



Angular - Interaction with REST APIs

- Aditya Kumar
Chief Technology Officer, edwisor.com

We will cover with the following concepts

- 1) Concept of REST API
- 2) Concept of Http requests and Http verbs
- 3) Making an http request
- 4) Concept of observables
- 5) Rx/Js
- 6) Exception handling
- 7) Handling subscriptions properly
- 8) Why use services for making Http calls.

Give it a REST !

- REST or Representational state transfer is an architectural style for writing web services
- In this style, resources are called using a combination of urls and then endpoints
- Lets see the example of some REST APIs - LinkedIn and Twitter

Give it a REST !

- As an Angular Developer or a Frontend Developer, you are expected to know -
 - How to test/check the APIs
 - Use the APIs in your application.
- Let's look at the documentation of the APIs, I have prepared for you -
<https://blogapp.edwisor.com/apiDoc/>
- Lets test these APIs using POSTMAN tool

API testing is critical!

- You must test your apis before starting the integration to make sure they are not buggy!
- API testing also gives you better clarity on the headers, body parameters, query parameters and url parameters.

Making Http requests in angular is easy



Observables may seem tricky, but are they?

- <http://reactivex.io/documentation/observable.html>
- Observables are kind of event driven equivalent of promises. They are specialized in scenarios where you have to make numerous concurrent requests.
- Lets understand the concept of observables using the example of search functionality that you see in a lot of website.
- Story of Observables and Rx/Js

Handling observables is very easy

Angular http module by default returns and observables.

Step 1

You have to handle the subscribe event in your component

Step 2

Do necessary operations for case of data received or error case

Step 3

Exception handling is important for robustness of the application.

- Let's perform excellent exception handling to make sure we are covered from all sides.

Are you wondering why make http calls in services?

- Services are instantiated once.
- Most of the times you can reduce the number of requests made to the server by using the facilities offered by services like one time call and data passing hence making our frontend application much faster.

The next steps are ...

Using Forms

- Two way data binding using NgModel
- Creating and updating information
- Handling State of things and responding to events