

1.Spring

1.1 什么是 spring?

- Spring 是一个 Java 开发框架，由 Rod Johnson 发起，旨在为企业级应用提供一个轻量级、易于集成、可移植的 JavaSE/JavaEE full-stack () 开发平台，为 Java 开发者提供一套完整的开发工具，以专注于业务逻辑的实现。
- Spring 旨在解决企业级应用开发中的许多问题，即简化 Java 开发。
- Spring 以事务为中心，为企业级应用提供了丰富的功能，但一些核心功能依赖于两个关键技术：依赖注入 (dependency injection, DI) 和面向切面编程 (aspect-oriented programming, AOP)。

为了降低 Java 开发成本，Spring 提供了以下4个核心功能 **

- 基于 POJO 的轻量级核心；
- 基于面向切面编程 (AOP) 的轻量级核心；
- 基于面向切面编程 (AOP) 的轻量级核心；
- 基于面向切面编程 (AOP) 的轻量级核心；

1.2 Spring

- IOC () :
 - 控制反转，也即依赖注入，他不会将对象的创建和生命周期管理交给开发者，而是交给 Spring 容器。在 Spring 容器中，对象的创建和生命周期管理是由 Spring 容器负责的，开发者只需要关注业务逻辑的实现。通过 IOC，Spring 容器可以管理对象的生命周期，并可以在需要时将对象注入到其他地方。这有助于提高代码的可测试性和可维护性。
- AOP ()

。个，为，他，义，些
中，中，：，不 AOP
中，中，但，了AOP 以，个
，以，他们，中，了

1.3 Spring 什么

- **Spring** : Spring为一个 ;
- **Spring** : JavaEE 中, POJO JavaBean , ,
OOP () ; Spring IOC ,
之 交 IOC , ;
- **Spring** : IOC AOP IOC POJO 以 他们之 ;
AOP以
- IOC 互 作 件 , AOP 于
件

1.4 Spring 优 什么?

优

- ,
Spring 个 , 以 , 交 Spring
- AOP
Spring , 以
- 事
以 事 ,
- Spring Junit4 , 以 Spring
- 优
Spring不 优 , 了 优 (: Struts
Hibernate MyBatis)
- 低JavaEE API
Spring JavaEE 中 些API (JDBC JavaMail) , 了 ,
些API 低
- Spring 个 , 人
- Spring ,
- , Spring

