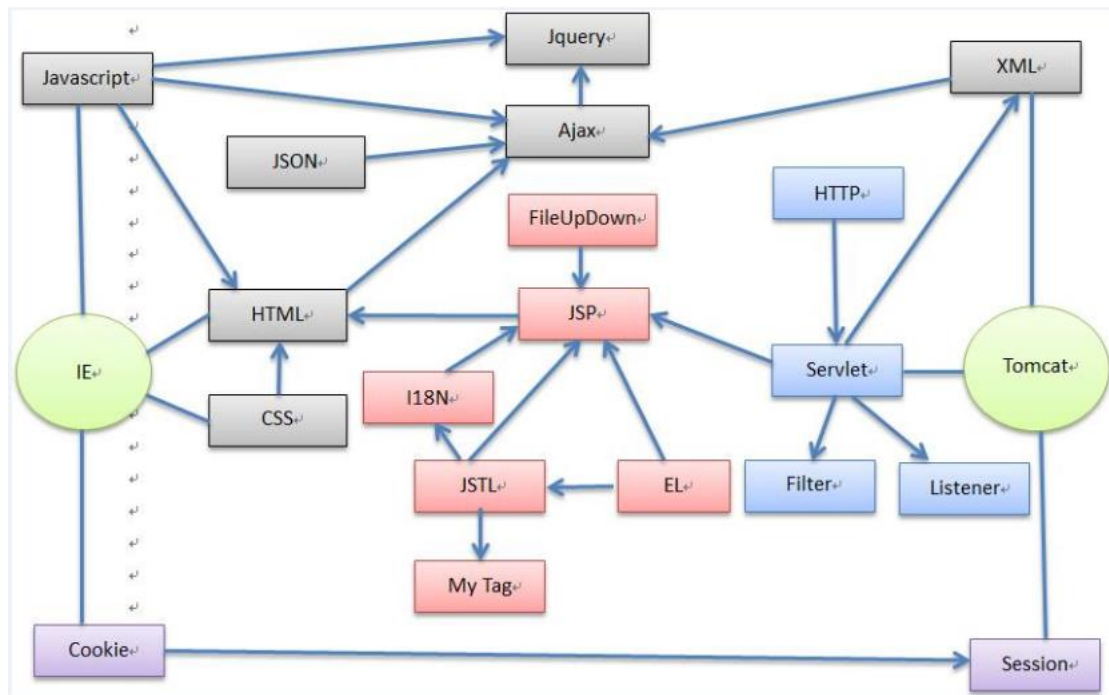


JavaWeb面试题

1.Javaweb技术的结构

1.1. Javaweb技术结构图



1.2.结构图说明:

整体分为四个部分:

- 1.黑线: JavaScript相关技术路线
- 2.蓝线: Servlet相关技术路线
- 3.红线: Jsp相关技术路线
- 4.紫线: Web会话相关技术路线

2.JavaScript相关技术路线(黑线)

此部分包括: JavaScript, JQuery, Ajax, XML, JSON和 HTML等技术.

2.1.列举 BOM中常用的几个全局变量和全局方法?

全局对象: window

全局变量 : document location history navigator screen

全局方法 : alert() confirm() prompt() open() close()

2.2.在 js中如何创建一个对象?

```
var p1 = {name: "Tom", "my age" : 12};

function Person(name, age) {
    this.name = name;
    this.age = age;
}
var p2 = new Person("Jack", 14);
```

