**Spring Java Based Configuration**

1. create maven java project:

mvn archetype:generate -DgroupId=com.spring.java.conf -DartifactId=SpringJavaConfDemo -Dversion=1.0-SNAPSHOT -Dpackage=com.spring.java.configuration -DarchetypeArtifactId=maven-archetype-quickstart -DinteractiveMode=false

2. Make it Eclipse compatible:

mvn eclipse:eclipse -Dwtpversion=2.0

3. Import into eclipse and add spring-core, spring-context maven dependencies. Following is the 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/maven-v4\_0\_0.xsd"*>  <modelVersion>4.0.0</modelVersion>  <groupId>com.spring.java.conf</groupId>  <artifactId>SpringJavaConfDemo</artifactId>  <packaging>jar</packaging>  <version>1.0-SNAPSHOT</version>  <name>SpringJavaConfDemo</name>  <url>http://maven.apache.org</url>    <properties>  <spring.version>4.2.5.RELEASE</spring.version>  </properties>  <dependencies>  <dependency>  <groupId>junit</groupId>  <artifactId>junit</artifactId>  <version>3.8.1</version>  <scope>test</scope>  </dependency>    <!-- spring dependencies -->  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-core</artifactId>  <version>${spring.version}</version>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-context</artifactId>  <version>${spring.version}</version>  </dependency>    </dependencies>  <build>  <pluginManagement>  <plugins>  <plugin>  <groupId>org.apache.maven.plugin</groupId>  <artifactId>maven-compiler-plugin</artifactId>  <version>3.5.1</version>  <configuration>  <source>1.8</source>  <target>1.8</target>  </configuration>  </plugin>  </plugins>  </pluginManagement>  </build>  </project> |

4. Write Java based configuration: SpringJavaConfiguration.java

|  |
| --- |
| **package** com.spring.java.configuration;  **import** org.springframework.beans.factory.annotation.Value;  **import** org.springframework.context.annotation.Bean;  **import** org.springframework.context.annotation.ComponentScan;  **import** org.springframework.context.annotation.Configuration;  **import** org.springframework.context.annotation.PropertySource;  **import** org.springframework.context.support.PropertySourcesPlaceholderConfigurer;  @Configuration  @ComponentScan("com.spring.java.configuration")  @PropertySource("classpath:/test.properties")  **public** **class** SpringJavaConfiguration {  @Value("${key1}")  **private** String key1;  @Value("${key2}")  **private** String key2;  @Value("${key3}")  **private** String key3;  @Bean  **public** Foo foo() {  System.***out***.println(key1 + ", " + key2 + ", " + key3);  **return** **new** Foo("Spring Java Based Configuration");  }  @Bean  **public** **static** PropertySourcesPlaceholderConfigurer propertySourcesPlaceholderConfigurer() {  **return** **new** PropertySourcesPlaceholderConfigurer();  }  } |

5. Foo.java

|  |
| --- |
| **package** com.spring.java.configuration;  **public** **class** Foo {  **private** String message;  **public** Foo(String message) {  **this**.message = message;  }  **public** String getMessage() {  **return** message;  }  } |

6. test.properties

|  |
| --- |
| key1=value1  key2=value2  key3=value3 |

7. Test.java

|  |
| --- |
| **package** com.spring.java.configuration;  **import** org.springframework.context.annotation.AnnotationConfigApplicationContext;  **import** org.springframework.context.support.AbstractApplicationContext;  **public** **class** Test {  **public** **static** **void** main(String[] args) {  **try** (AbstractApplicationContext context = **new** AnnotationConfigApplicationContext(SpringJavaConfiguration.**class**);) {  Foo foo = context.getBean(Foo.**class**);  System.***out***.println(foo.getMessage());  Foo foo2 = (Foo) context.getBean("foo");  System.***out***.println(foo2.getMessage());  }  }  } |