

# Example01~06

## CCR Ports and PortSets

### Example 01: Posting Items

```
1
2
3
4 using System;
5 using System.Collections.Generic;
6 using System.Text;
7 using Microsoft.Ccr.Core;
8
9 namespace Example
10 {
11     class Program
12     {
13         //Example 01: Posting Items
14         static void Main(string[] args)
15         {
16             // Create port that accepts instances of System.Int32
17             Port<int> portInt = new Port<int>();
18
19             // Display number of items to the console
20             Console.WriteLine(portInt.ItemCount);
21
22             // Add the number 10 to the port
23             portInt.Post(10);
24
25             // Display number of items to the console
26             Console.WriteLine(portInt.ItemCount);
27         }
28     }
29 }
```

### Example 02: Retrieving Items (passive queue)

```
30
31 using System;
32 using System.Collections.Generic;
33 using System.Text;
34 using Microsoft.Ccr.Core;
35
36 namespace Example
37 {
38     class Program
39     {
40         //Example 02: Retrieving Items (passive queue)
```

```

41     static void Main(string[] args)
42     {
43         /// <summary>
44         /// 取值的方式：
45         /// 被動取值(passive queue, Example 02)：將port當queue用，可以接受使用者用以非同步取值。
46         /// 主動取值(active queue, Example 03)：使用Arbiter主動接收，為CCR ports主要的使用方式。
47         /// </summary>
48
49         // Create port that accepts instances of System.Int32
50         var portInt = new Port<int>();
51
52         // Add the number 10 to the port
53         portInt.Post(10);
54
55         // Display number of items to the console
56         Console.WriteLine(portInt.ItemCount);
57
58         // retrieve the item using Test
59         int item;
60         // 如果取值成功，Test會回傳true，反之回傳false。
61         // 取得資料放於item內。
62         var hasItem = portInt.Test(out item);
63         if (hasItem)
64         {
65             Console.WriteLine("Found item in port:" + hasItem + " " + item);
66         }
67         int item2 = 3;
68         var hasItem2 = portInt.Test(out item2); // 再次取值，取值失敗，回傳null。
69         Console.WriteLine("Found item in port:" + hasItem2 + " " + item2 + "\n");
70
71         portInt.Post(11);
72         // alternative to using Test is just assignment of port to variable using
73         // implicit operator
74         Console.WriteLine(portInt.ItemCount);
75         //var nextItem = portInt; // 不指定變數型態，nextItem為port。
76         int nextItem = portInt; // 指定變數型態，nextItem為port中的資料，會將port裡的資料取出。
77         Console.WriteLine(portInt.ItemCount);
78         Console.WriteLine("Found item in port:" + nextItem);
79     }
80 }
81 }

```

## 82 Example 03: Retrieving Items (active queue)

```

83 using System;
84 using System.Collections.Generic;
85 using System.Text;
86 using Microsoft.Ccr.Core;
87
88 namespace Example

```

```

89 {
90     class Program
91     {
92         //Example 03: Retrieving Items (active queue)
93         static void Main(string[] args)
94         {
95             // Create port that accepts instances of System.Int32
96             var portInt = new Port<int>();
97
98             // Add the number 10 to the port
99             portInt.Post(10);
100
101             // Display number of items to the console
102             Console.WriteLine(portInt.ItemCount);
103
104             // create dispatcher and dispatcher queue for scheduling tasks
105             // Dispatcher與DispatcherQueue會在「Scheduling Tasks section」中再介紹
106             Dispatcher dispatcher = new Dispatcher();
107             DispatcherQueue taskQueue = new DispatcherQueue("sample queue", dispatcher);
108
109
110             // retrieve the item by attaching a one time receiver
111             // Arbiter.Activate method is used to register a simple receiver arbiter to the port
112             // Receiver與Arbiter之後再介紹
113             Arbiter.Activate(
114                 taskQueue,
115                 portInt.Receive(delegate(int item) // user delegate, anonymous method
116                 {
117                     // this code executes in parallel with the method that
118                     // activated it
119                     Console.WriteLine("Received item:" + item + "\n");
120                 }
121             ));
122             // any code below runs in parallel with delegate
123
124             // Inspecting Port State : useful inspecting methods and properties on the Port<>
125             //Console.WriteLine("ToString():\n" + portInt.ToString()); // The number of items in the
126             port and the hierarchy of the receivers associated with the port
127             //Console.WriteLine("ItemCount:\n" + portInt.ItemCount); // Number of items currently
128             queued
129         }
130     }
131 }

```

## 132 Example 04: Constructing a PortSet Instance

```

133 using System;
134 using System.Collections.Generic;
135 using System.Text;
136 using Microsoft.Ccr.Core;