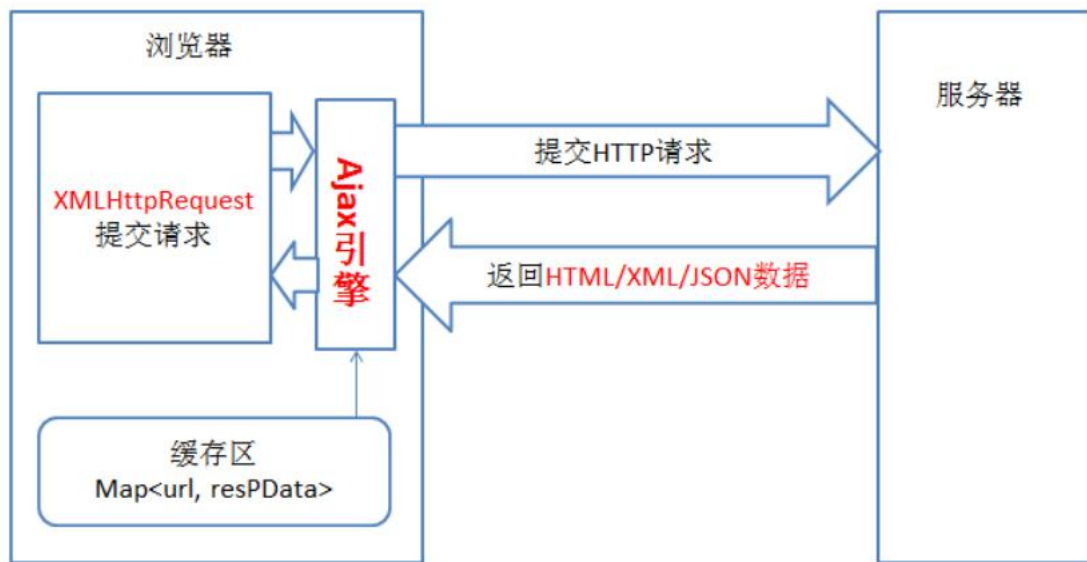
2.3.在 js中如何得到对象的属性?

```
var age = p2.age
//alert(age);
age = p1["my age"];
alert(age);
```

2.4.谈谈 Ajax技术

Ajax原理

Ajax的原理简单来说通过 XMLHttpRequest对象来向服务器发异步请求，从服务器获得数据，然后用 javascript来操作 DOM而更新页面的局部显示。



Ajax的优点：

- 1.最大的一点是页面无刷新，给用户的体验非常好。
- 2.使用异步方式与服务器通信，不需要打断用户操作，具有更加迅速的响应能力。
- 3.ajax的原则是“按需取数据”，最大程度的减少冗余请求，减少服务器的负荷。

Ajax的缺点：

- 1.破坏浏览器后退按钮的正常行为。在动态更新页面后，用户无法回到前一个页面的状态。
- 2.使用 JavaScript作 Ajax的引擎，JavaScript的兼容性和 Debug本身就让人头大。

Ajax的应用场景：

- 1.文本输入提示（自动完成）的场景(注册)
- 2.对数据进行联动过滤的场景(三级联动)

2.5.你觉得 jquery有哪些好处？

jQuery是轻量级的 javascript框架

强大的选择器

出色的 DOM操作的封装

可靠的事件处理机制

完善的 ajax封装

出色的浏览器的兼容性

支持链式操作，隐式迭代

支持丰富的插件

jquery的文档也非常的丰富

2.6. jquery对象和 dom对象如何转换？

1. jquery转 DOM对象：

jQuery对象是一个数组对象，可以通过[index]的丰富得到 DOM对象还可以

通过 `get[index]`去得到相应的 DOM对象。

2. DOM对象转 jQuery对象:

`$(DOM对象)`

2.7. jquery中\$.get()提交和\$.post()提交的区别？

1. `$.get()`方法使用 GET方式提交请求,而`$.post()`使用 POST方式。
2. GET方式传输的数据大小不能超过 2KB而 POST要大的多
3. GET方式请求的数据会被浏览器缓存起来, 因此有安全问题。

2.8. \$(document).ready()方法和 window.onload区别？

答:两个方法有相似的功能, 但是在实行时机方面是有区别的。

- 1 `window.onload`方法是在网页中所有的元素(包括元素的所有关联文件)完全加载到浏览器后才执行的。
- 2 `$(document).ready()`方法可以在 DOM载入就绪时就对其进行操纵, 并调用执行绑定的函数。

2.9. xml有哪些解析技术?区别是什么?

