===============

What is Spring

===============

=> It is java based framework developed by interface21 company.

Note: Now spring is under license of VMWare.

=> Spring is free and open source framework.

=> Spring is called as Application development framework.

Note: We can develop all the layers of application using spring framework.

=> Spring Framework developed by "Rod Johnson"

=> Spring 1.x released in 2004

=> The current version of spring is 6.x version

=================================

What we can develop using Spring

==================================

1) Stand-alone apps (desktop)

2) Web Applications (C2B)

3) Distributed Applications (B2B)

=====================

Spring Architecture

=====================

=> Spring developed in modular fashion

Spring 1.x => 7 modules

Spring 2.x => 6 modules

Spring 3.x to 6.x => 20+ modules....

Note: Spring is loosely coupled framework.

==================

Spring Modules

==================

1) Spring Core (IOC & DI)

2) Spring Context

3) Spring AOP (Aspect Oriented Programming)

4) Spring JDBC / Spring DAO

5) Spring ORM (Object relational mapping)

6) Spring Web MVC (C 2 B)

7) Spring REST (B 2 B)

8) Spring Security (Authentication & Authorization)

9) Spring Cloud (Microservices)

10) Spring Batch (bulk operation)

11) Spring Schedular

12) Spring Data JPA

====================================

Developing First Spring Application

====================================

Step-1 : Setup STS IDE

Step-2 : Create Maven Project using STS IDE

Step-3 : Configure Spring-Context dependency in pom.xml

Step-4 : Create required java classes

Step-5 : Configure Java classes as Spring-Beans (xml)

Step-6 : Create class to test IOC behaviour

================== Beans.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"

xsi:schemaLocation="

http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd">

<bean id="de" class="in.ashokit.beans.DieselEngine"/>

<bean id="pe" class="in.ashokit.beans.PetrolEngine"/>

</beans>

=======================================================

public class MyApp {

public static void main(String[] args) {

ApplicationContext ctxt = new ClassPathXmlApplicationContext("Beans.xml");

DieselEngine de = ctxt.getBean(DieselEngine.class);

de.start();

}

}

=========================================================

==============

Assignment

===============

1) Eclipse Shortcuts : https://www.youtube.com/watch?v=TvYMey5SYa8

2) Debugging : https://www.youtube.com/watch?v=2WxsClYhreE

===============

What is IOC ?

===============

=> IOC stands for Inversion of control.

=> It is responsible to manage and colloborate classes in our application.

=> IOC will take care of objs creation and dependency injection.

=> The java classes which are managed by IOC are called as Spring Beans.

=> To represent java class as spring bean we are using <bean/> tag.

<bean id="" class="">

</bean>

==============================

What is Dependency Injection

==============================

=> Injecting dependent class obj into target class object is called as Dependency Injection.

=> We can perform Dependency Injection in 3 ways

1) Constructor Injection (CI)

2) Setter Injection (SI)

3) Field Injection (FI)

=> The process of injecting dependent bean obj into target bean obj using target bean constructor is called as constructor injection.

=> The process of injecting dependent bean obj into target bean obj using target bean setter method is called as setter injection.

=> The process of injecting dependent bean obj into target bean obj using target bean variable directley is called as Field injection.

-----------------------------------------------------

public class PwdService {

public PwdService() {

System.out.println("PwdService::Constructor");

}

public String encodePwd(String pwd) {

Encoder encoder = Base64.getEncoder();

String encodedPwd = encoder.encodeToString(pwd.getBytes());

return encodedPwd;

}

}

------------------------------------------------------

public class UserService {

private PwdService pwdService;

public UserService() {

System.out.println("UserService::Constructor");

}

// SI

public void setPwdService(PwdService pwdService) {

System.out.println("setPwdService() -- called..");

this.pwdService = pwdService;

}

public void saveUser(String uname, String pwd, Long phno) {

String encodePwd = pwdService.encodePwd(pwd);

System.out.println("Encoded pwd :: " + encodePwd);

// TODO:: insert into db

System.out.println("Record inserted in DB...");

}

}

-------------------------------------------------------

<?xml version="1.0" encoding="UTF-8"?>

<beans xmlns="http://www.springframework.org/schema/beans"

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">

<bean id="pwd" class="in.ashokit.beans.PwdService"/>

<bean id="user" class="in.ashokit.beans.UserService">

<property name="pwdService" ref="pwd" />

</bean>

</beans>

-------------------------------------------------------

public class MyApp {

public static void main(String[] args) {

ApplicationContext ctxt = new ClassPathXmlApplicationContext("Spring-Beans.xml");

UserService userService = ctxt.getBean(UserService.class);

userService.saveUser("ashok", "ashok@123", 123456l);

}

}

------------------------------------------------------