

10.1. write a spring API using Application context to load a bean by name Hello with property message and load the property using spring bean xml

Code:- config.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<beans
xmlns="http://www.springframework.org/schema/beans"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:context="http://www.springframework.org/schema/beans"
xsi:schemaLocation="http://www.springframework.org/sc
hema/beans
https://www.springframework.org/schema/beans/spring-
beans.xsd
http://www.springframework.org/schema/context
https://www.springframework.org/schema/context/spring
-context.xsd">
  <bean class="com.SpringCore.SpringCore.hello"
name="H1">
    <constructor-arg>
      <value>
        hello constructor Dependency called for setting
message
      </value>
    </constructor-arg>
    <!--<property name="message">
      <value>
        hello dear, Successfully returning bean property
      </value>
    </property-->
  </bean>

</beans>
```

Code:-App.java

```
package com.SpringCore.SpringCore;

import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;

public class App
{
    public static void main( String[] args )
    {
        System.out.println( "Hello World!" );
        ApplicationContext ac=new ClassPathXmlApplicationContext("config.xml");
        hello h1=(hello)ac.getBean("H1");
        System.out.println(h1);
        //h1.display();
    }
}
```

Code:-Hello.java

```
package com.SpringCore.SpringCore;

public class hello {
    private String message;

    public String getMessage() {
        return message;
    }
    public void setMessage(String message) {
        System.out.println("setter dependecy");
        this.message = message;
    }

    public hello() {
        super();
        // TODO Auto-generated constructor stub
    }

    public hello(String message) {
        super();
        System.out.println("constructor called");
        this.message = message;
    }

    @Override
    public String toString() {
        return "hello [message=" + message + "]";
    }
    public void display()
    {
        System.out.println("Hello Bean Called");
    }
}
```

Code:pom.xml

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>com.SpringCore</groupId>
  <artifactId>SpringCore</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <packaging>jar</packaging>
  <name>SpringCore</name>
  <url>http://maven.apache.org</url>
  <properties>
    <project.build.sourceEncoding>UTF-
8</project.build.sourceEncoding>
  </properties>

  <dependencies>
<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-core</artifactId>
  <version>5.2.3.RELEASE</version>
</dependency>
<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-context</artifactId>
  <version>5.2.3.RELEASE</version>
</dependency>

  <dependency>
    <groupId>junit</groupId>
    <artifactId>junit</artifactId>
    <version>3.8.1</version>
    <scope>test</scope>
  </dependency>
</dependencies>
</project>
```

Search Project Run Window Help

Console × config.xml AppTest.java hello.java

<terminated> App (1) [Java Application] D:\TYBCA\ eclipse-jee-2023-03-R-win32-x86_64\ eclipse\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.6.v20230204-1729\jre\bin\javaw.exe (Dec 14, 2024, 3:25:31 PM)

```
Hello World!  
constructor called  
hello [message=  
    hello constructor Dependency called for setting message  
]
```