```
We can calve the ambiguity problems by using one of the three solutions.
                   gi salag @Primary (ancosator) diven digit or primary "mas" artifluts of visions discontidition digit
(i) rolog (Challies) acconting (and

    By realisting target spring lover 1965 & property name with time of the possible dependent spring lover til/curve

of thing @introny concention
                   validations are level and the second contractions.

    Hilliget vom pleam dam dij is firsting att tigde over the dependents, we need to place differency on
top of over dependent typing fear in the efficiency powers are been on effect a mediend level.

                      m K var place ij Primary samointism on the top of multiple possible dependent, spring isom de finite describes we again get
                                km kiguine problem
                      in imagen. Spring hoses
                                                                                                                                                                                                                 In Configuration days
                                                                                                                                                                                                 (b) particulation (A girl components).
(Note: bose fibrate basefuldinger = "(best dube ene") patric class Applicating).
       ghtersperson("web")
     give represent (with )
public; and Vendorst index of
COntamination (Vendorst index)
price of handfolio state;
giver and investigation
prices are benefit in a street;
                                                                                                                                                                                                                                         System out prints ("Applicable: Operate constructor");
                                                                                                                                    British C Nimery
the British C Name
method below
young beam of the p
                □ kanadasik
                                                                                                                                                                                                                                           Open delimed show as the spring boson.
                                                                                                                                                                                                                                        @ Sees (seemed liberal)
glibiosocy
                                                                                                                                                                                                                                        public local Salar constatitutari (
                                                                                                                                                                                                                                                      By the manufactor (App Configuration Enter) (i) 
return (Local Community); (flags state)
                                                                                                                                                                                                                                       () from Junium "Maked")
patrille Land Safer (maked Safed)) }
thysic manus priorite; (Applicating account Safer Is)*();
manus (Lond Safer Is)*(Applicating account Safer Is)*();
manus (Lond Safer Is)*(Applicating account Safer Is)*();
                                                                                                                                                                                                                                        gi limes (names "littere")
public Local Time (names Littere) (
                                                                                                                                                                                                                                                        System.cut.prartie (Appointingsperset) med (*).
Indian Docs Times pref (*).
                                                                                                                                                                                                 1
                C) Long (VO. afterly) amount on
                       --> this sentencing applicable of classified, this level, mentod level, pears level and section to the state of the sta
                                   Minutes Spring bean class
                                   (Component (Codf)
public show Ministry Senter (
spanished (Claid injuries)
singual fina? (date (C))
                                                                                                                                                                                                                           Configurations law
                                                                                                                                                                                                                              (Configuration A) externation matter.
(Company Charl Jaco Padages = "compilates em.")
                                                     middle stars SourConfle (
                                                                                                                                                                                                                                                public representation.
                                                          All Committees in the process of the committee of the process of the committee of the commi
                                                                                                                                                                                                                                                                     System cetternitri ("Apetonia), departur consocioni b
                                                                                                                                                                                                                                                                      Open defined class as the spring boars
                                                                                                                                                                                                                                                                     gi pean (seemas-"block")
                                                                                                                                                                                                                                                                     paties (compare simulational) :

Pytic manufaction (Applicating Secretari Enter) (ig.
com an accord arm accord); //aya class
                                                                                                                                                                                                                                                                   Client junt on "Maled")

Antille land fate around Scientify |

dynamics applicate; high antily count item in "it
cause accordance (high antily count item in "it
cause (high antily count it
c
                                                                                                                                                                                                                                                                     dillow (march 1869)
                                                                                                                                                                                                                                                                              where percent the content (see (f)).
Subtract the content (see (f)).
Subtract the MacConfigure (set) see (f)).
reform Docs/Times (set)).
                                c) By matching the target spring has a later HASA property name with lane of the possible dependent oping beautiful as shown
                                                             42 Services recent of using \langle \xi| C_{\rm P} dR(n) \rangle I and \langle \xi| C_{\rm P} decay are stable as
                                                             at the just ment in realist target spring branch 1985 A property some with Grand the Department
                                                                             Spring beam for
                                          Target spring been.
                                                                                                                                                                                                                                                                                       Арарсанабідірізма
                                   #Companion(Twiff)
                                 public dan Wedtfoyfinder (
                                                                                                                                                                                                                                                        #Nonfiguration // #Nonepowert→
#Nonepoment/car(base/adaps = "consecutiones")
                                                   @Material (/Pietr Injection)
                                                      private total bala dalar.
                                                                                                                                                                                                                                                          public class Associary) of
                                                      distance wired
                                                       prieme Local Time times
```

Hough there are two reflectings the second of them rectand broad condition along the brightness

