, so you have to do some coding on the server.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=165.59846153846152) [Starting with the FriendService we have to add a findByFirstName and LastName.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=169.428) [And in the controller make the request parameters not](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=175.548) [required by setting them to false, and the other one also,](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=178.8845454545454) [and then we have to create an if statement with different possibilities,](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=187.56914285714282) [its first and last name, it's only the firstName,](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=195.46799999999993) [only the lastName, or it's none.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=200.75549999999998) [If first and last name and find by first and last name on the friendService,](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=204.549) [if only the first name and find by firstName,](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=208.62709999999998) [if only the last name then find by lastName,](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=213.326) [and if none we do a findAll that's a design decision,](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=215.88079999999988) [but I think it's a good one.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=220.21816666666672) [Restart the server and over to Postman.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=223.906) [Let's try this one.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=226.82563636363642) [Yes, it's working now.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=228.494) [Let's do it with the last name.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=230.7150909090909) [Also working.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=234.602) [Let's make an error and get nothing. That's right, and now let's do define all, the none, and this is also working.](https://app.pluralsight.com/course-player?clipId=56899280-001e-4100-b312-94a03dce6949&startTime=235.55711111111114)

[JSON Mapping and Relations](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774)

[Look at JSON mapping now.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=0.63) [First the JSON type, then a mapping between JSON and Java,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=2.2384615384615385) [some JSON annotations, maps on relations,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=6.624333333333333) [and embedded relation and a @OneToMany relation.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=10.249) [JSON Types.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=13.501) [JSON stands for JavaScript Object Notation,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=14.508000000000001) [so it's JavaScript, and JavaScript knows the following types,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=17.529) [a string, and that's double or single quotes,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=21.116) [a number, a Boolean true or false, a list in square brackets,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=24.805600000000013) [and an object in curly brackets, and a Null is also available.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=29.286571428571435) [Then JSON to Java this is string and then Java is the string,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=34.475) [number, and in Java it's an integer,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=39.837) [and when it's a floating point number we can do a double in Java.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=42.64615384615383) [This is a Boolean in Java, object in JSON,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=49.15971428571427) [and it's a class in Java, a list of strings,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=53.66238461538462) [and in Java is a list of strings.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=58.407) [A null is an object null.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=61.62528571428574) [There are also some annotations,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=64.039) [@JsonProperty("first") will rename a property and](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=66.32462500000003) [@JsonIgnore will ignore this property, and we can ignore more properties.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=68.90145454545456) [We can do a @JsonInclude, so it will include or exclude nulls,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=73.66324999999999) [empty, or default failures, and these ones are used for one-to-many relations,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=79.657) [a parent-child relation, and a child-parent relation.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=85.49110000000002) [Let's start a demo.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=89.64) [We will rename some properties, we will add some properties,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=91.18953846153848) [and we'll add a relation that will be an Address class.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=95.03333333333333) [First is @Embedded, and then is a @OneToMany relation.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=98.964) [Let's extend our Friend class.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=103.426) [The first and last name properties are renamed,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=104.81308333333334) [and integer age is added, and a Boolean married,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=107.21079999999999) [but this one is ignored.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=111.31299999999996) [Let's go over to IntelliJ.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=113.592) [Set the modification to our Friend class,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=115.85824999999998) [change this one, this is our new code, generate getters and setters.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=119.031) [Let's try it at the Postman.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=128.029) [Add the first-name, and the last-name, and then age.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=130.65850000000006) [See what's happening, localhost, and it's added to the database.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=135.73449999999997) [Next will add a JSON object.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=143.442) [In Java it's a class.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=145.77763636363636) [This class is called Address, and it has two properties,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=147.724) [a street, and a city with the getters and setters.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=153.992) [We have to add an annotation, it's @Embeddable,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=163.993) [over to our Friend, and here we add an Address property,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=168.24633333333333) [embed it in this case, and also the getters and setters.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=175.09449999999995) [Okay, over to Postman.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=184.05479999999997) [We add an address, with the through property street, and the city.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=187.78) [Okay, so this is working.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=198.076) [This is the post, it's added, but we do not want one address,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=200.12599999999998) [but we want a list of addresses.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=205.38369999999998) [So here it will be a List of Address.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=208.085) [We call it addresses, generate getters and setters,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=213.46353846153843) [and this is the annotation @OneToMany, and cascade all.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=219.05400000000006) [Okay?](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=225.7283076923077) [And the address will be a real entity now,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=226.35307692307694) [so we copied the id from the Friend, put it in the Address,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=231.976) [and this one is an Entity, and now the address will be addresses,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=239.822) [and we've got square brackets because it's a list,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=246.817) [and see if this is working.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=251.3818000000001) [Yes, it works.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=254.425) [Let's add an extra address, copy this object, comma, next object.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=256.7385454545455) [Yes, this is working.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=266.6365714285714) [Let's go a GET, see what's inside the database; three friends there.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=271.608) [Let's delete one, the third one, back to the GET.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=281.206) [Yes.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=292.382) [Everything is working.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=293.514) [Sometimes you want to backReference a @ManyToOne and a @OneToMany.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=296.91) [Then you have to add some JSON annotations,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=300.765) [a @JsonManagedReference, and a @JsonBackReference,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=304.38980000000004) [otherwise you will end in an infinite loop,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=307.06109090909086) [and in Postman everything stays the same.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=311.768) [In summary, this was a project, we created the CRUD, Create Read Update Delete.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=316.292) [We added some Finders and FindBy id, and a FindBy first and last name,](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=323.054) [and we did some JSON mapping, especially some relations and embedded relation, and a one to many.](https://app.pluralsight.com/course-player?clipId=6b15c4e9-79c3-47f5-a77b-a33596109774&startTime=328.233)