1.5 Spring 些

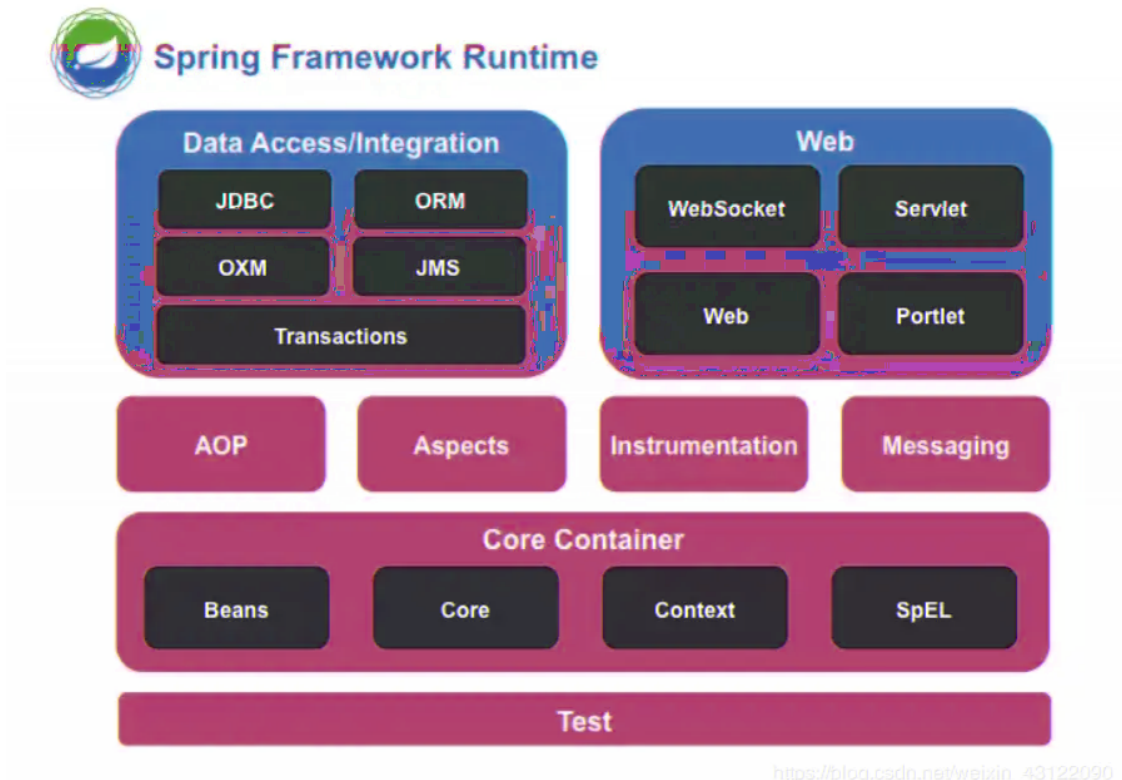
- : JavaEE企业 , SSH SSM

Spring价值:

- Spring 代 ;
- Spring 个 , POJO , 与 ;
- Spring , 了代 ;

1.6 Spring 些 ?

- Spring 20 个 , 1300 个不 件 些 件
- Core Container AOP Aspect Oriented Programming
- Instrumentation Data Access/Integration Web
- Messaging Test 6 个 中 以下 Spring 5 :



https://blog.csdn.net/weixin_43122090

- spring core: 了 (Inversion of Control, IOC) (Dependency Injection, DI)
- spring beans: 了 BeanFactory, 个 , Spring 为 Bean
- spring context: 于 core 上 context , 了
- spring jdbc: 了 个 JDBC , 了 JDBC 代 , 于 JDBC
- spring aop: 了 , 以 义
- spring Web: 了 Web , 件上传, servlet listeners ioc Web ApplicationContext
- spring test: 主 为 , JUnit TestNG Spring 件

1.7 Spring 中 到了 些 ?

1. : BeanFactory 体 , ;
2. : Bean 为
3. 代 : Spring AOP 了JDK 代 CGLIB ;

4. : 代 . RestTemplate, JmsTemplate, JpaTemplate
5. : 义 , 个 , 于
会 , Spring中listener -ApplicationListener

1.8 下 (spring context 上下)

- Spring , spring , BeanFactory 任何以spring为
Spring 之上, Spring 为 个
- Bean 个 , 了 , 从
代 中 org.springframework.beans.factory.xml.XmlBeanFactory ,
XML 件中 义 beans 从XML 件 个

1.9 Spring 中 些不同 事件

- Spring 了以下5 事件:
 1. 上下 事件 (ContextRefreshedEvent) : ConfigurableApplicationContext
中 refresh()
 2. 上下 事件 (ContextStartedEvent) : ConfigurableApplicationContext
Start() / 事件
 3. 上下 事件 (ContextStoppedEvent) : ConfigurableApplicationContext
Stop() 事件
 4. 上下 事件 (ContextClosedEvent) : ApplicationContext 事件
, Bean
 5. 事件 (RequestHandledEvent) : Web 中, 个http (request)
事件 个bean 了ApplicationListener , 个ApplicationEvent
以 , bean会

1.10 Spring 些不同 件?

Spring 以下 件:

- - 义
- Bean - , setter getter ,
- Bean 件 - 以 何 们
- Spring (AOP) -
- -

1.11 使 Spring 些 ?

使 Spring 以下 :

- 作为 个 Spring Web
- 作为 三 Web , Spring Frameworks 中
- 作为企业 Java Bean, 以 POJO (Plain Old Java Objects)
- 于

2. Spring 制反 (IOC)

2.1 什么 Spring IOC ?

- IOC (Inversion of Control), 传上代交, 件 “ ” 件, 从代了
- Spring IOC , ((DI) , , 且些个

2.2 制反 (IOC) 什么作

- 不件事, ,
- , 体 ,
- 了产以, 们交, 产中些, 何代, 代,

2.3 IOC 优 什么?

- IOC 代低
- , 不 JNDI
- 代价以
- IOC

2.4 Spring IOC 制

- Spring 中 IOC

- :

```
interface Fruit {
    public abstract void eat();
}

class Apple implements Fruit {
    public void eat(){
        System.out.println("Apple");
    }
}

class Orange implements Fruit {
    public void eat(){
        System.out.println("Orange");
    }
}

class Factory {
    public static Fruit getInstance(String className) {
        Fruit f=null;
        try {
            f=(Fruit)Class.forName(className).newInstance();
        } catch (Exception e) {
            e.printStackTrace();
        }
        return f;
    }
}

class Client {
    public static void main(String[] a) {
        Fruit f=Factory.getInstance("io.github.dunwu.spring.Apple");
        if(f!=null){
            f.eat();
        }
    }
}
```

