## 事件例子

namespace test1

{

class Program

{

static void Main(string[] args)

{

father AAA = new father();

son BBB = new son();

AAA.shi += new father.handler(BBB.abc);

//strcase.abc();

AAA.ddd();

Console.ReadLine();

}

}

public class father

{

public delegate void handler();

public event handler shi;

public virtual void abc()

{

Console.WriteLine("father method");

}

public void ddd ()

{

shi();

}

}

public class son : father

{

public override void abc()

{

Console.WriteLine("son method");

}

}

}

## 启动浏览器 打开特定页面

//打开浏览器

Unityorder: "start \"C:\Program Files (x86)\Google\Chrome\Application\chrome.exe\" \"http://172.30.17.107/SWMX/unity/index.html\"",

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

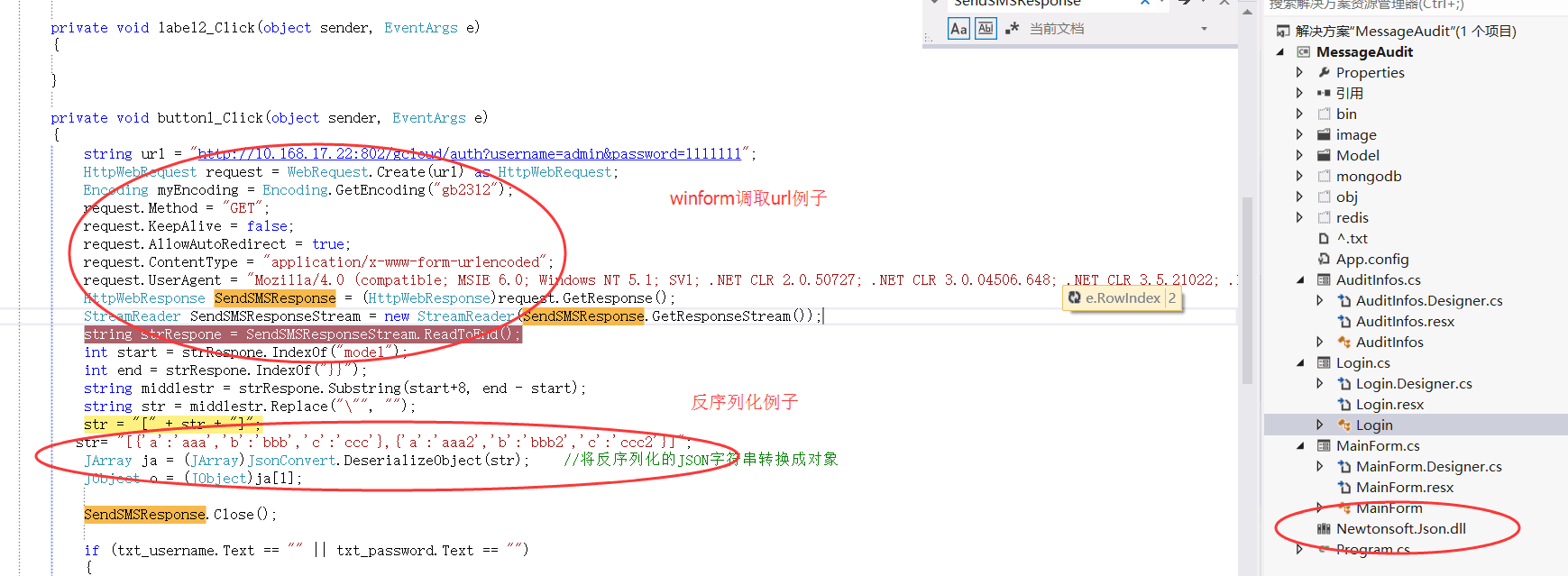
var cmd = new ActiveXObject("WScript.Shell");

var str = Robin.Setting.GlobalSetting.Unityorder;

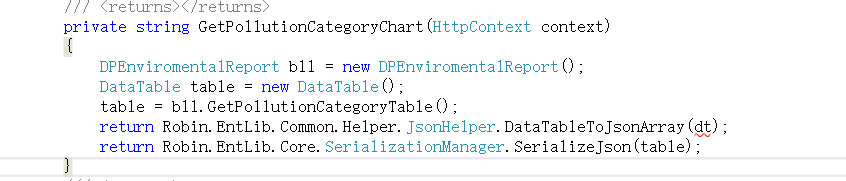
cmd.run("cmd.exe /c" + str);

## });

## Winform 调取url获得html 页面元素



## 序列化返回json字符串两种方式



## 代码授权过期 可注释下段代码

GSCP.Impower.Core.IMParam param = GSCP.Impower.Core.CheckAuthorization("ESIACTGSHWDPGEBOCGCZBEIXCHAFBCGTJIFYHLHKHF");

if (param != null && !param.IsPass)

{

Response.Redirect("/Users/Account/ErrPage?num=" + param.MachineKey);

//return View();

}

## jsPanel弹出框

//服务地址获取

function getRestServiceURL(tableId, groupId, tableName, tableGuid) {

var pamas = "?TABLEID=" + tableId + "&GROUPID=" + groupId + "&TABLENAME=" + tableName + "&TABLEGUID=" + tableGuid;

$.jsPanel({

resizable: "disabled",//是否可以改变大小

// draggable: "disabled",//是否可以拖动

title: "数据服务相关地址",

content: '<iframe src="../../Cloud/EntityConfig/entityConfigRestUrl.html' + pamas + '" width="100%" height="100%" frameborder="0" tabindex="9999"></iframe>',

position: 'center',

size: { width: 200, height: 200 },

controls: { buttons: "closeonly" }

});

}

## Infopanel例子

//增加

function addTenant() {

//top.Robin.Window.InfoPanel("添加使用单位", '<iframe src="../manhole/Business/Tenant/AddTenant.htm" width="100%" height="100%" frameborder="0"></iframe>',

// {

// id: 'addTenantid', size: { width: 900, height: 550 }

// });

var url = '<iframe src="AddTenant.htm" width="100%" height="100%" frameborder="0"></iframe>';

projWin = $.jsPanel({

resizable: "disabled",

draggable: "disabled",

renderTo: 'body',

title: "添加使用单位",

content: url,

position: 'center',

size: { width: 900, height: 550 },

controls: { buttons: "closeonly" }

});

}

function addTenantRoleUser(jsonDataRoleUser, TENANTID) {

var url = Robin.Setting.GlobalSetting.AuthorityService + "TenantRoleUserAdd/v1";

jsonDataRoleUser.TENANTID = TENANTID;

$.ajax({

type: "POST",

url: url,

async: false,

dataType: 'json',

data: jsonDataRoleUser,

success: function (dataroleuser) {

if (dataroleuser) {

noty({ text: '添加成功!', type: 'success', layout: "topCenter", timeout: 2000 });

//var listpanel = parent.Robin.Window.GetInfoPanelByID("tenantManagementid");

//listpanel.normalize();

//listpanel.reloadContent();

//window.setTimeout(function () {

// top.Robin.CloseOneInfoPanel("addTenantid");

//}, 1000);

parent.initTable();

window.setTimeout(function () {

parent.projWin.close();

}, 1000);

} else {

noty({ text: '添加失败!', type: 'error', layout: "topCenter", timeout: 2000 });

}

},

error: function (data) {

noty({ text: '添加失败!', type: 'error', layout: "topCenter", timeout: 2000 });

}

});

}

## Jspanel 例子二 第三层frame弹出到top

例一

function edit(id) {

var pro = top.$.jsPanel({

resizable: "disabled",

// draggable: "disabled",

title: "实体信息修改",

content: '<iframe src="EntityConfig/entityConfigEdit.html?Id=' + id + '" width="100%" height="100%" frameborder="0" tabindex="9999"></iframe>',

position: 'center',

size: { width: 1100, height: 600 },

controls: { buttons: "closeonly" }

});

top.Robin.Window.instances.push({

id: "entityConfigEdit",

obj: pro

});

}

关闭：

$.ajax({

url: url,

method: "POST",

data: formdata,

dataType: "json",

success: function (result) {

if (result == true) {

top.noty({ text: "编辑成功！", type: "success", layout: "topCenter", timeout: 1000 });

setTimeout(function () {

//关闭 刷新

top.iframePage.frmrightChild.location.reload();

$.each(top.Robin.Window.instances, function (i, item) {

if (!item || !item.obj) {

return true;

}

var pnl = item.obj;

pnl.close();

});

}, 1000);

} else {

noty({ text: "编辑失败！", type: "error", layout: "topCenter", timeout: 1000 });

}

},

error: function (ex) {

var exception = ex;

}

});

例二

function add() {

top.Robin.Window.InfoPanel("企业信息添加", '<iframe src="DataManagement/ggztsj/frxxAdd.html" width="100%" height="100%" frameborder="0" tabindex="9999"></iframe>',

{ id: 'frxxadd', size: { width: 900, height: 500 }, minimizeOthers: false });

}

关闭：

$.ajax({

url: urlAdd + pamas,

method: "POST",

dataType: "json",

async: false,

success: function (result) {

if (result.Success == true) {

top.noty({ text: "保存成功！", type: "success", layout: "topCenter", timeout: 1000 });

setTimeout(function () {

//关闭 刷新

$.each(top.Robin.Window.instances, function (i, item) {

if (item) {

var pnl = item.obj;

pnl.close();

}

});

}, 1000);

var a = parent.document.getElementById('iframePage').contentWindow.document.getElementById('qyxxview').contentWindow;

a.InitData();

} else {

noty({ text: "保存失败！", type: "error", layout: "topCenter", timeout: 1000 });

}

},

error: function (ex) {

var exception = ex;

}

});

## 地图js核心代码

Robin.Map.GetGraphicLayer(Robin.Map.Map2DControl, "GasStationListLayer");//创建id为GasStationListLayer的层

p=Robin.Map.GetPoint(StationInfo.BMX, StationInfo.BMY);//获取标注点位置

Robin.Map.Fly2Geometry(Robin.Map.Map2DControl, p);//移动到标注点。

Robin.Map.Fly2Geometry(Robin.Map.Map2DControl, p, function () {});移动到标注点后回掉方法

Robin.Map.ClearLayer(Robin.Map.Map2DControl, "GasStationListLayer");//清除标注

Robin.Map.MapWindow.Show(StationInfo.Id, Robin.Map.Map2DControl, p, newhtml, true);//标注点弹窗

