Ssh 整合

Struts + Spring + Hibernate三者各自的特点都是什么？

Struts 的MVC设计模式可以使我们的逻辑变得很清晰，主要负责表示层的显示。

Spring 的IOC和AOP可以使我们的项目在最大限度上解藕。

hibernate的就是实体对象的持久化了, 数据库的封装。

web.xml 文件配置

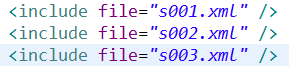


在Java Resources下的src目录下创建一个struts.xml文件



告知Struts2运行时使用Spring来创建对象就要加入一句：

<constant name="struts.objectFactory" value="spring" />

当然也可以包括多个xml文件，让项目分工明确，减少各个人的代码冲突

配置hibernate.cfg.xml



**appliCation.xml配置文件**

将 hibernate.cfg.xml 文件内的信息写入到该文件中：<bean id="sessionFactory" class="org.springframework.orm.hibernate5.LocalSessionFactoryBean">

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

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

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xmlns:p="http://www.springframework.org/schema/p"

xmlns:aop="http://www.springframework.org/schema/aop"

xmlns:context="http://www.springframework.org/schema/context"

xmlns:jee="http://www.springframework.org/schema/jee"

xmlns:tx="http://www.springframework.org/schema/tx"

xsi:schemaLocation="

http://www.springframework.org/schema/aop http://www.springframework.org/schema/aop/spring-aop-4.2.xsd

http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans-4.2.xsd

http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context-4.2.xsd

http://www.springframework.org/schema/jee http://www.springframework.org/schema/jee/spring-jee-4.2.xsd

http://www.springframework.org/schema/tx http://www.springframework.org/schema/tx/spring-tx-4.2.xsd">

<!-- 引入外部属性文件 -->

<context:property-placeholder location="classpath:jdbc.properties"/>

<!-- 类似于财务部门一样，类就是钱，所有需要类的实例都由srping去管理 -->

<bean id="myIndexAction" class="ssh.action.IndexAction" scope="prototype">

<!-- setIs(myIndexService) -->

<property name="is" ref="myIndexService"/>

</bean>

<!-- myIndexService = new ssh.service.IndexServiceImpl() -->

<bean id="myIndexService" class="ssh.service.IndexServiceImpl" scope="prototype">

<property name="id" ref="myIndexDao"/>

</bean>

<bean id="myIndexDao" class="ssh.dao.IndexDaoImpl" scope="prototype">

<!-- 晚点再注入能用的seesionFactory -->

<property name="sessionFactory" ref="mySessionFactory"></property>

</bean>

<!--

错误的做法，new org.hibernate.internal.SessionFactoryImpl()

不可以，需要configuration来创建

bean id="mySessionFactory" class="org.hibernate.internal.SessionFactoryImpl"></bean

-->

<bean id="mySessionFactory" class="org.springframework.orm.hibernate5.LocalSessionFactoryBean">

<!-- 注入连接池，包含了数据库用户名，密码等等信息 -->

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

<!-- 配置Hibernate的其他的属性 -->

<property name="hibernateProperties">

<props>

<prop key="hibernate.dialect">org.hibernate.dialect.MySQL5Dialect</prop>

<prop key="hibernate.show\_sql">true</prop>

<prop key="hibernate.format\_sql">true</prop>

<prop key="hibernate.connection.autocommit">false</prop>

<!-- 开机自动生成表 -->

<prop key="hibernate.hbm2ddl.auto">update</prop>

</props>

</property>

<property name="mappingResources">

<list>

<value>ssh/entity/BookCard.hbm.xml</value>

</list>

</property>

</bean>

<bean id="myDataSource" class="com.mchange.v2.c3p0.ComboPooledDataSource">

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

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

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

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

<!-- 每300秒检查所有连接池中的空闲连接 -->

<property name="idleConnectionTestPeriod" value="300"></property>

<!-- 最大空闲时间,900秒内未使用则连接被丢弃。若为0则永不丢弃 -->

<property name="maxIdleTime" value="900"></property>

<!-- 最大连接数 -->

<property name="maxPoolSize" value="2"></property>

</bean>

</beans>

<bean id="myIndexAction" class="ssh.action.IndexAction" scope="prototype">

<!-- setIs(myIndexService) -->

<property name="is" ref="myIndexService"/>

</bean>  
  
id="myIndexAction" 实例化对象的名字

class="ssh.action.IndexAction" 实例化对象的路径（包名类名）

scope="prototype" 非单例

name="is" 实例化对象里面属性的名字

ref="myIndexService" 引用