答:有 DOM,DOM4j,SAX,PULL等

DOM:一次性将整个文档加载到内存中,生成一个对象树,在处理大型文件时其性能下降的非常厉害。

DOM4J:对 DOM的进一步封装, API使用更简洁

SAX:基于事件驱动的方法回调机制。每读取一小部分数据时就会回调事件处理器对象的方法,但解析一旦开始就不能停止。

PULL:也是基于事件驱动,只是需要手动控制读取下一部分数据,这样得到想要的
数据后就可以停止解析.

2.10.你在项目中用到了 xml技术的哪些方面?如何实现的?

答:用到了数据存贮, 信息配置两方面。在做数据交换平台时, 将不能数据源的数据组装成 XML文件, 然后将 XML文件压缩打包加密后通过网络传送给接收者, 接收解密与解压缩后再同 XML文件中还原相关信息进行处理。在做软件配置时, 利用 XML可以很方便的进行, 软件的各种配置参数都存贮在 XML文件中。

2.11.说说你对 JSON的理解

JSON(JavaScript Object Notation)是一种轻量级的数据交换格式。它基于标准 JavaScript的一个子集,是一个 Js对象或数组结构的字符串

JSON有三类数据

1.单个数据

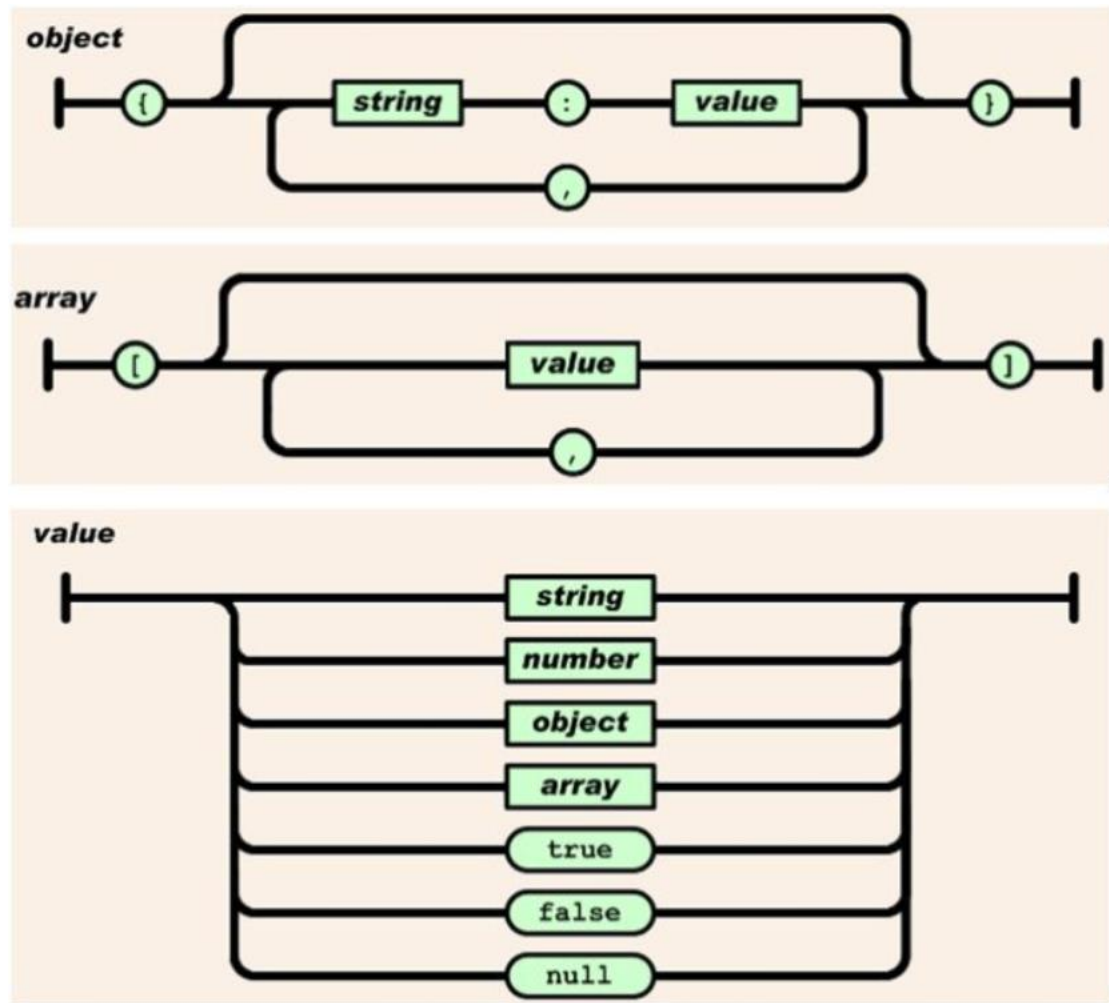
有 number, string, boolean和 null四种类型数据

2.多个有序的数据:数组

用[]包含起来,其元素可以是三类数据中的任意一种,元素之间用,号隔开

3.多个无序的数据:对象

用{}包含起来,其元素必须由 key-value组成, key是一个字符串, value可以是任意类型数据, key与 value之间用:号隔开,两个 key-value之间用,号隔开.



3.Servlet相关技术路线(蓝线)

此部分包括: Servlet, Filter, Listener和 HTTP协议

3.1.解释一下什么是 **servlet**?

答:我们可以从下面二个方面去看 Servlet:

1. API:有一个接口 Servlet,它是 Servlet规范中定义的用来处理客户端请求的程序需要实现的顶级接口