2.5 Spring IOC 一些功

- Spring IOC 以下 :
- 中, , 从XML 上 , ref Spring RuntimeBeanReference
- 于IOC , Bean , Bean

2.6 BeanFactory ApplicationContext 什么区别?

- BeanFactory ApplicationContext Spring 两 , 以 Spring 中 ApplicationContext BeanFactory
-

- BeanFactory: Spring 了 Bean 义, bean , bean , bean 之
- ApplicationContext 作为BeanFactory , 了 BeanFactory , 了 :
 - MessageSource,
 - 件
 - 中 bean 事件
 - 个 件
 - 个 () 上下 , 个上下 专 于 个 , web
- - BeanFactory Bean , 个Bean (getBean()), Bean , 们 不 些 Spring Bean 个 , BeanFactory , getBean 会
 - ApplicationContext, , 了 Bean , , 们 以 Spring中 , 于 ApplicationContext Bean, bean , , 不 , 为 们 了

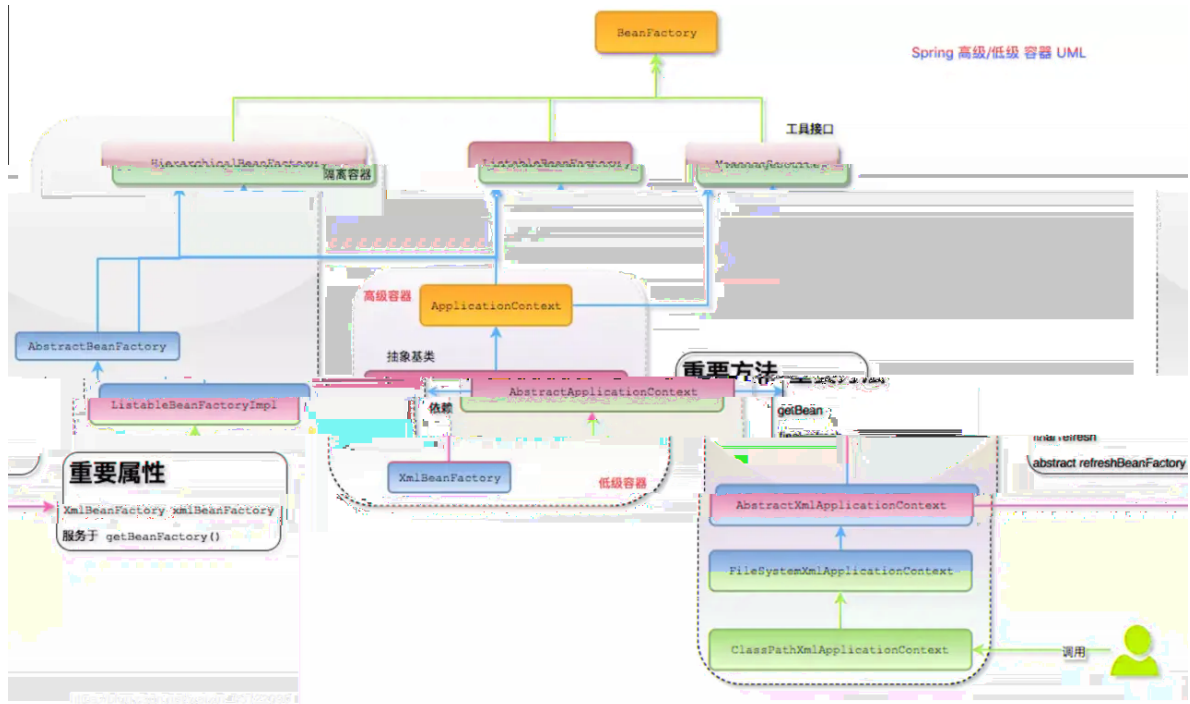
BeanFactory	ApplicationContext	Bean

- - BeanFactory 以 , ApplicationContext 以 , ContextLoader
- - BeanFactory ApplicationContext BeanPostProcessor BeanFactoryPostProcessor , 但两 之 : BeanFactory , ApplicationContext

2.7 Spring 何 , BeanFactory ApplicationContext 关

- Spring 作 Rod Johnson 了两个 以
 - BeanFactory
 - ApplicationContext
- BeanFactory , 以 为 个 HashMap, Key BeanName, Value Bean (put) , (get) 两个 们 以 之为“低 ”
- ApplicationContext 以 之为“ ” 为他 BeanFactory 了 他 了 个 了 , (JSP tag) , BeanFactory 了 以 他 , 不 BeanFactory 之 了, “ 上下 ”, 代 个 义了 个 refresh , Spring 人 , 于 个 , / bean
- , 了 两个 , 他 , 不介 他们了
- BeanFactory ApplicationContext

