

152. Java based Config

```
AppConfig.java
1 package com.chinmay.configuration;
2
3 import org.springframework.context.annotation.Bean;
4 import org.springframework.context.annotation.Configuration;
5
6 import com.chinmay.javabasedconfig.Desktop;
7
8 @Configuration
9 public class AppConfig {
10
11     @Bean
12     public Desktop desktop()
13     {
14         return new Desktop();
15     }
16 }
17
18
```

Configuration File Will be Annotated with @Configuration Annotation

```
App.java
3 import org.springframework.context.ApplicationContext;
4 import org.springframework.context.annotation.AnnotationConfigApplicationContext;
5 import org.springframework.context.support.ClassPathXmlApplicationContext;
6
7 import com.chinmay.configuration.AppConfig;
8
9 public class App
10 {
11     public static void main( String[] args )
12     {
13         ApplicationContext context=new AnnotationConfigApplicationContext(AppConfig.class)
14
15         System.out.println("Inside Java Based Config");
16         Desktop desktop=context.getBean(Desktop.class);
17         desktop.compile();
18
19         // System.out.println("Inside Java Based Config");
20         // Desktop d=context.getBean("com",Desktop.class);
21         // Computer com=context.getBean(Computer.class);
22         // Desktop d1=context.getBean(Desktop.class);
23
24
```

Creating Bean in Configuration File: Will be Annotated with @Bean Annotation

```
@Bean
public Desktop desktop()
{
    return new Desktop();
}
```

We will use AnnotationConfigApplicationContext as the context rather than ClassPathXmlApplicationContext

Console

```
<terminated> App (7) [Java Application] C:\Users\THANMAY_PC\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_23.0.1.v20241024-1700\jre\bin\javaw.exe (5 Jan 2025, 12:20:29 pm - 12:20:30 pm) [pid: 2]
Desktop Object/bean Created
Inside Java Based Config
Desktop Code Compilation is in Progress
```

Ex: 3_1_Java_Based_Config

153 : Bean Name :

By Default if don't specify any name to the bean method name will become Bean name.

```
AppConfig.java
1 package com.chinmay.configuration;
2
3 import org.springframework.context.annotation.Bean;
4
5
6
7
8 @Configuration
9 public class AppConfig {
10
11     @Bean
12     public Desktop desktop1()
13     {
14         return new Desktop();
15     }
16 }
17
18
```

By Default if don't specify any name to the bean --> method name will become Bean name.

```
App.java
1 package com.chinmay.beanname;
2
3 import org.springframework.context.ApplicationContext;
4
5 public class App
6 {
7     public static void main( String[] args )
8     {
9         ApplicationContext context=new AnnotationConfigApplicationContext(AppCor
10
11         System.out.println("Inside Java Based Config");
12         Desktop desktop=context.getBean("desktop1",Desktop.class);
13         desktop.compile();
14
15     }
16 }
17
18
```

So we could see here we were able to get the Desktop type with desktop1 as it is the name associated with that bean.

Console

```
<terminated> App (8) [Java Application] C:\Users\THANMAY_PC\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_23.0.1.v20241024-1700\jre\bin\javaw.exe (5 Jan 2025, 1:11:17 pm - 1:11:18 pm) [pid: 8804]
Desktop Object/bean Created
Inside Java Based Config
Desktop Code Compilation is in Progress
```

We can also specify a name to bean in the below way

```
1 package com.chinmay.configuration;
2
3 import org.springframework.context.annotation.Bean;
4
5 @Configuration
6 public class AppConfig {
7
8     @Bean(name="desktop")
9     // @Bean(name= {"desktop","com1","Beast"})
10    public Desktop desktop1()
11    {
12        return new Desktop();
13    }
14 }
15
```

