

Nutz 未添加

1.*****

一组 IocBean 的配置信息，这个JS文件的格式通常为

```
{  
    "对象名称": {  
        type : "对象类型",  
        parent : "被继承的对象名称",  
        events : {  
            fetch : "触发器的类型或者函数名",  
            create : "触发器的类型或者函数名",  
            depose : "触发器的类型或者函数名"  
        },  
        args : [  
            参数1, 参数2 ...  
        ],  
        fields : {  
            "字段名称1": 字段值1,  
            "字段名称2": 字段值2,  
            ...  
        },  
        factory : "xxx.yyy.zzz#create" //工厂类#工厂方法  
    }  
}
```

2.*****

用户查询

```
//查询单个用户--登录  
@At  
@Ok("json")  
@Filters()  
public Object doLogin(@Param("uid")String uid, @Param("pwd") String pwd,  
    @Param("role")String role, @Param("bver") String bver,  
    HttpSession session,HttpServletRequest req){  
    NutMap re = new NutMap();  
    User user = dao.fetch(User.class, uid);  
    if (user == null || !user.getRole().equals(role)) {  
        re.put("ok", false);  
        re.put("msg", "考号不存在");  
        return re;  
    }  
    String p = lrwCode(pwd, "");  
    if (!p.equals(user.getPwd())) {  
        re.put("ok", false);  
    }
```

```
re.put("msg", "密码不正确");
return re;
}

////登录成功后...记录登录日志
Loginlog log = new Loginlog();
log.setUid(user.getId());
log.setLoginip(getIpAddr(req));
log.setLogintime(new Date());
log.setBver(bver);
dao.insert(log);
session.setAttribute("me", user);
////登录成功后...不同角色将请求跳转到不同页面
if(role.equals("5")) {
    re.put("ok", true);
    re.put("msg", "user/goIndex");
    return re;
}
else if(role.equals("1")) {
    re.put("ok", true);
    re.put("msg", "user/goAdmin");
    return re;
}
else {
    re.put("ok", false);
    re.put("msg", "系统错误！");
    return re;
}
}

//查
@At
@Ok("beetl:web/userlist.html")//跳转到用户信息列表
public void goList(HttpServletRequest session, HttpServletResponse req){
}

@At
@Ok("raw")//分页查询指定用户或所有用户
public String listUser(@Param("page") int curPage, @Param("rows") int pageSize,
                      @Param("s_name") String s_name, HttpSession session) {
    Criteria cri = Cnd.cri();
    if (!Strings.isBlank(s_name)) {
        cri.where().andLike("uid", s_name).orLike("xm", s_name);
    }
    else cri.where().andEquals("1", 1);
    cri.getOrderBy().asc("uid");
    return listPageJson(dao, User.class, curPage, pageSize, cri);
}
```

```

}

@At
@Ok("beetl:web/index.html")//考生登录成功，跳转到考试信息页面
public void goIndex(HttpServletRequest session,HttpServletResponse req){
    User loginuser=(User)session.getAttribute("me");
    req.setAttribute("user", loginuser);
}

@At
@Ok("beetl:web/admin.html")//管理员登录成功，跳转到管理信息页面
public void goAdmin(HttpServletRequest session,HttpServletResponse req){
    User loginuser=(User)session.getAttribute("me");
    req.setAttribute("user", loginuser);
}

@At
@Ok(">>:/")//登出系统
public void doLogout(HttpServletRequest session) {
    session.invalidate();
}

///查看登录日志
@At
@Ok("beetl:web/loglist.html")
public void goLog() {
}

@At
@Ok("raw")//分页查询所有登录日志
public String listLog(@Param("page") int curPage,@Param("rows") int pageSize) {
    Cnd cnd=(Cnd) Cnd.where("1","=", "1").desc("logintime");
    return listPageJson(dao, Loginlog.class, curPage, pageSize, cnd);
}

