

Spring Support for Java 8 default methods

Starting from 4.2, Spring annotations can be processed when used on Java 8 default methods as well. This allows for a lot of flexibility in creating complex bean implementations and configuration classes (the class with @Configuration). We can even take the advantage of Java 8 default method's multiple inheritance behavior pattern.

Let's see how to use Spring annotations on Java 8 default methods.

Example

An interface for bean

```
public interface IMyBean {  
  
    @PostConstruct  
    default void init() {  
  
        System.out.println("post construct: "+this.getClass().getSimpleName());  
  
    }  
}
```

The implementation

```
public class MyBean implements IMyBean {  
  
    public void showMessage(String msg) {  
  
        System.out.println(msg);  
  
    }  
}
```

An interface for configuration

```
public interface IMyConfig {  
  
    @Bean  
  
    default MyBean myBean() {  
        return new MyBean();  
    }  
}
```

The implementation

```
@Configuration  
  
public class MyConfig implements IMyConfig {  
  
    public static void main(String[] args) {  
  
        ApplicationContext context =  
            new AnnotationConfigApplicationContext(MyConfig.class);  
  
        MyBean bean = context.getBean(MyBean.class);  
  
        bean.showMessage("a test message");  
  
    }  
}
```

Output

post construct: MyBean

a test message