We have named the bean as desktop using the same name we were able to fetch the bean details from context.

```
1 package com.chinmay.beannname;
2
3 import org.springframework.context.ApplicationContext;
4
5 public class App
6 {
7
8     public static void main( String[] args )
9     {
10        ApplicationContext context=new AnnotationConfigApplicationContext(AppConfig.class);
11        System.out.println("Inside Java Based Config");
12        Desktop desktop=context.getBean("desktop",Desktop.class);
13        desktop.compile();
14    }
15 }
16
```

terminated> App (8) [Java Application] C:\Users\THANMAY_PC\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_23.0.1.v20241024-1700\jre\bin\javaw.exe (5 Jan 2025, 1:19:05 pm - 1:19:06 pm) [pid: 8764]
Desktop Object/bean Created
inside Java Based Config
Desktop Code Compilation is in Progress

```
1 package com.chinmay.configuration;
2
3 import org.springframework.context.annotation.Bean;
4
5 @Configuration
6 public class AppConfig {
7
8     // @Bean(name="desktop")
9     @Bean(name= {"desktop","com1","Beast"})
10    public Desktop desktop1()
11    {
12        return new Desktop();
13    }
14 }
15
```

```
1 package com.chinmay.beannname;
2
3 import org.springframework.context.ApplicationContext;
4
5 public class App
6 {
7
8     public static void main( String[] args )
9     {
10        ApplicationContext context=new AnnotationConfigApplicationContext(AppConfig.class);
11        System.out.println("Inside Java Based Config");
12        Desktop desktop=context.getBean("desktop",Desktop.class);
13        Desktop desktop1=context.getBean("com1",Desktop.class);
14        Desktop desktop2=context.getBean("Beast",Desktop.class);
15
16        System.out.println(desktop.hashCode());
17        System.out.println(desktop1.hashCode());
18        System.out.println(desktop2.hashCode());
19    }
20 }
21
```

We have named our bean multiple names we can access them using any of the names. ----> desktop,com1,Beast

<terminated> App (8) [Java Application] C:\Users\THANMAY_PC\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_23.0.1.v20241024-1700\jre\bin\javaw.exe (5 Jan 2025, 1:23:49 pm - 1:23:51 pm) [pid: 2124]
Desktop Object/bean Created
Inside Java Based Config
30578394
30578394
30578394

154: Scope Annotation:

By Default Singleton will be default scope .When we use Single ton scope same bean will shared every time we access.

Where as For prototype scope a new bean instance will be returned every time we try to access the bean from context.

```
1 package com.chinmay.beannname;
2
3 import org.springframework.context.ApplicationContext;
4
5 public class App
6 {
7     public static void main( String[] args )
8     {
9         ApplicationContext context=new AnnotationConfigApplicationContext(AppCon
10
11         System.out.println("Inside Java Based Config");
12         Desktop desktop=context.getBean("desktop",Desktop.class);
13         Desktop desktop1=context.getBean("com1",Desktop.class);
14         Desktop desktop2=context.getBean("Beast",Desktop.class);
15
16         System.out.println(desktop.hashCode());
17         System.out.println(desktop1.hashCode());
18         System.out.println(desktop2.hashCode());
19
20     }
21 }
22
23
24
25
26
27
```

```
1 package com.chinmay.configuration;
2
3 import org.springframework.context.annotation.Bean;
4
5 @Configuration
6 public class AppConfig {
7
8     // @Bean(name="desktop")
9     @Bean(name= {"desktop","com1","Beast"})
10     @Scope("prototype")
11     public Desktop desktop1()
12     {
13         return new Desktop();
14     }
15 }
16
17
18
19
20
21
```

Console ×

<terminated> App (9) [Java Application] C:\Users\THANMAY_PC\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_23.0.1.v20241024-1700\jre\bin\javaw.exe (5 Jan 2025, 2:44:26 pm – 2:44:27 pm) [pid: 2196]

Inside Java Based Config
Desktop Object/bean Created
Desktop Object/bean Created
Desktop Object/bean Created
1018953501
1423561005
943870983

155:Auto Wire:

