**CNKIPageJs分页控件使用说明**

CNKIPageJs分页控件里面使用了jQuery,这样为了注册相关事件,适应各个版本的浏览器,因此在使用时必须引入jQuery。

CNKIPageJs有如下部分组成:

CNKIPager.css 分页组件的样式文件。

CNKIPager.js 分页组件的核心代码。

1. **效果图如下**

****

1. **使用说明如下：**
   1. 在使用的页面上引入jQuery.js，cnkiPager.css，cnkiPager.js 文件。
   2. 在需要分页的地方添加一个div容器,设置容器的id和样式class=”cnkipager”。

<div id="topPager" class="cnkipager">

</div>

* 1. 在页面写入js代码,生成分页控件。

<script type="text/javascript">

var topPager = new CNKIPager(

"topPager",

10);

topPager.totalRecords = 50;

topPager.isShowGoPage = true;

topPager.generPagerControl(topPager);

</script>

1. **分页控件CNKIPageJs 方法说明**

|  |  |  |  |
| --- | --- | --- | --- |
| **方法名称** | **参数** | **方法说明** | **使用案例** |
| selectPage | 数字类型,第几页 | 不刷新页面直接手动调用选中某一页码,参数为第几页 | topPager.selectPage(1); |
| click | 数字类型,第几页 | 页码单击事件处理函数,参数n为页码 | topPager.click=function(n){}  一般情况click方法会和selectPage两者结合使用  topPager.click = function(n){  topPager.selectPage(n);} |
| currentPageIndex | 无 | 获取当前的页码 | var pageIndex=  topPager.currentPageIndex(); |
| currentPageSize | 无 | 获取每页显示的行数,  如果需要每页显示的行数,直接调用这个方法即可。这样设计是为了以后如果修改了每页显示的行数,不必修改多个地方 | var pageSize=  topPager.currentPageSize() |
| generPagerControl | 生成控件的配置项,传递实例名称即可 | 生成分页控件，显示到页面上 | topPager.generPagerControl(topPager); |
|  |  |  |  |

# 4．使用案例(按照上图进行)

使用jQuery、knockoutjs、cnkiPager 进行讲解。

前台页面 CNKIPagerTest.aspx

<head runat="server">

<title></title>

<%-- 引入相关JS库--%>

<script src="script/jQuery.js" type="text/javascript"></script>

<script src="script/cnkiPagerJs/cnkiPager.js" type="text/javascript"></script>

<link href="script/cnkiPagerJs/cnkiPager.css" rel="stylesheet" type="text/css" />

<script src="script/knockoutjs.js" type="text/javascript"></script>

</head>

<body>

<form id="form1" runat="server">

<%-- 设置顶部的分页控件容器--%>

<div id="topPager" class="cnkipager">

</div>

<%--显示数据的table --%>

<table>

<tr>

<th>

编号

</th>

<th>

姓名

</th>

</tr>

<tr data-bind="visible:isDataLoading">

<td colspan="2" align="center">

<img src="images/loading.gif" />数据加载中...

</td>

</tr>

<%--<!-- ko foreach: searchResultList -->--%>

<tbody data-bind="foreach: searchResultList">

<tr>

<td data-bind="text:Id">

</td>

<td data-bind="text:Name">

</td>

</tr>

</tbody>

<%-- <!-- /ko -->--%>

</table>

<%-- 设置底部的分页控件容器--%>

<div id="bottomPager" class="cnkipager">

</div>

<input type="button" value="查询" data-bind="click:search" style="width: 80px; height: 20px" />

</form>

<script type="text/javascript">

var topPager = new CNKIPager("topPager", 10);

topPager.isShowGoPage = true;

topPager.generPagerControl(topPager);

var bottomPager = new CNKIPager("bottomPager", 10);

bottomPager.generPagerControl(bottomPager);

//定义表格数据模型

function tableDataModel() {

var self = this;

self.Id = ko.observable("");

self.Name = ko.observable("");

}

function viewDataModel() {

var self = this;

self.url = "CNKIPagerTest.aspx";

self.Id = ko.observable("");

self.Name = ko.observable("");

this.searchResultList = ko.observableArray([]);

this.searchParams = {};

this.totalCount = ko.observable(0);

this.isDataLoading = ko.observable(true);

//表格顶部分页器

self.topPager = topPager;

self.topPager.click = function(n) {

self.pageIndex = n;

self.topPager.selectPage(n);

self.bottomPager.selectPage(n);

self.getResult(function() {

});

};