- 为了 “低 ” “ ” “ ” ,
ClassPathXmlApplicationContext 个 UML



- ? 不 , 下
- 上 BeanFactory, 下 3 个 , , 不
- 下 ApplicationContext “ ” , “低 ” ,
不 他 “低 ” `getBean` : 不
, 以 件 , 事件 (Observer)
- “ ” 但 BeanFactory 也 !
- “低 ” , Bean, Bean 他
上 `refresh` Bean , 事件

- 了 么 , 不 Spring IOC? 下: IOC Spring , 低
以 , 2 个 :

1. 件, BeanDefinition Map
 2. `getBean` , 从 BeanDefinition Map , Class , ,
 , `getBean` —
- 上 Spring 低 (BeanFactory) IOC
 - 于 ApplicationContext, 他 了低 , 他 `refresh`
, 个 Bean 作为 , 了 , 他不仅仅
IOC 他 不 , BeanFactory , 件 ,
事件 ,

2.8 ApplicationContext 什么?

- **FileSystemXmlApplicationContext** : 从 个XML 件中 beans 义, XML Bean 件
- **ClassPathXmlApplicationContext**: 也从 个XML 件中 beans 义, ,
classpath 为 个 classpath bean
- **WebXmlApplicationContext**: 个XML 件, 件 义了 个WEB bean

2.9 什么 Spring 依 入?

- IOC 个 , 以 不 主 两 :
: 于IOC , (DI) 了IOC
(Dependency Injection) , 件之 , 也
中 个 件之中 件不
位 , Java

2.10 依 入 原则

- : 件不 他 作 作
IOC , “ ” 从 件 代 中 , 交 IOC
件 , 会 (JavaBean中 setter)
传

2.11 依 入 什么优势

- 之 以 为 : , 件
JavaBean setter , 以
与 , 主 优 为:
 - 位 作与 代
 - 不 于 API, 以 任何 以
 - 不 , 以 不

2.12 些不同 依 入 ?

- 下 IOC , 为 (Interface Injection) , Setter
(Setter Injection) (Constructor Injection) 三 中 于
, 从Spring4
- 依 入: 个 , ,
个 代 个 他
- Setter 入: Setter static bean之
, bean setter , 了 于setter

2.13 依 入 Setter 入 区别

函 入	setter 入
不会 setter	会 setter
任 会 个	任 不会 个
于	于

- 两 以 , Setter , setter

3. Spring Beans

3.1 什么 Spring beans?

- Spring beans 些 Spring 主 java 们 Spring IOC , , 些beans 中 , 以XML 件中 义

3.2 个 Spring Bean 义 包含什么?

- 个Spring Bean 义 , 何 个bean,

3.3 何 Spring 供 元 ? Spring 几

- 三 Spring
 - XML 件
 - 于
 - 于java

3.4 Spring 件包含了 些信

- Spring 件 个XML 件, 个 件 了 , 了 何 们, 以 何 互

3.5 Spring 于xml 入bean 几

1. Set ;
2. :
3. index 位 ;
4. type ;
5. ;
6. ;

3.6 你 义 作 ?

- 义 个 Spring , 们 个bean 个作 以 bean 义中 scope 义 , Spring 产 个 bean , bean scope 为prototype , 个bean 个 , 个bean scope 为 singleton

3.7 Spring 几 bean 作

Spring 以下五 bean 作 :

- **singleton** : bean 个Spring ioc 中 个
- **prototype**: 个bean 义 以 个
- **request**: http 会 个bean, 作 仅 于web Spring ApplicationContext 下
- **session**: 个HTTP Session中, 个bean 义 个 作 仅 于web Spring ApplicationContext 下
- **global-session**: 个 HTTP Session中, 个bean 义 个 作 仅 于web Spring ApplicationContext 下

: Spring bean 作 Singleton prototype 作 , 为 bean 会

3.8 Spring 中 单例bean 全 吗?

- 不 , Spring 中 bean不
- spring 中 bean , spring bean
- 上 spring bean (dao) , 上 bean 也 , 但 bean (view model) , 了 , bean 作 , "singleton" 为"prototype", bean 于 new Bean()了, 以 以 了
 -
 - 不会

