Spring AutoWire – Java

@Autowired

This annotation allows Spring to resolve and inject collaborating beans into our bean.

Enabling @Autowired Annotaions

* Spring framework enables automatic dependency injection.
* By declaring all the bean dependencies in a Spring Config file, spring container can autowire relationships between collaborating beans called Spring bean autowiring

the <context:annotation-config> annotation is mainly used to activate the dependency injection annotations in Spring XML files.

Spring Boot introduces the @SpringBootApplication annotation. This single annotation is equivalent to using @Configuration, @EnableAutoConfiguration, and @ComponentScan

Using Autowired

After enabling annotation injection,

We can use autowiring on properties, setters and constructors.

Examples:-

1. @Autowired on Properties

@Component("abcPlace")

public class AbcPlace {

public String format() {

return "abc";

}

}

@Component

public class AbcService {

@Autowired

private AbcPlace abcPlace;

}

1. @Autowired on Setters

public class AbcService {

private AbcPlace abcPlace;

@Autowired

public void AbcPlace (AbcPlace abcPlace) {

this. abcPlace = abcPlace;

}

}

1. @Autowired on Constructor

public class AbcService {

private AbcPlace abcPlace;

@Autowired

public AbcService (AbcService abcPlace){

this.abcPlace= abcPlace;

}

}

1. Autowiring By Name

public class AbcService {

@Autowired

private word abcPlace;

}

If more than one bean of the same type is available in the container, the framework will throw a fatal exception.