[Error Handling](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67)

[Introduction](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67)

[This module is about error handling.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=0) [Why do we need error handling?](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=4.89) [What are possible solutions?](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=7.4807999999999995) [We will call a happy path and exception path, and then we do a demo.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=9.208) [We're going to use the PUT mapping, the POST mapping,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=13.791) [and we're going to use the validation API The problem,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=16.204846153846145) [why error handling?](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=20.766) [When I send the wrong data the database gets polluted,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=22.506) [and in the worst case becomes inconsistent,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=25.7436923076923) [and the client doesn't know that something is wrong,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=28.767400000000006) [so it cannot do a retry or cannot send better data.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=33.354142857142854) [Solution, we already got the happy path,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=38.884) [that's the default scenario, but now we can have exception paths,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=41.90927272727272) [sometimes called Sad Paths or Bad Paths.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=46.75950000000002) [The first thing you have to do is block write actions,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=49.881) [so your database won't get polluted, and send an error message to the client.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=52.551) [An example.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=59.464) [The PUT mapping it should be an update of an existing friend,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=60.12815384615385) [but if the friend does not exist in the database it will act as an insert.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=64.09364285714285) [It adds a new friend.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=69.51185714285715) [The solution is only update an existing friend by checking your findById,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=71.76828571428574) [and otherwise send an error message.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=77.70099999999996) [The second problem, POST can add empty friends.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=80.926) [Solution; only add a complete friend by validating the data,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=85.881) [otherwise, send an error message.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=90.873) [We've got a few types of error messages; HTTP Status Codes,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=93.675) [text messages, and JSON messages.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=98.854) [JSON messages can be Java classes.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=101.61854545454548) [For example, we are using an ErrorMessage class.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=104.936) [HTTP Status codes.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=109.644) [This is a list of Wikipedia.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=113.119) [The 100 numbers are informational responses,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=115.06463636363637) [the 200 numbers are success or you often see 200OK,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=117.24563636363636) [the 300 numbers are redirections, the 304 is often used,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=122.842) [the 400 numbers are errors, 400 itself is a bad request,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=127.468) [and 404 is a not found, and there are a lot more 400 numbers,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=132.0894666666666) [and there are 500 numbers error is known.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=140.28214285714276) [We often send a 500 internal server error.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=145.03833333333336) [Spring classes, which can help us.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=149.341) [ResponseEntity can send the happy path and the exception path.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=151.68809090909093) [An @ExceptionHandler is a separate method,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=155.84799999999998) [which can handle the exception, return an error message,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=158.4962) [and there is a @ControllerAdvice for global exception handling.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=162.494) [All the functions inside the advice will be available in all the controllers,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=166.398) [and we can send an error message here.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=170.23955555555557) [Error handling.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=173.804) [The first form the controller has the method and the happy path returns a](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=174.631) [ResponseEntity and the exception path returns a ResponseEntity.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=180.53666666666666) [The other form, the controller has this method,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=183.98900000000003) [the exception is handled in an @ExceptionHandler method,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=187.44677777777778) [so the controller can return the data and the](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=191.981) [@ExceptionHandler can return an ErrorMessage,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=195.0139999999999) [and the third form,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=197.30980000000002) [the @ExceptionHandler now is added to a separate](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=199.47220000000007) [class annotated by @ControllerAdvice,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=203.43622222222228) [so this class acts as a global exception handler.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=205.31) [Over to IntelliJ, restart the server,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=209.796) [check what's in the database with localhost:8080/friend.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=213.2727272727273) [Let's upgrade the second one that's a PUT to Body, raw, Text, JSON.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=217.60959999999997) [Copy the second one, paste it.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=227.52436363636366) [This will be Linus Torvalds.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=232.512) [Okay, that seems okay.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=237.1275555555555) [Id is 2.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=240.82) [Now let's do an id 0 and see what's happening now.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=242.95985714285715) [This is an id 3, so it's added to the database.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=250.806) [Let's check it with a GET, and now there are three elements in the database.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=254.485) [That's not okay.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=259.816) [Over to IntelliJ.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=261.763) [This is the @PutMapping, and we have to add an if,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=264.5028571428572) [if id is present, then return, but it doesn't work anymore.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=274.549) [We have to change the return type to ResponseEntity with a generic Friend.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=280.907) [Now return new ResponseEntity.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=286.909) [This is the data.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=290.1650000000001) [The HttpStatus, OK, and then the else, that's the exceptionPath,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=293.421) [and return another ResponseEntity, change the message to a friend,](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=304.48) [the current friend, and now tell that it's a BAD\_REQUEST.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=309.463) [Let's try.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=314.935) [PUT, same id, Send, but you get a Bad Request, and that's okay. And it's not added to the database.](https://app.pluralsight.com/course-player?clipId=7b46ab45-74f3-4f78-a2bb-668ec5c91b67&startTime=318.905)

