

1 问题:

struts2 在 float double 两类类型中, 前台和后台采用 get set 方法接收设置 0 时, 会报错。

解决方法:

自定义类型转换。

1: Src 目录下加上 xwork-conversion.properties 文件:

```
java.lang.Double=com.utils.DoubleConvert
java.lang.Float=com.utils.DoubleConvert
```

2: domain 的 javaBean 里面的 float 改为大写 Float

3: 加入工具类 DoubleConvert 转换的内容:

```
package com.utils;
import java.util.Map;
import org.apache.struts2.util.StrutsTypeConverter;
public class DoubleConvert extends StrutsTypeConverter {
    @Override
    public Object convertFromString(Map arg0, String[] arg1, Class arg2) {
        // TODO Auto-generated method stub
        if(Double.class==arg2){
            String str = arg1[0];
            if(str.equals("")){
                str="0";
            }
            return Double.parseDouble(str);
        }
        else if (Float.class==arg2){
            String str = arg1[0];
            if(str.equals("")){
                str="0";
            }
            return Float.parseFloat(str);
        }else if(Integer.class==arg2){
            String str= arg1[0];
            if(str.equals("")){
                str="0";
            }
            return Integer.parseInt(str);
        }
        return 0;
    }
}
```

```

    }
    @Override
    public String convertToString(Map arg0, Object arg1) {
        // TODO Auto-generated method stub
        return arg1.toString();
    }
}

```

2 编程经验：空的考虑

在从数据库读数据(采用 HQL 语言)时，在从 session 里面读存储的值时，一定考虑当读出的数据量是空的情况，结合 if else 解决。

3: 问题:

登陆速度比较慢（在部署 linux 服务器后 不能进入主体页面）。

这是由 struts 或 hibernate 的 xml 标签引用了 www 网的资源 而服务器端限制了防火墙

解决方法:

将 struts 或 hibernate 的 xml 标签改成引用本地标签库

4: 编程经验 easyui 不能显示

Easyui 中的 javascript 代码太多，多出逗号符号等 严格检查

5: 编程经验 integer 与 int

Integer 是对象，int 是基本数据类型。

Integer 不能直接等于 int。而是 Integer.equals(int)

7: 问题：easyui 中兼容性问题(模板加载影子问题)

在一个 jsp 含有多个 div(部件)时，且第一次加载页面会有影子现象产生。

解决方法:

```
<body class="easyui-layout" style="visibility: hidden;">
```

body 页面采用隐藏，适当时候(datagrid 加载成功时候)显示

8: 问题:数据库写插入语句时 自动增加的 ID 什么时候可以不写

解决办法:

```
insert into 表(属性列表) values(值列表)
```

无需写 id 会自动增加

9: dom4j 解析 xml 问题(xmpp)

```
Packet copyPacket = packet.createCopy();
```

```
(1) IQ iq = (IQ) copyPacket;//获得 IQ 包
```

