Wednesday, June 29, 2022 9:34 AM

Spring Framework -------

API Library = HttpSession , HttpServletRequest ,……..
Framework = An outline of application design is given , we have to FILL IN THE BLANKS and create our application!!!
Less experienced can quickly develop an application of GOOD DESIGN

Spring Framework is based on COMPONENT integration .
Many components [We write + Spring provides + Third party Provides]
Spring framework will Integrate the components to produce the application

Spring Components are called as BEANS !!!!!

What are beans in spring ? POJO classes !!! Plain Old Java Objects .
POJO consists ---- **packaged class + private properties + public constructors + getters + setters**
With some kind of responsibility / functionality /feature

Spring Framework provides Life Cycle Management of the Beans

1. Create bean objects = Instantiation
2. Inject properties of the bean = set properties = Dependency Injection
3. Call life cycle methods of the bean
4. Make the bean available to whoever wants it
5. Destroy the bean / make the bean available to GC

The Spring Container is called as **ApplicationContext**
How will the Spring Container know about the bean class?
How will it know how to instantiate the class ?
How will it know how to set the properties of the instance ?

Programmer Informs the Container using different configurations of XML or Annotation

---

Spring Framework comes with MANY JAR files !!!!
YOU need to manage the JAR files -

1. Download the jar files ---- Different Versions ? Are all the versions compatible ?
   Dependencies ---
   JAR 1 needs jarA, Jar B, Jar C
   jarA needs Jarq , jar p
   Jarq needs jart
2. Add them in your project in the build path

To simplify the above TASK BUILD tools are USED !! Example - ANT , MAVEN , GRADLE ,….

We will use a build tool = **MAVEN** !!!

MAVEN maintains Repositories of JAR files.

GLOBAL Repository = on the net = mvnrepositary
LOCAL Repository = on your machine = .m2 folder on our machine

Maven has a configuration file = POM.xml ---- we can specify the dependency tags of all the JARS that we may need.

We will get a Spring Project Template !!!!

Maven file structure is as follows : -

Usually in non MAVEN core java project source folder is src
src--
study
Book.java
test
Student.java

in MAVEN the folder structure is
Source folder is src/main/java
------- study
Book.java
--------- study.users
User.java

Exercise --- DemoApplication --- connect to data base and insert a row in product table

---

STS
Downloaded a template project of MAVEN and Spring
We imported that project in STS
We added a dependency of mysql connector in pom.xml
We executed a DB insert code

ApplicationContext of Spring and Beans !!!
HOW to inform the Spring Container about the Beans that we create ?
We can use Configuration in 3 WAYS
1. Using beans.xml
2. Using Java configuration class
3. Using annotations.

Annotation ----MARKER , STICKER

Sticker = Annotation [ready made in the JAR files]

Motor Cycle = Target of the Annotation [Type=class/interface , method , property, constructor , method parameter] ---- WE WRITE the code and apply ANNOTATION to it

Watchman = Container (Servlet Container , Spring Container , Hibernate Container ,.....)
[ready made JAR files]

Depending on if sticker is present or not present
Watchman/Container will do different things

Annotation = is used to communicate between the programmer who writes the components and the container

---

Write a bean Message -properties message, senderName
Constructor, getter setter