

```
1 檔案
2  classManage.aspx
3
4  <%@ Page Language="C#" AutoEventWireup="true" CodeFile="classManage.aspx.cs"
5     Inherits="classManage" %>
6
7  <!DOCTYPE html>
8
9  <html xmlns="http://www.w3.org/1999/xhtml">
10 <head runat="server">
11     <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
12     <title></title>
13     <script src="js/jquery.min.js"></script>
14     <script src="js/kendo.all.min.js"></script>
15     <link href="styles/kendo.default.min.css" rel="stylesheet" />
16     <link href="styles/kendo.common.min.css" rel="stylesheet" />
17     <style>
18         .btn-destroy {
19             background-color: #fbc2be;
20         }
21
22         .btn-update {
23             background-color: #b6ff00;
24         }
25
26         #stClass_grid tbody {
27             text-align: center;
28         }
29
30         #stClass_grid th.k-header {
31             text-align: center;
32             font-weight: bold;
33             color: #3f0380;
34         }
35
36         /* 設定滑鼠移到列上換色*/
37         #stClass_grid tbody tr:hover {
38             background: #fefdc3;
39         }
40
41         .border_div{
42             margin:10px;
43             padding:10px;
44             border:1px solid #ff0000;
45             border-radius:10px;
46         }
47
48         fieldset{
49             background-color:#f3faca;
50             padding:10px;
51         }
52     </style>
53     <script type="text/javascript">
54         $(function(){
55             //group_grid元件程式
56             var groupdataSrc = new kendo.data.DataSource({
57                 transport: { //資料操作的CRUD設定
58                     read: {
59                         type: "POST",
60                         url: "./ASHX/group_grid.ashx",
61                         dataType: "json"
62                     },
63                     destroy: {
64                         type: "POST",
65                         url: "./ASHX/group_grid.ashx",
66                         dataType: "json",
67                         data: { Operator: "Delete" }
68                     }
69                 }
70             });
```

```

68     },
69     schema: {
70         data: function (d) { return d.Data; },
           //取出資料陣列，d為最上層物件，d之中的Data包含顧客資料表的資料
71         total: function (d) { return d.TotalCount; },
           //取出資料總筆數(計算頁數用)，d之中的TotalCount為Data的筆數
72         model: { //設定資料模型
73             id: "cl_id",
74             fields: {
75                 cl_id: { editable: false },
76                 cl_year: { editable: false },
77                 cl_name: { editable: false }
78             }
79         }
80     },
81 },
82
83     pageSize: 10, //Grid的每頁資料筆數
84     serverPaging: true,
           //使用server端的分頁，點選分頁將會重新綁定DataSource給Grid
85     serverSorting: true
           //使用server端的排序，點選排序將會重新綁定DataSource給Grid
86
87 });
88
89 $("#stClass_grid").kendoGrid({ //對div的套用Grid widget
90     autoBind: true,
91     width: "100%",
92     dataSource: groupdataSrc, //指定資料來源
93     scrollable: true,
94     pageable: {
95         input: true,
96         numeric: false,
97         refresh: true,
98         messages: {
99             display: "顯示{0}-{1}筆，共{2}筆",
100            empty: "沒有資料",
101            page: "頁",
102            of: "/ {0}",
103            itemsPerPage: "筆/頁",
104            first: "第一頁",
105            previous: "前一頁",
106            next: "下一頁",
107            last: "最後一頁",
108            refresh: "刷新"
109        }
110    },
111    columns: //Grid的資料欄位
112    [
113        {
114            field: "cl_year",
115            width: "12%",
116            title: "學年度",
117            attributes: {
118                style: "text-align:center;"
119            }
120        },
121        {
122            field: "cl_name",
123            width: "30%",
124            title: "班級名稱",
125            attributes: {
126                style: "text-align:center;"
127            }
128        }
129    ],

```

```

129
130     {
131         command: [
132             //{ className: "btn-update", name: "edit",
133             text: "修改" }, // 使用預設的修改按鈕
134             { className: "btn-destroy", name: "destroy",
135             text: "刪除" } //使用預設的刪除按鈕
136         ],
137         // 設定指令欄位的標題名稱
138         title: " ",
139         width: "10%"
140     },
141
142     ],
143     editable: {
144         mode: "popup", //使用Popup的模式編輯資料
145         confirmation: "您確定要刪除此班級資料嗎?",
146         update: true, //
147         //若設定為True,當按下Update即更新server資料
148     }
149 });
150
151 $("#stClass_grid").on("click", ".k-grid-edit , .k-grid-add",
152 function () { //修改Popup視窗中的Button文字
153     $('.k-grid-update').text('儲存');
154     $('.k-grid-cancel').text('取消');
155     $('.k-window-title').text('班級管理');
156 });
157
158
159 function class_save(){
160     var stclass=$("#txt_lyear").val();
161     var stcname=$("#txt_cname").val();
162
163     if(stcname != "")
164     {
165         var gc_obj= new Object(); //定義一個物件
166         gc_obj.st_class=stclass; //給予各欄位值
167         gc_obj.st_cname=stcname;
168
169         var array=[]; //建立一個陣列
170         array.push(gc_obj); //將物件放入陣列中
171
172         $.ajax({
173             type: "POST",
174             url: "../ASHX/insert_card.ashx",
175             contentType: "application/json; charset=utf-8",
176             dataType: "json",
177             data: JSON.stringify(array), //陣列轉換成json格式
178             success: function (data) {
179
180                 alert(data.Message);
181             },
182             error: function (jqXHR, textStatus, errorThrown) {
183
184             }
185         });
186     }
187     else{
188         alert("請輸入姓名!!");
189     }
190 }
191 </script>
192 </head>