Robin.Map.MapWindow.Show(item.BMID, Robin.Map.Map2DControl, p, null, true);//弹窗参数为null则显示红点

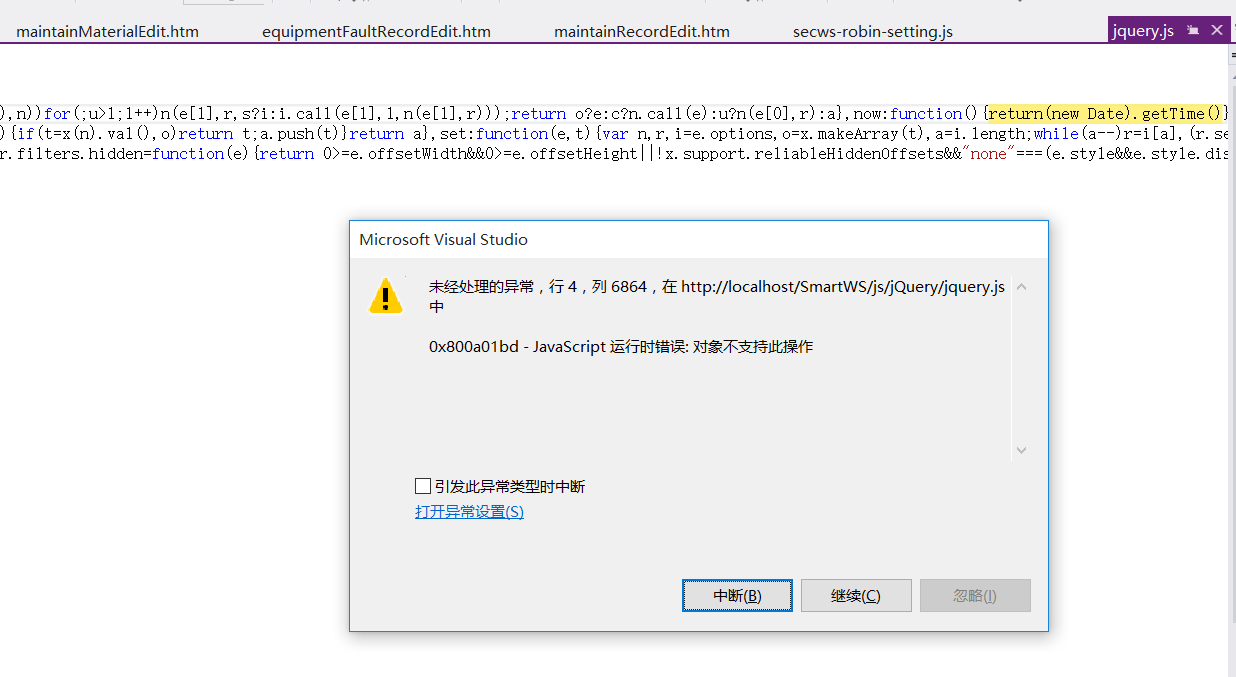
Robin.Map.ShowGraphic(StationListLayer, item.BMMC, item.BMX, item.BMY, "../images/gas/index/gstation.png", item, symbolConfig);// 地图渲染图标

var extent = new esri.geometry.Extent(parseFloat(x) - R \* 1.5, parseFloat(y) - R \* 1.5, parseFloat(x) + R \* 1.5, parseFloat(y) + R \* 1.5,new esri.SpatialReference({ wkid: Robin.Setting.GlobalSetting.wkid }));

map.setExtent(extent);//缓冲区定位后，调整显示范围

http://localhost:891/SYS/APP\_DATA\_TABLE\_QueryListByDatasource/v

## 错误及解决方案1



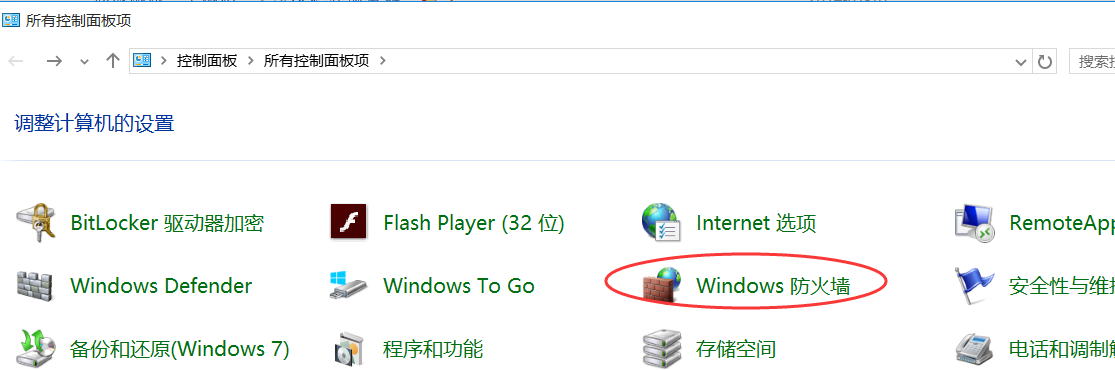


阴影部分修改为 加了延时处理，ajax的异步问题



## 数据查询in (MBriBasicinfo.\_.NAME.Contains(bridgeName));

## 其他电脑无法访问部署的服务，关闭防火墙。



## 页面刷新location.replace

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

var currenttime = $("#queryDate").val();

location.replace("ProjectReport.html?projId=" + projId + "&projName=" + projName + "&currenttime=" + currenttime)

});

## Js 自定义事件及触发方式

var $x = $(document.getElementById("stationInfo"));

/\*关闭panel事件.\*/

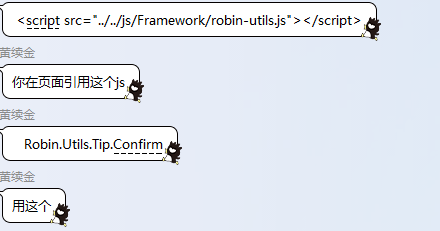
$x.on("unloadname", function (event, id) {

alert(id);

});

$x.trigger("unloadname", ["1111"]);

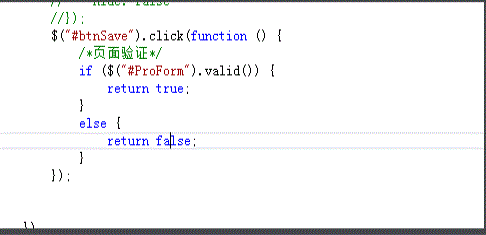
## 验证模块



<script src="../../js/Bootstrap/Version3/js/bootstrap.min.js"></script>

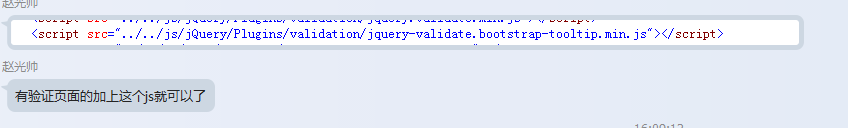
<script src="../../js/jQuery/Plugins/validation/jquery.validate.min.js"></script>

<script src="../../js/jQuery/Plugins/validation/jquery-validate.bootstrap-tooltip.min.js"></script>









$('input:text:first').focus();表单元素选择

$('p).toggleClass(“class”);切换类。如果类名存在就删除，不存在就添加。

### Confirm noty 关闭窗口刷新table

function deleteOne(Id, name) {

confirm("<div class='notyContent'>确定要删除【" + name + "】吗？</div>",

"information", function () {

$.ajax({

url: "InvestmentHandler.ashx?Action=Delete&Id=" + Id,

success: function (result) {

if (result == "true") {

noty({ text: "删除成功！", type: "success", layout: "topCenter" });

oTable.fnClearTable(0);

oTable.fnDraw();

} else {

noty({ text: "删除失败！", type: "success", layout: "topCenter" });

}

}

});

}, null,

'确定', '取消');

}

//关闭窗口并刷新table

function closeAdd() {

projWin.close();

oTable.fnDraw();

}

/\*添加成功失败，返回弹窗提示\*/

function alertInfo(content, typeInfo) {

noty({ text: content, type: typeInfo, layout: "topCenter" });

}

/\*\*

\* confirm提示.需要提前加载noty.这个方法放在utils里会报错

\* 需要引用js/Bootstrap/Version3/css/bootstrap-button.css，js/Noty/packaged/jquery.noty.packaged.min.js

\* @param message 提示的内容

\* @param type 类型information,alert,error,success,warning

\* @param okCallback yes执行的方法

\* @param cancelCallback no执行的方法

\* @param lblok yes显示的内容

\* @param lblcancel no显示的内容

\* @constructor

\*/

function confirm(message, type, okCallback, cancelCallback, lblok, lblcancel) {

if (!noty) {

alert("请检查插件是否加载成功！");

return;

}

var \_default =

{

text: message || "您确定要删除所选对象吗?",

type: type || "alert",

dismissQueue: true,

modal: true,

layout: "center",

buttons: [

{

addClass: 'btn btn-primary',

text: lblok || "确定",

onClick: function ($noty) {

$noty.close();

if (okCallback && $.isFunction(okCallback)) {

okCallback();

}

}

},

{

addClass: 'btn btn-warning',

text: lblcancel || "取消",

onClick: function ($noty) {

$noty.close();

if (cancelCallback && $.isFunction(cancelCallback)) {

cancelCallback();

}

}

}

]

};

noty(\_default);

}

## zTree

var setting = {

treeId: "tree",

async: {

url: "ProCityHandler.ashx?action=CityTreeWithoutPro"

}

};

//区域列表数据初始化--jln

Robin.Tree.Init(setting);

## ZTree2

var serviceUrl = Robin.Setting.GlobalSetting.RestAPIService + "/dataservice/city/default?callback=?";

var treeData = [];

var setting = {

treeId: "tree",

check: {

enable: true,

chkStyle: 'radio',

radioType: 'all'

},

data: {

simpleData: {

enable: true

}

}

//callback: {

// // beforeClick: openURL,

// onNodeCreated: onNodeCreatedEvent

//}

};

