```
> IDC container uses, parameterized constructor to create target spring, bean class objaind to assign \ell
                                   inject, dependent spring bean class object values, to target spring bean class obj.
                   Or Porthis injection, we need to place reconstructor againgto for "of times to use "of parameters tracted."
                                              for constructor injection, while working with smill driven eigh-
                      to For this injection, we need to place givintoevired on the top of I parameterized constructor in precipition
                    with this injection. The spring bean class object always be created using parameterized constructor.
                    no. This is called Notest injection becausi the dependent values/disjects will be injected to larget spring
                          been deas objidening the object creation process.
                                                                                                                                                                (spring bean class etc.)

    In setter injection, the order of creating spring been class objects.

                                                                        a) target spring class class obj. b) bependent spring bean class obj.
                                                                   [ ICC contains Playart set (kss), i having dependent obj. so the expiration]
                         to in constructor injection , the order of creating spring bear class objuste-
                                                                        all Dependent spring beam date obj. (i) Target spring beam date obj.
                                                                    [10C container uses parameterized constructor having dependent obj(s) as the arg
                                                                             values. To conside the target spring bean above obj.]
           Burnale weet a sing constructor injection
           (Childring aged amendon jures [larged spring books store)
partings consum account;
             import (availment ocal Date)
           publis de la Michide supplient race (
private social line times () studios property
private socialiste elette (//b/de/yproperty
                                                                                                                                                                                                                                                                                                                                                                                                # IA/The JO Aspendency Injection assessmental particle and the s
                                                                                                                                                                                                                                                                                                                                                                                                          4. White John Library Barris and
                                     A parameterization structure for constructor rejection public to street supplementation confidence translations and street confidence described.
                                                                                                                                                                                                                                                                                                                                                                                                             note and object
                                                                                                                                                                                                                                                                                                                                                                                                                                 All many representations in the
                                                                                                                                                                                                                                                                                                                                                                                                          * I Consider the State of the S
                                                           Bildinering
Bildinering
Byromion signification against a section of spaces
                                                                                                                                                                                                                                                                                                                                                                                                                                    CMMAN april maki (asa
                                                                                                                                                                                                                                                                                                                                                                                                   in the Person and Common
                                   //L. method
poleli. Bidny obser#AhitA sought bidny ment (
Spaint and polelic [Withbidition of Commun.
// type current boar of the day.
                                                                                                                                                                                                                                     tandered likelikerange (m° time ("....." stete)
                                                              et leve-tree, see seerfic
                                                          M_{\rm present derivative message bound consummations of the day <math display="inline">0\,({\rm loc}\approx 33)
                                                           ericum (Cond Marriage (1999))
des Bibes (MC)
                                                          steen aftern mass;

without his cold Affections (Musel),

where is "Small Persons," it was so

where is been disjoint factors.
             appellusibun Contient, and
       de est acon ann 1670 a conding (1707-170)
desse sportige Parra, france, sportigen aconste amplica e majorane
a condingue Parra, france a condition (1707-170) and aconstraine
a desse a condition (1707).
                         ill_{\mathcal{F}}(f) = \max_{i \in \mathcal{F}} ill_{\mathcal{F}}(f) = \min_{i \in \mathcal{F}(f)} ill_{\mathcal{F}}(f) = \min_{i \in \mathcal{F}(f)} ill_{\mathcal{F}}(f) = \min_{i \in \mathcal{F}(f)} ill_{\mathcal{F}(f)}(f) = \min_{i \in \mathcal{F}(f)} ill_{
                       A la Deparation spring bean afg.-x.

    -composition in described the continuent (conversation from it.)
    -composition in described the control in belong resident in control in the control in the control in the control in co
                       \epsilon \mapsto torpet upring bean class <math display="inline">dg \rightarrow c
                       "Count for "ang," Alain "Arman's formal Mill Million application for " or 
count data for any country March 1 and Mills ("); " a construction of produced and selection of pro
                                                                                                                 combinator Separated games some spring beam to
           Dependency niection estables their date / client Apol
       nport imparemights need facer best support. Hindyroen Kerlógy, Listen Centest.
Impart contributed Alph Messagethmentary
```

