目录

[1. 修改默认样式 2](#_Toc452469474)

[2. 修改【样式更改】方法 2](#_Toc452469475)

[3. 扩展dwz.navTab.js，增加getCurrentTabId方法，获取当前navTab的tabid，具体方法实现如下 2](#_Toc452469476)

[4. 扩展dwz.navTab.js，增加getTitle方法，获取当前navTab的title，具体方法实现如下 3](#_Toc452469477)

[5. 修改dwz.navTab.js的\_reload方法，解决JUI刷新乱码的问题 3](#_Toc452469478)

[6. 扩展rel属性值，增加【currentNavTab】关键字 4](#_Toc452469479)

[7. 扩展navTabAjaxDone返回值属性扩展 6](#_Toc452469480)

[8. 扩展dwz.navTab.js，新增loadUrl方法 6](#_Toc452469481)

[9. 扩展dwz.ui.js的[target=dialog]，为标记出此dialog是由哪个按钮打开的做准备 7](#_Toc452469482)

[10. 扩展dwz.dialog.js，为pdialog增加openedObj属性，标记出此dialog是由哪个按钮打开的。 8](#_Toc452469483)

[11. 扩展A标签，增加before属性，在A标签的页面跳转前执行before中的方法 9](#_Toc452469484)

[12. 增加submitForm方法，用于提交表单数据 11](#_Toc452469485)

[13. 增加在Table布局中错误信息的提示方式 13](#_Toc452469486)

# 修改默认样式

将首页（index.jsp）的如下部分

<link href=*"../../frame/themes/default/style.css"* rel=*"stylesheet"* type=*"text/css"* media=*"screen"*/>

改为

<link href=*"../../frame/themes/azure/style.css"* rel=*"stylesheet"* type=*"text/css"* media=*"screen"*/>

# 修改【样式更改】方法

修改【dwz.theme.js】文件的【setTheme】方法

将

$("head").find("link[href$='style.css']").attr("href",\_themeHref.replace("#theme#", themeName));

改为

$("head").find("link[href$='style.css']").attr("href","../../frame/"+\_themeHref.replace("#theme#", themeName));

# 扩展dwz.navTab.js，增加getCurrentTabId方法，获取当前navTab的tabid，具体方法实现如下

//anlb新增

getCurrentTabId:function(){

return this.\_getTabs().eq(this.\_currentIndex).attr("tabid");

}

# 扩展dwz.navTab.js，增加getTitle方法，获取当前navTab的title，具体方法实现如下

//anlb新增

getTitle:function(){

return this.\_getTabs().eq(this.\_currentIndex).find("span").text();

}

# 修改dwz.navTab.js的\_reload方法，解决JUI刷新乱码的问题

\_reload: function($tab, flag){

flag = flag || $tab.data("reloadFlag");

var url = $tab.attr("url");

if (flag && url) {

$tab.data("reloadFlag", null);

var $panel = this.getPanel($tab.attr("tabid"));

if ($tab.hasClass("external")){

navTab.openExternal(url, $panel);

}else {

//获取pagerForm参数

var $pagerForm = $("#pagerForm", $panel);

var args = $pagerForm.size()>0 ? $pagerForm.serializeArray() : {}

//修改前

//$panel.loadUrl(url, args, function(){navTab.\_loadUrlCallback($panel);});

//anlb修改后

$panel.ajaxUrl({

type:"POST", url:url, data:args, callback:function(response){

navTab.\_loadUrlCallback($panel);

}

});

}

}

}

# 扩展rel属性值，增加【currentNavTab】关键字

在rel属性的值中加入【currentNavTab】关键字，含义为在当前navTab中点击超链接时，不重新打开新的navTab，刷新本身navTab。

在dwz.ui.js的JS中更新如下代码

// navTab

$("a[target=navTab]", $p).each(**function**(){

$(**this**).click(**function**(event){

**var** $this = $(**this**);

**var** title = $this.attr("title") || $this.text();

**var** tabid = $this.attr("rel") || "\_blank";

//anlb增加

**if**(tabid=="currentNavTab") {tabid=navTab.getCurrentTabId();title=navTab.getTitle();};

**var** fresh = eval($this.attr("fresh") || "true");

**var** external = eval($this.attr("external") || "false");

**var** url = unescape($this.attr("href")).replaceTmById($(event.target).parents(".unitBox:first"));

DWZ.debug(url);

**if** (!url.isFinishedTm()) {

alertMsg.error($this.attr("warn") || DWZ.msg("alertSelectMsg"));

**return** **false**;

}

navTab.openTab(tabid, url,{title:title, fresh:fresh, external:external});

event.preventDefault();

});

});

