Spring & Microservices Training

1. Spring Core
2. Hibernate
3. Java 8
4. GIT
5. Spring Boot & Microservices

Spring Framework

Framework: It allows you to quickly develop the applications by providing lot of inbuilt features like Type-conversion, Exception Handling, Design Patterns, Transaction, Security, Internationalization and so on.

Spring Framework: It allows you to develop different types of applications like desktop, web applications, enterprise applications, mobile application, cloud based application, it provides lot of modules to develop the applications

Some spring framework modules

1. Core Spring: Basic Building block for all the spring technologies/modules
2. Spring MVC: It is used to develop web applications
3. Spring REST: It is used to develop REST API’s
4. Spring Boot: It will automate the spring development
5. Spring Cloud & Microservices: It used to deploy spring applications on cloud

Spring Core:

It provides you the alternate way of creating the dependent objects which is nothing but dependency injection, here the objects that are dependent on other objects are maintained by the spring and supplies to the dependents with very less changes in the application code.

Dependency Injection: It is an alternative way of creating the dependencies that are supplied to other dependent object

Environment setup

Java 8  
Eclipse  
MySQL  
Postman client  
Git.

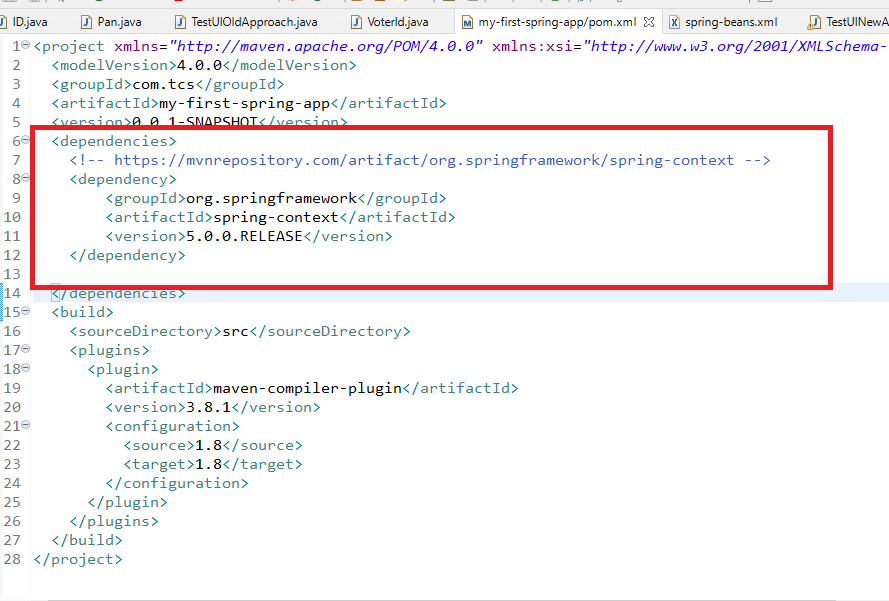
Spring Framework helps in resolving the object dependencies through Dependency Injection feature where it creates the dependencies and maintains in its container called IoC(Inversion of Control)

IoC: Since the operation is inverted its called as Inversion of Control

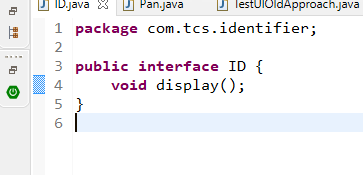
How to implement the Dependency Injection in Spring Framework

1. Create Java Maven Project
2. Add the Spring Libraries in pom.xml: to configure the jar dependencies
3. Spring Configuration XML file: to configure the object dependencies.