$.getJSON(serviceUrl, function (data) {//跨域访问

$.each(data, function (i, v) {

treeData.push({

id: v.CITYCODE, pId: v.PID, name: v.CITYNAME

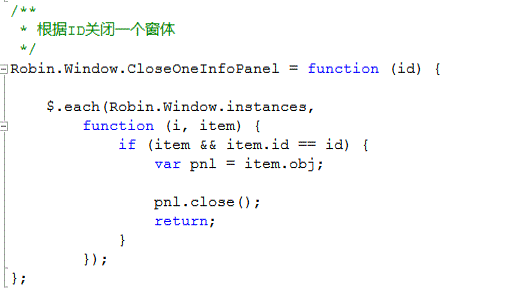
});

});

$.fn.zTree.init($("#tree"), setting, treeData);

});

## Panel按名称关闭CloseOneInfoPanel("TotalInvestAdd");



top.Robin.Window.CloseOneInfoPanel = function (id) {

$.each(top.Robin.Window.instances, function (i, item) {

if (item && item.id == id) {

var pnl = item.obj;

pnl.close();

return;

}

});

};

//noty({ text: content, type: typeInfo, layout: "topCenter" });

function closeAddWind(content, typeInfo) {

noty({ text: content, type: typeInfo, layout: "topCenter" });

if (typeInfo == 'success') {

window.setTimeout(function () {

top.Robin.Window.CloseOneInfoPanel("TotalInvestAdd");

}, 1000);

}

}

确认提示框



## 关闭panel 并刷新table

function closeAdd() {

window.setTimeout(function () { projWin.close(); }, 1000)

oTable.fnDraw();

}

//初始化表格

if (oTable != null) {

oTable.fnDestroy();

oTable = null;

};

## 导出数据到word –项目管理中有例子

function exportToWord(controlId) {

var control = document.getElementById("reportTable");

try {

var oWD = new ActiveXObject("Word.Application");

var oDC = oWD.Documents.Add("", 0, 1);

var oRange = oDC.Range(0, 1);

var sel = document.body.createTextRange();

try {

sel.moveToElementText(control);

} catch (notE) {

alert("导出数据失败，没有数据可以导出。");

window.close();

return;

}

sel.select();

sel.execCommand("Copy");

oRange.Paste();

oWD.Application.Visible = true;

oDC.saveAs("d:\\ba.doc");

// window.close();

}

catch (e) {

alert("导出数据失败，需要在客户机器安装Microsoft Office Word(不限版本)，将当前站点加入信任站点，允许在IE中运行ActiveX控件。");

try { oWD.Quit(); } catch (ex) { }

//window.close();

}

}

## 导出数据到excel（两种方式）管网云项目有例子—全库管点统计模块

$("#btnExport").click(function (i, v) {//对参数长度有要求

//有返回值的方式

var temp = JSON.stringify(queryData);

$.ajax({

type: "POST",

cache: false,

url: "ReportHandler.ashx?Action=Export&jsonData=" + encodeURIComponent(temp),

// dataType: 'string',

success: function (result) {

if (result != null && result != "") {

window.open(result);

}

}

});

后台代码：

string result = string.Empty;

string data = context.Server.UrlDecode(context.Request.QueryString["jsonData"]);//url传参

//无返回值的方式 //对参数长度无要求，但需要隐藏变量

var temp = JSON.stringify(exportDataList);

$("#hidData").val(temp);

$("#form1").attr("action", "reportHandler.ashx?Action=Export");

$("#form1").submit();

后台代码：

string data = context.Request.Form["hidData"];

List<Report> list = JsonHelper.DeserializeObject(data, typeof(List<Report>)) as List<Report>;//序列化

encodeURIComponent(bridgeName),

context.Server.UrlDecode(context.Request["bridgeName"]);

## 文件流查询删除

string[] rootFiles = Directory.GetFiles(url.Replace("projectDetailInfo.doc", ""), "\*.doc");//不带文件名的路径

DirectoryInfo dir = new DirectoryInfo(url.Replace("projectDetailInfo.doc", ""));

if (dir.Exists)

{

FileInfo[] fileInfos = dir.GetFiles();

if (fileInfos.Length > 0)

{

foreach (FileInfo item in fileInfos)

{

if (item.Name == "projectDetailInfo.doc")

{

File.Delete(item.FullName);

}

}

}

}

## 字符串判空

if (string.IsNullOrEmpty(context.Request.QueryString["NVFID"]))

## 确认框

if (!confirm("确认要删除？")) {

return false;

}

## Oracle 日期转换

string where = "to\_char(plandate,'yyyy-mm')='" + planDate.Trim() + "'";

string where += "and COSTDATE=to\_date(substr('" + proName + "',1,10),'yyyy-mm') ";

## .net 后台日期加减

//方法内数据可以为负数

DateTime dt = DateTime.Now.AddDays(Convert.ToInt32(YJSJ) -Convert.ToInt32(yhzq));

where += " and EQCATEGORYID='"+eqClassCode+"'";

where += " and to\_char(ZHYHSJ,'yyyy-mm-dd')<='" + dt.ToShortDateString() + "'";

## js 日期加减

var d = new Date(data);

d.setDate(d.getDate() + yhzq);

## .net 后台时间样式设置

sheet.Range["A2"].Text = DateTime.Parse(time).ToString("yyyy年MM月dd日");

// DateTime.Parse参数必须为字符串，将DateTime?做判空处理后转换为字符串

string strdate = list[j].ZHYHSJ == null ? "" : DateTime.Parse(list[j].ZHYHSJ.ToString()).ToString("yyyy-MM-dd");

sheet.Range[4 + j, 5].Text = yhzq;

// 时间加法及格式处理

sheet.Range[4 + j, 6].Text = DateTime.Parse(strdate).AddDays(Convert.ToInt32(yhzq)).ToString("yyyy-MM-dd");

## js 日期处理

//根据时间戳生成的时间对象

function GetDate(TIME) {

if (TIME) {

var a = TIME.indexOf("(");

var b = TIME.indexOf(")");

var str = TIME.substring(a + 1, b);

var c = str.indexOf("-");

var timestamp = new Date(parseInt(str.substring(0, c)));

var d = new Date(timestamp); //根据时间戳生成的时间对象

var date = (d.getFullYear()) + "-" +

Appendzero((d.getMonth() + 1)) + "-" +

Appendzero((d.getDate()));

return date;

} else {

return "";

}

}

function Appendzero(obj) {

if (obj < 10) return "0" + "" + obj;

else return obj;

}

## 设置当前日期为当日和前一天

方法一

var dt = new Date();

var eTime = $("#endYear").val(dt.Format("yyyy-MM-dd"));// HH:mm:ss

dt.setDate(dt.getDate() - 1);

var sTime = $("#startYear").val(dt.Format("yyyy-MM-dd"));// HH:mm:ss

Date.prototype.Format = function (fmt) {

var o = {

"M+": this.getMonth() + 1, //月份

"d+": this.getDate(), //日

"h+": this.getHours() % 12 == 0 ? 12 : this.getHours() % 12, //小时

"H+": this.getHours(), //小时

"m+": this.getMinutes(), //分

"s+": this.getSeconds(), //秒

"q+": Math.floor((this.getMonth() + 3) / 3), //季度

"S": this.getMilliseconds() //毫秒

};

var week = {

"0": "/u65e5",

"1": "/u4e00",

"2": "/u4e8c",

"3": "/u4e09",

"4": "/u56db",

"5": "/u4e94",

"6": "/u516d"

};

if (/(y+)/.test(fmt)) {

fmt = fmt.replace(RegExp.$1, (this.getFullYear() + "").substr(4 - RegExp.$1.length));

}

if (/(E+)/.test(fmt)) {

fmt = fmt.replace(RegExp.$1, ((RegExp.$1.length > 1) ? (RegExp.$1.length > 2 ? "/u661f/u671f" : "/u5468") : "") + week[this.getDay() + ""]);

}

for (var k in o) {

if (new RegExp("(" + k + ")").test(fmt)) {

fmt = fmt.replace(RegExp.$1, (RegExp.$1.length == 1) ? (o[k]) : (("00" + o[k]).substr(("" + o[k]).length)));

}

}

return fmt;

}

方法二

var myDate = new Date();

var year = myDate.getFullYear();

var month = myDate.getMonth() + 1;

var day = myDate.getDate();

if (month <= 9) {

month = "0" + month;

}

if (day <= 9) {

day = "0" + day + 1;

}

var beginTime = year + "-" + month + "-" + day;

$("#txtTime").val(beginTime);

## 滚动条添加

<script src="../js/jQuery/Plugins/slimScroll/jquery.slimscroll.min.js"></script>//引入js库

<div class="widget-body no-padding" id="maintianscroll">//div设置id

<table id="maintainnWarningRecord" class="table table-striped table-bordered" style="width: 95%; height: 250px; cellspacing: 0px; align: center; margin-left: -1px"></table>

</div>

$("#maintianscroll").slimScroll({height:250});

## 点击panal列表 定位地图后添加图标

function locateXY(x, y) {

var x = parseFloat(x);

var y = parseFloat(y);

var listpanel = top.Robin.Window.GetInfoPanelByID("attendancedataid");

listpanel.smallify();//panel 最小化

top.Robin.Map.ClearLayer(Robin.Map.Map2DControl, "dwLayer");

var jgLayer = top.Robin.Map.GetGraphicLayer(top.Robin.Map.Map2DControl, "dwLayer");

var symbolConfig = {

font:

{ 'size': "13", 'style': 'normal' },

color: [0, 89, 149],

pic: {

src: '../images/manhole/gstation.png',

width: 40,

height: 40

},

offset: {

x: 0,

y: -30

}

}

//经纬度信息转换为web墨卡托

var mercator = top.Robin.Map.Tool.lonlatToMercator(x, y);

//定义井盖信息属性数据

var attribute = {};

//地图渲染图标

top.Robin.Map.ShowGraphic(jgLayer, "", mercator.x, mercator.y, "../images/Map/point.png", attribute, symbolConfig);

var p = top.Robin.Map.GetPoint(mercator.x, mercator.y);

top.Robin.Map.Fly2Geometry(top.Robin.Map.Map2DControl, p);

}

## 下拉框例子（一）aspx控件

1. 前端页面

<asp:DropDownList ID="ACCIDENTTYPENAME" runat="server" CssClass="form-control required" style="width:90%;">

<asp:ListItem Text="-请选择-" Value="" Selected="True"></asp:ListItem>

</asp:DropDownList>

固定item例子：

<asp:DropDownList ID="RESPONSELEVEL" runat="server" CssClass="form-control required" style="width:90%;">