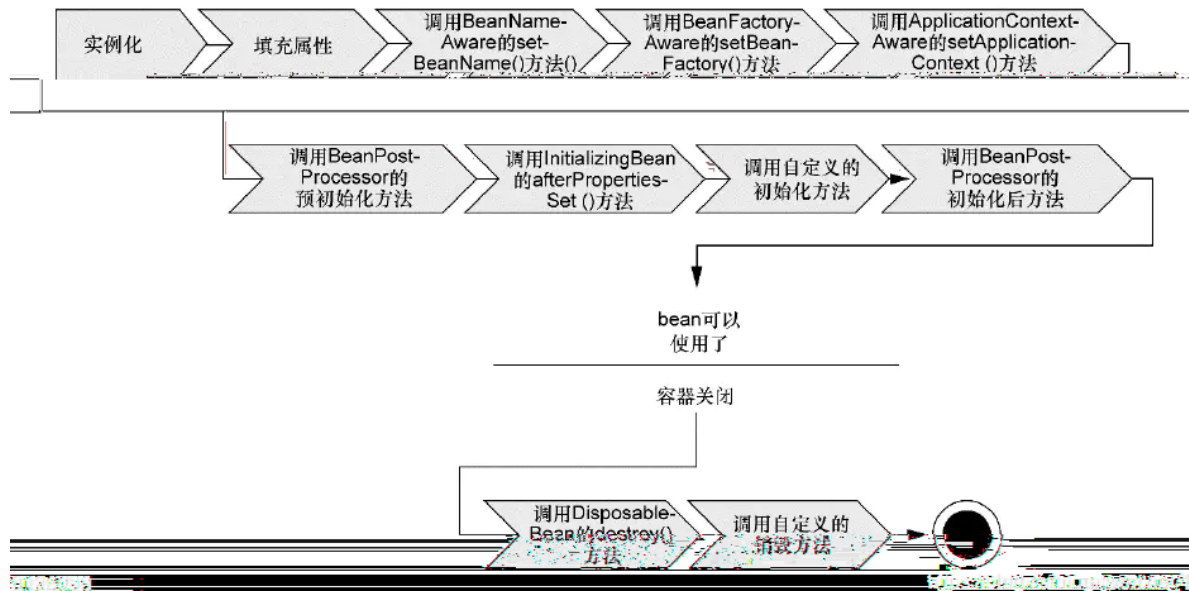
3.9 Spring 何 发 ?

- 下, Bean 以 下 享, Spring中, Bean 以 为singleton作 , 为Spring 些Bean中 ThreadLocal

- ThreadLocal 为了 中 了
“ ” , 仅 份 , 不 ,
ThreadLocal 了“ ”
- ThreadLocal会为 个 个 , 从 了 个
为 个 , 从 也 了 ThreadLocal
了 享 , 代 , 以 不 ThreadLocal

3.10 Spring 中bean 命周

- 传 Java 中, bean Java new bean ,
bean 以 了 bean不 , Java 之下, Spring
中 bean 了 Spring bean , 为
Spring 义bean 下 了bean Spring 上
下 中 个



- bean Spring 中从 了 , 以 Spring 何 bean
个
- , bean 之 , bean 了

们 上 :

- Spring bean ;
- Spring bean bean 中;
- bean 了BeanNameAware , Spring bean ID传 setBean-Name() ;
- bean 了BeanFactoryAware , Spring setBeanFactory() , BeanFactory 传 ;
- bean 了ApplicationContextAware , Spring setApplicationContext() , bean 上下 传 ;
- bean 了BeanPostProcessor , Spring 们 post-ProcessBeforeInitialization() ;
- bean 了InitializingBean , Spring 们 after-PropertiesSet() 似 , bean initmethod 了 , 也会 ;
- bean 了BeanPostProcessor , Spring 们 post-ProcessAfterInitialization() ;

- `bean` 以 `了` 们 上下 中,
- `bean` 了 `DisposableBean` , `Spring` `destroy()` , `bean` `destroy-method` 了 , 也会

Spring

Spring DI()

Spring

3.11 些 bean 命周 ? 你 们吗?

- 两个 bean , 一个 setup , 一个 bean teardown
- bean 两个 (init-method destroy-method) 们 以 们也 (@PostConstruct @PreDestroy)

3.12 什么 Spring 肉 bean? 什么 Spring inner beans?

- Spring 中, 一个bean仅 作 一个bean , 为 一个 bean
bean 以 setter " " " " , bean
, 们 Scope prototype

3.13 什么 bean ?

- , bean Spring 中 bean , bean
何 们

3.14 什么 bean 动 ?