(2) //获得 IQ 下面的子元素

```
iq.getChildElement()
```

//获得 IQ 下子元素的子元素的属性

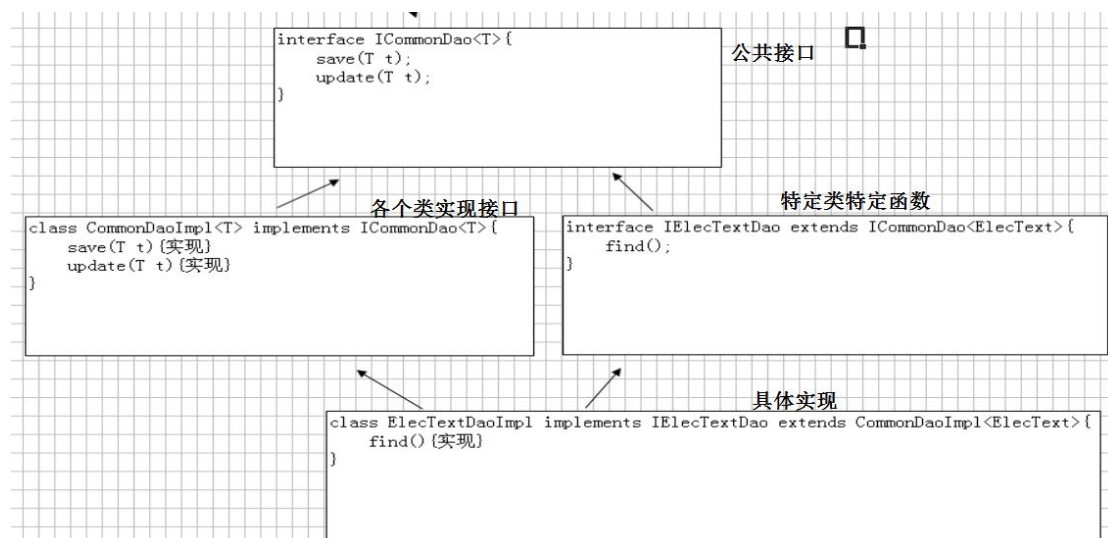
```
iq.getChildElement().element("file").attributeValue("name")
```

```
<iq id="E47Am-45" to="2013220605418i8058/Spark 2.6.3" from="2013210603188i8058/Spark 2.6.3" type="set">
  <si xmlns="http://jabber.org/protocol/si" id="jsi_3667021831713276868" profile="http://jabber.org/protocol/si/profile/file-transfer">
    <file>
      <desc>Sending file</desc>
    </file>
    <feature xmlns="http://jabber.org/protocol/feature-neg">
      <x xmlns="jabber:x:data" type="form">
        <field var="stream-method" type="list-single">
          <option>
            <value>http://jabber.org/protocol/bytestreams</value>
          </option>
          <option>
            <value>http://jabber.org/protocol/ibb</value>
          </option>
        </field>
      </x>
    </feature>
  </si>
</iq>
```

10: 一个标准的 JavaBean 应该具有如下几个特点

- 1、必须是一个公开的类，类访问权限必须是 public 的。
- 2、必须具有一个无参数的构造方法。如果定义了自定义的有参构造方法，就必须添加一个无参数构造方法，否则将无法设置属性，如果没有定义自定义的有参构造方法，则可以利用编译器自动添加的无参构造方法。
- 3、一般将属性设置成私有的，使用 getXXX() 方法和 setXXX() 方法来进行属性的设置。

11: dao 层框架



12: HibernateDaoSupport 类的使用

1. 继承了 HibernateDaoSupport 类的类获取 session 时，已不可用 SessionFactory.OpenSession 的形式来获取 Session 了，由于 HibernateDaoSupport 本身已有获取 session 的方法 getSession(), 所以直接用 Session se=this.getSession(); 来获取，

2. 在依据 hql 获取用户信息时, 继承了 HibernateDaoSupport 类的类中不能使用 Query 类了, 而是用 `List<Ssh> list = this.getHibernateTemplate().find(hql);` 形式来获取实体类集合

13:PO 对象 VO 对象区别

PO: persistent object 持久对象, 可以看成是与数据库中的表相映射的 java 对象。最简单的 PO 就是对应数据库中某个表中的一条记录, 多个记录可以用 PO 的集合。PO 中应该不包含任何对数据库的操作

VO: value object 值对象。通常用于业务层之间的数据传递, 和 PO 一样也是仅仅包含数据而已。但应是抽象出的业务对象, 可以和表对应, 在 web 上传递。

14:ModelDriven

实现了 ModelDriven 就必须实现 getModel 这个方法, 该方法返回的是你的 ModelDriven 泛型中设定的实体类。其实是为了实现代码的重用, 不要再定义一次你在数据访问层定义的属性了, 直接用那个类就行了

15:JQuery 中使用 Ajax 赋值给全局变量异常解决方案

在用 JQuery 的 Ajax 从后台提取数据后想把它赋值给全局变量, 但是却怎么都赋不进, 为什么呢?