//表格底部分页器

self.bottomPager = bottomPager;

self.bottomPager.click = function(n) {

self.pageIndex = n;

self.topPager.selectPage(n);

self.bottomPager.selectPage(n);

self.getResult(function() {

});

};

self.load = function() {

self.searchParams = { type: 1 }

self.getResult(function() {

topPager.totalRecords = bottomPager.totalRecords = self.totalCount();

topPager.generPagerControl(topPager);

bottomPager.generPagerControl(bottomPager);

});

}

self.search = function() {

self.searchParams = { type: 1, Id: 1, Name: "cnki" };

self.getResult(function() {

topPager.totalRecords = bottomPager.totalRecords = self.totalCount();

topPager.generPagerControl(topPager);

bottomPager.generPagerControl(bottomPager);

});

};

self.getResult = function(callback, params) {

var url = self.url;

self.searchParams.pn = topPager.currentPageIndex();

self.searchParams.ps = topPager.currentPageSize();

self.isDataLoading(true);

self.totalCount(0);

$.get(

url,

self.searchParams,

function(result) {

var json = eval(result).cnkiResult;

self.totalCount(json[0].recordCount);

json.shift();

self.isDataLoading(false);

self.searchResultList([]);

$.each(json, function(index, item) {

var row = new tableDataModel();

row.Id(item.id);

row.Name(item.name);

self.searchResultList.push(row);

});

if (typeof callback === 'function') {

callback(params);

}

},

"json"

).fail(function() { alert("出现异常"); self.isDataLoading(false); });

}

};

var viewModel = new viewDataModel();

viewModel.load();

ko.applyBindings(viewModel);

</script>

</body>

后台处理CNKIPagerTest.aspx.cs

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

{

switch (Convert.ToString(Request.Params["type"]))

{

case "1":

GetResultList();

break;

}

}

private void GetResultList()

{

Response.Clear();

Response.ContentType = "text/json";

int pn =Convert.ToInt32( Request.Params["pn"]);

int ps = Convert.ToInt32(Request.Params["ps"]);

int recordsCount = 50;

string id = Request.Params["id"] ?? "";

string name = Request.Params["name"] ?? "";

int total = pn \* ps;

pn = (pn - 1) \* ps + 1;

if (total > recordsCount)

total = recordsCount;

DataTable dt = new DataTable();

dt.Columns.Add(new DataColumn() { DataType=typeof(int), ColumnName="id"});

dt.Columns.Add(new DataColumn() { DataType=typeof(string),ColumnName="name"});

DataRow dr = null;

if (id.Length > 0&& name.Length > 0)

{

recordsCount = 1;

dr = dt.NewRow();

dr["id"] =id;

dr["name"] =name;

dt.Rows.Add(dr);

}

else

{

for (int i = pn; i <= total; i++)

{

dr = dt.NewRow();

dr["id"] = i;

dr["name"] = "cnki" + i;

dt.Rows.Add(dr);

}

}

Response.Write(GetJsonFromDataTable(dt,recordsCount,"cnkiResult",""));

Response.End();

}

/// <summary>

/// dataTable转换json

/// </summary>

/// <param name="dt">格式化的DataTable</param>

/// <param name="recordCount">总数</param>

/// <param name="jsonName">json的名字</param>

/// <param name="timeFormat">时间格式化,默认为yyyy-MM-dd HH:mm:ss,如果没有时间类型可以选择不填写</param>

/// <returns>json串</returns>

private string GetJsonFromDataTable(DataTable dt, int recordCount, string jsonName, string timeFormat)

{

if ((timeFormat + "").Length == 0)

timeFormat = "yyyy-MM-dd HH:mm:ss";

StringBuilder Json = new StringBuilder();

Json.Append("{\"" + jsonName + "\":[");

Json.Append("{\"recordCount\":" + recordCount + "}");

if (dt != null && dt.Rows.Count > 0)

{

Json.Append(",");

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

{

Json.Append("{");

for (int j = 0; j < dt.Columns.Count; j++)

{

Json.Append("\"" + dt.Columns[j].ColumnName.ToString() + "\":\"");

if (dt.Columns[j].DataType == typeof(DateTime))

{

Json.Append(string.Format("{0:" + timeFormat + "}", dt.Rows[i][j]) + "\"");

}

else

{

Json.Append(Convert.ToString(dt.Rows[i][j]) + "\"");

}

if (j < dt.Columns.Count - 1)

{

Json.Append(",");

}

}

Json.Append("}");

if (i < dt.Rows.Count - 1)

{

Json.Append(",");

}

}

}

Json.Append("]}");

return Json.ToString();

}