Spring 面向接口编程----》上转型对象的实现

1. spring的初始化：
   1. (1)ApplicationContext ctx=new ClassPathXmlApplicationContext(“xxxxx.xml’)

(2)String path[]={“xx.xml”,”c.xml”,”zz.xml”}

ApplicationContext ctx=new ClassPathXmlApplicationContext(path)

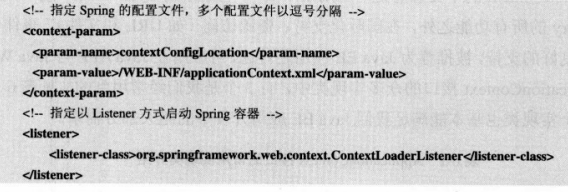
* 1. FileSystemXmlApplicationContext()

String path[]={“c:/bean.xml”,”c:/bean2/xml”}

ApplicationContext ctx=new FileSystemXmlApplicationContext(path)

* 1. 现在一边都是基于Listener实现的，也可以给予Servlet实现

XmlWebApplicationContext()



1. 常用bean整理

(2-1)国际化消息bean

第一种

<bean id=*"messageSource"*

class=*"org.springframework.context.support.ReloadableResourceBundleMessageSource"*>

<property name=*"parentMessageSource"* ref=*"bizMessageSource"* />

<property name=*"basenames"*>

<list>

<value>resources.cls-web-resources</value>

<value>resources.cls-web-resources-definitions</value>

<value>resources.cls-web-resources-menu</value>

</list>

</property>

</bean>

第二种

<bean id=*"messageSource"*

class=*"org.springframework.context.support.ReloadableResourceBundleMessageSource"*>

<property name=*"parentMessageSource"* ref=*"bizMessageSource"* />

<property name=*"fallbackToSystemLocale"*>

<value>false</value>

</property>

<property name=*"basenames"*>

<list>

<value>classpath:resources/cls-web-resources</value>

<value>classpath:resources/cls-web-resources-definitions</value>

<value>classpath:resources/cls-web-resources-menu</value>

</list>

</property>

</bean>

(2-2)properies属性文件配置

<bean id=*"propertyConfigurer"*

class=*"org.springframework.beans.factory.config.PropertyPlaceholderConfigurer"*>

<property name=*"location"*>-----》可以配置多个 用list标签包裹

<value>conf/sqlmap/jdbc.properties</value>

</property>

<property name=*"fileEncoding"*>

<value>UTF-8</value>

</property>

</bean>

1. 数据库连接池

<bean id="dataSource" destroy-method="close">

<property name="driverClassName"><value>${jdbc.driverClassName}</value></property>

<property name="url"><value>${jdbc.url}</value></property>

<property name="username"><value>${jdbc.username}</value></property>

<property name="password"><value>${jdbc.password}</value></property>

</bean>

1. 会话工厂

<bean id="sessionFactory"

<property name="dataSource"><ref local="dataSource"/></property>

<property name="mappingResources">

<value>mapping.xml</value>

</property>

<property name="hibernateProperties">

<props>

<prop key="hibernate.dialect">${hibernate.dialect}</prop>

</props>

</property>

</bean>