|  |  |
| --- | --- |
|  | **Spring Containers** |
|  | **Introduction** |
|  | **1.** The central component of a Spring application is the containers. |
|  | **2.** Spring containers manage the life cycle of the bean in the applications,and are responsible for wiring of these beans. |
|  | **3.** Spring provides two interfaces that act as containers,namely **BeanFactory** and **ApplicationContext.** |
|  |  |
|  | **BeanFactory** |
|  | **1.** BeanFactory is an interface and available in org.springframework.beans.factory package. |
|  | **2.** BeanFactory container is the root container that loads all the beans and provides dependency injection. |
|  | **3.** BeanFactory is the basic container. |
|  | **4.** BeanFactory is a lazy container,which means that it instantiate the bean and configures it only when the getBean() method is called. |
|  |  |
|  | **Loading the configuration file using BeanFactory** |
|  | BeanFactory factory=new XmlBeanFactory(new ClassPathResource("beans.xml")); |
|  |  |
|  | **Example:** |
|  | **1.** Create the java project. |
|  | **2.** Go to configure build path option. |
|  | **3.** Add the spring jar files to the project using build. |
|  | **4.** Create a pojo class named Message.java |
|  | **5.** Create the Spring bean configuration file and add Message class as bean entry. |
|  | **6.** Create client program and run it. |
|  | **7.** Final project structure. |
|  | **Implementation** |
|  | **1.** Create the java project. |
|  | **1.png** |
|  | **2.png** |
|  |  |
|  | 2.Configuring build path |
|  | 3.png |
|  | 1.png |
|  |  |
|  | 3. Add the spring jar files to the project using build. |
|  | 2.png |
|  | 3.png |
|  |  |
|  | 4. Create a pojo class named Message.java |
|  | 4.png |
|  |  |
|  | 5. Create the Spring bean configuration file and add Message class as bean entry. |
|  | 5.png |
|  | 6. Create client program and run it. |
|  | 6.png |
|  | BeanFactory factory=new XmlBeanFactory(new ClassPathResource("Spring-Core.xml")); |
|  | 7.png |
|  |  |
|  | 7. Final Project structure. |
|  | 8.png |
|  |  |
|  | **Output:** |
|  | 9.png |
|  |  |
|  | **ApplicationContext** |
|  | 1. ApplicationContext is an interface and available in org.springframework.contexty package. |
|  | 2. ApplicationContext is the sub interface of BeanFactory. |
|  | 3. ApplicationContext used as a container in the enterprise applications with a number of features. |
|  | 4. ApplicationContext container instantiate all the beans while loading the configuration file. |
|  |  |
|  | **Loading the configuration file using ApplicationContext container.** |
|  | Use the above example for the ApplicationContext container demo except client program. |
|  |  |
|  | **Example:** |
|  | 10.png |
|  | ApplicationContext applicationContext=new ClassPathXmlApplicationContext("Spring-Core.xml"); |
|  | 11.png |
|  |  |
|  | **Output:** |
|  | 12.png |
|  |  |