[ErrorHandler](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a)

[The second example, Add Error Handling to the Create Method, the POST.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=2.179) [First we demo the problem by adding an empty friend,](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=6.966) [use the @ErrorHandler method, and we're going to create an ErrorMessage class.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=10.496) [Let's add an empty friend, clear everything,](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=17.201) [do a Send, and see what's happening.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=21.94499999999999) [This is a strange friend.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=26.689) [That's not what we want.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=29.278000000000006) [Let's look at the GET.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=31.867) [The fourth friend is edited with no name and no last name.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=34.09092857142857) [Over to the Java code.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=41.129) [It's in the create method we need an if statement here.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=43.06515384615384) [If the friends id is 0 and the friend first name is not null,](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=48.223571428571425) [and the last name is not null, then save the friend,](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=58.409571428571425) [else throw an exception, throw a new ValidationException with a message.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=68.055) [Now we add an @ExceptionHandler with the annotation ExceptionHandler.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=82.035) [The ExceptionHandler listens to the ValidationException.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=88.554) [It returns a ResponseEntity with the string,](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=93.289) [in this case, and we call our method exceptionHandler,](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=97.36999999999999) [and the input is the ValidationException,](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=102.02190909090908) [and here return a ResponseEntity with the message from the exception.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=105.55522222222223) [This is also a Bad Request.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=117.48100000000001) [Restart and over to Postman.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=123.505) [Back to the POST, the Body, and retry this one.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=129.142) [Send, and now we get a 400 Bad Request with an error message.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=134.206) [This is not exactly what we want.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=140.474) [We want a JSON text error message, and that will be the next step.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=142.8205000000001) [Back to IntelliJ.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=148.301) [We add a new class, an ErrorMessage class with two fields; status,](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=149.75853846153845) [as a string, and the message text.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=156.6818888888889) [Generate the getters and setters, and generate the constructor.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=163.909) [This is ready.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=172.451) [Back to the controller.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=174.5087) [Return types ErrorMessage now, and return a new ErrorMessage to the status 400,](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=177.25230000000002) [and the message is the same message from the exception.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=188.62923076923084) [We don't need HTTP status.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=194.83399999999997) [We will add to the HTTP status to the @ResponseStatus, and the HttpStatus.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=197.269) [BAD\_REQUEST.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=206.3135) [Like the Postman, we do a POST, the same empty friend.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=207.668) [This is okay.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=215.988) [We get a JSON message now.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=217.70799999999997) [The next demo we move the @ErrorHandler to a separate class,](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=221.148) [add a @ControllerAdvice annotation, and it returns an error message.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=226.521) [Create a new class, a ControllerExceptionHandler,](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=232.803) [add a @ControllerAdvice annotation,](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=238.5) [and copy the @ExceptionHandler from the controller in the advice,](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=242.13400000000001) [and don't forget the @ResponseBody, and then we run this in Postman. Everything stays the same.](https://app.pluralsight.com/course-player?clipId=97ffc9b1-f662-431c-a138-799ae4b8cc6a&startTime=252.51036363636368)

[Validation API](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89)