- Spring 中, 件中 bean 个, Spring 互 作 bean, 不 , Bean bean之 作 Spring 以 Bean Factory中 bean之 以 个bean上, 也 以 bean上

3.15 不同 动 , spring 动 bean 些 ?

- spring中, 与 他 , 互 作
予 个 , autowire
- Spring xml 中 5 :
 - no: 不 , ref bean
- byName: bean , 个bean property与 bean name
,
- byType:
- constructor: , 且 byType

- autodetect: , , construct , , byType

3.16 使 @Autowired 动 ?

- @Autowired bean @Autowired 之 Spring 件 , <context:annotation-config />
- spring IOC , 了 个AutowiredAnnotationBeanPostProcessor , @Autowired @Resource @Inject , 会 IOC bean, @Autowired , 中 bean:
 - 为 个, bean @Autowired ;
- 不 个, 么@Autowired会 ;
 - 上 为 , 么会 , required=false

3.17 动 些 ?

- :
 - 写: 仍 义 ,
- : 不 , String 串,
 - : 不 ,

3.18 你可以 Spring中 入 个null 个 串吗?

- 以

4.Spring

4.1 什么 于Java Spring ? 些 例

- 于Java , Java 下, Spring XML 件
- 以@Configuration 为 , 以 个bean 义, Spring IOC
- 个 @Bean , 个 , 作为 个bean Spring 上 下

```
@Configuration
public class StudentConfig {
    @Bean
    public StudentBean myStudent() {
        return new StudentBean();
    }
}
```

4.2 启动？

- 一下不，为了，他们 Spring 件中
`<context:annotation-config/>`

4.3 @Component, @Controller, @Repository, @Service 何区别？

- @Component: java 为 bean 任何 Spring 件 spring 件中 以 中
- @Controller: 个 为 Spring Web MVC Bean 会 IOC 中
- @Service: 件 不会 @Component 任何 他 为 以 中 @Service 不 @Component, 为 以 了
- @Repository: 个 似 @Component 为 DAO 了 DAO IOC , 为 Spring DataAccessException

4.4 @Required 什么作

- 个 bean , 个bean 义 , @Required bean , BeanInitializationException :

```
public class Employee {  
    private String name;  
    @Required  
    public void setName(String name){  
        this.name=name;  
    }  
    public string getName(){  
        return name;  
    }  
}
```

4.5 @Autowired 什么作

- @Autowired , 下 (以 required 为false) @Autowired 了 , 何 以 何 个 @Required , setter 任 / 个 PN

```
public class Employee {
    private String name;
    @Autowired
    public void setName(String name) {
        this.name=name;
    }
    public String getName(){
        return name;
    }
}
```

4.6 @Autowired @Resource之区别

- @Autowired @Resource 于: Setter
- @Autowired @Resource之 于
 - @Autowired , 下 (以 required 为false)
 - @Resource , 不 与 bean 会

4.7 @Qualifier 什么作

- 个 bean 仅 中 个 bean , 以 @Qualifier @Autowired 个 bean 义

4.8 @RequestMapping 什么 ?

- @RequestMapping 于 HTTP 中 / 于两个 :
 - : URL
 - : URL 以 HTTP

5.Spring

5.1 /关

- Spring ORM , 们 JDBC之上 个 / (ORM) , Spring 主 ORM , Hibernate, JDO MyBatis, JPA, TopLink, JDO, OJB Spring 事 以上 ORM JDBC

5.2 Spring 中 何 使 JDBC?

- Spring JDBC , 代价 会 以 statements queries从 , JDBC也 以 Spring 下 , 个 JdbcTemplate

5.3 JDBC DAO

- JDBC DAO , 代 , , 不 之上, 了 个 Spring AOP Spring 中 事

5.4 spring DAO 什么 ?

- Spring DAO () JDBC, Hibernate JDO 以 作 久 之 代 , 不

5.5 spring JDBC API 中 些 ?

- JdbcTemplate
- SimpleJdbcTemplate
- NamedParameterJdbcTemplate
- SimpleJdbcInsert
- SimpleJdbcCall

5.6 JdbcTemplate 什么

- JdbcTemplate 了 , 作 , 义

5.7 使 Spring 什么 Hibernate? 使 Spring Hibernate 些?

- Spring中 两 Hibernate:
 - Hibernate
 - HibernateDAOSupport AOP

5.8 何 HibernateDaoSupport Spring Hibernate 合 ?