```

3.*****

数据封装成json

```

@At
@Ok("raw")
public String menutree(HttpServletRequest req, HttpSession session) {
    User user = (User) session.getAttribute("me");
    String role=user.getRole();
    //父级菜单
    List<Cmenu> menulist=dao.query(Cmenu.class, Cnd.where("id", "like", "%").asc("id")); //id=01,02,03
    List<EasyUITree> eList = new ArrayList<EasyUITree>();
    if(menulist.size() != 0){
        for (int i = 0; i < menulist.size(); i++) {
            Cmenu t = menulist.get(i);
            if(!t.getPermission().contains(role))continue; //无权限,不显示相应菜单
    }
}
```

```

EasyUITree e = new EasyUITree();
e.setId(t.getId());
e.setText(t.getName());
List<EasyUITree> eList2 = new ArrayList<EasyUITree>();
List<Cmenu> menu2 = dao.query(Cmenu.class, Cnd.where("id", "like", t.getId() + "__").asc("id") );
for (int j = 0; j < menu2.size(); j++) {//二级菜单
    Cmenu t2 = menu2.get(j);
    if(!t2.getPermission().contains(role))continue;
    Map<String, Object> attributes = new HashMap<String, Object>();
    attributes.put("url", t2.getUrl());
    attributes.put("role", t2.getPermission());
    EasyUITree e1 = new EasyUITree();
    e1.setAttributes(attributes);
    e1.setId(t2.getId());
    e1.setText(t2.getName());
    e1.setState("open");
    eList2.add(e1);
}
e.setChildren(eList2);
e.setState("closed");
eList.add(e);
}
}
return Json.toJson(eList);
}

```

4.*****

DataGrid数据封装

为了能让前端easyui datagrid获取正确的数据，需要在后端从数据库按查询条件cnd获取数据后，再次封装成包含分页curPage、pageSize的JSON格式数据。

```

public <T> String listPageJson(Dao dao, Class<T> obj, int curPage, int pageSize, Condition cnd) {
    Map<String, Object> jsonobj = new HashMap<String, Object>();
    Pager pager = dao.createPager(curPage, pageSize);
    List<T> list = dao.query(obj, cnd, pager);
    pager.setRecordCount(dao.count(obj, cnd)); //记录数需手动设置
    jsonobj.put("total", pager.getRecordCount());
    jsonobj.put("rows", list);
    return Json.toJson(jsonobj);
}

```

用户信息处理（前端）

能够查询显示用户信息userlist、添加新用户useradd、删除用户和修改用户信息useredit的前端页面。后台管理，基本上采用了easyui控件，所以网页中引用了相关的css和js文件，其中\${ctxPath}为beetl模板中定义的全局变量ContextPath，例如“/exam”，它不包含末尾的“/”，所以在使用时，需要在其后补充：

```

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<link rel="stylesheet" type="text/css" href="${ctxPath}/include/css/themes/default/easyui.css">

```

```

<link rel="stylesheet" type="text/css" href="${ctxPath}/include/css/themes/icon.css">
<script type="text/javascript" src="${ctxPath}/include/js/jquery.min.js"></script>
<script type="text/javascript" src="${ctxPath}/include/js/jquery.easyui.min.js"></script>
<script type="text/javascript" src="${ctxPath}/include/js/easyui-lang-zh_CN.js"></script>

```

后台布局

后台管理页 **admin.html** 采用 easyui 布局 + easyui 标签页，所以在网页中除了引用上述文件外，又引用了 **tabpage.js**。

布局中的 north 区域放置了系统 logo、登录用户名称、退出； west 区域加载树形菜单； south 区域显示版权信息； center 区域使用 tab 来展示内容窗口；暂时没有使用 east 区域。

```

<body class="easyui-layout">
    <div data-options="region:'north',border:false"
        style="background:#B3DFDA;padding:0 10px 0 10px;vertical-align: middle;">
        
        <div style="float:right;line-height:50px;margin-right:10px;">
            <a href="#" id="logout">退出</a>
        </div>
        <div style="float:right;line-height:50px;margin-right:10px;">登录用户： ${me.xm} |</div>
    </div>
    <div data-options="region:'west',split:true,title:'系统菜单'" style="width:180px;padding:10px;">
        <ul id="menutree" class="easyui-tree"></ul>
    </div>
    <div data-options="region:'south',border:false"
        style="height:30px;padding:5px;text-align:center;font-family: arial;
color: #A0B1BB;">Copyright © 2017 JavaEE. All Rights Reserved.
    </div>
    <div data-options="region:'center'">
        <div id="tabs" class="easyui-tabs" fit="true" border="false">
            <div title="" id="home">
                <iframe id="mainframe" name="mainframe" scrolling="auto"
                    height="99%" width="99%" frameborder="0" src=""></iframe>
            </div>
        </div>
    </div>
</body>

```

使用以下的js代码，在网页左侧(west)加载树形菜单，菜单数据来源于后台以json形式封装的层级菜单数据。如果点击一个菜单项，则在tab中打开相应的页面\$("#mainframe").attr("src", "\${ctxPath}/" + node.attributes.url)，并在标签上面显示相应的名称setTabName(node.text)；如果点击一个父级菜单，则收缩collapse其它打开的父级菜单，展开expand当前点击的菜单。

```

$("#menutree").tree(
{
    url : "${ctxPath}/menutree",
    animate : true,
    onClick : function(node) {
        if (!node.attributes) {
            if (!opened_node) {