2. 组件:服务器端用来处理客户端请求的组件,需要在 web.xml请求中配置

3.2.说一说 Servlet的生命周期?

答: Servlet生命周期分为三个阶段:

- 1, 初始化阶段调用 `init()`方法
- 2, 响应客户请求阶段 调用 `service()`方法-`doGet/doPost()`
- 3, 终止阶段 调用 `destroy()`方法

3.3.区别请求的转发与重定向?

答:可以从以下三个方面进行比较

1.地址栏:

转发:显示的是请求的 URL

重定向:显示的不是请求的 URL,而是重定向指向的新的 URL

2.浏览器发了几次请求?

转发: 1次请求

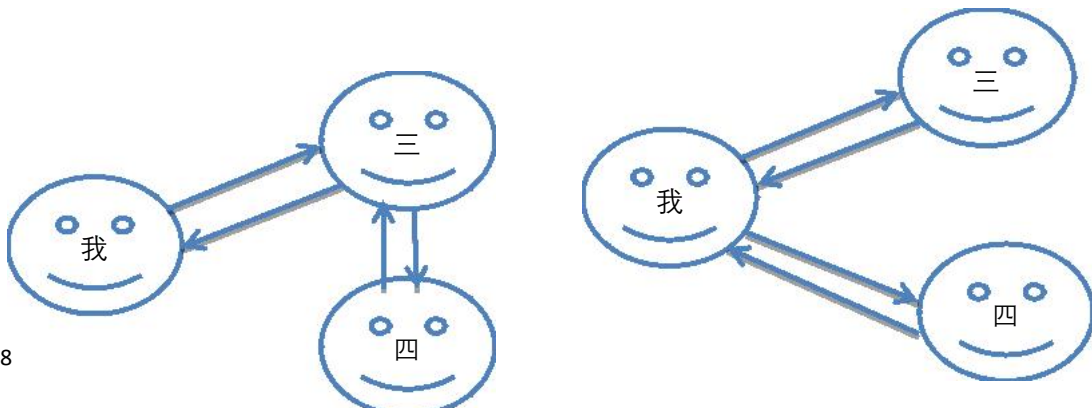
重定向: 2次请求

3.是否可以共享 Request的数据共享?

转发:两个资源之间是同一个 request对象,可以共享 request中的数据

重定向:两个资源之间不是同一个 request对象,不可以共享

经典现实案例:



3.4. HTTP请求的 GET与 POST方式的区别

答:可以从以下几个方面去回答:

1.携带请求参数的方式

GET:通过请求行携带参数,参数会显示在地址栏

POST:通过请求体来携带参数,参数不会显示在地址栏

2.服务器端处理请求的方法

GET:会调用 Servlet的 doGet()来处理请求

POST:会调用 Servlet的 doPost()来处理请求

3.数据大小与安全性

GET:大小有限制(小于 2k),不安全

POST:大小没有限制,安全

3.5.比较一下 Servlet与 Filter

1. Filter是一种特别的 Servlet,它们的作用是完全不一样的. Servlet是用来处理请求的,而 Filter是用来过滤检查请求的.

经典现实案例:

假如我们要去坐地铁去天安门,我们需要先在检票机上刷票后才能进

站坐上地铁,请求问:在这个实际业务中,哪个是

Servlet?哪个是

Filter呢?