# 扩展navTabAjaxDone返回值属性扩展

扩展ajaxtdod回调函数的返回值，navTabId增加currentNavTab关键字，当navTabId：currentNavTab时刷新本身页面

if (json.navTabId){ //把指定navTab页面标记为需要“重新载入”。注意navTabId不能是当前navTab页面的

//anlb修改

if(json.navTabId=="currentNavTab")

navTab.reloadFlag(navTab.getCurrentTabId());

else

navTab.reloadFlag(json.navTabId);

} else { //重新载入当前navTab页面

var $pagerForm = $("#pagerForm", navTab.getCurrentPanel());

var args = $pagerForm.size()>0 ? $pagerForm.serializeArray() : {}

navTabPageBreak(args, json.rel);

}

# 扩展dwz.navTab.js，新增loadUrl方法

具体实现如下：

//anlb新增

loadUrl:function(url){

var $panel=this.\_getPanels().eq(this.\_currentIndex);

this.\_getTabs().eq(this.\_currentIndex).attr("url",url);

$panel.loadUrl(url, {}, function(){navTab.\_loadUrlCallback($panel);});

}

# 扩展dwz.ui.js的[target=dialog]，为标记出此dialog是由哪个按钮打开的做准备

$("a[target=dialog]", $p).each(function(){

$(this).click(function(event){

var $this = $(this);

var title = $this.attr("title") || $this.text();

var rel = $this.attr("rel") || "\_blank";

var options = {};

var w = $this.attr("width");

var h = $this.attr("height");

if (w) options.width = w;

if (h) options.height = h;

options.max = eval($this.attr("max") || "false");

options.mask = eval($this.attr("mask") || "false");

options.maxable = eval($this.attr("maxable") || "true");

options.minable = eval($this.attr("minable") || "true");

options.fresh = eval($this.attr("fresh") || "true");

options.resizable = eval($this.attr("resizable") || "true");

options.drawable = eval($this.attr("drawable") || "true");

options.close = eval($this.attr("close") || "");

options.param = $this.attr("param") || "";

//anlb增加

options.clickBtn=$this;

var url = unescape($this.attr("href")).replaceTmById($(event.target).parents(".unitBox:first"));

DWZ.debug(url);

if (!url.isFinishedTm()) {

alertMsg.error($this.attr("warn") || DWZ.msg("alertSelectMsg"));

return false;

}

$.pdialog.open(url, rel, title, options);

return false;

});

});

# 扩展dwz.dialog.js，为pdialog增加openedObj属性，标记出此dialog是由哪个按钮打开的。

jDContent.ajaxUrl({

type:options.type||'GET', url:url, data:options.data || {}, callback:function(){

jDContent.find("[layoutH]").layoutH(jDContent);

$(".pageContent", dialog).width($(dialog).width()-14);

$("button.close").click(function(){

$.pdialog.close(dialog);

return false;

});

//anlb增加

$.pdialog.openedObj=options.clickBtn;

}

});

# 扩展A标签，增加before属性，在A标签的页面跳转前执行before中的方法

**使用方法：**

<a class="edit" href="../demo/listDemo1!inputEdit.action?id={id}" target="navTab" rel="currentNavTab" before="a(this);"><span>修改</span></a>

**使用效果：**

在页面跳转到../demo/listDemo1!inputEdit.action?id={id}前，执行js函数a(this)

**实现方法：**

修改dwz.ui.js的如下代码

$("a[target=navTab]", $p).each(function(){

$(this).click(function(event){

var $this = $(this);

//anlb增加

var beforeClickEvent=$this.attr("before");

if(beforeClickEvent)

eval(beforeClickEvent);

var title = $this.attr("title") || $this.text();

var tabid = $this.attr("rel") || "\_blank";

//anlb增加

if(tabid=="currentNavTab") {tabid=navTab.getCurrentTabId();title=navTab.getTitle();};

var fresh = eval($this.attr("fresh") || "true");

var external = eval($this.attr("external") || "false");

var url = unescape($this.attr("href")).replaceTmById($(event.target).parents(".unitBox:first"));

DWZ.debug(url);

if (!url.isFinishedTm()) {

alertMsg.error($this.attr("warn") || DWZ.msg("alertSelectMsg"));

return false;

}

navTab.openTab(tabid, url,{title:title, fresh:fresh, external:external});

event.preventDefault();

});

});

# 增加submitForm方法，用于提交表单数据

**使用方法：**

$(function(){

$("#saveBtn").click(function(){

if($("#userName").val()==""){

alertMsg.error("请输入工号");

return;

}

submitForm($("#frm"),navTabAjaxDone);

})

})

**使用效果：**

直接提交FORM表单

**实现方法：**

在dwz.ajax.js中增加如下代码

/\*\*

\* 普通ajax表单提交

\* anlb增加

\* @param {Object} form

\* @param {Object} callback

\* @param {String} confirmMsg 提示确认信息

\*/

function submitForm(callback, confirmMsg){

var $form=navTab.getPanel(navTab.getCurrentTabId()).find("FORM");

var \_submitFn = function(){

$form.find(':focus').blur();