<asp:ListItem Text="-请选择-" Value="" Selected="True"></asp:ListItem>

<asp:ListItem value="甲类">甲类</asp:ListItem>

<asp:ListItem value="乙类">乙类</asp:ListItem>

</asp:DropDownList>

后端代码

protected void Page\_Load(object sender, EventArgs e)

{

if (!IsPostBack)

{

LoadAccidentType();

}

}

private void LoadAccidentType()

{

DAccidentType bllOrganization = new DAccidentType();

List<MAccidentType> listOrganization = bllOrganization.GetEntityList(WhereClip.None);

if (listOrganization != null)

{

for (int i = 0; i < listOrganization.Count; i++)

{

ACCIDENTTYPENAME.Items.Add(new ListItem(listOrganization[i].ACCIDENTTYPE.ToString(), listOrganization[i].ACCIDENTTYPECODE.ToString()));

}

}

}

保存

model.ACCIDENTTYPECODE = ACCIDENTTYPENAME.SelectedItem.Value;

model.ACCIDENTTYPENAME = ACCIDENTTYPENAME.SelectedItem.Text;

编辑加载

ACCIDENTTYPENAME.SelectedValue = model.ACCIDENTTYPECODE;

## 下拉框例子（二）select2插件应用多选

<link href="../../js/jQuery/Plugins/select2/select2.css" rel="stylesheet" />

<script src="../../js/jQuery/Plugins/select2/select2.min.js"></script>

前端加载数据

$('#dll\_connect').select2();

$.ajax({

url: "Handler.ashx?Action=GetAccidentForDropdown",

success: function (result) {

if (result != null) {

result = eval('(' + result + ')');

$.each(result, function (i, item) {

$("#dll\_connect").append('<option value="' + item.ACCIDENTTYPECODE + '">' + item.ACCIDENTTYPE + '</option>');

});

}

}

});

点击后隐藏变量赋值

$("#dll\_connect").on("click", function () {

var count = $(this).children('option:selected').length;

var txt = "";

var val = "";

for (var i = 0; i < count; i++) {

txt += $(this).children('option:selected')[i].text + "#";

val += $(this).children('option:selected')[i].value + "#";

}

$("#sglxtxt").val(txt);

$("#sglxval").val(val);

});

前端页面html

<td>关联事故类型

<asp:HiddenField ID="sglxtxt" runat="server" />

<asp:HiddenField ID="sglxval" runat="server" />

</td>

<td>

<select id="dll\_connect" multiple="multiple" style="width: 300px; height: 30px;">

</select>

</td>

后端取隐藏变量值

string[] sglxvals = sglxval.Value.Split(new char[] { '#' });

string[] sglxtxts = sglxtxt.Value.Split(new char[] { '#' });

for (int i = 0; i < sglxtxts.Length-1; i++)

{

MExpertLinkAccidentType modelLink = new MExpertLinkAccidentType();

modelLink.ID = Guid.NewGuid().ToString();

modelLink.EXPERTCODE = model.EXPERTCODE;

modelLink.EXPERTNAME = EXPERTNAME.Text.Trim();

modelLink.ACCIDENTTYPECODE = sglxvals[i];

modelLink.ACCIDENTTYPENAME = sglxtxts[i];

bllLink.Insert(modelLink);

}

编辑页面赋默认值

$.ajax({

url: "Handler.ashx?Action=GetAccidentForDropdown",

success: function (result) {

if (result != null) {

result = eval('(' + result + ')');

$.each(result, function (i, item) {

$("#dll\_connect").append('<option value="' + item.ACCIDENTTYPECODE + '">' + item.ACCIDENTTYPE + '</option>');

});

$("#dll\_connect").select2("val", $("#sglxval").val().split('#'));

}

}

});

## 下拉框例子（二）select2插件应用单选

$('#txtName').select2();

$.ajax({

url: "../Expert/Handler.ashx?Action=GetAccidentForDropdown",

success: function (result) {

if (result != null) {

result = eval('(' + result + ')');

$("#txtName").append('<option value="">' + "全部" + '</option>');

$.each(result, function (i, item) {

$("#txtName").append('<option value="' + item.ACCIDENTTYPECODE + '">' + item.ACCIDENTTYPE + '</option>');

});

}

}

});

$("#txtName").on("click", function () {

var txt = $(this).children('option:selected').val();

$("#ddl\_sglx").val(txt);//隐藏变量赋值

});

页面html <select id="txtName" style="width: 150px; height: 30px;">

</select> <input type="hidden" id="ddl\_sglx" />

前端编辑页面加载默认值

$.each(v.stationData, function (i, v) {

if (i == 0) {

//赋值，默认第一项 或者也可以-- $('#layersname').select2().select2('val', changedNodes[1].id);

$('#select2-chosen-1').html(v.BMMC);

$("#stationList").append('<option value="' + v.BMID + '">' + v.BMMC + '</option>');

} else {

$("#stationList").append('<option value="' + v.BMID + '">' + v.BMMC + '</option>');

}

});

## 获取下拉框的值

$('#DISABLED option:selected').text();

$('#DISABLED option:selected').val();

<div class="col-md-4 col-sm-4 col-xs-12">

<select id="DISABLED" name="DISABLED" class="form-control col-md-7 col-xs-12" required="required">

<option value="1">启用</option>

<option value="0">禁用</option>

</select>

</div>

<select autocomplete="off">

    <option>是</option>

    <option selected="selected">否</option>

</select>

## Select ajax 绑定数据

例子一

//加载主题分类下拉框数据

function initTABLETOPIC() {

$.ajax({

url: Robin.Setting.GlobalSetting.EntityConfigWeb + Robin.Setting.GlobalSetting.APP\_CODE\_QueryByGROUPNAME + "?GROUPNAME=TABLETOPIC",

method: "GET",

dataType: "json",

success: function (result) {

if (result != null && result.length > 0) {

$.each(result, function (i, item) {

if (typeof (item.PARENTCODE) == "undefined" || item.PARENTCODE == null) {

if (item.CODENAME == "业务系统数据") {

$("#TABLETOPIC").append('<option value="' + item.CODE + '" selected="selected" >' + item.CODENAME + '</option>');

}

else {

$("#TABLETOPIC").append('<option value="' + item.CODE + '">' + item.CODENAME + '</option>');

}

}

else {

$("#TABLETOPIC").append('<option value="' + item.CODE + '">' + "&nbsp;&nbsp;&nbsp;" + item.CODENAME + '</option>');

}

});

}

}

});

}

例子二

$.ajax({

url: Robin.Setting.GlobalSetting.EntityConfigWeb + Robin.Setting.GlobalSetting.APP\_CODE\_QueryByGROUPNAME + "?GROUPNAME=TABLETOPIC",

method: "GET",

dataType: "json",

success: function (result) {

if (result != null && result.length > 0) {

$.each(result, function (i, item) {

if (typeof (item.PARENTCODE) == "undefined" || item.PARENTCODE == null) {

if (item.CODENAME == "业务系统数据")

{

$("#TABLETOPIC").append('<option value="' + item.CODE + '" selected="selected" >' + item.CODENAME + '</option>');

}

else {

$("#TABLETOPIC").append('<option value="' + item.CODE + '">' + item.CODENAME + '</option>');

}

}

else {

$("#TABLETOPIC").append('<option value="' + item.CODE + '">' + "&nbsp;&nbsp;&nbsp;" + item.CODENAME + '</option>');

}

});

}

}

});

## 单选控件

<asp:RadioButton ID="RbtnY" GroupName="DISABLED" Text="男" runat="server" />

<asp:RadioButton ID="RbtnN" GroupName="DISABLED" Text="女" Checked="true" runat="server" />

保存

if (this.RbtnN.Checked)

{

myMGCUser.SEX = "1";

}

else

{

myMGCUser.SEX = "0";

}

编辑

if (myMGCUser.SEX == "0")

{

this.RbtnY.Checked = true;

}

else

{

this.RbtnN.Checked = true;

}

## Js前端json转字符串后传入后端excel导出

var temp = JSON.stringify(exportDataList);

$("#hidData").val(temp);

$("#form1").attr("action", "reportHandler.ashx?Action=Export");

$("#form1").submit();

## Js前端字符串转json

queryjson = JSON.parse(querystr);

## ajax post 传递过长参数

前端

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

var checkedItems = [];

var gridid = 'dg';

checkedItems = $("#" + gridid).datagrid('getSelections');

var temp = JSON.stringify(checkedItems);

var param = '?Action=ExportTasks&now=' + new Date().getTime();

$.ajax({

type: 'POST',

cache:false,

url: "Handler.ashx" + param,

data: { "taskdata": temp },

success: function (data) {

},

dataType: 'JSON'

});

});

后端获取参数

string tasks = context.Request.Params["taskdata"];

## 获取当前目录文件路径两种方式

HttpContext context = HttpContext.Current;

if (context != null)

{//方式一

filename = context.Server.MapPath("~/ConfigurationFiles/ftppathxml.xml");

if (!File.Exists(filename))

{

filename = context.Server.MapPath("~/ConfigurationFiles/ftppathxml.xml");

}

}

else

{//方式二

filename = Path.Combine(AppDomain.CurrentDomain.BaseDirectory, "/ConfigurationFiles/ftppathxml.xml");

}

if (!File.Exists(filename))

{

throw new Exception("发生错误: config目录下没有正确的ftppathxml.xml文件");

}

## .net 后端发起request请求和接收响应

/// <summary>

/// 打印地图后将地图图片保存到服务器上

/// </summary>

/// <param name="context"></param>

/// <param name="printhref"></param>

/// <returns></returns>

private string SavePrintImage(HttpContext context,string printhref)

{

string result = string.Empty;

try

{

HttpWebRequest webrequest = (HttpWebRequest)WebRequest.Create(printhref);

HttpWebResponse webresponse = (HttpWebResponse)webrequest.GetResponse();

if (webresponse.StatusCode == HttpStatusCode.OK)

{

System.Drawing.Image image = System.Drawing.Image.FromStream(webresponse.GetResponseStream());

//string dirpath = @"Images";

//制定根路径

string imgPath = context.Server.MapPath("upload/images/");