```

```

193 <body>
194     <form id="form1" runat="server">
195         <div>
196             <fieldset>
197                 <legend>新增班級資料</legend>
198                 <label>學年度:</label>
199                 <input id="txt_lyear" class="k-textbox" />
200                 <label>班級名稱:</label>
201                 <input id="txt_cname" class="k-textbox" />
202                 <button id="btn_save" type="button" class="k-button" onclick=
                "class_save()">儲存</button>
203             </fieldset>
204
205         </div>
206         <div class="border_div">
207             <div id="stClass_grid"></div>
208         </div>
209
210     </form>
211 </body>
212 </html>
213
214 檔案
215  .\ASHX\insert_card.ashx
216
217 <%@ WebHandler Language="C#" Class="insert_card" %>
218
219 using System;
220 using System.Web;
221 using System.Collections.Generic;
222 using System.Linq;
223 using Newtonsoft.Json;
224 using Newtonsoft.Json.Linq;
225 using System.Web.Script.Serialization;
226
227
228
229 public class insert_card : IHttpHandler
230 {
231     public class stClassObj
232     {
233         public string st_class;
234         public string st_cname;
235     }
236
237     static List<stClassObj> _stClassObj = new List<stClassObj>();
238
239     //建立資料庫物件
240     studentDataContext sdc = new studentDataContext();
241
242     public void ProcessRequest(HttpContext context)
243     {
244         context.Response.ContentType = "text/plain";
245         string json = string.Empty;
246         string message = string.Empty;
247         bool result = false;
248
249         using (var reader = new System.IO.StreamReader(context.Request.
                InputStream))
250         {
251             json = reader.ReadToEnd();
252         }
253
254         if (!string.IsNullOrEmpty(json))
255         {
256             _stClassObj = JsonConvert.DeserializeObject<List<stClassObj>>(
                json);
257         }

```

```

258
259     var _c = _stClassObj.FirstOrDefault();
260
261
262     stClass sc = new stClass(); //建立實體資料表
263     sc.cl_year = Convert.ToInt32(_c.st_class);
264     sc.cl_name = _c.st_cname;
265
266     sdc.stClass.InsertOnSubmit(sc);
267     sdc.SubmitChanges();
268
269
270     result = true;
271     this.SendResponse(context, result, "成功建立一個班級類別資料!!");
272
273 }
274
275 public bool IsReusable
276 {
277     get
278     {
279         return false;
280     }
281 }
282
283 private void SendResponse(HttpContext context, bool result, string
message)
284 {
285     context.Response.Write(JsonConvert.SerializeObject(
286         new
287         {
288             Result = result,
289             Message = message
290         }));
291
292     context.Response.End();
293 }
294
295 }
296
297
298 檔案
299 .\ASHX\group_grid.ashx
300
301 <%@ WebHandler Language="C#" Class="teacher_grid" %>
302
303 using System;
304 using System.Web;
305 using System.Linq;
306 using Newtonsoft.Json;
307 using Newtonsoft.Json.Linq;
308 using System.Collections.Generic;
309 using System.IO;
310
311 public class teacher_grid : IHttpHandler {
312     studentDataContext sdc = new studentDataContext();
313     //模擬資料物件
314     public class classInfo
315     {
316         public int cl_id;
317         public int cl_year;
318         public string cl_name;
319     }
320
321     public class ResultData
322     {
323         public object Data;
324         public int TotalCount;

```

```

325     }
326
327     static List<classInfo> _classInfo = null;
328     public void ProcessRequest (HttpContext context) {
329
330         //刪除功能
331         if(context.Request["Operator"] !=null)
332         {
333             string _operator = context.Request["Operator"];
334
335             switch (_operator)
336             {
337                 case "Delete":
338                     {
339                         var clid = Convert.ToInt32(context.Request["cl_id"]);
340                         var search_query = from g in sdc.stClass
341                                             where g.cl_id==clid
342                                             select g;
343
344                         if(search_query.FirstOrDefault() !=null)
345                         {
346                             //刪除記錄
347                             sdc.stClass.DeleteOnSubmit(search_query.
348                                 SingleOrDefault());
349                             sdc.SubmitChanges();
350
351                         }
352                     }
353                 }
354             }
355
356         //查詢指令
357         var listBoard = from g in sdc.stClass
358                         orderby g.cl_id descending //遞減排序
359                         select g;
360
361         if (listBoard.FirstOrDefault() != null)
362         {
363             _classInfo = listBoard.Select(bo => new classInfo()
364             {
365                 cl_id=bo.cl_id,
366                 cl_year=Convert.ToInt32(bo.cl_lyear),
367                 cl_name=bo.cl_name
368             }).ToList();
369         }
370         else
371         {
372             _classInfo = listBoard.Select(bo => new classInfo()
373             {
374                 cl_id=0,
375                 cl_year=0,
376                 cl_name=""
377             }).ToList();
378         }
379
380
381         int pageSize = 20, take = 20, skip = 0;
382         //分頁設定預設值：分頁資料筆數、取出筆數、當前頁位置
383         int.TryParse(context.Request["pageSize"], out pageSize);
384         //取得分頁的資料筆數設定
385         int.TryParse(context.Request["take"], out take);
386         int.TryParse(context.Request["skip"], out skip);
387         var paged = _classInfo.Skip(skip).Take(take);
388
389         ResultData rd = new ResultData();
390         rd.Data = paged.ToList();
391         rd.TotalCount = _classInfo.Count();

```

```
390         string jsonString = JsonConvert.SerializeObject(rd);
391         context.Response.Write(jsonString);
392     }
393
394     public bool IsReusable {
395         get {
396             return false;
397         }
398     }
399
400 }
401
```