4. Jsp相关技术路线(红线)

此部分包括: JSP, EL, JSTL, My Tag, I18N, FileUpDown

3.1. jsp有哪些内置对象?作用分别是什么?

答:JSP共有以下 9个内置的对象：

request:客户端请求，此请求会包含来自 GET/POST请求的参数

response:网页传回客户端的回应

pageContext:网页的属性是在这里管理

session:与请求有关的会话期

application:与当前应用对应的 ServletContext对象,应用中只有一个

out:用来传送回应的输出 {}<%= %>

config:与 jsp配置对象的对象,一般无用

page: jsp对应的 Servlet对象

exception:针对错误网页，未捕捉的异常对象

3.2. jsp有哪些动作?作用分别是什么?

答:JSP共有以下 6种基本动作

jsp:include：在页面被请求的时候引入一个文件。

jsp:forward：把请求转到一个新的页面。

jsp:useBean：寻找或者实例化一个 JavaBean。

jsp:setProperty：设置 JavaBean的属性。

jsp:getProperty：输出某个 JavaBean的属性。

jsp:plugin：根据浏览器类型为 Java插件生成 OBJECT或 EMBED标记

3.3. JSP的常用指令

答:主要有下面 3种指令

1. page指令:指定页面的的一些属性,常用属性:

`contentType="text/html; charset=utf-8" //向浏览器端输出数据的编码`

`pageEncoding="utf-8" //jsp文件被编译成 java文件时所用的编码`

`session="true" //是否自动创建 session`



2. include指令:包含别一个 jsp页面
3. taglib指令:引入一个标签库

3.4. JSP中动态 INCLUDE与静态 INCLUDE的区别?

答:

- 1.动态包含:用<jsp:include>,包含的动作是在 jsp对应的 Servlet处理请求时去执行的,每次请求都会执行.
- 2.静态包含:用 include指令,包含的动作是在 jsp被编译成 java文件时执行的,只有第一次请求时执行.

4.5. JSP和 Servlet有哪些相同点和不同点, 他们之间的联系是什么?

答:

JSP的优点是擅长于网页制作，生成动态页面比较直观，缺点是不容易跟踪与排错。

Servlet是纯 Java语言，擅长于处理流程和业务逻辑，缺点是生成动态网页不直观。

3.5. EL的功能,为什么要用 EL?

EL的功能包括:

- 1.从四个域对象中取出属性数据显示
- 2.取出请求参数数据显示

为什么要用 EL?

在页面中用 jsp脚本和 jsp表达式来获取数据显示比较麻烦

- 1.需要条件判断
- 2.可能需要强转

3.6. JSTL的功能,为什么要用 JSTL?

JSTL的功能

JSTL全名为 JavaServer Pages Standard Tag Library,主要用于基本输入输出、流程控制、循环、XML文件剖析、数据库查询及国际化和文字格式标准化的应用等

为什么要用 JSTL?

在 jsp页面做条件判断或循环操作并输出时,比较费力

3.7.为什么要用自定义标签?, MyTag如何实现?

为什么要用?

- 1.不想在 Jsp中编写 java代码
2. JSTL标签库不能满足实际项目的需求

自定义标签定义和使用的流程

- 1.编写标签处理器类(SimpleTagSupport的实现类)

a)重写 doTag()

- 2.编写标签库文件(WEB-INF/xxx.tld)

a)整个文件的定义: <short-name> <uri>

b)标签的定义: <tag>

- 3.在 jsp页面使用标签:

a)导入标签库(xxx.tld/)

b)使用标签

5. Web会话相关技术路线(紫线)

此部分包括: Cookie和 Session技术

5.1.说说你对 Cookie与 Session技术的理解?