//利用时间的无重复性对图片命名

string rename = DateTime.Now.ToString("yyyyMMddHHmm");

if (!Directory.Exists(imgPath))

Directory.CreateDirectory(imgPath);

image.Save(imgPath + rename + ".png", ImageFormat.Png);

image.Dispose();

result = rename + ".png";

}

}

catch(Exception ex)

{

}

return result;

}

## 输入格式验证

Regex r = new Regex("^\\s\*([A-Za-z0-9\_-]+(\\.\\w+)\*@(\\w+\\.)+\\w{2,5})\\s\*$");

if (!r.IsMatch(myMGCUser.EMAIL))

{

ClientScript.RegisterClientScriptBlock(this.GetType(), "this", "<script>alertInfo('邮件地址格式错误！','error');</script>");

return;

}

r = new Regex(@"^[1][3-8]\d{9}$");

if (!r.IsMatch(myMGCUser.PHONE))

{

ClientScript.RegisterClientScriptBlock(this.GetType(), "this", "<script>alertInfo('手机格式错误！','error');</script>");

return;

}

r = new Regex(@"^[0-9]\*$");

if (!r.IsMatch(myMGCUser.TELNODEPT))

{

ClientScript.RegisterClientScriptBlock(this.GetType(), "this", "<script>alertInfo('办公电话格式错误！','error');</script>");

return;

}

r = new Regex(@"^[0-9]\*$");

if (!r.IsMatch(myMGCUser.HOMEPHONE))

{

ClientScript.RegisterClientScriptBlock(this.GetType(), "this", "<script>alertInfo('家庭电话格式错误！','error');</script>");

return;

}

r = new Regex(@"^[A-Za-z0-9]+$");

if (!r.IsMatch(myMGCUser.PASSWORDS))

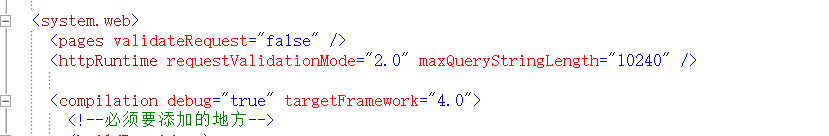
{

ClientScript.RegisterClientScriptBlock(this.GetType(), "this", "<script>alertInfo('密码格式错误，只能输入字母与数字！','error');</script>");

return;

}

## 文本编辑器 ueditor



<script src="../../js/jQuery/Plugins/ueditor/ueditor.config.js" charset="utf-8" type="text/javascript"></script>

<script src="../../js/jQuery/Plugins/ueditor/ueditor.all.js" charset="utf-8" type="text/javascript"></script>

<script src="../../js/jQuery/Plugins/ueditor/lang/zh-cn/zh-cn.js" charset="utf-8" type="text/javascript"></script>

<script type="text/javascript">

var ue = UE.getEditor('editor');

$(function () {

$('input:text:first').focus();

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

$('#CODEEXAMPLE').val(UE.getEditor('editor').getContent());

/\*页面验证\*/

if ($("#mainForm").valid()) {

return true;

}

else {

return false;

}

});

ue.addListener('ready', function (editor) {

// ue.setDisabled({});//是否只读

var content = $('#CODEEXAMPLE').val();

if (content) {

ue.setContent(content);

}

ue.execCommand('focus'); //编辑器加载完成后，让编辑器拿到焦点

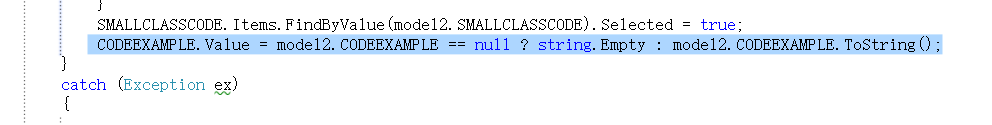
});

});

保存：



编辑加载



## 动态加载js文件

var oHead = document.getElementsByTagName('HEAD').item(0);

var oScript = document.createElement("script");

oScript.type = "text/javascript";

oScript.src = "../js/jquery.marquee.js";

oHead.appendChild(oScript);

window.setTimeout(function () {//ajax情况下加载文件需要时间

$('#marquee').kxbdSuperMarquee({

isMarquee: true,

isEqual: false,

scrollDelay: 20,

direction: 'up'

});

}, 1000);

## HttpCookie 设置与修改 js cookie

创建cookie

HttpCookie cookie = new HttpCookie("TokenCookieName");

cookie.Values.Add("token", token);

cookie.Expires = DateTime.Now.AddDays(Double.Parse(ConfigurationManager.AppSettings["Timeout"]));

Response.SetCookie(cookie);

获取cookie

HttpContext.Current.Request.Cookies["TokenCookieName"];

修改cookie值

Response.Cookies["TokenCookieName"].Value = "-1";

Js 代码：

<script src="js/jQuery/Plugins/jquery.cookie.js"></script>

创建cookie

$.cookie("address", adress, { path: '/' }); $.cookie("loginUserName", name, { expires: 7 });

$.cookie("lastTime", Times);

获取cookie值

var userName = $.cookie('userName');

修改cookie值

$.cookie("ISTENANTUSER", s);

## HttpCookie

设置

HttpCookie cookie = new HttpCookie("TokenCookieName");

cookie.Values.Add("token", tokenvalue);

cookie.Expires = DateTime.Now.AddDays(Double.Parse(ConfigurationManager.AppSettings["Timeout"]));

HttpContext.Current.Response.SetCookie(cookie);

获取 var Token =HttpContext.Current.Response.Cookies["TokenCookieName"];

## serviceStack cookie

设置：

Cookie cookie = new Cookie("TokenCookieName", "token=" + AESHelper.AESEncrypt(token));

base.Response.AddHeader("X-CustomHeader", "Modify HTTP Response in Service");

base.Response.SetCookie(cookie);

获取：

Cookie token = base.Request.Cookies["TokenCookieName"];

## 浏览器cookie怎么直接看 直接在浏览器地址输入下面代码

javascript:alert(document.cookie)

## Session cookie 列子

### 后台

public ActionResult Index(string ptype, int infotype, string userid)

{

Session["userID"]=userid;

HttpCookie cookie = new HttpCookie("MyCookie");

cookie.Values.Add("userid", userid);

Response.SetCookie(cookie);

return View();

}

### 前台

<script src="../js/jQuery/Plugins/jquery.cookie.js"></script>

//设置cookie 全目录共享

$.cookie("address", adress, { path: '/' });

//删除cookie的值

$.cookie("address", null, { path: '/' });

//获取cookie值

var tenantid = $.cookie("TENANTID");

## bootstrap table 编辑后生成静态html页面

前端：

$("table tbody tr td").dblclick(function () {

var td = $(this);

var tdold = td[0].outerHTML;

// 根据表格文本创建文本框 并加入表表中--文本框的样式自己调整

var text = td.text();

var txt = $("<input type='text' class='form-control'>").val(text);

txt.blur(function () {

// 失去焦点，保存值。

var newText = $(this).val();

// 移除文本框,显示新值

$(this).remove();

td.text(newText);

var tdnew = td[0].outerHTML;

$.ajax({

type: 'POST',

async: false,

url: 'Handler/LoginAuthorize.ashx?Action=htmlDY',

data: {

TDold: tdold, TDnew: tdnew

},

success: function () {

}

});

});

td.text("");

td.append(txt);

});

后台：

private string htmlDY(HttpContext context)

{

try

{

string sourceurl="E:\\SafetyEvaluationXuWei\\trunk\\Smart.SafetyEvaluationXuWei.Web\\weight.htm";

System.Net.WebRequest myRequest = System.Net.WebRequest.Create(sourceurl);

System.Net.WebResponse myResponse = myRequest.GetResponse();

using (Stream stream = myResponse.GetResponseStream())

{

using (StreamReader sr = new StreamReader(stream, Encoding.GetEncoding("utf-8")))

{

string TDold = context.Request.Form["TDold"];

string TDnew = context.Request.Form["TDnew"];

string html= sr.ReadToEnd();

sr.Close();

html = Regex.Replace(html, TDold, TDnew, RegexOptions.IgnoreCase);

using (StreamWriter sw = new StreamWriter(sourceurl, false, Encoding.GetEncoding("utf-8")))

{

sw.WriteLine(html);

sw.Close();

}

return "true";

}

}

}

catch (Exception ex)

{ throw new Exception(string.Format("ReadHtml出错：{0}", ex.Message)); }

}

## Iframe 嵌套后有双重滚动条问题

里面不要滚动 外面只要垂直滚动 然后固定iframe垂直高度

子页面

<style>

html { overflow:hidden;}

</style>

主页面

<style>

html{ overflow-x:hidden;}

</style>

<iframe id="iframePage" name="iframePage" style="border: 0;width:100%;height:1480px">

</iframe>

## 内联样式修改菜单选中不选中



$(".link").click(function () {

$('.link').attr("style", "color:#30a5ff;background-color:white!important");

$(this).attr("style", "color:white!important;background-color:#30a5ff");

});

## datatable 日期为空处理

{

"mData": "SOLVETIME", "sTitle": "解决时间", "sWidth": "8%",

"mRender": function (data, type, full) {

if (data) {

var dtStart = full.SOLVETIME;

var dtStartWrapper = moment(dtStart);

return dtStartWrapper.format('YYYY-MM-DD');

} else {

return "";

}

}

},

## Ajax 调用rest服务 后绑定数据到dataTable

var url = "http://10.168.17.43:802/gcloud/operat/companyInfo/v1?format=json";

$.getJSON(url, function (result) {

var queryData = [];

if (result.status == 200) {

$.each(result.data, function (i, val) {

var temp = {};

temp.ID = val.ID;

temp.RECORDCREATETIME = val.RECORDCREATETIME;

temp.RECORDUPDATETIME = val.RECORDUPDATETIME;

temp.DELETEFLAG = val.DELETEFLAG;

temp.PERSONINCHARGE = val.PERSONINCHARGE;

temp.LEGALPERSON = val.LEGALPERSON;

temp.COMPANYADDRESS = val.COMPANYADDRESS;

temp.ORGANIZATIONCODE = val.ORGANIZATIONCODE;

queryData.push(temp);

})

InitTable(queryData);

}

else {

alert(result.data);

}

});