Creating the wiring between the beans.

```
17 this.com = com;
18
19
20 public int getAge() {
21     return age;
22 }
23
24 public void setAge(int age) {
25     //System.out.println("Inside the Setter Method");
26     this.age = age;
27 }
28
29 public Alien() {
30     System.out.println("Alien Object Created");
31 }
32
33 public void code()
34 {
35     System.out.println("Coding");
36     com.compile();
37 }
38 }
39
```

```
1 package com.chinmay.configuration;
2
3 import org.springframework.context.annotation.Bean;
4
5 @Configuration
6 public class AppConfig {
7
8     @Bean
9     public Alien alien()
10     {
11         return new Alien();
12     }
13
14     @Bean
15     public Desktop desktop()
16     {
17         return new Desktop();
18     }
19 }
20
21
22
23
24
```

Here we are getting Exception as the Computer is not linked with any implementation

Console ×

<terminated> App (10) [Java Application] C:\Users\THANMAY_PC\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_23.0.1.v20241024-1700\jre\bin\javaw.exe (5 Jan 2025, 3:00:09 pm – 3:00:10 pm)

Alien Object Created
Desktop Object/bean Created
Inside Auto Wired Exapmle
Coding
Exception in thread "main" java.lang.NullPointerException: Cannot invoke "com.chinmay.javabasedconfig.Computer.compile()" because "this.com" is null
at com.chinmay.javabasedconfig.Alien.code(Alien.java:36)
at com.chinmay.javabasedconfig.App.main(App.java:18)

The screenshot shows two Java files in the Eclipse IDE. On the left is `Desktop.java` with methods `getAge()`, `setAge(int age)`, `Alien()`, and `code()`. On the right is `AppConfig.java` which uses Spring annotations. The `@Bean` method `alien()` in `AppConfig` creates an `Alien` object and calls `al.setCom(desktop())`, where `desktop()` is another `@Bean` method that returns a new `Desktop` object. A red box with an arrow pointing to the `al.setCom(desktop())` line contains the text: "Here we could see as we have Desktop bean present we have called the Desktop bean and the same has been injected into Computer implementation. Due to which there is tight Coupling." The console at the bottom shows the output of the application, including "Alien Object Created", "Desktop Object/bean Created", and "Coding".

Injecting the Desktop into Alien Bean As Desktop is the implementation of Computer.

The screenshot shows the same Eclipse IDE environment but with different code. In `Desktop.java`, the `getCom()` method now returns `this.com` instead of creating a new object. In `AppConfig.java`, the `alien()` method is annotated with `@Bean` and takes a `Computer com` parameter. The `al.setCom(com)` line is highlighted, with a callout box stating: "Here it will check if there is any bean of Computer present and injects into Alien class." Another callout box points to the `al.setCom(desktop())` line in the commented-out code, stating: "We have loose coupling scenario here it will inject if there is any bean matching of Computer type present rather than Depending on Desktop bean as seen in the previous case." The console output is the same as the previous screenshot.

We don't have tight Coupling here as for injection if it finds any implementation of Computer class it will inject .

156.Primary and Qualifier

```

1 package com.chinmay.configuration;
2
3 import org.springframework.context.annotation.Bean;
4
5
6
7
8
9
10
11 @Configuration
12 public class AppConfig {
13
14     @Bean
15     public Alien alien(Computer com)
16     {
17         Alien al=new Alien();
18         al.setAge(25);
19         al.setCom(com);
20         return al;
21     }
22
23     @Bean
24     public Desktop desktop()
25     {
26         return new Desktop();
27     }
28
29     @Bean
30     public Laptop laptop()
31     {
32         return new Laptop();
33     }
34
35 }
36
37 Matching Beans

```

In all the previous cases we had only one implementation for Computer due to which it has not resulted in any Exception. but now we have 2 matching beans for Computer. Due to which it will cause Exception

