# Spring Boot答案

# 简述题：为什么导入dependency时不需要指定版本？

spring-boot-starter-parent依赖

spring-boot-starter-web依赖

# 编程题：使用Spring Boot整合Mybatis?

参考文档：03\_Spring Boot案例\_day02.docx

# 面试题：Spring Boot到底是如何进行自动配置？

Spring Boot应用的启动入口是@SpringBootApplication注解标注类中的main()方法

|  |
| --- |
| @SpringBootApplication  public class SpringbootDemoApplication {  public static void main(String[] args) {  SpringApplication.run(SpringbootDemoApplication.class, args);  }  } |

进入到@SpringBootApplication内，观察其做了哪些工作：

|  |
| --- |
| @Target({ElementType.TYPE}) //注解的适用范围,Type表示注解可以描述在类、接口、注解或枚举中  @Retention(RetentionPolicy.RUNTIME) //表示注解的生命周期，Runtime运行时  @Documented //表示注解可以记录在javadoc中  @Inherited //表示可以被子类继承该注解  @SpringBootConfiguration // 标明该类为配置类  @EnableAutoConfiguration // 启动自动配置功能  @ComponentScan(excludeFilters = { @Filter(type = FilterType.CUSTOM, classes = TypeExcludeFilter.class),  @Filter(type = FilterType.CUSTOM, classes = AutoConfigurationExcludeFilter.class) })  public @interface SpringBootApplication {  // 根据class来排除特定的类，使其不能加入spring容器，传入参数value类型是class类型。  @AliasFor(annotation = EnableAutoConfiguration.class)  Class<?>[] exclude() default {};  // 根据classname 来排除特定的类，使其不能加入spring容器，传入参数value类型是class的全类名字符串数组。  @AliasFor(annotation = EnableAutoConfiguration.class)  String[] excludeName() default {};  // 指定扫描包，参数是包名的字符串数组。  @AliasFor(annotation = ComponentScan.class, attribute = "basePackages")  String[] scanBasePackages() default {};  // 扫描特定的包，参数类似是Class类型数组。  @AliasFor(annotation = ComponentScan.class, attribute = "basePackageClasses")  Class<?>[] scanBasePackageClasses() default {};  } |

从上述源码可以看出，@SpringBootApplication注解是一个组合注解，前面 4 个是注解的元数据信息， 我们主要看后面 3 个注解：@SpringBootConfiguration、@EnableAutoConfiguration、@ComponentScan三个核心注解，关于这三个核心注解作具体说明即可。