1. 基于你对springboot的理解描述一下什么是springboot

SpringBoot是一种用于快速、高效地构建基于Spring框架及Spring生态体系的应用解决方案。它是一种微框架，也就是框架的框架。采用约定大于配置的方式减化了配置文件，从而实现Bean的自动装配，并提供Starter使开发人员在使用组件时能够做到开箱即用。

1. 约定优于配置指的是什么？

1、SpringBoot在创建项目时约定好目录结构，如：main/src存放源码文件,main/resources存放配置文件，默认打包方式为jar包。

2、spring-boot-starter-web中默认包含springmvc相关依赖以及内置的tomcat容器，使得构建一个web应用更加简单。

3、默认提供application.properties/yml文件。

4、默认通过spring.profiles.active属性来决定运行环境时读取的配置文件。

5. EnableAutoConfiguration默认对于依赖的starter进行自动装载。

3.@SpringBootApplication由哪几个注解组成，这几个注解分别表示什么作用

由@ComponentScan、@SpringBootConfiguration、@EnableAutoConfiguration等注解组成。

@ComponentScan注解用于扫描@Service、@Controller等Spring定义的注解Bean。

@SpringBootConfiguration相当于Configuration，通过注解方式装配Bean。

@EnableAutoConfiguration用于自动装配Bean，通过ImportSelector和ImportBeanDefinitionRegister接口以及@Conditional注解来实现动态装配。

1. springboot自动装配的实现原理

通过Spring SPI机制实现，@SpringBootApplication注解中同时包括了@ComponentScan、@SpringBootConfiguraion、@EnableAutoConfiguration注解，其中@EnableAutoConfiguration注解中又引入了@Import(AutoConfigurationImportSelector.class），在AutoConfigurationImportSelector.class中先扫描扩展组件包中的classpath/META-INF/spring-autoconfiguration-metadata.properties，再扫描扩展组件包中的classpath/META-INF/spring.factories，@EnableAutoConfiguration只加载key=EnableAutoConfiguration对应的配置项，并将此配置项中的对应的Configuration类自动加载到Spring IOC容器中。

（1）SpringApplication.run(AppConfig.class,args);初始化Spring IOC容器，其中refreshContext(context);方法会解析配置类上的标签，实现自动装配功能的注解@EnableAutoConfiguration。

（2）@EnableAutoConfiguration注解中@Import引入的动态配置类：AutoConfigurationImportSelector。

（3）AutoConfigurationImportSelector这个类中有这个方法.SpringFactoriesLoader.loadFactoryNames(getSpringFactoriesLoaderFactoryClass(), getBeanClassLoader());  
（4）SpringFactoriesLoader.loadFactoryNames的作用就是读取jar包中的/项目中的META-INF/spring.factories文件.  
（5）spring.factories配置了自动装配的类.

比如:

com.alibaba.druid.spring.boot.autoconfigure.DruidDataSourceAutoConfigure

（6）根据条件自动装配Bean。

5.spring中的spi机制的原理是什么？

SPI的全名为Service Provider Interface，为某个接口寻找服务实现的机制。

当服务的提供者，提供了服务接口的一种实现之后，在jar包的META-INF/services/目录里同时创建一个以服务接口命名的文件。该文件里就是实现该服务接口的具体实现类。而当外部程序装配这个模块的时候，就能通过该jar包META-INF/services/里的配置文件找到具体的实现类名，并装载实例化，完成模块的注入。通过这个约定，就不需要把服务放在代码中了，通过模块被装配的时候就可以发现服务类了。