Jan 05, 2025 3:26:46 PM org.springframework.context.support.AbstractApplicationContext refresh
 WARNING: Exception encountered during context initialization - cancelling refresh attempt:
 org.springframework.beans.factory.UnsatisfiedDependencyException: Error creating bean with name 'alien' defined in com.chinmay.configuration.AppConfig: Unsatisfied dependency expressed through method 'alien' parameter 0: No qualifying bean of type 'com.chinmay.javabasedconfig.Computer' available: expected single matching bean but found 2: desktop,laptop
 Exception in thread "main" org.springframework.beans.factory.UnsatisfiedDependencyException:
 Error creating bean with name 'alien' defined in com.chinmay.configuration.AppConfig:
 Unsatisfied dependency expressed through method 'alien' parameter 0: No qualifying bean of type 'com.chinmay.javabasedconfig.Computer' available: expected single matching bean but found 2: desktop,laptop
 at
 org.springframework.beans.factory.support.ConstructorResolver.createArgumentArray(ConstructorReso
 lver.java:884)

```

1 package com.chinmay.configuration;
2
3 import org.springframework.beans.factory.annotation.Qualifier;
4
5
6
7
8
9
10
11 @Configuration
12 public class AppConfig {
13
14     @Bean
15     public Alien alien(@Qualifier("desktop") Computer com)
16     {
17         Alien al=new Alien();
18         al.setAge(25);
19         al.setCom(com);
20         return al;
21     }
22
23     @Bean
24     public Desktop desktop()
25     {
26         return new Desktop();
27     }
28
29
30     @Bean
31     public Laptop laptop()
32     {
33         return new Laptop();
34     }
35
36 }

```

Here we are specifying to consider bean with name desktop as the bean for injecting. So it will check if there exists a bean with name desktop and injects.

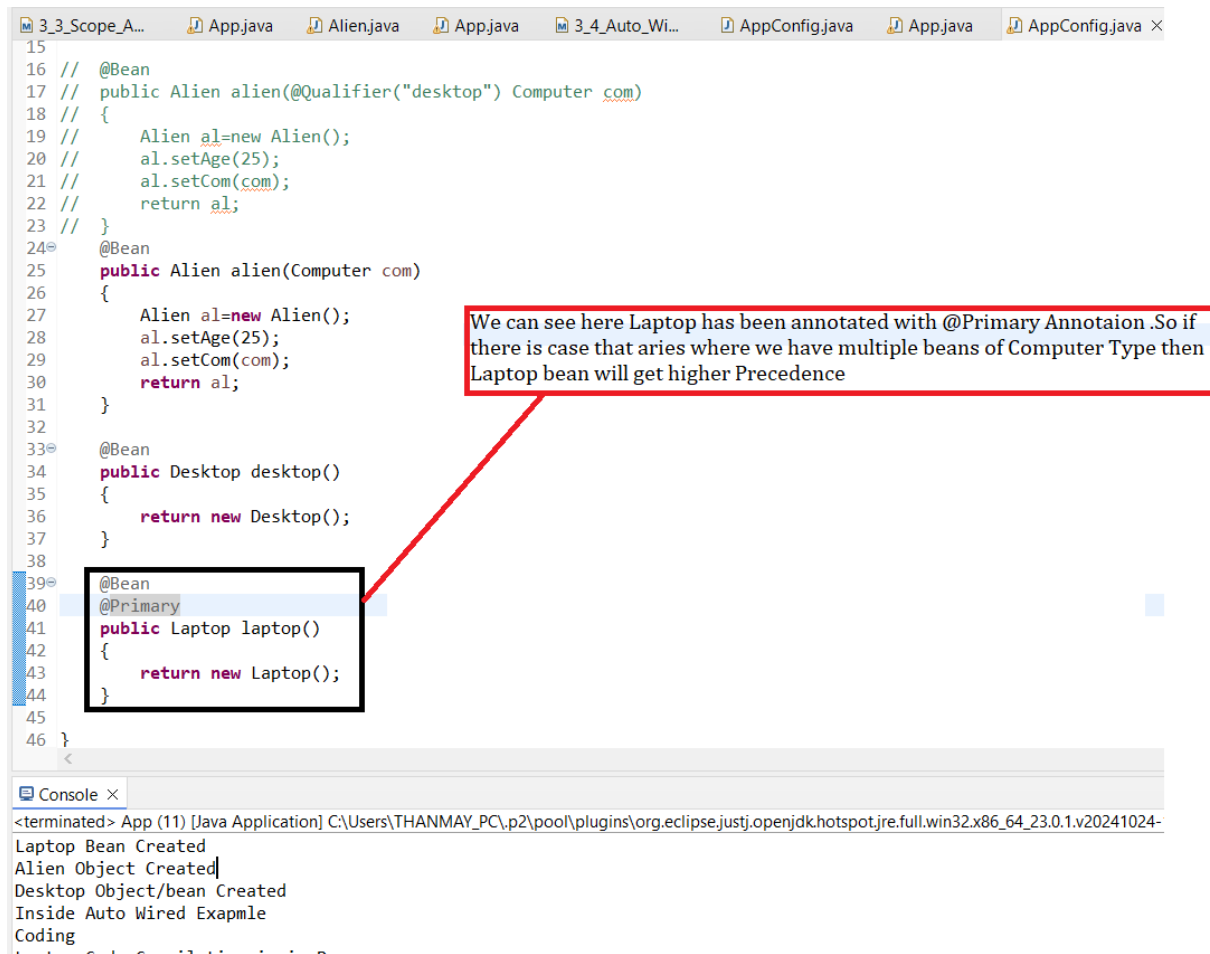
Console ×

```

<terminated> App (11) [Java Application] C:\Users\THANMAY_PC\p2\poo\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_23.0.1.v20241024-1700\jre\bin\javaw.exe (5 Jan 2025, 3:35:13 pm - 3:35:15 pm) [pid:
Desktop Object/bean Created
Alien Object Created
Laptop Bean Created
Inside Auto Wired Exapmle
Coding
Desktop Code Compilation is in Progress