[There's another possibility to validate input failures, The Java Validation API.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=1.159) [Validation is added to the model class,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=6.639454545454545) [and constraints are added as annotation to the properties.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=10.695) [In the controller a valid annotation is added to the input,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=14.286) [and when something goes wrong a MethodArgumentNotValidException is thrown.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=17.421) [Here's a list of constraints; @NotNull checks if a property is not null,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=23.404) [@AssertTrue checks if a Boolean value is true,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=27.766416666666668) [there's also an @AssertFalse,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=31.04975) [@Min and @Max checks the minimum and the maximum values of a number,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=32.796454545454544) [but it could also be a string representing a number,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=39.21676923076924) [@Size is the length of a string or the length of a collection,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=43.436) [@Digits represents a number with an integer and a fraction,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=48.469) [and a @Pattern is a regular expression,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=53.29) [@NotEmpty means the property is not null or empty,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=56.73336363636363) [and @NotBlank contains no whitespaces, @Positive means with a positive number,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=62.083000000000006) [there's also a @Negative,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=67.78339999999997) [and there's also a @PositiveOrZero and a @NegativeOrZero,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=69.329) [@Email checks if the email address is valid,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=74.502) [@Past checks if a date is in the past, there's also @PastOrPresent,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=77.1762857142857) [and @Future checks if a date is in the future,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=81.8145) [and there's also a @FutureOrPresent.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=86.459) [Over to the demo.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=88.974) [We add a validation to the friend class, a @NotBlank.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=90.89600000000002) [In the controller we add the @Valid and @ExceptionHandler](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=95.701) [on the MethodArgumentNotValidException.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=98.54500000000002) [In IntelliJ we don't need the validation in the controller anymore.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=102.605) [It moves to the model to the friend.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=107.372) [The only thing that's left here is an annotation @Valid that will do the job.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=111.68628571428569) [In the Friend class we add an annotation to the firstName, @NotBlank.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=120.299) [This is all.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=127.707) [Then in the controller this throws an exception,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=129.0943636363636) [and we need an @ExceptionHandler with a MethodArgumentNotValidException.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=132.794) [We retake a method name, then call it the exceptionHandler,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=140.667) [and the input argument is the exception,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=144.919) [and here we ask the BindingResults,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=147.88700000000006) [and in the BindingResults there are the FieldErrors.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=150.78355555555555) [We put that in a local variable, and this one we're going to map to a new class,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=156.996) [the FieldErrorMessage class with two fields,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=167.14724999999999) [the field name and the error message.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=173.142) [Getters and setters and the constructor.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=181.58669230769235) [Okay, and we use the stream API here, so the fieldErrors.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=188.825) [stream.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=197.6080769230769) [map.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=198.40653846153845) [One fieldError is mapped to a new FieldErrorMessage class.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=199.205) [First argument, the field name, and the second argument defaultMessage,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=204.33) [and we have to convert it to a list.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=209.09629999999999) [Let's collect this to a list.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=214.427) [This returns a list of fieldErrorMessages.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=219.72000000000006) [Okay, now we have to add the return type, but IntelliJ can do that for me.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=225.013) [There's still an error.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=231.995) [That's not okay.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=234.7939090909091) [It should be.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=236.89309090909092) [class.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=238.99227272727273) [Add a @ResponseStatus, HTTP BAD\_REQUEST.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=239.692) [Use an empty friend again.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=248.272) [Don't look at the URL.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=250.92800000000005) [Send.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=253.584) [We got a BAD\_REQUEST with a field name firstName,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=254.0175) [and the message is, must not be blank.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=257.919) [That's okay.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=261.55899999999986) [A reminder.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=262.469) [Error handling is part of the software architecture.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=263.752) [Architectural decisions should be consistent.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=268.884) [Examples of decisions are the name of the exceptions,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=272.062) [the shape of the error messages, the flow of the exception handling,](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=275.1085000000001) [is it global or is it local, and so on.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=279.564) [In summary, error handling is about the happy path and the exception path.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=285.076) [Spring has many annotations and classes to help us. When it comes to error handling, be consistent, think like an architect.](https://app.pluralsight.com/course-player?clipId=fd6709a3-e9ed-4d5b-84aa-188761b7df89&startTime=291.464)

[Testing](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827)

[Introduction](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827)