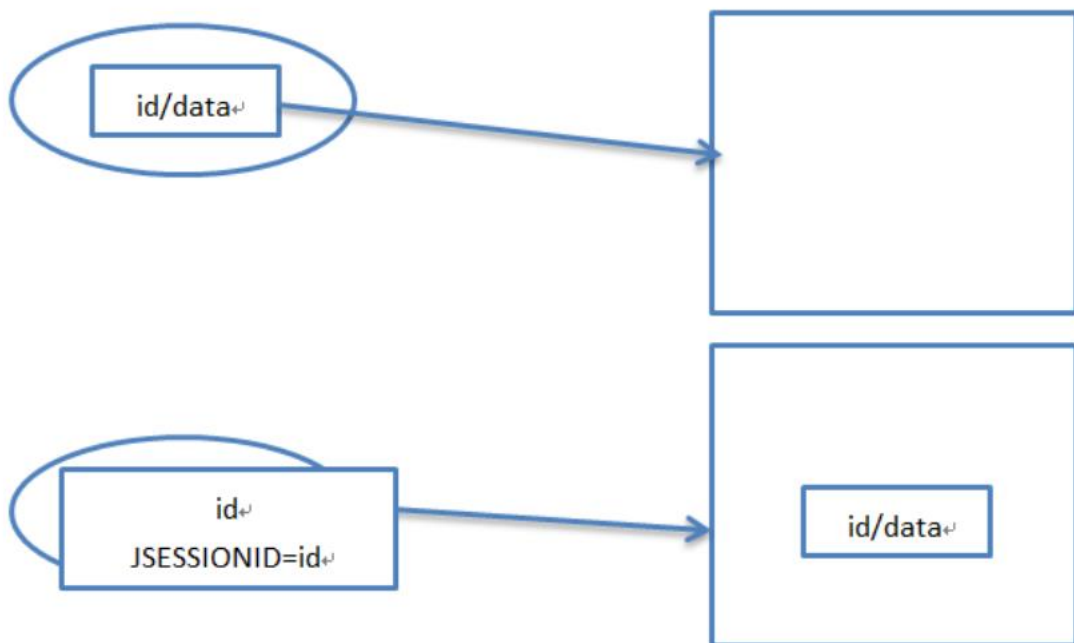
1. cookie是一种浏览器端的缓存技术,而 Session是一种服务器端的缓存技术(依赖 cookie)

经典现实案例：

某咖啡厅推出了一个优惠活动：累计喝五杯咖啡可以免费赠送一杯。他们该如何实现呢？

方法一:咖啡厅办卡(id,count),交给消费者,消费者下次再来消费时,必须带上卡,消费一次由咖啡厅来更新卡上的数据,再次交给消费者

方法二:咖啡厅办卡(id), id和 count都保存在咖啡厅的电脑中的表中,将卡(id)交给消费者;消费者下次再来消费时,必须带上卡,消费一次由咖啡厅来更新表中的数据,再次交给消费者



5.2.说说自动登陆功能的编码实现?

- 1.登陆功能是用 Session实现的,就是向 Session对象中保存当前用户的对象
- 2.自动的功能用 Cookie实现,就是登陆时将用户的信息保存为持久化 Cookie
- 3.下次访问时,读取请求中如果有用户信息的 Cookie就可以自动登陆