Constructor injection

public data beyanda-cytej-com Natió

Constructor Injection

=======

=======

- => IOC container uses parameterized constructor to create target spring bean class obj and to assign/ inject dependent spring bean class objs or values to target spring bean class obj
- => For this injection, we need to place <constructor-arg> tags for "n" times to use "n" for constructor Injection while working with xml driven cfgs

param constructor

- => For this injection, we need to place @Autowired on the top of parameterized constructor in annotation driven cfgs
- => In this injection, the spring bean class obj will always be created using parameterized constructor
- => This is called Fastest injection becoz the dependent values/objects will be injected to target spring bean class obj during the object creation process.

(spring bean class obj)

- => In setter Injection, the order of creating spring bean class objs are
- a) target spring class class obj b) Dependent spring bean class obj

calls

[IOC containerrget.setXxx(-) having dependent obj as the arg value]

=> In constructor Injection, the order of creating spring bean class objs are a) Dependent spring bean class obj b) Target spring bean class obj

[IOC container uses parameterized constructor having dependent obj(s) as the arg values to create the target spring bean class obj]

Example app using constructor Injection

//WishMessageGenerator.java (Target spring bean class) package com.nt.sbeans

```
import java.time.LocalDate;
import java.time.LocalTime;
public class Wish MessageGenerator {
  private LocalTime time; //HAS-A property
  private LocalDate date; //HAS-A property

IOCProj02-DependencyInjection-constructorInjection-xml cfgs
// parameterized constructor for constructor Injection
  public WishMessageGenerator(LocalTime time, LocalDate date) {
  }
  this.time-time;
  this.date=date;
```

System.out.println("Wish MessageGenerator:: 2-param constructor");

```
> JRE System Library [JavaSE-21]
src
com.nt.cfgs
applicationContext.xml

√ com.nt.main

DependencyInjectionTest.java
com.nt.sbeans
> WishMessageGenerator.java Referenced Libraries
//b.method
public String showWishMessage(String user) {
System.out.println("Wish MessageGenerator.showWishMessage()::"+time+"....."+date);
// get current hour of the day
int hour-time.getHour();
//generate wish message based on current hour of the day
if(hour<12)
return "Good Morning::"+user;
else if(hour<16)
return "Good Afternoon:"+user;
else if(hour<20)
return "Good Evening:"+user;
else
return "Good Night:"+user;
}
applicationContext.xml
<?xml version="1.0" encoding="UTF-8"?>
<beans xmIns="http://www.springframework.org/schema/beans"</pre>
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="
http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans.xsd">
<!-- Dependent spring bean cfg -->
<bean id="Itime" class="java.time.LocalTime" factory-method="now" />
<bean id="Idate" class="java.time.LocalDate" factory-method="now"/>
<!-- calls LocalTime.now() method to create LocalTime class obj having system time -->
<!-- Target spring bean class cfg -->
<bean id="wmg" class="com.nt.sbeans.WishMessageGenerator">
<constructor-arg name="time" ref="ltime"/>
```

```
<constructor-arg name="date" ref="ldate"/>
constructor Injection cfgs
</bean>
</beans>
constructor Dependent param name spring bean id
Dependencylnjection Test.java (main class/ Client App)
_____
//DependencyInjectionTest.java
package com.nt.main;
import org.springframework.context.support.FileSystemXmlApplicationContext;
import com.nt.sbeans.WishMessageGenerator;
public class DependencyInjection Test {
public static void main(String[] args) {
System.out.println("DependencyInjection Test.main()");
//crete IOC Container
FileSystemXmlApplicationContext ctx=
new FileSystemXmlApplicationContext("src/com/nt/cfgs/applicationContext.xml");
//get Target Spring bean class obj ref
Object obj=ctx.getBean("wmg");
WishMessageGenerator generator=(WishMessageGenerator)obj;
//invoke the b.method
String result-generator.showWishMessage("raja");
System.out.println("Result is ::"+result);
//close the IOC cotnainer
ctx.close();
System.out.println("DependencyInjection Test.main()");
Spring Bean cfg file
=>For target spring bean, if want to inject another spring bean class obj using xml driven cfgs there we need
to use "ref" attribute of roperty> tag or <constructor-arg> tag.
=>For Dependent spring bean, if want to inject simple values/wrapper values/String values using xml driven
cfgs there we need to use "value" attribute of cfgs there we need to use "value" attribute of cfgs there we need to use "value" attribute of cfgs there we need to use "value" attribute of cfgs there we need to use "value" attribute of cfgs there we need to use "value" attribute of cfgs there we need to use "value" attribute of cfgs there we need to use "value" attribute of cfgs there we need to use "value" attribute of cfgs there we need to use "value" attribute of cfgs there we need to use "value" attribute of cfgs there we need to use "value" attribute of cfgs there we need to use "value" attribute of cfg the to use "v
=> A spring bean class can have the following types of bean properties
a) HAS -A properties (Object type/ Ref type properties)
b) Simple type properties (primitive datatype/wrapper data type/String type properties)
c) Array type properties
d) Collection type properties
```