[This module is about software testing.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=0) [Why do we need testing?](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=4.766) [What type of tests are there?](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=6.557818181818181) [And we'll talk about the testing levels.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=8.708) [In the demos we will write tests on the different levels.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=10.977076923076925) [On every test level another library from Spring Boot is used.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=14.671090909090905) [You could see it as a cookbook on different tests.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=18.868999999999993) [Why testing?](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=23.067) [We want to be sure that the software meets the requirements,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=23.766) [responds correctly to input, that's input validation,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=27.764875) [performs an acceptable time, that's a performance test,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=31.292) [it can be installed and run, the deployment test,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=33.72618181818181) [and achieves the result for the stakeholders.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=36.714538461538474) [These could be busy results, but could also be non-function required,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=39.136) [like security, usability, maintainability,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=42.889) [and other kinds of ilities.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=45.164111111111104) [Here is an example of test types.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=48.008) [A smoke and sanity test tests if the software will run even in a basic form.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=50.323500000000024) [Continuous testing means that the tests are running in every iteration.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=56.18) [When we run Maven this is the case.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=59.515) [Regression tests where we add new programming code or alter existing code.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=62.1292857142857) [We want to be sure that the other code keeps on working.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=65.92507692307689) [Performance steps are meant to measure the time it takes to run the software,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=70.25327272727272) [and acceptance steps are used to check if the](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=75.34692307692303) [stakeholders are happy with the software,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=78.25759999999998) [and that they are willing to pay the bill.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=80.309) [Testing Levels.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=83.867) [Unit tests are the smallest building blocks.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=84.78211111111112) [Every class in the programming code has a companion unit test class.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=87.985) [The test is isolated from the other class by mocking the method call.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=91.782) [Integration tests are easier to implement.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=97.512) [We'll test a class with all the dependencies.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=100.558) [Here you can assure that the path through the software is working,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=103.155) [but when it fails we do not know which class is failing.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=106.627) [A system test checks the complete system,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=110.867) [including hardware operating system web service,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=114.12339999999998) [and so on.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=117.63300000000001) [This is a picture of the different test types used in](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=119.747) [Wired Brain Friends application.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=123.29999999999993) [Starting with the system test,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=124.592) [then the integration test of the controller with the surface later,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=126.21836363636366) [this is the unit test on the surface layer.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=130.5842857142857) [Next,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=134.93) [we'll test if the controller will stand alone and the service layer is mocked,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=135.15858333333335) [and the last two tests are exception to test the first controller,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=138.95149999999998) [and then the complete system.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=144.417125) [You could see it as a cookbook on different tests.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=147.134) [Spring Boot has a lot of test libraries.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=150.1150909090909) [JUnit is the defacto standard for all the testing libraries in the Java world.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=153.694) [Spring Test is the Spring Boot test support,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=158.181) [including the annotation we will be using.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=160.88572727272728) [AssertJ is a fluent assertion library.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=163.92639999999997) [Hamcrest are for matchers.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=166.966) [JSONPath is an XPath for JSON, so we can walk through the JSON structure,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=168.84199999999998) [and Mockito is a mocking framework.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=176.30709090909085) [In all the examples we will use the same test pattern.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=180.2) [First we create a friend,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=183.246) [then we list all the friends and asserted the friends in the list,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=184.87869230769232) [and after that we delete a friend, and assert if the list is empty.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=189.01100000000002) [We'll start with the Smoke and Sanity test.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=193.646) [We check if the Spring Context is running.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=195.732) [When Spring Initializer generates a product for us a test is included.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=199.629) [In this test we assert if the friend controller is alive.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=203.542) [This is the friend controller with the create,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=207.38) [read, update, and delete function.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=211.5992) [This is the friend with the id, first and last name,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=213.75799999999998) [and we will generate two constructors for convenience,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=218.4545454545454) [and we need an empty constructor for JPA.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=223.128) [Over to the test.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=227.714) [This is the generated test.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=229.89266666666668) [We add a FriendController, and here we assert that it's not null.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=232.616) [@RunWith means the JUnit is used, but the Spring will test annotation,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=238.495) [SpringBoot application is started inside the test.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=243.26175) [The Spring context will instantiate the FriendController,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=247.327) [and inside the Test function we check that the FriendController is not null.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=250.4) [Let's run the test.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=254.888) [Maven test.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=257.456) [Here we see 1 run, 0 failures.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=258.74) [We can also check a single test with the FriendsApp](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=263.234) [with a wildcard and then the test, and also 0 failures.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=266.45476923076905) [The next test is a system test.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=274.762) [It will test the complete FriendApplication.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=277.40261538461544) [We will use the Spring template as a client,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=279.666) [and it has the same role as Postman,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=281.8331999999998) [and the RestTemplate class has different methods for us.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=284.192) [On the GET this getForObject and a getForEntity.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=288.305) [On the POST is a postForObject and a postForLocation.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=292.067) [There's also a PUT and a DELETE,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=295.903) [and if we don't know we can also use exchange or execute.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=298.23618749999986) [In the demo we have to run the server and use the same](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=305.166) [test pattern with JUnit assertions,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=309.50539999999995) [and in the RestTemplate we do a postForEntity to add a friend,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=311.23979999999995) [a getForObject to list of Friends, and delete a friend.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=315.879) [This is the test.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=320.843) [Test function test create and read.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=322.33059999999995) [We use the restTemplate on the URL friend.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=324.562) [Create a new friend, and add it with the post for entity on the URL.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=328.675) [Then we do a GET for object on the restTemplate,](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=334.439) [and we assert that it contains a Friend with the first name Gordon.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=337.94633333333326) [Next, delete a friend on the same URL, and assert that the list is empty. Start the server, and then run the test to test serial failures.](https://app.pluralsight.com/course-player?clipId=091ac9a2-7634-4b66-a3ac-dd2f1c37e827&startTime=343.933)

[Integration and Unit Testing](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862)

