[**http://www.journaldev.com/7969/spring-boot-tutorial**](http://www.journaldev.com/7969/spring-boot-tutorial)

**Why Spring Boot?**

Assume that we are talking about an application that accepts request from any browser, do certain computing then persist data in specific type of database. A little bit more technical, we are talking about the flow of

Browser <> RESTful layer <> Middle layer <> Database

**Spring framework**is a Injection dependency framework at first (it's still as it is today) targeting managing life-cycle of Java components (what so-called ***beans***). Today, Spring framework is pretty bloated with tons facilities/helpers on top of it; but if you look at the big picture, it's still a framework that glue things together, a middle man to MVC frameworks (Struts 1,2, JSF etc), ORM frameworks (Hibernate, iBatis, JOOQ etc) and other necessary facilities (Quartz, Email, you can tell, whatever you need, most likely, there's a Spring support). Back to our assumed flow, it takes quite a lengthy tutorial to set Spring framework up and running because Spring framework nature is to provide flexibility of choices to **you.**

**Spring boot**on the other hand is built on a totally different *mantra*. It's basically a suite, pre-configured, pre-sugared set of frameworks/technologies to reduce boiler plate configuration providing you the shortest way to have a Spring web application up and running with smallest line of code/configuration ***out-of-the-box.*** As you can see from there [Spring Boot](http://projects.spring.io/spring-boot/) page, it took less than 20 LOC to have a simple RESTful application up and running with almost *zero*configuration.

Spring Boot provides the extensions of **configuration on convention** model.

1. Spring Boot reduces the need to write a lot of configuration and boilerplate code.
2. It has an opinionated view on Spring Platform and third-party libraries so you can get started with minimum effort.
3. Easy to create standalone applications with embedded Tomcat/Jetty/Undertow.
4. Provides metrics, health checks, and externalized configuration