Spring Boot RestTemplate shows how to use RestTemplate to create synchronous HTTP requests in a Spring application.

RestTemplate is a synchronous client to perform HTTP requests. It uses a simple, template method API over underlying HTTP client libraries such as the JDK HttpURLConnection, Apache HttpComponents, and others.

Since Spring 1.5.0, a new client WebClient is available that can be use do create both synchronous and asynchronous requests. In the future releases, RestTemplate will be deprecated in favour of WebClient.

Rest Template is used to create applications that consume RESTful Web Services. You can use the exchange() method to consume the web services for all HTTP methods. The code given below shows how to create Bean for Rest Template to auto wiring the Rest Template object.

REST stands for REpresentational State Transfer. REST specifies a set of architectural constraints. Any service which satisfies these constraints is called RESTful Service.

The five important constraints for RESTful Web Service are

Client - Server : There should be a service producer and a service consumer.

The interface (URL) is uniform and exposing resources.

The service is stateless.

The service results should be Cacheable. HTTP cache, for example.

Service should assume a Layered architecture. Client should not assume direct connection to server - it might be getting info from a middle layer - cache.