```

```

        $("#" + menutree).tree('expand', node.target);
        opened_node = node.target;
    } else if (opened_node != node.target) {
        $("#" + menutree).tree('collapse', opened_node);
        $("#" + menutree).tree('expand', node.target);
        opened_node = node.target;
    }
} else {
    $("#mainframe").attr("src", "${ctxPath}/" + node.attributes.url);
    setTabName(node.text);
}
},
onLoadSuccess : function(node, data) {
    $("#" + menutree).tree('expandAll');
}
});

```

设置进入后台管理页面后， 默认打开的标签页：

```
setTabName("考生管理");
```

```
$("#mainframe").attr("src", "${ctxPath}/user/goList");
```

退出系统功能的实现：

```
$("#logout").click(function() {
    dlg_h(300, 140, "退出系统？", "您确认要退出系统吗？", dlg_ok1, null, "question");
})
```

```
function dlg_ok1() {
```

```
    top.location.href = "${ctxPath}/user/doLogout";
```

```
}
```

布局框架中隐藏了已经封装的两种形式的公用对话框，一种只有“确认”按钮，另一种有“确认”和“取消”按钮，以下代码封装在`dlg.html`页面文件中，在`admin.html`文件中引用它`<% include("dlg.html")%>`

```
<div id="dlg" style="padding:10px;font-size:14pt;" ></div>
```

```
<script type="text/javascript">
```

```
/*
```

```
message-icon:error,info,question,warning
```

```
*/
```

```
function dlg_h(w,h,t,content,fun_ok,fun_cancel,icon){
```

```
    if(icon)$("#dlg").html("<div class=\"message-icon message-"+icon+"\"></div>" + content);
```

```
    else $("#dlg").html(content);
```

```
    if(w>800)w=800;
```

```
    else w='auto';
```

```
    if(h<400)h='auto';
```

```
    $("#dlg").dialog({
```

```
        width:w, height:h, title:t,
```

```
        iconCls: 'icon-tip',resizable: false, modal: true, closed: true,
```

```
        buttons: [{
```

```
            text:'确认',iconCls:'icon-ok',
```

```

handler:function(){
    $('#dlg').dialog('close');
    if(fun_ok)fun_ok();
}
},{
    text:'取消',iconCls:'icon-cancel',
    handler:function(){
        $('#dlg').dialog('close');
        if(fun_cancel)fun_cancel();
    }
}
]);
$('#dlg').dialog("open");
$('#dlg').dialog('center');
}

function dlg_ok(w,h,t,content,fun_ok,icon){
    if(icon)$("#dlg").html("<div class=\"messenger-icon messenger-"+icon+"\"></div>"+content);
    else $("#dlg").html(content);
    if(w>800)w=800;
    else w='auto';
    if(h<400)h='auto';
    $('#dlg').dialog({
        width:w, height:h, title:t,
        iconCls: 'icon-tip',resizable: false, modal: true, closed: true,
        buttons: [
            {
                text:'确认',iconCls:'icon-ok',
                handler:function(){
                    $('#dlg').dialog('close');
                    if(fun_ok)fun_ok();
                }
            }
        ]
    });
    $('#dlg').dialog("open");
    $('#dlg').dialog('center');
}
}

</script>

```

用户列表（查询+删除）

userlist.html页面中用easyui datagrid显示列表数据，加载后端封装传输过来的分页json数据时，分析数据后显示“修改”、“删除”等操作信息；在页面的输入框中输入待查询的关键字（考号或姓名）可以进行模糊查询；可以删除指定的用户，删除之前必须提示确认；可以请求进入添加用户页面；可以请求进入修改指定用户的页面。

```

<table id="dg" cellpadding="2"></table>
<div id="tb" style="padding:5px;">
    <input id="s_name" class="easyui-textbox" data-options="prompt:'考号或姓名...'" style="width:200,height:32">

```

```

<a id="s_user" href="#" class="easyui-linkbutton" data-options="iconCls:'icon-search'">查询</a>
<a id="a_user" href="#" class="easyui-linkbutton" data-options="iconCls:'icon-add'">添加</a>
</div>
<script type="text/javascript">
var s_name="";
function loadGrid(){
    s_name=$("#s_name").val();
    $("#dg").datagrid({
        width:800,height:500,nowrap:false,
        striped:true,border:true,collapsible:false,
        url:"${ctxPath}/user/listUser",
        queryParams:{"s_name":s_name},
        pagination:true,
        rownumbers:true,
        fitColumns:true,pageSize:20,
        loadMsg:'数据加载中...' ,
        columns:[[
            {title:'学院', field:'department',width:100},
            {title:'考号', field:'uid',width:80},
            {title:'姓名', field:'xm',width:80},
            {title:'操作', field:'role',width:100, formatter: function(value,row,index){
                var p="

