

BELOW CODE IS TO BE ADDED IN SpringCoreDemo project created

Step 1: @COMPONENT ANNOTATION

Create below classes within package com.demo.service:

1. Create an interface IMessageProvider as follows:

```
public interface IMessageProvider {  
    public String getMessage();  
}
```

2. Create ScannerMessageProvider as follows:

```
public class ScannerMessageProvider implements IMessageProvider {  
    @Override  
    public String getMessage()  
    {  
        return "From Scanner";  
    }  
}
```

3. Create StringMessageProvider as follows:

```
public class StringMessageProvider implements IMessageProvider {  
  
    private String message;  
    public void setMessage(String message)  
    {  
        this.message = message;  
    }  
    @Override  
    public String getMessage()  
    {  
        return message;  
    }  
}
```

4. Create a class NotificationService as follows:

```
public class NotificationService {  
  
    IMessageProvider provider;  
  
    public NotificationService(IMessageProvider provider)  
    {  
        this.provider = provider;  
    }  
}
```

```

public void setMessage(IMessageProvider provider)
{
    this.provider = provider;
}

public void sendMessage()
{
    System.out.println(provider.getMessage());
}
}

```

5. Add @Component annotation as follows, that informs spring to create object of the class

```

@Component
public class StringMessageProvider implements IMessageProvider {
    ....
}

```

Step 2: @COMPONENTSCAN ANNOTATION

6. Complete the App class which has main method as follows for spring to set the context and scan for all the classes within the same or sub packages for spring specific annotations. If spring creates objects of the class they are called as "SPRING MANAGED BEAN"

```

@ComponentScan
public class App
{
    public static void main( String[] args )
    {

        ApplicationContext context = new AnnotationConfigApplicationContext(App.class);

        // NOT A SPRING MANAGED BEAN
        // StringMessageProvider messageProvider = new StringMessageProvider();
        System.out.println();
        for(String beanName: context.getBeanDefinitionNames())
            System.out.println(beanName);
    }
}

```

Step 3: Get bean context

1. To get the reference of beans created by spring, we can specify classname:

```
StringMessageProvider messageProvider =  
context.getBean(StringMessageProvider.class);
```

2. To get the reference of beans created by spring, we can specify id as well which is auto-generated by spring using camel casing syntax as follows:

NOTE: typecasting is required

```
StringMessageProvider messageProvider = (StringMessageProvider  
                                         )context.getBean("stringMessageProvider");
```

3. You can give your id as well as follows:

```
@Component("ob")  
public class SringMessageProvider implements IMessageProvider {  
    .....  
}
```

```
StringMessageProvider messageProvider = (StringMessageProvider  
                                         )context.getBean("ob");
```

4. Update App class as follows:

```
StringMessageProvider messageProvider = (StringMessageProvider)  
context.getBean("stringMessageProvider");  
messageProvider.setMessage("Hey!!");  
System.out.println(messageProvider.getMessage());
```