... burnerthod

Section and printing "App Configuration or a construction" is

Water-defined class so the spring bean

We can solve the ambiguity problems by using one of the three solutions

- c) By matching target spring bean HAS-A property name with one of the possible dependent spring bean id/name
- a) Using @Primary annotation
- => It is class level, method level annotation
- => if target Spring bean class obj is having multiple possible dependents we need to place @Primary on top of one dependent Spring bean cfg either @Component class level or @Bean method level
- we again get Ambiguity problem

```
=> if we place @Primary annotation on the top of multiple possible dependent spring bean definitations then
In target Spring bean
@Component("wdf")
public class WeekDayFinder {
@Autowired //Field Injection private LocalDate date; @Autowired
private LocalTime time;
b.methods
In Configuration class
_____
@Configuration // @Component++
@ComponentScan (base Packages = "com.nt.sbeans")
public class AppConfig {
Beco of @Primary
the first @Bean method based
spring bean class obj
is injected
public AppConfig() {
}
System.out.println("AppConfig:: O-param constructor");
//pre-defined class as the spring bean
@Bean(name="Idate")
@Primary
public LocalDate createLDate() {
```

```
System.out.println("AppConfig.createLDate()");
return LocalDate.now(); //sys date
}
@Bean(name="|date1")
public LocalDate createLDate1() {
System.out.println("AppConfig.createLDate1()");
return LocalDate.of(2020,10,20);//custom date
}
@Bean(name="Itime")
public LocalTime createLTime() {
System.out.println("AppConfig.createLTime()");
return LocalTime.now();
}
b) Using @Qualifier(-) annotation
_____
=> This annotation is applicable at class level, filed level, method level, param level and etc...
=> This best solution to solve the ambiguity problem (NoUniqueBeanDefinitationException)
=> It has be placed in target spring bean class on the top HAS-A property (Field) or setter method or arbitrary
method or parameters of the parameterized constructor to specifying the our choice Dependent spring bean
//Target Spring bean class
_____
@Component("wdf")
public class WeekDayFinder { @Autowired //Field Injection @Qualifier("Idate1") private LocalDate date;
@Autowired
}
private LocalTime time;
....// B.methods
Since "late" is specified
Configuration class
==========
@Configuration // @Component++
@ComponentScan (base Packages = "com.nt.sbeans") public class AppConfig {
in @Qualifier(-) annotation we can say Second @Bean method that is returning LocalDate class obj will be
Injected
```

```
public AppConfig() {
}
System.out.println("AppConfig:: O-param constructor");
//pre-defined class as the spring bean
@Bean(name="Idate")
public LocalDate createLDate() {
System.out.println("AppConfig.createLDate()");
return LocalDate.now(); //sys date
@Bean(name="Idate1")
public LocalDate createLDate1() {
System.out.println("AppConfig.createLDate1()");
return LocalDate.of(2020,10,20);//custom date
}
@Bean(name="Itime")
public LocalTime createLTime() {
System.out.println("AppConfig.createLTime()");
return LocalTime.now();
}
c) By matching the target spring bean class HAS-A property name with one of the possible dependent spring
bean id as shown below
=> Here no need of using @Qualifer(-) and @Primary annotations
=> we just need to match target spring bean's HAS-A property name with One of the Dependent Spring bean
Target spring bean
@Component("wdf")
public class WeekDayFinder {
@Autowired //Field Injection
private LocalDate date;
@Autowired
private LocalTime time;
Though there are
AppConfig.java
======
@Configuration // @Component++
@ComponentScan (base Packages = "com.nt.sbeans") public class AppConfig {
```

```
public AppConfig() {
System.out.println("AppConfig:: O-param constructor");
}
b.method
two marchings
the second @Bean method based LocalDate
class obj will be injected
becoz the HAS-A propert name (date) and the
dependent bean id name(date) are matching
//pre-defined class as the spring bean @Bean(name="Idate")
public LocalDate createLDate() {
System.out.println("AppConfig.createLDate()");
return LocalDate.now(); //sys date
@Bean(name="date")
public LocalDate createLDate1() {
System.out.println("AppConfig.createLDate1()");
return LocalDate.of(2020,10,20);//custom date
}
@Bean(name="Itime")
public LocalTime createLTime() {
System.out.println("AppConfig.createLTime()");
return LocalTime.now();
=>@Qualifier(-) (2nd solution) and matching HAS-A property name with Dependent spring bean id (3rd
solution) are performing ByName mode of Autowiring (becoz the dependent spring bean is identified based
on its bean id/name)
=>@Primary (1st solution) is performing ByType mode of Autowiring (becoz the dependent spring bean is
identified based on its class name (type))
Q) if we apply all the 3 solutions at a time on single HAS-A property of Target spring bean class having 3
different spring beans of same type then can u tell me which solution will taken as the final solution?
Ans) @Qualifier(-) specified Dependent spring bean will be injected as the final Spring bean class obj to
target spring bean class object (becoz @Qualifier(-) based dependent spring bean class obj will be