```

```

url:"${ctxPath}/user/doDel?uid="+uid,
type:"post",
success: function(res){
    if(res=="true"){
        parent.dlg_h(300,140,"删除成功","您已删除考生: "+uid,null,null,"info");
        uid="";
        s_name="";
        loadGrid();
    } else {
        parent.dlg_h(300,140,"删除失败",res,null,null,"error");
    }
    return false;
},
error:function(res){
    parent.dlg_h(300,140,"系统提示","系统错误",null,null,"error");
}
})
}

$(function(){
    loadGrid();
    $("#s_user").click(function(){
        s_name=$("#s_name").val();
        loadGrid();
    });
    $("#a_user").click(function(){
        parent.$("#mainframe").attr("src","${ctxPath}/user/goAdd");
        var tab=parent.$("#tabs").tabs("getSelected");
        parent.$("#tabs").tabs("update",{tab:tab,options:{title:"添加考生信息"}});
    });
    $("#tb").bind("keydown",function(e){
        var theEvent = e || window.event; // 兼容FF和IE和Opera
        var code = theEvent.keyCode || theEvent.which || theEvent.charCode;
        if (code == 13) { //回车执行查询
            $("#s_user").click();
        }
    });
});
}
</script>

```

用户添加

useradd.html页面中的表单采用了自动校验form('enableValidation').form('validate')，如果标记为**required:true**的项没填写，则在保存时会自动提示，并用不一样的颜色提醒；设定登录密码时，需要输入两次，以确认输入密码的一致性。保存成功后或者返回，则自动跳转到用户列表页。

```

<div class="easyui-panel" style="width:580px;padding:30px 60px">
<form id="adduser_form" method="post">
<div style="margin-bottom:20px">

```

```

<input id="department" name="department" class="easyui-textbox" data-options="prompt:'班级',required:true" style="width:100%;height:32px">
</div>
<div style="margin-bottom:20px">
<input name="xm" class="easyui-textbox" data-options="prompt:'姓名',required:true" style="width:100%;height:32px">
</div>
<div style="margin-bottom:20px">
<input id="uid" name="uid" class="easyui-textbox" data-options="prompt:'考号',required:true" style="width:100%;height:32px;">
</div>
<div style="margin-bottom:20px">
<div>登录密码:</div>
<input name="pwd" id="pwd" class="easyui-validatebox easyui-textbox" type="password" data-options="icons:[{iconCls:'icon-lock'}],required:true" style="width:100%;height:32px" >
</div>
<div style="margin-bottom:20px">
<div>确认密码:</div>
<input id="pwd2" validType="equals['#pwd']" class="easyui-validatebox easyui-textbox" type="password" data-options="icons:[{iconCls:'icon-lock'}]" style="width:100%;height:32px" >
</div>
<div style="margin-bottom:20px">
<div>角色:</div>
<input type="radio" name="role" value="5" checked="checked" />考生
<input type="radio" name="role" value="1" />管理员
</div>
</form>
<div style="text-align:center;">
<a id="saveuser" href="#" class="easyui-linkbutton" iconCls="icon-ok" style="width:132;height:32px">保存</a>
<a id="back" href="#" class="easyui-linkbutton" iconCls="icon-back" style="width:132;height:32px">返回</a>
</div>
</div>
<script>
var uid;
var opened_node=null;
$(function(){
$('#saveuser').click(function(){
if(!($("#adduser_form").form('enableValidation').form('validate')))return;
var p=$('#pwd').val(), p2=$('#pwd2').val();
if(p!=p2){
parent.dlg_h(300,120,"校验失败！ ","密码不一致！ ",null,null,"warning");
return;
}
})
}

```

```

        uid=$("#uid").textbox('getValue');

        jQuery.ajax({
            type: 'POST',
            url : "${ctxPath}/user/saveAdd",
            data : $("#adduser_form").serialize(),
            success : function (res) {
                if(res=="true"){
                    parent.dlg_h(300,140,"添加成功！","你已添加用户:"+uid,dlg_close,null,"info");
                } else{
                    parent.dlg_h(300,140,"添加失败！","添加失败！",null,null,"error");
                }
                return false;
            },
            error : function(res) {parent.dlg_h(300,140,"系统提示","系统错误！",null,null,"error");}
        });
    });

    $('#back').click(function(){dlg_close();});

})

function dlg_close(){
    parent.$("#mainframe").attr("src","${ctxPath}/user/goList");
    var tab = parent.$("#tabs").tabs("getSelected");
    var title1="用户信息";
    parent.$("#tabs").tabs("update", {tab:tab,options: {title: title1}});
}
</script>