## Ajax 调用rest服务 添加实例(一)

$.ajax({

url: "http://10.168.17.43:802/gcloud/operat/companyInfoAdd/v1",

type: "GET",

data:{

ORGANIZATIONCODE: ORGANIZATIONCODE, COMPANYNAME: COMPANYNAME, NATUREOFCOMPANY: NATUREOFCOMPANY, LEGALPERSON: LEGALPERSON,

PIPETYPE: PIPETYPE, PERSONINCHARGE: PERSONINCHARGE, PIPELENGTH: PIPELENGTH, COMPANYFAX: COMPANYFAX,

CONTACTNUMBER: CONTACTNUMBER, COMPANYMAIL: COMPANYMAIL, COMPANYADDRESS: COMPANYADDRESS

},

dataType: "json",

success: function (result) {

if (result.status == 200) {

alert(result.data);

if (result.data=="")

noty({ text: "删除成功！", type: "success", layout: "topCenter", timeout: 1000 });

else

noty({ text: "删除失败！", type: "error", layout: "topCenter", timeout: 1000 });

}

else {

alert(result.data);

}

},

error: function (result) {

alert(result.data);

}

});

## Ajax 调用handler服务 添加实例(二)

E:\本人参与开发项目\综合管廊\Smart.ComprePipeGalleryXuWei\ams\pages\company

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

if (!$("#mainForm").valid()) {

return;

}

var datas = $("#mainForm").serialize();

$.ajax({

type: "post",

url: "companymsg.ashx?Action=Add",

cache: false,

dataType: "text",

data:datas,

success: function (data) {

if (data == "True") {

top.noty({ text: "添加成功！", type: "success", layout: "topCenter", timeout: 1000 });

setTimeout(function () {

//关闭 刷新

parent.location.reload();

$.each(top.Robin.Window.instances, function (i, item) {

if (!item || !item.obj) {

return true;

}

var pnl = item.obj;

pnl.close();

});

}, 1000);

} else {

top.noty({ text: "添加失败！", type: "error", layout: "topCenter" });

}

}

});

});

## dataTable 添加数据后刷新表格方式

1、 if (result.data == true) {

top.noty({ text: "添加成功！", type: "success", layout: "topCenter", timeout: 1000 });

setTimeout(function () {

//关闭 刷新

parent.location.reload();

$.each(top.Robin.Window.instances, function (i, item) {

if (!item || !item.obj) {

return true;

}

var pnl = item.obj;

pnl.close();

});

}, 1000);

} else {

top.noty({ text: "添加失败！", type: "error", layout: "topCenter" });

}

## dataTable 删除数据后刷新表格方式

1、success: function (result) {

if (result == "True") {

noty({ text: "删除成功！", type: "success", layout: "topCenter", timeout: 1000 });

oTable.fnClearTable(0);

oTable.fnDraw();

} else {

noty({ text: "删除失败！", type: "error", layout: "topCenter", timeout: 1000 });

}

}

2、if (typeInfo == 'success') {

var listpanel = parent.Robin.Window.GetInfoPanelByID("ProjectList");

listpanel.normalize();

listpanel.reloadContent();

window.setTimeout(function () {

top.Robin.CloseOneInfoPanel("DataManagerEdit");

}, 1000);

}

Robin.Window.normalizeInfoPanelByID=function(e){

$.each(Robin.Window.instances,function(t,n){

if(null!=n&&n.id==e){

var i=n.obj;

i.normalize()}})}

Robin.Window.GetInfoPanelByID=function(e){

var t=null;

return $.each(Robin.Window.instances,function(n,i){

return null!=i&&i.id==e?void(t=i.obj):void 0}),t}

## dataTable 带单选按钮例子

设置

columns: [

{

"mData": "NVFID", "sWidth": "5%", "sClass": "text-center", "fnCreatedCell": function (nTd, sData, oData, iRow, iCol) {

if (roleidList && roleidList.length > 0 && roleidList.in\_array(sData)) {

$(nTd).html("<input type='radio' name='checkList' checked='checked' value='" + sData + "' rolename='" + oData.DEPARTMENTNAME + "'>");

} else {

$(nTd).html("<input type='radio' name='checkList' value='" + sData + "' rolename='" + oData.DEPARTMENTNAME + "'>");

}

}

},

{ "mData": "ROLENAME", "sTitle": "角色名称", "sWidth": "20%" },

## dataTable 去掉x轴滚动条 "scrollable":false,

oTable = $('#table2').dataTable({

"oLanguage": {

"sUrl": "../../js/jQuery/Plugins/DataTables/dataTables.chinese.txt"

},

'bPaginate': true, //是否分页。

"bProcessing": true, //当datatable获取数据时候是否显示正在处理提示信息。

'bFilter': false, //是否使用内置的过滤功能。

"bSort": false,

"iDisplayLength": 8, //每页显示的行数

"bLengthChange": false,//允许用户从一个选择菜单中格式化页面大小(改变页面显示的数据数量)

//'sPaginationType': 'full\_numbers', //分页样式

//"sScrollY": "140",

//"sScrollX": "88%",

"sScrollX": "99%",

"sScrollXInner": "99%",

"scrollable":false,

"aoColumns": colomns,

"bAutoWidth": true,

"aoColumnDefs": columnDefs,

"aaData": result

});

datatable 表头禁止换行：

1. .dataTable th {
2. white-space: nowrap !important;
3. }

## dataTable 多选按钮例子

function multiConfig() {

var str = '';

$("input[name='checkList']:checked").each(function (i, o) {

str += $(this).val();

str += ",";

});

if (str.length > 0) {

var IDS = str.substr(0, str.length - 1);

var datasource = datasourceValue.replace(/-/g, "=");

var pamas = "&DATASOURCENAME=" + datasourceName + "&DATASOURCEVALUE=" + datasource + "&TOPICID=" + topicId + "&GROUPID=" + groupId;

var url = Robin.Setting.GlobalSetting.EntityConfigWeb + Robin.Setting.GlobalSetting.APP\_DATA\_TABLE\_MultiConfig + "?IDS=" + IDS+pamas;

$.ajax({

url: url,

method: "POST",

dataType: "json",

async: false,

success: function (result) {

if (result === "true") {

noty({ text: "批量配置实体成功！", type: "success", layout: "topCenter", timeout: 2000 });

GetAllTablesByDatasource(datasourceValue);

}

},

error: function (ex) {

var exception = ex;

}

});

} else {

alert("至少选择一条记录操作");

}

}

$("#checkAll").on("click", null, function () {

if ($(this).is(":checked")) {

$("input[name='checkList']").prop("checked", true);

} else {

$("input[name='checkList']").prop("checked", false);

}

});

## 获取复选框选中的值

function Link() {

var roleID = $("input[name='checkList']:checked").val();

//var roleNAME = $("input[name='checkList']:checked")[0].getAttribute('DEPARTMENTNAME');

if (roleID != null && roleID != "undefined" && roleID.length > 0) {

$.ajax({

type: 'POST',

url: top.Robin.Setting.GlobalSetting.AuthorityService + "TenantDepartmentUserAdd/v1",

dataType: 'json',

data: {

"TENANTID": TENANTID, "USERID": USERID, "DEPARTMENTID": roleID,

"RECORDCREATETIME": new Date(), "DELETEFLAG": 0

},

success: function (result) {

if (result == "true") {

noty({ text: "关联成功！", type: "success", layout: "topCenter", timeout: 2000 });

parent.location.reload();

//关闭本窗口

parent.closeWin();

//更新父窗口，可以调用父窗口的方法

} else {

noty({ text: "关联失败！", type: "error", layout: "topCenter", timeout: 2000 });

}

}

});

} else {

alert("至少选择一条记录操作");

}

## } dataTable 带多选按钮checkbox例子

设置

columns: [

{

"mData": "NVFID", "sWidth": "5%", "sClass": "text-center", "fnCreatedCell": function (nTd, sData, oData, iRow, iCol) {

if (roleidList && roleidList.length > 0 && roleidList.in\_array(sData)) {

$(nTd).html("<input type='checkbox' name='checkList' checked='checked' value='" + sData + "'>");

} else {

$(nTd).html("<input type='checkbox' name='checkList' value='" + sData + "'>");

}

}

},

{ "mData": "ROLENAME", "sTitle": "角色名称", "sWidth": "20%" },

获取值

function sure() {

var str = '';

$("input[name='checkList']:checked").each(function (i, o) {

str += $(this).val();

str += ",";

});

if (str.length > 0) {

var IDS = str.substr(0, str.length - 1);

//alert("你要关联的数据集id为" + IDS);

$.ajax({

url: "Handler.ashx?Action=LinkRoleSelect&idList=" + IDS + "&USERID=" + USERID,

success: function (result) {

if (result == "true") {

noty({ text: "关联成功！", type: "success", layout: "topCenter", timeout: 2000 });

//关闭本窗口

parent.closeWin();

//更新父窗口，可以调用父窗口的方法

} else {

noty({ text: "关联失败！", type: "error", layout: "topCenter", timeout: 2000 });

}

}

});

} else {

alert("至少选择一条记录操作");

}

}

## 递归方法

private string HandleDeptALLRequest(HttpContext context)

{

DGCDepart bll = new DGCDepart();

List<MGCDepart> ls = bll.GetLists(new WhereClip("1=1"));

List<MGCDeparts> list = new List<MGCDeparts>();

MGCDeparts model = null;

for (var i = 0; i < ls.Count; i++)

{

if (ls[i].PARENTID == "0")

{

model = new MGCDeparts(ls[i]);

model.childens = getdata(ls,ls[i].NVFID);

}

}

try

{

return JsonHelper.SerializeObject(model);

}

catch (Exception ex)

{

return string.Empty;

}

}

/// <summary>

/// 获取PID下的list

/// </summary>

/// <returns></returns>

private List<MGCDeparts> getdata(List<MGCDepart> data, string pid)

{

List<MGCDeparts> models = new List<MGCDeparts>();

foreach (var item1 in data)

{

if (item1.PARENTID == pid)

{

var model1 = new MGCDeparts(item1);

model1.childens = getdata(data, model1.NVFID);

models.Add(model1);

}

}

return models;

}

## js实现获取当前月份前的12个月份

var last\_year\_month = function () {

var d = new Date();

var result = [];

for (var i = 0; i < 12; i++) {

d.setMonth(d.getMonth() - 1);

var m = d.getMonth() + 1;

m = m < 10 ? "0" + m : m;