```

Approach 1: We manually specify the bean name using Qualifier



```
15
16 // @Bean
17 // public Alien alien(@Qualifier("desktop") Computer com)
18 // {
19 //     Alien al=new Alien();
20 //     al.setAge(25);
21 //     al.setCom(com);
22 //     return al;
23 // }
24 @Bean
25 public Alien alien(Computer com)
26 {
27     Alien al=new Alien();
28     al.setAge(25);
29     al.setCom(com);
30     return al;
31 }
32
33 @Bean
34 public Desktop desktop()
35 {
36     return new Desktop();
37 }
38
39 @Bean
40 @Primary
41 public Laptop laptop()
42 {
43     return new Laptop();
44 }
45
46 }
```

We can see here Laptop has been annotated with @Primary Annotation .So if there is case that arises where we have multiple beans of Computer Type then Laptop bean will get higher Precedence

Laptop Bean Created
Alien Object Created
Desktop Object/bean Created
Inside Auto Wired Example
Coding

Approach 2: Using primary Annotation

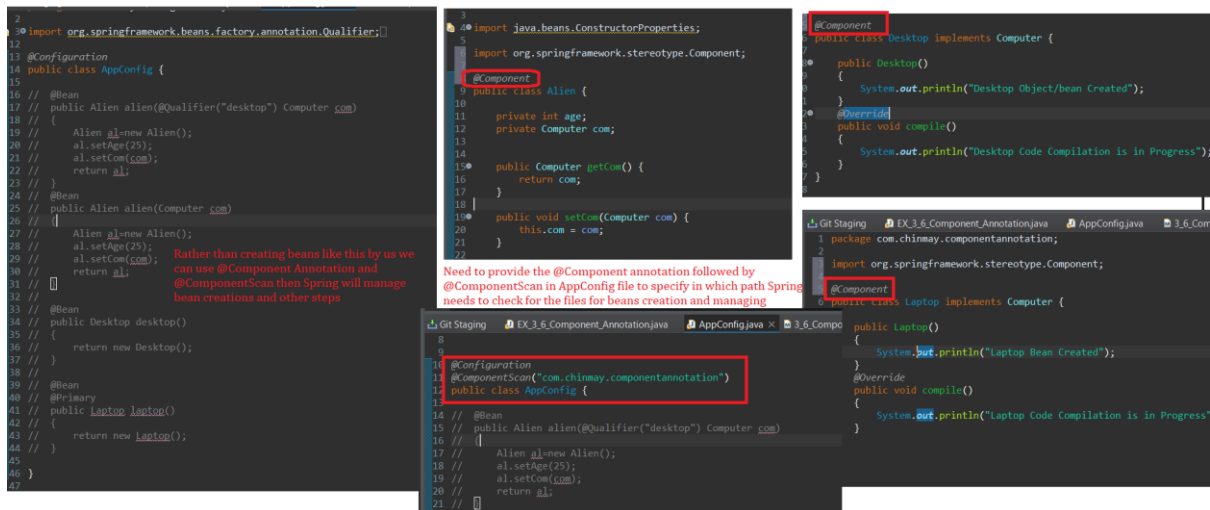
157:Component Stereo Type Annotation:

@Component:

Telling the Spring that this class has to be managed by the Spring Framework.