```

用户修改

修改用户信息的页面**useredit.html**与添加新用户的页面内容基本相同，不相同的是，需要加载指定用户的信息到页面中相应的位置；修改后保存时，提交数据的路径不一样。**html**的代码中，使用了**beeth**语法，如： `${user.uid}`，尤其是在判断用户角色时，使用了**if**结构。

```

<div class="easyui-panel" style="width:580px;padding:30px 60px">
<form id="edituser_form" method="post" action="#">
    <div style="margin-bottom:20px">
        <div style="margin-bottom:20px">
            <input id="uid" name="uid" value="${user.uid}" readonly="readonly" class="easyui-textbox" data-options="prompt:'考号',required:true" style="width:100%;height:32px;">
        </div>
        <div style="margin-bottom:20px">
            <input id="department" name="department" value="${user.department}" class="easyui-textbox" data-options="prompt:'部/厂',required:true" style="width:100%;height:32px;">
        </div>
        <div style="margin-bottom:20px">
            <input name="xm" value="${user.xm}" class="easyui-textbox" data-options="prompt:'姓名',required:true" style="width:100%;height:32px;">
        </div>
    </div>

```

```

</div>
<div style="margin-bottom:20px">
    <div>登录密码(不修改密码请留空):</div>
    <input name="pwd" id="pwd" class="easyui-validatebox easyui-textbox" type="password" data-options="icons:[{iconCls:'icon-lock'}]" style="width:100%;height:32px" >
</div>
<div style="margin-bottom:20px">
    <div>确认密码:</div>
    <input id="pwd2" validType="equals['#pwd']" class="easyui-validatebox easyui-textbox" type="password" data-options="icons:[{iconCls:'icon-lock'}]" style="width:100%;height:32px" >
</div>
<div style="margin-bottom:20px">
    <div>角色:</div>
    <input type="radio" name="role" value="5"
    <% if (user.role=='5') { %>
        checked="checked"
    <% } %>
    />考生
    <input type="radio" name="role" value="1"
    <% if (user.role=='1') { %>
        checked="checked"
    <% } %>
    />管理员
</div>
</form>
<div style="text-align:center;">
    <a id="saveuser" href="#" class="easyui-linkbutton" iconCls="icon-ok" style="width:132;height:32px">保
存</a>
    <a id="back" href="#" class="easyui-linkbutton" iconCls="icon-back" style="width:132;height:32px">返回</a>
</div>
<script>
    var uid="${user.uid}";
$(function(){
    $('#saveuser').click(function(){
        //提交前先校验
        if(! ($("#edituser_form").form('enableValidation').form('validate'))){return;
        var p=$('#pwd').val(), p2=$('#pwd2').val();
        if(p!=p2){
            parent.dlg_h(300,120,"校验失败！ ","密码不一致！ ",null,null,"warning");
            return;
        }
        jQuery.ajax({
            type: 'POST',

```

```

url : "${ctxPath}/user/saveEdit",
data : $("#edituser_form").serialize(),
success : function (res) {
    if(res=="true"){
        parent.dlg_h(300,140,"修改成功！","你已修改用户:"+uid,dlg_close,null,"info");
    } else{
        parent.dlg_h(300,140,"修改失败！","修改失败！",null,null,"error");
    }
    return false;
},
error : function(res) {parent.dlg_h(300,140,"系统提示","系统错误！",null,null,"error");}
}),
$('#back').click(function(){dlg_close();});
})
}

function dlg_close(){
parent.$("#mainframe").attr("src","${ctxPath}/user/goList");
var tab = parent.$("#tabs").tabs("getSelected");
var title1="用户信息";
parent.$("#tabs").tabs("update", {tab:tab,options: {title: title1}});
}

```

</script>

3.*****

登录成功

考生登录成功后跳转到页面web/index.html,此页显示一些与考试相关的新闻，如考试须知、考试题型分值等信息，阅读相关信息后，考生确认跳转到考试页面。页面中包含了倒计时代码，强制考生阅读考试须知，并确认已阅读考试须知。

管理员登录成功后跳转到页面web/admin.html,此页以easyui布局形式展现后台管理功能，布局左侧以easyui树形结构显示分级菜单，布局中心区域使用了easyui标签页，标签页使用了iframe框架，框架中引用的页面是真正互动实现对信息进行增删改查操作的界面，界面通常会用到一些输入框easyui TextBox、按钮

LinkButton...

3.*****

3.*****