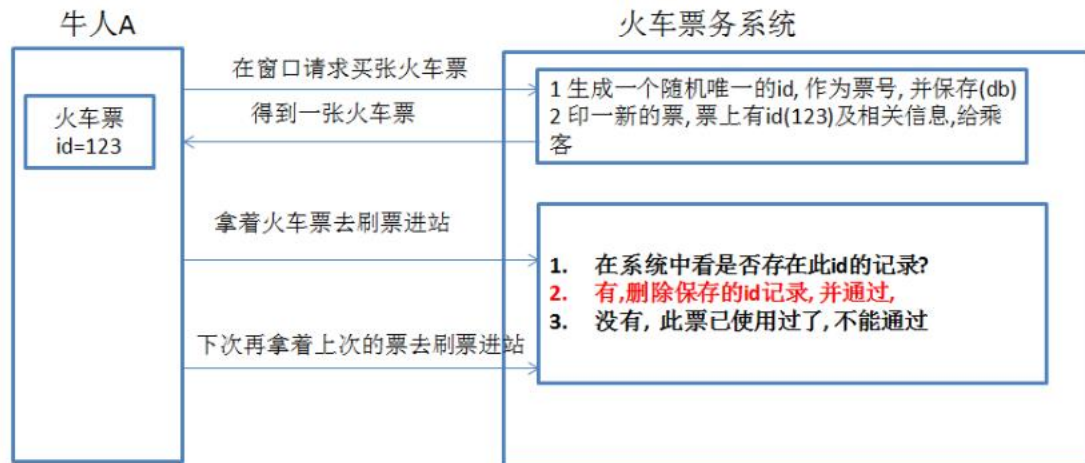
5.3.如何防止表单重复提交?

答:使用 Session技术:

- 1.在 regist.jsp页面中生成一个唯一随机值,将其保存到 Session中,同时将其保存为表单的隐藏域的值
- 2.在处理注册的请求时,获取 Session中值,获取请求参数的值,比较两者是否相同,如果相同说明不是重复提交,请求通过同时删除 session中保存的值,如果不相同则是重复提交,不能通过.

经典现实案例:

一位乘客在北京火车站买了一张去天津的火车票(直接刷的那种),他刷票进站坐火车去了天津,回来后过了几天,他又需要去天津这次他不想再买票,直接拿上次的票去进站口刷,检票机提示“此火车票已使用过了”,不能进站.



6.其它

此部分包括: MVC, Webservice和 Mybatis

6.1. MVC的各个部分都有那些技术来实现?如何实现?

答: MVC是 Model – View – Controller的简写。

Model代表的是应用的业务逻辑（通过 JavaBean, EJB组件实现），

View是应用的表示面（由 JSP页面产生），

Controller是提供应用的处理过程控制（一般是一个 Servlet），

通过这种设计模型把应用逻辑，处理过程和显示逻辑分成不同的组件实现。这些组件可以进行交互和重用。

6.2. WEB SERVICE相关名词解释

Web Service

Web Service是基于网络的、分布式的模块化组件，它执行特定的任务，遵守具体的技术规范，这些规范使得 Web Service能与其他兼容的组件进行互操作。

JAXM(Java API for XML Messaging)

是为 SOAP通信提供访问方法和传输机制的 API。

WSDL:

是一种 XML格式，用于将网络服务描述为一组端点，这些端点对包含面向文档信息或面向过程信息的信息进行操作。这种格式首先对操作和消息进行抽象描述，然后将其绑定到具体的网络协议和消息格式上以定义端点。相关的具体端点即组合成为抽象端点（服务）。

SOAP:

即简单对象访问协议(Simple Object Access Protocol)，它是用于交换 XML编码信息的轻量级协议。

6.3. WebService技术的本质是使用哪几种技术实现的？

HTTP + XML + Schema

6.4.如何编码发布一个 WebService？

- 1.定义 SEI:使用@Webservice和@Webmethod
- 2.定义 SEI的实现类:使用@Webservice
- 3.发布:使用 JDK中的 Endpoint,或者使用 CXF框架基于 Spring的配置来发布

6.5.如何编码请求一个 WebService？

- 1.根据 wsdl文档生成客户端代码.
- 2.利用客户端代码编写调用 webservice的代码.

6.6.比较一下 JDBC, dbutils, Mybatis和 Hibernate

1. JDBC:原生访问数据库的方式,其它三个都是对 JDBC不同程度的包装

访问数据库比较麻烦,代码重复度极高

2. dbutils:是对 jdbc进行了相对简单的包装,主要就是能自动封装查询结构集,需要在代码中写 sql语句
3. Mybatis:进一步封装 jdbc, Sql语句写在配置文件中,面向对象操作,有一二级缓存功能
4. Hibernate:对 jdbc封装得最彻底的框架,纯面向对象,可以不用写 SQL