

0624技术整理

title: 0624技术整理

author: 刘星烨

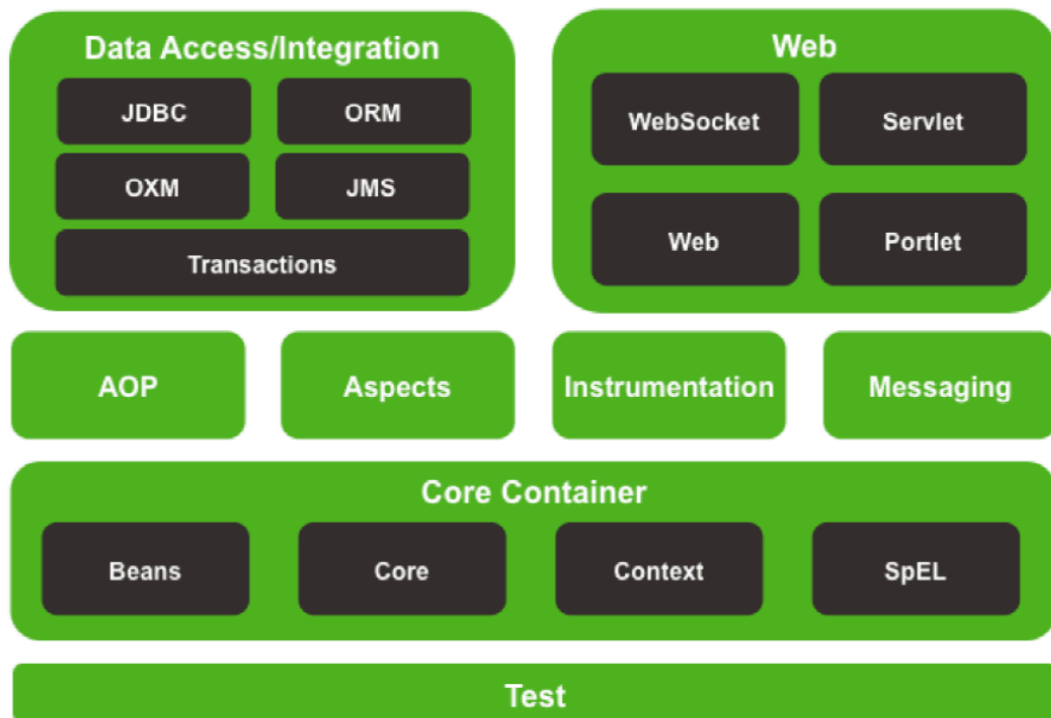
content: Spring框架

Spring概述

一个全栈开源框架，可以托管对象创建以及关系维护



Spring Framework Runtime



IOC

借助spring的IOC实现service层与dao层代码解耦合

- 导入spring坐标

```
<dependencies>
  <dependency>
    <groupId>org.springframework</groupId>    <artifactId>spring-
context</artifactId> <version>5.1.5.RELEASE</version> </dependency>
  <dependency> <groupId>junit</groupId> <artifactId>junit</artifactId>
<version>4.12</version>
    </dependency>
  </dependencies>
```

- 编写Dao接口和实现类

```
public interface UserDao {
    public void save();
}

public class UserDaoImpl implements UserDao {
    public void save() {
        System.out.println("保存成功");
    }
}
```

- 创建Spring核心配置文件

```
<?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"> </beans>
```

配置bean的相关标准，spring及xml

- 在spring配置文件中配置UserDaoImpl

```
<beans ...>
  <bean id="userDao" class="com.lagou.dao.impl.UserDaoImpl"></bean>
</beans>
```

在bean中注册userDao

- 使用spring相关API获得Bean实例

```
public class UserTest {  
    @Test  
    public void testSave() throws Exception {  
        ApplicationContext applicationContext = new  
        ClassPathXmlApplicationContext("applicationContext.xml");  
        UserDao userDao = (UserDao) applicationContext.getBean("userDao");  
        userDao.save();  
    }  
}
```

Spring API

BeanFactory

BeanFactory是 IOC 容器的核心接口，它定义了IOC的基本功能。

特点：在第一次调用getBean()方法时，创建指定对象的实例

ApplicationContext

代表应用上下文对象，可以获得spring中IOC容器的Bean对象。

特点：在spring容器启动时，加载并创建所有对象的实例

实现类：

- ClassPathXmlApplicationContext
它是从类的根路径下加载配置文件 推荐使用这种。
- FileSystemXmlApplicationContext
它是从磁盘路径上加载配置文件，配置文件可以在磁盘的任意位置。
- AnnotationConfigApplicationContext
当使用注解配置容器对象时，需要使用此类来创建 spring 容器。它用来读取注解。

Spring配置

```
<bean id="" class="" scope="" init-method="" destroy-method="" factory-bean  
= "" factory-method=""></bean>
```

依赖注入

- 构造方法

```
<bean id="userService" class="com.lagou.service.impl.UserServiceImpl">
  <constructor-arg name="userDao" ref="userDao"/>
</bean>
```

- set方法

```
<bean id="userService" class="com.lagou.service.impl.UserServiceImpl">
  <property name="userDao" ref="userDao"/>
</bean>
```

- P命名空间注入

```
<bean id="userService" class="com.lagou.service.impl.UserServiceImpl"
  p:userDao-ref="userDao"/>
```

首先需要引入P命名空间

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