@SpringBootApplication — это основная аннотация, используемая при работе с приложениями Spring Boot.

При работе над приложением Spring мы можем часто использовать следующие 3 аннотации:

@Configure

@EnableAutoConfiguration

@ComponentScan

Эти аннотации являются наиболее распространенными при работе с любым приложением Spring, поскольку они являются строительными блоками.

Основываясь на их частом использовании, Spring Boot предоставляет удобную альтернативу @SpringBootApplication.



1. @SpringBootApplication

Аннотация @SpringBootApplication — это удобная аннотация, эквивалентная объявлению @Configuration, @EnableAutoConfiguration и @ComponentScan. Вы можете использовать эту аннотацию вместо трех вышеперечисленных аннотаций.

1.1 Аннотация @Configuration

Указывает, что класс объявляет один или несколько методов @Bean и используется контейнером Spring для создания определений компонентов и запросов на обслуживание для этих компонентов во время выполнения. Это важно для поддержки конфигурации на основе Java вместо конфигурации XML.

1.2 @EnableAutoConfiguration

Эта аннотация включает функцию автоматической настройки Spring Boot.

1.3 @ComponentScan

Эта аннотация включает сканирование компонентов, так что классы веб-контроллера и другие компоненты автоматически обнаруживаются и регистрируются как bean-компоненты.

2. Параметры @SpringBootApplication

Аннотация @SpringBootApplication предоставляет возможность настроить его поведение, передав дополнительные параметры:

String[] excludeName — исключить определенные имена классов автоконфигурации.

String[] scanBasePackages — базовые пакеты для сканирования аннотированных компонентов.

Class<?>[] scanBasePackageClasses — типо-безопасная альтернатива scanBasePackages для указания пакетов для сканирования на наличие аннотированных компонентов.

Class<?>[] exclude — исключить определенные классы автоконфигурации.