@ComponentScan("com.example.beans") ---

This will scan all the files in the com.example.beans and creates and manages the classes which are annotated with @Component Annotation.



```

Console ×
<terminated> EX_3_6_Component_Annotation [Java Application] C:\Users\THANMAY_PC\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_23.0.1.v20241024-1700\jre\bin\javaw.exe (15 Jan 2024)
Alien Object Created
Desktop Object/bean Created
Laptop Bean Created
Inside Component Annotation Exapmle
Coding
Exception in thread "main" java.lang.NullPointerException: Cannot invoke "com.chinmay.componentannotation.Computer.compile()" because "this.com" is null
    at com.chinmay.componentannotation.Alien.code(Alien.java:39)
    at com.chinmay.componentannotation.EX_3_6_Component_Annotation.main(EX_3_6_Component_Annotation.java:18)

```

Even though the beans has been in the context as there is no linking b/w them it resulted in NullPointerException.

```

Console ×
<terminated> EX_3_6_Component_Annotation [Java Application] C:\Users\THANMAY_PC\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_23.0.1.v20241024-1700\jre\bin\javaw.exe (15 Jan 2024)
Alien Object Created
Desktop Object/bean Created
Laptop Bean Created
Inside Component Annotation Exapmle
Coding
Exception in thread "main" java.lang.NullPointerException: Cannot invoke "com.chinmay.componentannotation.Computer.compile()" because "this.com" is null
    at com.chinmay.componentannotation.Alien.code(Alien.java:39)
    at com.chinmay.componentannotation.EX_3_6_Component_Annotation.main(EX_3_6_Component_Annotation.java:18)

```

158: Autowired Field, Constructor ,Setter

159:Primary Annotation

```
1 package com.chinmay.primaryannotation;
2
3 import java.beans.ConstructorProperties;
4
5 @Component
6 public class Alien {
7     private int age;
8     @Autowired
9     private Computer com;
10
11     public Computer getCom() {
12         return com;
13     }
14
15     // @Autowired // Setter Injection
16     // @Qualifier("com1")
17     public void setCom(Computer com) {
18         this.com = com;
19     }
20
21     public int getAge() {
22         return age;
23     }
24
25     public void setAge(int age) {
26
27     }
28 }
```

```
1 package com.chinmay.primaryannotation;
2
3 import org.springframework.context.annotation.*;
4
5 @Component
6 @Primary
7 public class Desktop implements Computer {
8
9     public Desktop() {
10         System.out.println("Desktop Object Created");
11     }
12
13     @Override
14     public void compile() {
15         System.out.println("Desktop Code Compilation is in Progress");
16     }
17 }
```

```
1 package com.chinmay.primaryannotation;
2
3 import org.springframework.stereotype.Component;
4
5 @Component("com1")
6 public class Laptop implements Computer {
7
8     public Laptop() {
9         System.out.println("Laptop Bean Created");
10     }
11
12     @Override
13     public void compile() {
14         System.out.println("Laptop Code Compilation is in Progress");
15     }
16 }
```

Here we could see that even though 2 matching beans for Computer as we have kept @Primary Annotation on the Desktop class. If there is any ambiguity in such cases Desktop class will be considered.

