**My Learnings while Spring boot startup :-**

# Issue 1 : [@RestController in other package doesn't work](http://stackoverflow.com/questions/33039774/restcontroller-in-other-package-doesnt-work).

Solution : Using a @SpringBootApplication annotation is equivalent to using @Configuration, @EnableAutoConfiguration and @ComponentScan.

From the documentation:

ComponentScan configures component scanning directives for use with @Configuration classes. Provides support parallel with Spring XML's element.

One of basePackageClasses(), basePackages() or its alias value() may be specified to define specific packages to scan. If specific packages are not defined scanning will occur from the package of the class with this annotation.

You can either move it as you did or specify basePackages in @ComponentScan

ISSUE2 : How to switch off defalt error page, i.e WHITE LABLE ERROR page.

Solution :- You have to write server.error.whitelable.enabled = false property in the application.properties file. Which is under src/main/resources/application.properties.

**MAIN APPLICATION AND ANNOTATIONS:**

package hello;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class Application {

public static void main(String[] args) {

SpringApplication.run(Application.class, args);

}

}

**@SpringBootApplication**is a convenience annotation that adds all of the following:

**@Configuration**tags the class as a source of bean definitions for the application context.

**@EnableAutoConfiguration**tells Spring Boot to start adding beans based on classpath settings, other beans, and various property settings.

Normally you would add **@EnableWebMvc**for a Spring MVC app, but Spring Boot adds it automatically when it sees spring-webmvc on the classpath. This flags the application as a web application and activates key behaviors such as setting up a DispatcherServlet.

@ComponentScan tells Spring to look for other components, configurations, and services in the the hello package, allowing it to find the HelloController.

**The main()**method uses Spring Boot’s SpringApplication.run() method to launch an application. Did you notice that there wasn’t a single line of XML? No web.xml file either. This web application is 100% pure Java and you didn’t have to deal with configuring any plumbing or infrastructure.