[The next example is integration test.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=1.28) [We will test the FriendController together with the](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=2.6756) [FriendService directly in the Java code.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=5.926399999999999) [In the demo we use @SpringBootTest, and @Autowired FriendController,](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=9.407) [and we used the same test pattern with AssertJ assertions.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=14.113) [Again, I @RunWith, so JUnit can work with Spring,](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=18.98) [@SpringBootTest, so the Spring context is loaded in the test.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=23.262199999999996) [From the context we can get the FriendController with @Autowired,](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=26.319) [create a new Friend, and add it to the create on the friendController,](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=30.72) [then on the controller we read the friends,](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=36.321) [we get the Iterable, and I need to look at the first one,](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=39.22033333333334) [and ask if it has a field with the value firstName Gordon.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=42.50229411764708) [Next, a friend is deleted, and then we check if the list is empty.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=46.13166666666666) [Run the test.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=54.10433333333334) [Three tests, zero failures.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=56.09) [The next example is a JPA test.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=59.567) [We test the FriendService and the database.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=62.62250000000002) [We'll use a @DataJpaTest annotation and @Autowired FriendService and we'll](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=65.678) [use the same test pattern with the AssertJ assertions.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=70.84020000000001) [This is the @DataJpaTest.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=75.765) [This allows us to only test the data layer.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=77.30500000000002) [Because we're using MySQL we need to auto config the test database.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=80.77) [It won't be necessary with an embedded database,](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=84.884) [and we got the FriendService from the context.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=87.56933333333335) [Create a new friend, add it to the service with save,](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=90.69366666666664) [and then a findAll on the service,](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=95.32615384615384) [assert if it contains a friend with the first name Gordon,](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=97.82084615384613) [you will delete all the friends, and check if the list is empty.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=101.28569230769234) [Start a test; 4 tests, 0 failures.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=107.0251333333333) [The next test is a unit test.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=110.737) [We will test the controller in a standalone mode,](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=112.38300000000002) [and mock the FriendService using Mockito,](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=115.10725) [and we'll use the pattern of when and thenReturn,](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=118.342) [and on the assertion we'll use JSONPath,](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=121.17976923076928) [this is an XPath for JSON, so it can walk the JSON instruction.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=123.50618750000001) [The demo, it's a @WebMvcTest that's another Spring context,](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=128.122) [and we @Autowired MockMvc, and we use a @MockBean.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=133.315) [Use the same test pattern,](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=137.973) [and on the MockMvc there is a perform and andExpect that we](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=139.25800000000004) [can ask for the status and the jsonPath.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=142.82881818181818) [Again, the @RunWith.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=146.183) [The @WebMvcTest allows us to call URLs on the test context.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=147.55485714285714) [It will mock the friendService, so it is a @MockBean,](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=152.585) [and here we get the MockMvc from the context.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=155.72499999999994) [Here we'll create the mock data, a friend is added to the list,](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=159.22554545454548) [and when the friendService.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=163.93069230769228) [findAll is called we return that list, get on the list,](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=165.08023076923072) [and normally if it turns list of friends from the database,](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=168.69250000000005) [and now we return the mock data.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=172.61753846153852) [We check if it's okay, and on the jsonPath we check the size of the list,](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=175.228) [which is one, and we check the first element should be firstName Gordon.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=180.04314285714284) [Start a test. If I look, 5 tests, 0 failures.](https://app.pluralsight.com/course-player?clipId=d003baad-1bfb-4324-a7d7-6d7285ad5862&startTime=185.583)

[Testing Error Handling](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9)

[In this example we will test the exceptions on the FriendController.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=1.657) [We will use @SpringBootTest.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=7.353) [In the friend controller we create the method,](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=8.964272727272727) [somethingIsWrong, that throws a ValidationException.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=12.440166666666666) [And in the test we check if the exception is thrown.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=15.721) [Of the new FriendController add a new method,](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=20.262) [and that's mapped on the wrong URL, and it throws a ValidationException.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=23.183999999999997) [Over to Integration Tests,](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=29.309) [and here we add a new test method with the @Test and the expected exception,](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=30.64214285714286) [and we call friendController.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=35.793124999999996) [somethingIsWrong.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=37.5608) [We run the test, and you will see 6 runs, 0 failures.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=38.2376) [In this example we do the error handling on system level with the RestTemplate,](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=46.779) [and assert the HTTP response to be a bad request.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=50.45009090909091) [In this demo we run the server, run the test template,](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=54.171) [the getForEntity, which returns the ResponseEntity.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=60.315) [We will add an @ExceptionHandler, which catches the ValidationException,](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=64.964) [and then returns a message, and the HTTP error code should be BAD\_REQUEST.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=68.0475) [Over to the test, the SystemTests.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=74.5146) [We add a new method, and we'll use the RestTemplate on the URL role.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=77.715) [Do a getForEntity.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=83.594) [In this case, we have to try a catch on the HttpClientException,](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=84.3870714285714) [and inside the catch we asserted that the status code is a BAD\_REQUEST.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=87.7402) [Run the test.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=94.60533333333335) [Seven runs and zero failures.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=96.74908333333336) [Let's have a look at Spring Rest Testing.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=100.322) [On the system level we have the RestTemplate,](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=102.358) [we have the GET, POST, PUT, and DELETE,](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=104.45133333333337) [and when something goes wrong that gets an exception,](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=106.94066666666664) [in the integration test we will use the @SpringBootTest annotation,](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=110.1353) [and that will start a Spring Boot application,](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=113.235) [so we can get the elements from the context with @Autowired.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=115.76300000000002) [In a unit test you want to use mocking to isolate](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=120.354) [your class from the environment,](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=122.89499999999994) [and we will use the @WebMvcTest environment with Mockito and @MockBean.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=124.32191666666667) [Let's have a look at assertions, in other words matchers.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=129.854) [There's JUnit, Hamcrest, and AssertJ.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=133.641) [As you can see, they're all doing the same job in nearly the same way.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=136.78877777777777) [The only difference is JUnit is the standard,](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=142.511) [Hamcrest is the most popular, and AssertJ is the most active.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=144.9423333333334) [And I want to make a remark.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=149.783) [Tests should be readable because non-programmers should](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=151.92558333333332) [be able to read or change a test.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=154.0522) [In an Agile team programmers work together with testers and analysts,](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=156.437) [and there the tests and the specifications are the common ground,](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=161.198) [so everybody should be able to read the test and even](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=165.106) [alter the tests when necessary.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=169.3471874999999) [In summary, Spring Boot has a lot of testing libraries.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=171.275) [For every level a library is available.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=176.419) [There are also a lot of assertion libraries to make your tests as readable as possible.](https://app.pluralsight.com/course-player?clipId=142ce3db-a0b9-4b8d-87fa-7072a47af3c9&startTime=178.80707692307692)