//在这里可以自定义输出的日期格式

// result.push(d.getFullYear() + “年" + m+”月“);

result.push(d.getFullYear() + "-" + m);

}

return result;

}

var month = last\_year\_month().sort();

## 统计图 ajax获取数据后绑定12个月份的数据

var dataDone = {

name: '未处理案件',

data: []

}

$.ajax({

type: 'GET',

url: Robin.Setting.GlobalSetting.SZGLDDAPIService + "/Authority/AJXXDoneGroupByMonth/v1",

dataType: 'json',

async: false,

success: function (resultDone) {

if (resultDone != null && resultDone.length > 0) {

for (var j = 0; j < 12; j++) {

var exist=false;

$.each(resultDone, function (i, v) {

if (v.TIME == month[j])

{

dataDone.data.push(parseInt(v.COUNT));

exist = true;

}

});

if (exist == false)

{ dataDone.data.push(0); }

}

}

},

error: function (er) {

alert(er);

}

});

## js 列表数据去掉重复 查询distinct数据（数据库分组查询可简化）

var tagKeys = [];

$.each(result, function (i, v) {

var flag = false;

$.each(tagKeys, function (ii, vv) {

if (vv.TAGKEY == v.TAGKEY) {

flag = true;

return false;

}

});

if (!flag) {

tagKeys.push(v);

}

});

## Json属性名称转换成大写 大小写转换

$.each(reply, function (i) {

reply[i] = upperJSONKey(reply[i]);

});

function upperJSONKey(jsonObj) {

for (var key in jsonObj) {

jsonObj[key.toUpperCase()] = jsonObj[key];

delete (jsonObj[key]);

}

return jsonObj;

}

## 去掉字符串最后一个字符

sql = sql.Substring(0, sql.Length - 1);

## 选取坐标后 赋值到页面 坐标选点

$('#btnSelect').click(function () {

var listPanel = parent.Robin.Window.GetInfoPanelByID("accident");

if (listPanel != undefined) {

listPanel.minimize();

}

var page = parent.Robin.Window.GetInfoPanelByID("accidentAdd");

if (page != undefined) {

page.smallify();

}

parent.Robin.MapTool.getMapPoint(function (point) {

if (point) {

var x = point.x.toFixed(2);

var y = point.y.toFixed(2);

$('#X').val(x);

$('#Y').val(y);

page.normalize();

}

});

});

Robin.MapTool = {

//点击地图获取点坐标

getMapPoint: function (a) {

handler = dojo.connect(Robin.Map.Map2DControl, "onClick", function (event) {

Robin.Map.ClearLayer(Robin.Map.Map2DControl, "PointLayer");

try {

a(event.mapPoint);

dojo.disconnect(handler);//事件值执行一次

} catch (err) { }

})

},

## 菜单动态变换样式

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

var $this = $(this), id = $this.attr('id');

var obj = $('.mainMenu.selected'), haveId = false;

$.each(obj, function (i) {

//如果上一个id和当前选中id是同一个，操作的是自己

if (obj[i].id == id) {

haveId = true;

}

});

if (haveId) {

$this.removeClass('selected');

} else {

$this.addClass('selected');

}

});

**例子二**

$('.link').on('click', function () {

var $this = $(this), id = $this.attr('id');

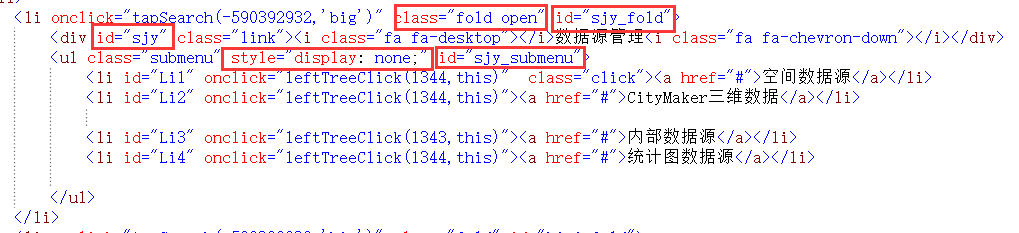
$('.fold').removeClass("open");

$('.submenu').hide();

$("#"+id+"\_fold").toggleClass("open");

$("#"+id+"\_submenu").fadeToggle();

});



<li class="fold open" id="sjy\_fold">

<div id="sjy" class="link"><i class="fa fa-desktop"></i>数据源管理<i class="fa fa-chevron-down"></i></div>

<ul class="submenu" style="display: none;" id="sjy\_submenu">

<li id="Li1" onclick="leftTreeClick(1344,this)" class="click"><a href="#">空间数据源</a></li>

<li id="Li2" onclick="leftTreeClick(1344,this)"><a href="#">CityMaker三维数据</a></li>

<li id="Li3" onclick="leftTreeClick(1343,this)"><a href="#">内部数据源</a></li>

<li id="Li4" onclick="leftTreeClick(1344,this)"><a href="#">统计图数据源</a></li>

</ul>

</li>

## Toggle用法

$('#tucengid').on('click', function () {

$("#foldid").toggleClass("open");

$("tucengid").toggleClass("link");

$("#submenu").fadeToggle();

});

## $(window).height()

/\*计算页面高度宽度,自动改变布局\*/

$(window).resize(function () {

//$("#Main\_Map\_DIV").height($(window).height() - $(".b\_header").height() - 50);

$("#Main\_Map\_DIV").height($(window).height());

});

//$("#Main\_Map\_DIV").height($(window).height() - $(".b\_header").height() - 2);

$("#Main\_Map\_DIV").height($(window).height());

//$("#Main\_Map\_DIV").width($(window).width());

## 前端静态页面序列化 $("#form").serializeJson();

/\*\*

\* 将form里面的内容序列化成json

\* 相同的checkbox用分号拼接起来

\* @param {obj} 需要拼接在后面的json对象

\* @method serializeJson

\* \*/

$.fn.serializeJson = function (otherString) {

var serializeObj = {},

array = this.serializeArray();

$(array).each(function () {

if (serializeObj[this.name]) {

serializeObj[this.name] += ';' + this.value;

} else {

serializeObj[this.name] = this.value;

}

});

if (otherString != undefined) {

var otherArray = otherString.split(';');

$(otherArray).each(function () {

var otherSplitArray = this.split(':');

serializeObj[otherSplitArray[0]] = otherSplitArray[1];

});

}

return serializeObj;

};

/\*\*

\* 将form里面的内容序列化成QueryString

\* @method serializeQuerystring

\* \*/

$.fn.serializeQuerystring = function () {

var str = this.serialize();

return str;

};

/\*\*

\* 将form里面的内容序列化成数组

\* @method serializeList

\* \*/

$.fn.serializeList = function () {

var t = $('form').serializeArray();

return t;

}

/\*\*

\* 将josn对象赋值给form

\* @param {jsonValue} 需要给form赋值的json对象

\* @method serializeJson

\* \*/

$.fn.setForm = function (jsonValue) {

var obj = this;

$.each(jsonValue, function (name, ival) {

var $oinput = obj.find("input[name=" + name + "]");

if ($oinput.attr("type") == "checkbox") {

if (ival !== null) {

var checkboxObj = $("[name=" + name + "]");

var checkArray = ival.split(";");

for (var i = 0; i < checkboxObj.length; i++) {

for (var j = 0; j < checkArray.length; j++) {

if (checkboxObj[i].value == checkArray[j]) {

checkboxObj[i].click();

}

}

}

}

}

else if ($oinput.attr("type") == "radio") {

$oinput.each(function () {

var radioObj = $("[name=" + name + "]");

for (var i = 0; i < radioObj.length; i++) {

if (radioObj[i].value == ival) {

radioObj[i].click();

}

}

});

}

else if ($oinput.attr("type") == "textarea") {

obj.find("[name=" + name + "]").html(ival);

}

else {

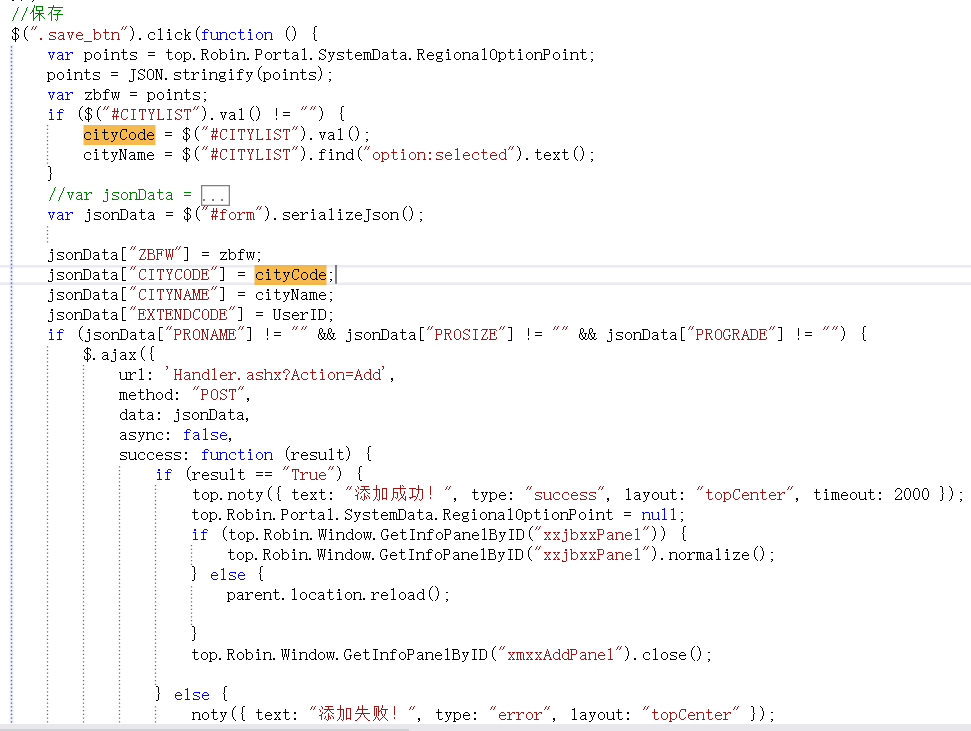
obj.find("[name=" + name + "]").val(ival);

obj.find("label[name=" + name + "]").text(ival);

}

})