Q) if we configure both setter Injection and constructor injection to the same property of spring bean class

```
can u tell me which injection value will be taken as the final value?
Ans) Since the setter Injection takes place after the constructor injection we can say
setter injection value overrides the constructor injection value
<!-- Target spring bean class cfg -->
<bean id="wmg" class="com.nt.sbeans.WishMessageGenerator"> <constructor-arg name="time"</pre>
ref="Itime"/>
</bean>
//Injects mukesh as the final value(setter Injection)
to the "name" property
What Circular Injection or Cyclic Injection in spring programming?
Ans) if two spring beans are dependent to each other then we can say there in cyclic injection or Circular
Injection (A is dependent to B and B is dependent to A) =>This is not possible with constructor Injection
(application goes to inifinite loop)
and possible with setter Injection
Sample code
========
//A.java
package com.nt.sbeans;
public class A{
private B b;
public void setB(B b) {
System.out.println("A.setB()");
this.b=b;
}
/* public A(B b) {
System.out.println("A::1-param constructor");
this.b=b;
}*/
//toString()
public String toString() {
return "from A spring bean";
}
//B.java
```

package com.nt.sbeans;

public class B {

```
private A a;
public void setA(A a) {
System.out.println("B:: setA(-)");
this.a=a;
//toString()
public String toString() {
return "From B spring bean";
applicationContext.xml
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"</pre>
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:context="http://www.springframework.org/schema/context"
xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans.xsd">
<!-- spring beans cfg -->
<bean id="a1" class="com.nt.sbeans.A">
property name="b" ref="b1"/>
</bean>
<bean id="b1" class="com.nt.sbeans.B">
property name="a" ref="a1"/>
</bean>
</beans>
Client App/ Main class
//CyclicInjection Test.java
package com.nt.main;
import org.springframework.context.support.FileSystemXmlApplicationContext;
import com.nt.sbeans.A;
public class CyclicInjection Test {
public static void main(String[] args) {
//create IOC container
FileSystemXmlApplicationContext ctx=
new FileSystemXmlApplicationContext("src/com/nt/cfgs/applicationContext.xml");
// get spring bean class obj ref
A a1 =(A)ctx.getBean("a1");
```

```
System.out.println(a1);
//close the container ctx.close();
Few Eclipse short cut keys
ctrl+shift+o: To import the packages
ctrl+shift+c :: To enable or disable single line commenting (toggle key)
ctrl+shfit+ / To enable multiline commenting
alt+shift+s,r ---> For setters and getters
ctrl+shfit+:: To disable multiline commenting
systrace +ctrl+space :: System.out.println(-) with default message
alt+shift+s,o ---> for parameterized constructors alt+shift+s,s ----> for toString()
sysout +ctrl+space :: System.out.println(-) with no message
ctrl+space :: For getting possible suggestions
and etc..
How Disable block commenting problem (which disturbs the structure of the code) while working multi-line
commenting?
Ans) window menu--->preferences ---> search for formatter ---> new formatter name :: f1 --->
go to comments section ---> enable ---> deselect the following checkbox
▼ Comments
Maximum width for comments:

✓ Count width from comment's starting position

Enable Javadoc comment formatting
Enable block comment formatting
```