[Hateoas](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475)

[Hateoas: Basic](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475)

[This module is about HATEOAS.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=0) [First we'll look at what's HATEOAS, and then we do three demos,](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=5.643) [a simple demo, the second is about the one-to-many relation,](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=10.485) [and the third is about a many-to-many relation.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=13.926) [We'll look at Wikipedia, and it says,](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=18.614) [HATEOAS is Hypermedia As The Engine Of Application State and it's a](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=21.811249999999998) [compliment of the REST application architecture.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=27.546538461538464) [What does that mean?](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=30.505) [It's part of REST, and uses hypermedia and hypermedia are hyperlinks.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=31.926666666666666) [In other words, it's JSON data with hyperlinks.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=36.80018181818181) [Here's an example.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=42.214) [This is the friend JSON message with the first and last name,](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=43.21600000000001) [and we see two links, the self and the friend.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=47.611999999999995) [In the Richardson Maturity Model HATEOAS is the highest level of maturity.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=52.892) [When we look at the different levels level 0,](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=57.205) [Swamp of POX, these are plain old XML messages sent over remote procedure call.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=60.32874999999997) [Mostly post messages over the same URL.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=65.556) [Next are resources.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=69.12669999999996) [In the URL the different messages are sent,](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=70.657) [for example, an update message is sent over an update URL.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=73.00100000000005) [The next level, HTTP Verbs.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=76.823) [These ones we used in the previous example,](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=78.7100833333333) [these are REST messages sent over the same URL and](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=81.70775) [HTTP methods are the differentiator, and HATEOAS is REST with hyperlinks.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=85.26525000000001) [Over to the first demo.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=92.686) [First, we run the basic CRUD example using the third module,](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=94.9683076923077) [and in Postman we look at the output,](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=99.67257142857142) [and then we will refactor this example to HATEOAS,](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=102.47942857142851) [and first we add the spring-rest-data to the POM,](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=105.89) [and then we delete the FriendController,](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=109.50620000000002) [and we look in Postman what's happening.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=111.983) [Over to IntelliJ.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=114.485) [This is the friend class with first and last name.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=115.41615384615386) [This is the FriendService, and this is the FriendController with the create,](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=118.52) [read, update, and delete methods.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=123.70783333333333) [We start the server, and we look at localhost friend.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=127.357) [In this result we got Ada Lovelace and Charles Flint with id 1 and id 2.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=132.029) [Next, we add the Spring REST Data to the POM,](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=139.093) [and we delete the FriendController.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=145.703) [Restart the server, and now friend has become friends,](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=149.008) [it's pluralized, and here we see the result,](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=155.331) [two friends with the links to self and friend.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=158.5856666666667) [All the CRUD operations are available, so let's add a friend, Gordon Moore.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=163.0911818181818) [This is friend number three.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=172.0359166666666) [And we do a GET again.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=175.958) [We see three friends.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=178.76179999999997) [Next, let's delete a friend.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=180.631) [That's also available.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=183.53975) [Friend number three.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=185.285) [Let's do the GET again on the friends, and we click on the first one. We get a findById, friend number one, and that's Ada Lovelace.](https://app.pluralsight.com/course-player?clipId=822cb44e-11fb-4b48-97c3-e86b0eed7475&startTime=187.79763636363634)

[Hateoas: One-to-many](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173)