- Spring SessionFactory LocalSessionFactory 三 :
 - the Hibernate SessionFactory
 - HibernateDaoSupport 个DAO
 - AOP 事 中

5.9 Spring 事务，spring 事务 些？

- Spring 两 事：
 - 事务：事，，但
 - 事务：以业代事，XML 事

5.10 Spring 事务 原

- Spring 事，spring 事，binlog redo log 事交

5.11 下Spring 事务传 为

spring 事 传 为，个事，spring 何 些事 为

- ① PROPAGATION_REQUIRED: 事，个事，事，事，
- ② PROPAGATION_SUPPORTS: 事，事，事，不事，以事
- ③ PROPAGATION_MANDATORY: 事，事，事，不事，
- ④ PROPAGATION_REQUIRES_NEW: 事，不事，事
- ⑤ PROPAGATION_NOT_SUPPORTED: 以事作，事，事
- ⑥ PROPAGATION_NEVER: 以事，事，
- ⑦ PROPAGATION_NESTED: 事，事，事，REQUIRED

5.12 下 spring 事务 ？

- spring 五，为 ISOLATION_DEFAULT ()，他个：
 1. ISOLATION_DEFAULT: 什么 什么；
 2. ISOLATION_READ_UNCOMMITTED: 交，低 事 交，他事 (会 不)；
 3. ISOLATION_READ_COMMITTED: 交，个事 交 他事 (会 不)，SQL server ；
 4. ISOLATION_REPEATABLE_READ: 个，事 事 交 (会)，MySQL ；

5. ISOLATION_SERIALIZABLE: , 代价 ,

不

- : 个事 个事 中 交 , 个事 A, 事 交, 个事 了 A
- 不可 : 个事 ,
- : 个事 不 个事 A n 也 , 但 二 件下 n+1 , 产 了 个事 了, 个事 了 个事 , 个

5.13 Spring 事务 些优 ?

- 为不 事 API JTA, JDBC, Hibernate, JPA JDO, 个不
- 为 事 了 API 不 些 事 API
- 事
- Spring

5.14 你 倾向 事务 ?

- Spring 事 , 为 代 , 个 事 事 优于 事 , 事 (代 事) 了 不 , 作 , 事 以作 代

6.Spring 向切 (AOP)

6.1 什么 AOP

- OOP(Object-Oriented Programming) , 义 , 但 于 义 , 了 代 , 不 于 个
- AOP(Aspect-Oriented Programming), 为 , 作为 , 于 些与业 , 但 个 产 为 , 为 个 , 个 为" " (Aspect) , 中 代 , 低了 , 了 于 事

6.2 Spring AOP and AspectJ AOP 什么区别? AOP 些 ?

- AOP 于代 , AOP代 主 为 代 代 代 代 为 AspectJ; 代 以Spring AOP为代
 - (1) AspectJ 代 , 代 , AOP 会 AOP代 , 也 为 , 他会 AspectJ() Java 中, 之 AOP

- (2) Spring AOP 代理，代理 AOP 不会，
中临时为，一个AOP，一个AOP了，且
了，

6.3 JDK动态代理 CGLIB动态代理 区别

- Spring AOP中代理主要有两种，JDK代理 CGLIB代理：
 - JDK代理代理，不代理 InvocationHandler Proxy
， InvocationHandler invoke() 中代理，
业；， Proxy InvocationHandler 一个
， 代理
 - 代理 InvocationHandler，么Spring AOP会 CGLIB代理
CGLIB (Code Generation Library)，一个代，以
一个，中代，从 AOP
CGLIB 代，一个为final，么
CGLIB 代
- 代理与代理于AOP代理不，AspectJ代理
，但 AspectJ，Spring AOP

InvocationHandler invoke(Object proxy,Method method,Object[] args): proxy
代理；method 代理 个体；args 代理 个体
，

6.4 下Spring AOP 几个名词

- (1) (Aspect)：定义了
Spring AOP中，以 (于) 中以 @AspectJ
- (2) (Join point)：，Spring AOP中，一个代一个
以 些为 中
一个 一个以 一个代以
些 之中，为
- (3) (Advice)：AOP 中，作为
- (4) (Pointcut)：义会 一个 一个 们
， 义 些
- (5) (Introduction)：们
- (6) (Target Object)：一个 一个 (aspect) (advise)
个代也人 (advised) Spring AOP
代，一个 一个代 (proxied)
- (7) (Weaving)：代
一个以：
 - ：AspectJ 以
 - ：JVM，以
 - 之 AspectJ5 以
 - ：一个 下，，AOP 会为
个代 SpringAOP 以

6.5 Spring

- 代码中，Spring bean 中代码了，bean 之，会
- 代码 bean，Spring 代码 ApplicationContext，ApplicationContext 从 BeanFactory 中 bean，Spring 会代码为 Spring 代码，以们不 SpringAOP

