**Spring Boot introduces the**[**@SpringBootApplication**](https://www.baeldung.com/spring-boot-annotations#spring-boot-application)**annotation**. This single annotation is equivalent to using @Configuration, @EnableAutoConfiguration, and @ComponentScan.

As a result, when we run this Spring Boot application, **it will automatically scan the components in the current package and its sub-packages**. Thus it will register them in Spring's Application Context, and allow us to inject beans using @Autowired.

In Spring, the objects that form the backbone of your application and that are managed by the Spring IoC container are called beans. A bean is an object that is instantiated, assembled, and otherwise managed by a Spring IoC container. The life cycle of Bean is controlled by Spring.

 the BeanFactory provides the configuration framework and basic functionality, and the ApplicationContext adds more enterprise-specific functionality. The ApplicationContext is a complete superset of the BeanFactory and is used exclusively in this chapter in descriptions of Spring’s IoC container. For more information on using the BeanFactory instead of the ApplicationContext, see the section covering the [BeanFactory API](https://docs.spring.io/spring-framework/docs/current/reference/html/core.html" \l "beans-beanfactory).