}



**后端对静态页面进行反序列话**

**例子一**

//静态帮助类

public static class UtilityHelper

{

/// <summary>

/// 将HTTP上下文表单内容转为实体对象

/// </summary>

/// <typeparam name="T"></typeparam>

/// <param name="context"></param>

/// <returns></returns>

public static T RequestToModel<T>(this HttpContext context) where T : new()

{

T model = new T();

Type type = model.GetType();

PropertyInfo[] Fields = type.GetProperties();

for (int i = 0; i < Fields.Count(); i++)

{

try

{

var name = Fields[i].Name;

Type vtype = Fields[i].PropertyType;

string typeFullName = vtype.FullName.ToLower();

var value = context.Request[name];

if (value != null)

{

if (typeFullName.IndexOf("nt") >= 0 )//int

{

var v = Convert.ToInt32(value);

Fields[i].SetValue(model, v);

continue;

}

if (typeFullName.IndexOf("ecimal") > 0)//Decimal

{

var v = Convert.ToDecimal(value);

Fields[i].SetValue(model, v);

continue;

}

if (typeFullName.IndexOf("tring") >= 0)//string

{

var v = Convert.ToString(value);

Fields[i].SetValue(model, v);

continue;

}

if (typeFullName.IndexOf("ate") >= 0)//datetime

{

var v = Convert.ToDateTime(value);

Fields[i].SetValue(model, v);

}

}

}

catch (Exception ex)

{

LogManager.DefaultLogger.Error(ex);

}

}

return model;

}

}



**例子二**

public static class RequestModelHelper

{

/// <summary>

///

/// </summary>

/// <typeparam name="T"></typeparam>

/// <param name="M"></param>

/// <returns>T</returns>

public static T RequestToModel<T,M>(M m) where T : new()

{

T model = new T();

Type Ttype = typeof(T);

Type Mtype = typeof(M);

PropertyInfo[] mFieldInfo = Mtype.GetProperties();

PropertyInfo[] tFieldInfo = Ttype.GetProperties();

for (int i = 0; i < tFieldInfo.Length; i++)

{

try

{

var name = tFieldInfo[i].Name;

Type vtype = tFieldInfo[i].PropertyType;

string typeFullName = vtype.FullName.ToLower();

if (mFieldInfo.Length > 0)

{

for (int j = 0; j < mFieldInfo.Length; j++)

{

string MName = mFieldInfo[j].Name;

if (name == MName)

{

var value = mFieldInfo[j].GetValue(m);

if (value != null)

{

if (typeFullName.IndexOf("nt") >= 0)//int

{

var v = Convert.ToInt32(value);

tFieldInfo[i].SetValue(model, v);

continue;

}

if (typeFullName.IndexOf("ecimal") > 0)//Decimal

{

var v = Convert.ToDecimal(value);

tFieldInfo[i].SetValue(model, v);

continue;

}

if (typeFullName.IndexOf("tring") >= 0)//string

{

var v = Convert.ToString(value);

tFieldInfo[i].SetValue(model, v);

continue;

}

if (typeFullName.IndexOf("ate") >= 0)//datetime

{

var v = Convert.ToDateTime(value);

tFieldInfo[i].SetValue(model, v);

}

}

}

}

}

}

catch (Exception ex)

{

LogManager.DefaultLogger.Error(ex);

}

}

return model;

}

}



## Prettify 代码块添加样式

$("pre").addClass("prettyprint");

prettyPrint();

<!--代码高亮开始-->

<script src="js/jQuery/Plugins/prettify/prettify.js"></script>

<link href="js/jQuery/Plugins/prettify/prettify.css" rel="stylesheet" />

## jQuery 动态筛选

$('#paTable tbody tr').hide().filter(":contains('" + cxtj + "')").show();

## Ajax.BeginForm提交页面后，onSuccess函数无反应问题解决；

**Mvc 异步提交 需要引用特定版本的jquery**





## 打开新窗体

window.showModalDialog("manhole/Business/TenantApprove/AddTenantApprove.htm", "approveid", "dialogHeight:600px; dialogWidth:900px; help: No; resizable: no; status: No; scroll:no; dialogTop:" + (screen.height - 600) / 2 + "px;dialogLeft:" + (screen.width - 900) / 2 + "px;");

window.open("manhole/Business/TenantApprove/AddTenantApprove.htm", "tenantApproveid", "height=600, width=800,top=150,left=350")

## json数据绑定动态colomn的dataTable

//淹没分析—小区

InitDataTable('../data/WaterMap/community\_report.json', 'community\_report');

//初始化绑定table

function InitDataTable(url, tableid) {

$('#' + tableid).empty();

var oTable = null;

var colomns = [];

$.getJSON(url, function (result) {

if (result != null) {

$.each(result[0], function (m, data) {

var attr = { "mData": m.toUpperCase(), "sTitle": m.toUpperCase(), "sClass": "text-center", "width": "77px" };

colomns.push(attr);

});

}

if (oTable != null) {

oTable.fnDestroy();

oTable = null;

};

oTable = $('#' + tableid).dataTable({

"oLanguage": {

"sUrl": "../js/jQuery/Plugins/DataTables/dataTables.chinese.txt"

},

'bPaginate': false, //是否分页。

"bProcessing": true, //当datatable获取数据时候是否显示正在处理提示信息。

'bFilter': false, //是否使用内置的过滤功能。

"bSort": false,

"iDisplayLength": 4, //每页显示的行数

"bLengthChange": false,//允许用户从一个选择菜单中格式化页面大小(改变页面显示的数据数量)

'sPaginationType': 'full\_numbers', //分页样式

"sScrollY": "160",

"sScrollX": false,

"aoColumns": colomns,

"bAutoWidth": false,

"aaData": result

});

});

}

## Getjson 同步设置 ajax 缓存

$.ajaxSettings.async = false;

$.getJSON("../ResourceManagement/treedata.json", function (result) {

if (result.length > 0) {

treedata = result;

}

})

## Mvc 提交方式总结

@using (Ajax.BeginForm("Update", "Facility", new AjaxOptions

{

OnSuccess = "Show",

HttpMethod = "Post"

}, new { @class = "smart-form", @id = "mainForm", enctype = "multipart/form-data" }))

@using (Html.BeginForm("Update", "Facility", FormMethod.Post, new { @class = "smart-form", id = "mainForm", enctype = "multipart/form-data" }))

后台处理

string \_RedirectJSString = @"

<script src='../../Scripts/jquery-1.7.1.js'></script>

<script src='../../Scripts/Noty/packaged/jquery.noty.packaged.min.js'></script>

<script type='text/javascript'>

$(function(){

setTimeout(function ()window.location.href = '/Facility/Index'; }, 4000);

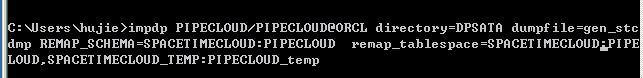
{ noty({ text: '保存成功', type: 'success', layout: 'topCenter', timeout: 2000 });

})</script>";

Response.Write(\_RedirectJSString);

## 数据库导表 包括表关系 index 等信息 需要在数据库服务器上操作

C:\Users\JiangLina\Documents\Tencent Files\498687387\Image\C2C\Image1\LCUZ1}95V{ZW`OI%H77R[CC.png



expdp SPACETIMECLOUD/SPACETIMECLOUD@172.30.17.212:1521/orcl dumpfile=gen\_stc2.dmp directory=DPSATA tables=APP\_CODE,APP\_CODEGROUP,APP\_DATA\_ENUM,APP\_DATA\_ENUMGROUP, APP\_DATA\_FIELD, APP\_DATA\_TABLE, APP\_DATA\_TABLEGROUP,SYS\_RESOURCE,SYS\_RESOURCE\_TOPIC

impdp PIPECLOUD/PIPECLOUD@172.30.17.212:1521/orcl directory=DPSATA dumpfile=gen\_stc2.dmp REMAP\_SCHEMA=SPACETIMECLOUD:PIPECLOUD REMAP\_TABLESPACE=SPACETIMECLOUD:PIPECLOUD,SPACETIMECLOUD\_TEMP:PIPECLOUD\_TEMP

后台拼sql 模糊查询

public class MGCResourceQueryService : Service

{

DGCResource dal = new DGCResource();

public List<GEN\_PC\_RESOURCE> Get(MGCResourceQuery Requestmodel)

{

List<GEN\_PC\_RESOURCE> list = null;

switch (Requestmodel.filter)

{

case -1:

{

list = dal.GetListBySQL("1=1");

;

}; break;

case 0:

{

string filter = Request.QueryString["sql"];

string sql = " instr(SOURCENAME, '" + filter + "') > 0 or";

sql += " instr(TYPENAME, '" + filter + "') > 0 or";

sql += " instr(ADDRESSURL, '" + filter + "') > 0 or";

sql += " instr(SERVICETYPE, '" + filter + "') > 0 or";

sql += " instr(CITYNAME, '" + filter + "') > 0 or";

sql += " instr(ISSAFE, '" + filter + "') > 0 or";

sql += " instr(CREATETIME, '" + filter + "') > 0 or";

sql += " instr(UPDATETIME, '" + filter + "') > 0";

list = dal.GetListBySQL(sql);

}; break;

case 1:

{

JavaScriptSerializer json = new JavaScriptSerializer();

string sql = Request.QueryString["sql"];

QueryCondition query = json.Deserialize<QueryCondition>(sql);

sql = query.CreateSql();

list = dal.GetListBySQL(sql);

}; break;

default: return null;

}

return list;

}

}

## 定时执行事件

System.Timers.Timer t = new System.Timers.Timer(Double.Parse(ConfigurationManager.AppSettings["Timeout"]));//实例化Timer类，设置间隔时间为10000毫秒；

t.Elapsed += new System.Timers.ElapsedEventHandler(pBsso.ClearTokenValue);//到达时间的时候执行事件；

t.AutoReset = true;//设置是执行一次（false）还是一直执行(true)；

t.Enabled = true;//是否执行System.Timers.Timer.Elapsed事件；

## c# 合并数组

public string[] MergerArray(string[] First, string[] Second)

{

string[] result = new string[First.Length + Second.Length];

First.CopyTo(result, 0);

Second.CopyTo(result, First.Length);

return result;

}

## c# 数组去重

listPro = listPro.Where((x, i) => listPro.FindIndex(z => z.PRODUCTID == x.PRODUCTID) == i).ToList();