6.6 Spring 只 别

- 为 Spring 于代码，以 Spring Spring，且

6.7 Spring AOP 中，关键 区别 什么？ spring aop 中 concern cross-cutting concern 不同之

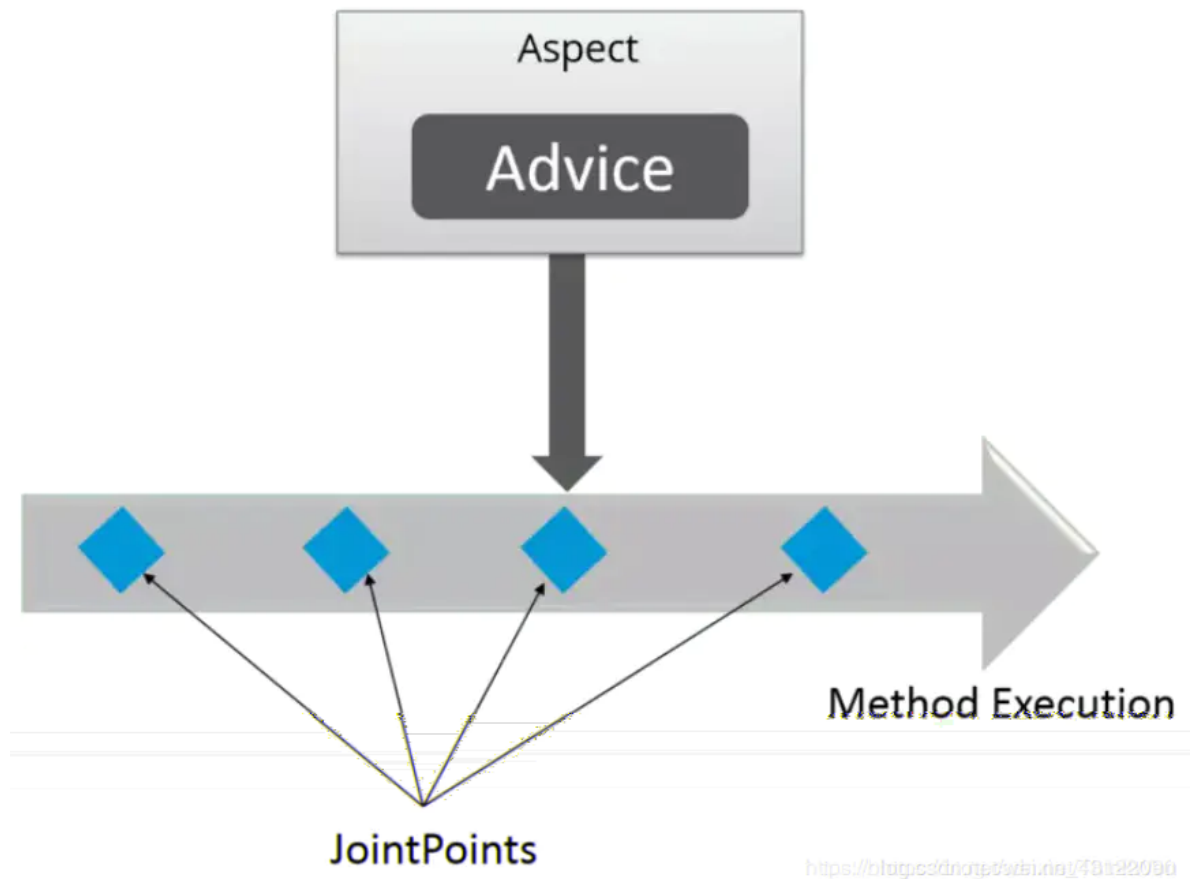
- (concern) 中个为，个会义个们个
- (cross-cutting concern) 个，个会些，于

6.8 Spring 些 ？

- AOP 中，作为，上 SpringAOP 代码
- Spring 以 5：
 1. (Before)：之；
 2. (After)：之，不会什么；
 3. (After-returning)：之；
 4. (After-throwing)：；
 5. (Around)：了，之之义为

6.9 什么 切 Aspect?

- aspect pointcut advice，了义，也了义。Spring AOP 中，AOP 作于何上，两个作：
 - 何 pointcut advice 位 joinpoint 上
 - 何 advice 中代码。
- 以 为，@Aspect。



6.10 于XML Schema 切

- 下, 以 于XML

6.11 于 切

- 下(于@AspectJ), 与 java5 java