[In the second demo a OneToMany relation is added.](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=2.241) [We will use Address from the third module and an AddressService,](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=6.718) [and we take a look at the data structure in the database](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=11.452) [because the hyperlinks in the JSON messages will be mirrored](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=14.119428571428573) [to the primary and foreign keys,](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=16.81666666666667) [and in Postman we will add an address with the hyperlink to a friend,](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=18.601833333333335) [then we'll list all the friends to see that the address is added.](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=22.901142857142858) [Create a new class Address with an id, a street, and a city.](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=27.829) [Getters and setters, and the annotations for JPA,](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=35.438) [and to the friend that will be a set of addresses called addresses,](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=40.04600000000001) [so a @OneToMany relation with the getters and setters.](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=46.54175) [Back to the Address.](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=52.72639999999999) [Add a Friend back reference called friend.](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=55.918) [This is a @ManyToOne, and also getters and setters.](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=61.206) [Make it to the Friend, and here is a mappedBy friend from the other side.](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=67.715) [Restart the server.](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=77.305) [Lookup the database, the tables, address, and friends.](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=78.70075) [We also look at friend and id, first, and last name,](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=81.93) [and this is the address, id, city, street,](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=85.48618181818183) [and friend id, and that's the foreign key.](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=88.548) [Over to Postman.](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=92.415) [We see these are the friends and the addresses are empty,](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=94.0832307692308) [and we have to create an AddressService that extends](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=100.54863636363636) [the CrudRepository to an address, and the id is an integer.](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=108.69036363636357) [We add an address, Parklane, Little Town,](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=116.971) [and here we get the id, we copy this one from the link of Ada Lovelace.](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=120.7100769230769) [Okay, that is not okay because it's addresses, it's pluralized.](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=132.576) [Make it to the friends.](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=138.069) [Now the address link is added, and when we follow that link we see that the rest is there.](https://app.pluralsight.com/course-player?clipId=558b14f2-b71b-457a-896d-76836dd96173&startTime=140.35542857142858)

[Hateoas: Many-to-many](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817)

[In the next demo we'll see the hyperlink to foreign key](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=0.16) [mapping is also valid for the @ManyToMany.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=2.709461538461538) [First, we add the @ManyToMany to the Friend,](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=4.508416666666667) [then the @ManyToMany to the Address.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=6.828857142857143) [Then we'll take a look at the data structure.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=9.534) [In Postman we add a friend, add an address,](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=12.431) [and add a link from the friends to the address.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=15.339199999999995) [Over to IntelliJ.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=19.906571428571432) [This becomes a @ManyToMany.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=21.878) [This will become a Set of Friends called friends,](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=24.55066666666667) [getters and setters, over to the Friend class.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=30.71922222222222) [This is also a @ManyToMany.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=37.305) [It maps to friends now.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=41.369000000000014) [Restart the server.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=45.433) [Let's have a look at the database, the tables,](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=47.30825000000001) [an address, a friend, and a coupling table,](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=52.934) [address\_friends.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=56.80066666666667) [The friend is standard, id, first and last name,](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=57.284) [the address the same, id, city, and street,](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=59.87253846153847) [and here is the coupling table with an addresses\_id and a friends\_id.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=62.273666666666664) [Over to Postman.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=69.08814285714286) [These are the friends.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=72.323) [These are the addresses.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=73.4835) [Both are empty, and we add a friend, Ada Lovelace.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=74.644) [We have to edit to friends.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=80.301) [That's the first 1 with the id 1.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=83.86242857142854) [Then we add an address to the addresses, Parklane, Little Town.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=88.611) [This is the address, address with id 1.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=97.965) [Let me follow this link, and see that the addresses are still empty,](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=102.31576923076918) [and now we pull from the other side, from the @ManyToMany,](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=107.08357142857145) [from the address friend side we also see that this is](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=111.93000000000004) [empty because we have to add a link.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=118.78766666666664) [That's a POST message.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=123.235) [Do a special header type.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=125.13455555555554) [The Content-Type is a text/uri-list, and here we put the friends number 1.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=127.509) [This is the coupling table.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=139.489) [Anyway, we look at the friends now, we go to the addresses friends,](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=141.22941666666668) [we see when we click on this link and follow that one here](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=145.95800000000003) [we see the addresses are visible now, and we look inside the database.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=151.86727272727276) [We see the first one, the first friend, Ada Lovelace,](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=158.41800000000006) [then address, Parklane, Little Town, and this is the coupling table,](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=164.17850000000004) [addresses\_id 1, friends\_id 1.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=171.084) [Next, we will serve the data structure starting with the friends.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=173.084) [This is a list of friends.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=177.864) [We can follow this link.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=179.5909090909091) [We get the first friend.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=181.03) [In the first friend we can go to the addresses,](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=182.44785714285717) [and get all the addresses from the first friend,](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=185.7577) [and the addresses we can go back to the friend,](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=192.577) [Ada Lovelace again, and we can follow the self-link again.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=196.383923076923) [As you can see, you can come everywhere by following the links.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=200.684) [It's just like the hyperlinks on the world wide web.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=203.906) [In summary, HATEOAS is JSON with hyperlinks.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=208.252) [You get them for free with the Spring-Data-REST library,](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=212.65) [and hyperlinks to mirror the primary and the foreign keys of the data model,](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=217.006) [so now the data model and the JSON message are closely related, and you can surf them like the world wide web.](https://app.pluralsight.com/course-player?clipId=3bfaf446-d30a-4e6e-8080-baed4d7b0817&startTime=220.71938461538463)