Console Output:
Alien Object Created
Desktop Object/bean Created
Laptop Bean Created
Inside Primary Annotation Example
Coding
Desktop Code Compilation is in Progress

Ex: @Primary Annotation

```
1 package com.chinmay.primaryannotation;
2
3 import java.beans.ConstructorProperties;
4
5 @Component
6 public class Alien {
7     private int age;
8     @Autowired
9     @Qualifier("com1")
10     private Computer com;
11
12     public Computer getCom() {
13         return com;
14     }
15
16     // @Autowired // Setter Injection
17     // @Qualifier("com1")
18     public void setCom(Computer com) {
19         this.com = com;
20     }
21
22     public int getAge() {
23         return age;
24     }
25
26     public void setAge(int age) {
27
28     }
29 }
```

```
1 package com.chinmay.primaryannotation;
2
3 import org.springframework.context.annotation.*;
4
5 @Component
6 @Primary
7 public class Desktop implements Computer {
8
9     public Desktop() {
10         System.out.println("Desktop Object Created");
11     }
12
13     @Override
14     public void compile() {
15         System.out.println("Desktop Code Compilation is in Progress");
16     }
17 }
```

```
1 package com.chinmay.primaryannotation;
2
3 import org.springframework.stereotype.Component;
4
5 @Component("com1")
6 public class Laptop implements Computer {
7
8     public Laptop() {
9         System.out.println("Laptop Bean Created");
10     }
11
12     @Override
13     public void compile() {
14         System.out.println("Laptop Code Compilation is in Progress");
15     }
16 }
```

Even though we have @Primary Annotation on top of Desktop but the @Qualifier Annotation has been provided while injecting so that will be given higher precedence due to which we could see here Laptop class implementation is taken into consideration.

Console Output:
Alien Object Created
Laptop Bean Created
Desktop Object/bean Created
Inside Primary Annotation Example
Coding
Laptop Code Compilation is in Progress

Ex: @Primary Annotation with @Qualifier as well.

160:Scope and Value Annotation

```
1 package com.chinmay.scopevalueannotation;
2
3 import org.springframework.context.annotation.Scope;
4 import org.springframework.stereotype.Component;
5
6 @Component("com1")
7 @Scope("prototype")
8 public class Laptop implements Computer {
9
10     public Laptop()
11     {
12         System.out.println("Laptop Bean Created");
13     }
14     @Override
15     public void compile()
16     {
17         System.out.println("Laptop Code Compilation is in Progress");
18     }
19 }
20`
```

As we have setted the scope for Laptop as prototype Bean will not be created until there is an usage due to which we could see here no bean is created even if it has @Component.

Git Staging ×

Console × Computer.java AppConfig.java AppConfig.java EX_3_9_Scope_Va

<terminated> EX_3_9_Scope_Value_Annotation [Java Application] C:\Users\THANMAY_PC\p2\pool\plu

Alien Object Created
Desktop Object/bean Created
Inside Scope and Value Annotation Example
Coding
Desktop Code Compilation is in Progress

```
1 package com.chinmay.scopevalueannotation;
2
3 import org.springframework.context.annotation.Scope;
4 import org.springframework.stereotype.Component;
5
6 @Component("com1")
7 @Scope("prototype")
8 public class Laptop implements Computer {
9
10     public Laptop()
11     {
12         System.out.println("Laptop Bean Created");
13     }
14     @Override
15     public void compile()
16     {
17         System.out.println("Laptop Code Compilation is in Progress");
18     }
19 }
20
```

Git Staging X

Console X Computer.java AppConfig.java AppConfig.java EX_3_9_Scope_Va

<terminated> EX_3_9_Scope_Value_Annotation [Java Application] C:\Users\THANMAY_PC\p2\pool\plu

Alien Object Created
Desktop Object/bean Created
Inside Scope and Value Annotation Example
Coding
Desktop Code Compilation is in Progress

```
11 @Component
12 public class Alien {
13     @Value("21")
14     private int age;
15     @Autowired
16     //@Qualifier("com1")
17     private Computer com;
18
19     public Computer getCom() {
20         return com;
21     }
22 }
23
```

```
<terminated> EX_3_9_Scope_Value_Annotation [Java Applic
Alien Object Created
Desktop Object/bean Created
Inside Scope and Value Annotation Example
Coding
Desktop Code Compilation is in Progress
Age : 21
```

@Scope and @Value Annotation Examples.