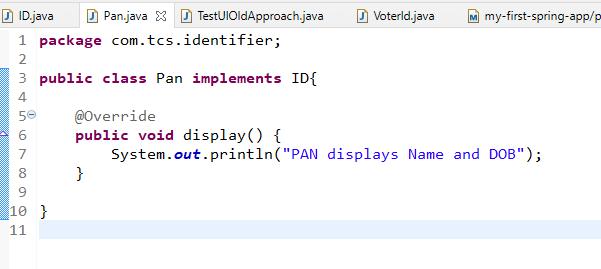
pom.xml



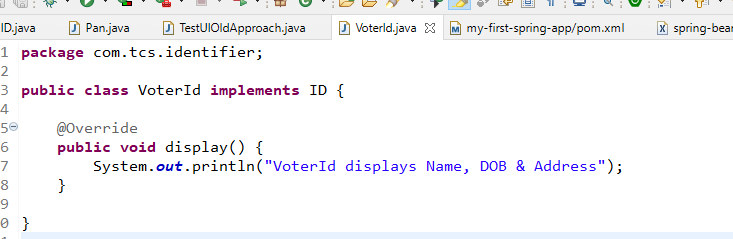
ID.java



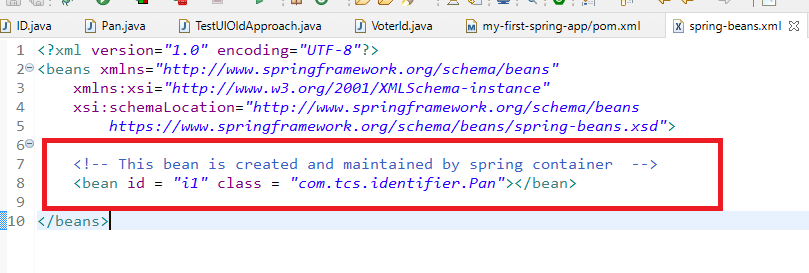
Pan.java



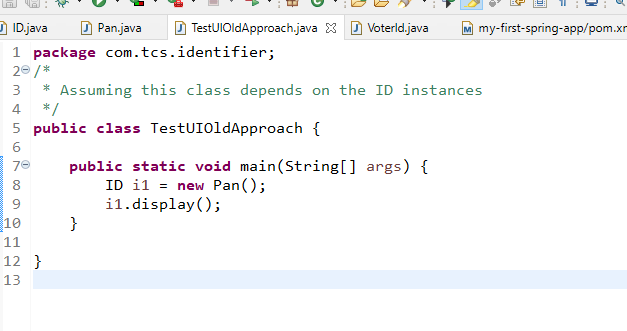
VoterId.java



spring-beans.xml

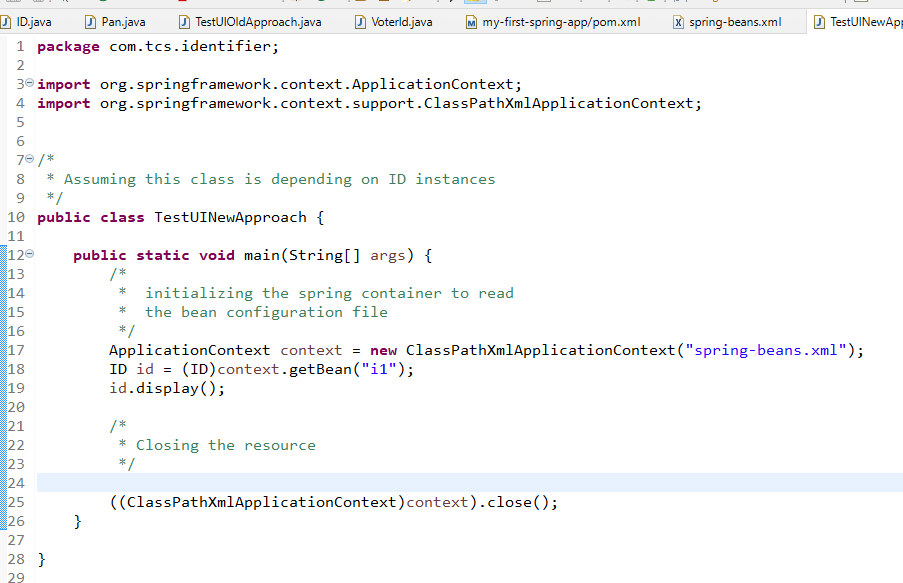


TestUIOldApproach.java



Here if you want to use the another implementation object you will change the code at 8th line but with new approach of using the dependency injection you don’t have to change the client code

TestUINewApproach.java



Here the spring framework is going to create the objects based on the configuration done in the xml file and getBean() returns the object having the id from the container, now you can change the implementation instance in the XML file without changing the code in TestUINewApproach