@EnableAutoConfiguration is an annotation given by Spring boot. This annotation is given to do default configuration in spring.

In this @EnableAutoConfiguration gives the default configuration to our spring application like creating JdbcTemplate bean.

The @**SpringBootApplication**does much more than what@EnableAutoConfiguration do.

The @**SpringBootApplication**actually a combination of three annotations:

1.@**Configuration**which is used in Java-based configuration on Spring framework

2. @**ComponentScan**to enable component scanning of components you write e.g. @Controller classes

3. @**EnableAutoConfgiuration**itself, which is used to enable auto-configuration in Spring Boot application.

**Purpose**

The clear purpose of @**EnableAutoConfiguration**is to enable automatic configuration feature of Spring Boot application which automatically configures things if certain classes are present in Classpath

It's not mandatory to put @**SpringBootApplication**to create a Spring Boot application, you can still use @**Configuration**and@**EnableAutoConfiguration**individually

@**EnableAutoConfiguration**annotation allows you to selectively exclude certain classes from auto-configuration

@Configuration

@EnableAutoConfiguration(exclude={DataSourceAutoConfiguration.class})

public class MyConfiguration {

  //.. Java code

}

The @**SpringBootApplication**is a combination of three annotations @**Configuration**(used for Java-based configuration), @**ComponentScan**(used for component scanning), and @**EnableAutoConfiguration**(used to enable auto-configuration in Spring Boot).