Struts国际化

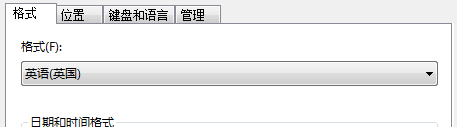
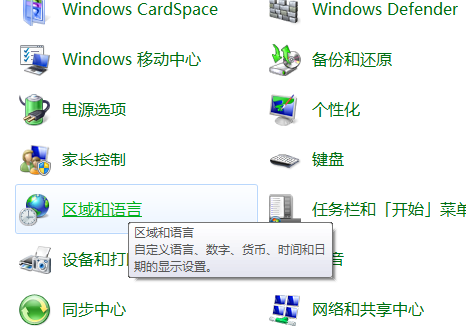
1. java国际化

国际化（I18N）:

Java util包中两个类： 1： locale 代表国家与语言的

2： ResourceBundle 取国际化资源文件

修改操作系统的语言与区域：



修改以后进入的网站就是修改后的语言

Java项目：

1.Test.java:

**public** **class** Test {

**public** **static** **void** main(String[] args) {

// Locale locale = new Locale("ja","JP");

Locale locale = **new** Locale("zh","CN");

//Locale locale = new Locale("en","US");

//Locale.getDefault();//去虚拟机上的语言

//System.out.println(locale.getCountry());//返回此语言环境的国家/地区代码

ResourceBundle rb = ResourceBundle.*getBundle*("MessageResource", locale);//取得properties文件

System.*out*.println(rb.getString("k1"));

MessageFormat mf = **new** MessageFormat(rb.getString("k1"));

mf.format(**new** Object[]{"admin"});//必须为Object的数组

System.*out*.println(mf.format(**new** Object[]{"admin","kjk"}));

}

}

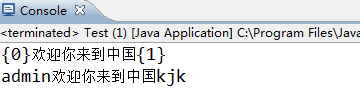
1. MessageResource\_zh\_CN.properties文件

k1=**{0}**\u6B22\u8FCE\u4F60\u6765\u5230\u4E2D\u56FD**{1} 中文自动编码**

k1=welcome you chain!

{0}占位符，每个语言相通，用MessageFormat来填充占位符

3.结果：



2、Struts国际化：

静态国际化与动态国际化：

整体配置 MessageResource\_zh\_CN.properties文件

MessageResource\_en\_US.properties 文件

login.username = username

login.password = password

login.submit = submit

login.succes = **{0}**login succes!

login.found = user**{0}** is not fount

login.error =password error\!

1.<struts-config>

必须加这样的配置：才能自动接收properties

<message-resources parameter=*"MessageResource"*></message-resources>

2.login.jsp:

用<bean:message key=*"login.username"*/>来接收参数

<form action=*"login.do?command=login"* method=*"post"*>

<bean:message key=*"login.username"*/>： <input type=*"text"* name=*"username"* /> <br>

<bean:message key=*"login.password"*/>： <input type=*"text"* name=*"password"* /> <br>

&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<input name=*"submit"* type=*"submit"* value=*"*<bean:message key=*"login.submit"*/>*"* />

</form>

3.UtilAction:



改文字语言的格式：

**public** ActionForward changLunage(ActionMapping mapping, ActionForm form,

HttpServletRequest request, HttpServletResponse response) **throws** Exception {

DynaActionForm daf = (DynaActionForm)form;

String lan = daf.getString("lan");

**if**("en".equals(lan)){

// request.getSession().setAttribute(Globals.MESSAGE\_KEY, new Locale("en","US"));

**this**.setLocale(request, **new** Locale("en","US"));

}**else** **if**("zh".equals(lan)){

// request.getSession().setAttribute(Globals.MESSAGE\_KEY, new Locale("zh","CN"));

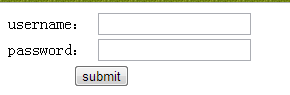
**this**.setLocale(request, **new** Locale("zh","CN"));

}

**return** mapping.findForward("login");

}

结果：

**动态改变语言：**

4.UserAction

**public** ActionForward login(ActionMapping mapping, ActionForm form,

HttpServletRequest request, HttpServletResponse response) **throws** Exception {

DynaActionForm daf = (DynaActionForm)form;

String username = daf.getString("username");

String password = daf.getString("password");

ActionMessages ams = **new** ActionMessages();

ActionMessage am = **new** ActionMessage("login.succes",username);

ActionErrors ae = **new** ActionErrors();

String path = "msg";

**try**{

UserServiceImpl.*getStance*().login(username, password);

}**catch** (UsernameNotFormException e){

am = **new** ActionMessage("login.found",username); //使用这个可以占位符，username

}**catch** (PasswordErrorEcxeption e){

am = **new** ActionMessage("login.error"); }

ae.add("msg",am);

ams.add("msg", am);

**this**.addMessages(request, ae);//传递给页面

**this**.addMessages(request, ams);

**return** mapping.findForward(path); }

5.msg.jsp

<html:messages id=*"msg"* message=*"true"* >

${msg }

</html:messages>

<html:errors/>//显示错误信息

迭代出保存在ActionMessages上文字

Message=”true” 就是可以显示出ActionMessages信息，而struts1默认为ActionError信息，但是ActionError除了可以格式标准输出有用，其余还是用ActionMessages最好

预定读取：properties=”a”预定读取