我们用的 Ajax 是异步操作, 也就是说在你赋值的时候数据还没提取出来, 你当然赋不进去, 所以只要改成同步操作就行~。

方法 1: 直接用 \$.ajax

```
$.ajax({
    type : "post",
    url :地址,
    data : "参数" + 参数的值,
    async : false,
    success : function(data) {
        //赋值给全局变量;
    }
});
```

16:json 对象转换成 json 字符串

```
var temp = json.str;
value = eval('(' + temp + ')');
Json 字符串转 json 对象
value = eval("(" + json + ")");
```

17:hibernate 排序问题

问题：用 hibernate 排序，如果 order by 非主键 asc 当数据量超过几十条 返回的数据出错（Hibernate 分页功能数据重复问题）

原因：

oracle 中的分页机制通过 rownum 函数来实现的，rownum 顺序号的生成是排序后生成的

例如：select * from web_user where rownum > 0 and rownum <= 10 order by name
这个排序中使用了 name 字段作为排序，但是数据不能保证 name 是唯一的，所以就会出现重复数据。当然如果将 name 设定为唯一索引那么就可以了

解决办法：所以如果 oracle 中需要分页排序的话，那么就要使用唯一键排序，
如果使用了外键 hql 和 sql 语句就应写成：order by 外键，本表主键

18: java.lang.NoSuchMethodError:

javax.persistence.OneToOne.orphanRemoval()Z

原因及解决办法：

在 hibernate3.5 之前，使用 Annotation 几个 jar 包：

hibernate Annotation 的包：hibernate-annotations.jar

持久化：ejb3-persistence.jar

javax.persistence.OneToOne.class 在 ejb3-persistence.jar 中。

Hibernate3.5 以及之后的版本使用 Annotation 只需加

hibernate-jpa-2.0-api-1.0.0.Final.jar

javax.persistence.OneToOne.class 在 hibernate-jpa-2.0-api-1.0.0.Final.jar

19:hibernate 查询数据表 char 类型字段只返回一个字符

问题描述：

hibernate 查询数据表 char 类型字段只返回一个字符。在 oracle 或 sqlServer 下会遇到，查询数据表字段类型为 char，如果使用 Hibernate 原生 SQL 去执行查询，往往只会返回一个字符。原因是 char 在 java 中只有两个字节。

解决办法：

1: select cast(列 as varchar2(列大小)) from 表;或者改为 select trim(列)
from 表

2. 指定列返回的类型。

SQLQuery query = session.createSQLQuery();

query.addScalar(columnAlias, Type);

20:JUnit 测试时，Hibernate 懒加载报错

问题描述：

Spring 等都有正常加载，一访问懒加载的属性就报 no Session。目前用的 openSessionInViewFilter 将 session 的活动周期延长到 view，而默认的 hibernateTemplate 是用完就关的。

解决办法：

要继续用 JUnit 测试只能自己构造一个 SessionFactory，详细可以参考上面那个链接，这里不再赘述。

21AOP 时 java.lang.ClassCastException: \$Proxy14 cannot be cast

问题描述： Spring AOP 部分使用 JDK 动态代理或者 CGLIB 来为目标对象创建代理。如果被代理的目标实现了至少一个接口，则会使用 JDK 动态代理。所有该目标类型实现的接口都将被代理。若该目标对象没有实现任何接口，则创建一个 CGLIB 代理

解决办法： 如果用 JDK 动态代理，就必须为被代理的目标实现一个接口

```
BuildDepartTreeService buildDepartTreeServiceImpl = (BuildDepartTreeService)ac.getBean("buildDepartTreeServiceImpl");
buildDepartTreeServiceImpl.listBuildService(1);
```

必须是接口 不能是实现类

如果使用 CGLIB 强制代理，就必须事先将 CGLIB 包导入项目，设 beanNameAutoProxyCreator 的 proxyTargetClass 属性 为 true。