//Target spring bean class
@Component("wdf")
public class WeekDayFinder {
```

```
injected at the end)
@Autowired //Field Injection @Qualifier("Idate2") private LocalDate date; @Autowired
private LocalTime time;
b.methods
Finally @Qualifier(-) based spring bean class obj will be injected
@Configuration // @Component++
@ComponentScan (base Packages = "com.nt.sbeans") public class AppConfig {
public AppConfig() {
System.out.println("AppConfig:: O-param constructor");
//pre-defined class as the spring bean @Bean(name="Idate")
@Primary (using Solution1)
public LocalDate createLDate() {
System.out.println("AppConfig.createLDate()"); return LocalDate.now(); //sys date
@Bean(name="Idate2") (using Solution2) public LocalDate createLDate2() {
System.out.println("AppConfig.createLDate2()"); return LocalDate.of(2000,10,20); //sys date
@Bean(name="date") (Using Solution3)
public LocalDate createLDate1() {
System.out.println("AppConfig.createLDate1()"); return LocalDate.of(2020,10,20);//custom date
@Bean(name="Itime")
public LocalTime createLTime() {
System.out.println("AppConfig.createTime()");
return LocalTime.now();
}
if we apply all the four injections (field Injection, setter Injection, constructor Injection and arbitrary method
Injection on single Spring bean property of target spring bean class having four different spring bean objs of
same type can u tell me which injection will be taken as the final injection?
Ans) if setter method of setter Injection is placed after arbitrary method of arbitrary method injection then
setter injection value/object will be taken as the final value /object
if arbitrary method of arbitrary Injection is placed after setter method of setter injection then
arbitrary method injection value/object will be taken as the final value /object
//WeekDayFinder.java (Target spring bean class)
package com.nt.sbeans;
import java.time.LocalDate;
import java.time.LocalTime;
import org.springframework.beans.factory.annotation.Autowired; import
```

```
org.springframework.beans.factory.annotation.Qualifier; import org.springframework.stereotype.Component;
@Component("wdf")
public class WeekDayFinder { @Autowired //Field Injection
@Qualifier("Idate")
Field Injection
private LocalDate date;
@Autowired
private LocalTime time;
@Autowired
public WeekDayFinder(@Qualifier("Idate3") LocalDate date, LocalTime time) {
this.date=date;
this.time=time;
System.out.println("WeekDayFinder:: 2-param cosntructor");
@Autowired
@Qualifier("Idate1")
public void setDate(LocalDate date) {
System.out.println("WeekDayFinder.setDate()");
this.date=date;
@Autowired
@Qualifier("Idate2")
public void putDate(LocalDate date) {
System.out.println("WeekDayFinder.putDate()");
this.date=date;
@Autowired
public void setTime(LocalTime time) {
System.out.println("WeekDayFinder.setTime()");
this.time-time;
@Autowired
public void assignTime(LocalTime time) {
System.out.println("WeekDayFinder.assignTime()");
this.time-time;
Sette Injection
```

```
Constructor Injection
Arbitrary method Injection
// b.method
public String showMessage(String user) {
System.out.println("WeekDayFinder.showMessage()::" + date + "...." + time);
// get current week day number
int number = date.getDayOfWeek().getValue();
// generate the message
if (number >= 1 && number <= 5)
return " Work Hard to build Stroing IT Career:" + user;
return "Take a Break and Enjoy ur week end:" + user;
//Configuration class
@Configuration // @Component++
@ComponentScan (base Packages = "com.nt.sbeans")
public class AppConfig {
public AppConfig() {
System.out.println("AppConfig:: 0-param constructor");
//pre-defined class as the spring bean
@Bean(name="Idate")
public LocalDate createLDate() {
System.out.println("AppConfig.createLDate()");
return LocalDate.now(); //sys date
@Bean(name="Idate2")
public LocalDate createLDate2() {
System.out.println("AppConfig.createLDate2()");
return LocalDate.of(2000,10,20); //sys date
@Bean(name="Idate1")
public LocalDate createLDate1() {
System.out.println("AppConfig.createLDate1()");
return LocalDate.of(2020,10,20);//custom date
```

```
@Bean(name="Idate3")
public LocalDate createLDate3() {
System.out.println("AppConfig.createLDate3()");
return LocalDate.of(1990,10,20);//custom date
@Bean(name="Itime")
public LocalTime createLTime() {
System.out.println("AppConfig.createLTime()");
return LocalTime.now();
}
if all the 4 injections are applied at a time on to the single property of
target spring bean class then injections takes in the following order
a) constructor Injection
b) Field Injection
c) Setter Injection
d) Arbitrary method Injection
if arbitrary method is
placed followed by setter method
(or)
a) constructor Injection
b) Field Injection
c) Arbitrary method Injection
if setter method is placed
d) Setter Injection
followed by arbitrary method
Q) Why @Qualifier(-) is best solution to solve the ambiguity Problem?
Ans) The reasons are
a) The bean id required in @Qualifier(-) can be gathered from properties file or xml file to make
Code loosely coupled code
b) if we apply multiple solutions (@Primary, @Qualifier(-), name matching) on the the single HAS -a property
to solve the ambiguity problem .. the @Qualifier(-) solution will be taken
as the final solution
```

}