$.ajax({

type: $form[0].method || 'POST',

url:$form.attr("action"),

data:$form.serializeArray(),

dataType:"json",

cache: false,

success: callback || DWZ.ajaxDone,

error: DWZ.ajaxError

});

}

if (confirmMsg) {

alertMsg.confirm(confirmMsg, {okCall: \_submitFn});

} else {

\_submitFn();

}

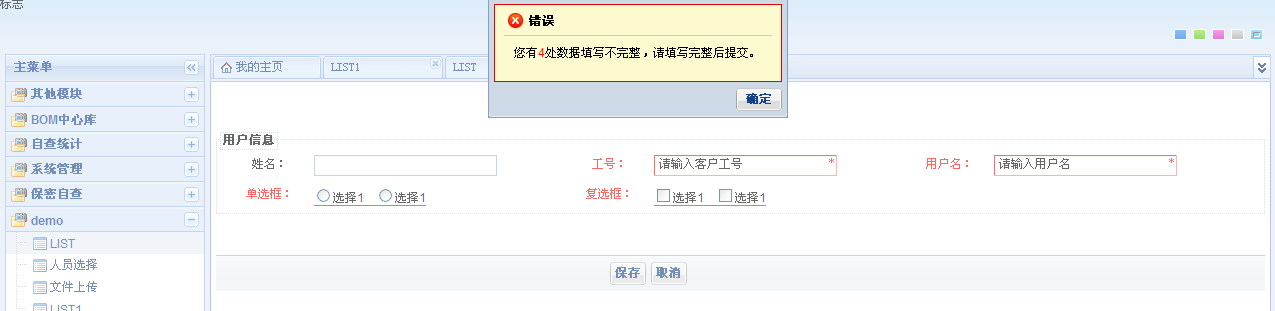
# }更改系统中的验证逻辑，废除JUI框架中自带的表单验证功能

**使用方法：**

使用方法与JUI之前的使用方法相同，在<form>标签的class中增加required-validate

**使用效果：**

点【提交】或【保存】按钮后提示【您有2处数据填写不完整，请填写完整后提交。】，同时文本框及文字变成红色



**实现方法：**

1、在dwz.ajax.js中增加/修改如下代码：

/\*\*

\* anlb增加

\* @memberOf {TypeName}

\*/

function validInit(){

//为INPUT框添加事件

$("[type=text].required",".required-validate").blur(function(){

if($(this).val()!=""){

$(this).removeClass("validError");

$(this).parent().prev().removeClass("validError");

$(this).attr("validError","false");

}else{

$(this).addClass("validError");

$(this).parent().prev().addClass("validError");

$(this).attr("validError","true");

}

})

$("span.required",".required-validate").blur(function(){

if($(":checked",$(this)).length==0){

$(this).addClass("validError");

$(this).parent().prev().addClass("validError");

$(this).attr("validError","true");

}

})

$("span.required",".required-validate").click(function(){

$(this).removeClass("validError");

$(this).parent().prev().removeClass("validError");

$(this).attr("validError","false");

})

}

/\*\*

\* anlb新增

\* @param {Object} $form

\* @return {TypeName}

\*/

function \_valid($form){

validInit();

var $element=$(".required",$form);

var errors=0

var i;

for(i=0;i<$element.length;i++){

errors+=\_validProxy($element.eq(i));

}

if(errors!=0){

alertMsg.error("您有<font color='red'>"+errors+"</font>处数据填写不完整，请填写完整后提交。");

return false;

}else

return true;

}

/\*\*

\* anlb新增

\* @param {Object} $element

\* @return {TypeName}

\*/

function \_validProxy($element){

$element.blur();

if($element.attr("validError")=="true"){

return 1;

}else{

return 0;

}

}

/\*\*

\* 普通ajax表单提交

\* @param {Object} form

\* @param {Object} callback

\* @param {String} confirmMsg 提示确认信息

\*/

function validateCallback(form, callback, confirmMsg) {

var $form = $(form);

if (!\_valid($form)) {

return false;

}

var \_submitFn = function(){

$form.find(':focus').blur();

$.ajax({

type: form.method || 'POST',

url:$form.attr("action"),

data:$form.serializeArray(),

dataType:"json",

cache: false,

success: callback || DWZ.ajaxDone,

error: DWZ.ajaxError

});

}

if (confirmMsg) {

alertMsg.confirm(confirmMsg, {okCall: \_submitFn});

} else {

\_submitFn();

}

return false;

}

2、在style的css中增加如下样式

/\* anlb增加 \*/

.edit\_table th,.edit\_table td{padding: 5px;margin: 3px;}

.edit\_table th.validError,dt.validError{color: red;}

.edit\_table input.validError,.edit\_table textarea.validError,dd input.validError,dd textarea.validError { border-color:#F80C11 #FB7E81 #FB7E81 #F80C11;}

.edit\_table span.validError{border-bottom: 1px solid #F80C11;}