```

```

137
138 namespace Example
139 {
140     class Program
141     {
142         //Example 04: Constructing a PortSet Instance
143         static void Main(string[] args)
144         {
145             /// <summary>
146             /// There are two ways to construct a PortSet instance:
147             /// 1. Use the generic type arguments to define at compile time.
148             /// 2. Use the initialization constructor that takes a parameter list of type arguments.
149             /// Compile-time即決定type，高效率 + 穩定，但受限於「desktop CLR最多20個type、Compact系統最
150             多8個參數」
151             /// Run-time決定，損失效能，但不受數量限制。須使用PostUnknownType或TryPostUnknownType。
152             /// </summary>
153
154             // Create a PortSet using generic type arguments
155             var genericPortSet = new PortSet<int, string, double>();
156             genericPortSet.Post(10);
157             genericPortSet.Post("hello");
158             genericPortSet.Post(3.14159);
159             Console.WriteLine("genericPortSet:" + genericPortSet.ToString()); // Port Information
160
161             // Create a runtime PortSet, using the initialization
162             // constructor to supply an array of types
163             PortSet runtimePortSet = new PortSet(
164                 typeof(int),
165                 typeof(string),
166                 typeof(double)
167             );
168
169             // 須使用PostUnknownType或TryPostUnknownType
170             runtimePortSet.PostUnknownType(10);
171             runtimePortSet.PostUnknownType("hello");
172             runtimePortSet.PostUnknownType(3.14159);
173             runtimePortSet.TryPostUnknownType(10);
174             runtimePortSet.TryPostUnknownType("hello");
175             runtimePortSet.TryPostUnknownType(3.14159);
176             Console.WriteLine("runtimePortSet:" + runtimePortSet.ToString()); // Port Information (無
177             結果)
178         }
179     }
180 }

```

## 181 Example 05: Handler

```

182 using System;
183 using System.Collections.Generic;
184 using System.Text;

```

```

185 using Microsoft.Ccr.Core;
186
187 namespace Example
188 {
189     class Program
190     {
191         //Example 05: Handler
192         static void Main(string[] args)
193         {
194             /// <summary>
195             /// 提高可讀性 · 非同步接收 · 指派Handler
196             /// </summary>
197
198             Dispatcher dispatcher = new Dispatcher();
199             DispatcherQueue taskQueue = new DispatcherQueue("sample queue", dispatcher);
200             // 使用CcrConsoleService.Create建立CcrConsolePort型態的變數ccrPort
201             CcrConsolePort ccrPort = CcrConsoleService.Create(taskQueue);
202
203             ccrPort.Post(10);
204             ccrPort.Post("hello");
205             ccrPort.Post(3.14159);
206         }
207
208         #region Example 05: function
209         /// <summary>
210         /// PortSet that accepts items of int, string, double
211         /// </summary>
212         // 宣告CcrConsolePort繼承PortSet<int, string, double>
213         public class CcrConsolePort : PortSet<int, string, double>
214         {
215         }
216         /// <summary>
217         /// Simple example of a CCR component that uses a PortSet to abstract
218         /// its API for message passing
219         /// </summary>
220         public class CcrConsoleService
221         {
222             CcrConsolePort _mainPort;
223             DispatcherQueue _taskQueue;
224
225             /// <summary>
226             /// Creates an instance of the service class, returning only a PortSet
227             /// instance for communication with the service
228             /// </summary>
229             /// <param name="taskQueue"></param>
230             /// <returns></returns>
231             public static CcrConsolePort Create(DispatcherQueue taskQueue)
232             {
233                 var console = new CcrConsoleService(taskQueue); // 呼叫「private CcrConsoleService」
234                 console.Initialize();
235                 return console._mainPort;
236             }

```

```

237
238     /// <summary>
239     /// Initialization constructor
240     /// </summary>
241     /// <param name="taskQueue">DispatcherQueue instance used for scheduling</param>
242     private CcrConsoleService(DispatcherQueue taskQueue)
243     {
244         // create PortSet instance used by external callers to post items
245         _mainPort = new CcrConsolePort();
246         // cache dispatcher queue used to schedule tasks
247         _taskQueue = taskQueue;
248     }
249
250     private void Initialize()
251     {
252         // Activate three persisted receivers (single item arbiters)
253         // that will run concurrently to each other,
254         // one for each item/message type
255         Arbiter.Activate(_taskQueue,
256             // Arbiter.Receive<type>(bool persist, Port port, Handler handler)
257             Arbiter.Receive<int>(true, _mainPort, IntWriteLineHandler),
258             Arbiter.Receive<string>(true, _mainPort, StringWriteLineHandler),
259             Arbiter.Receive<double>(true, _mainPort, DoubleWriteLineHandler)
260         );
261     }
262
263     void IntWriteLineHandler(int item)
264     {
265         Console.WriteLine("Received integer:" + item);
266     }
267     void StringWriteLineHandler(string item)
268     {
269         Console.WriteLine("Received string:" + item);
270     }
271     void DoubleWriteLineHandler(double item)
272     {
273         Console.WriteLine("Received double:" + item);
274     }
275     }
276     #endregion
277 }
278 }

```

## 279 Example 06: Demonstrates the use of the implicit 280 operator

```

281 using System;
282 using System.Collections.Generic;

```

```

283 using System.Text;
284 using Microsoft.Ccr.Core;
285
286 namespace Example
287 {
288     class Program
289     {
290         //Example 06: Demonstrates the use of the implicit operator
291         static void Main(string[] args)
292         {
293             /// <summary>
294             /// PortSet轉至Port，並註冊於Arbiter下
295             /// 1. Assigning the correct instance of a Port<> within a PortSet, to another Port<>
296 variable
297             /// 2. Extracting the correct instance of a Port<> so it can be used to register an arbiter
298             /// </summary>
299
300             Dispatcher dispatcher = new Dispatcher();
301             DispatcherQueue taskQueue = new DispatcherQueue("sample queue", dispatcher);
302
303             var portSet = new PortSet<int, string, double>();
304
305             // the following statement compiles because of the implicit assignment operators
306             // that "extract" the instance of Port<int> from the PortSet
307             Port<string> portInt = portSet; // 將portSet的String型式放到portInt
308             //Port<int> portInt = portSet; // 將portSet的int型式放到portInt
309
310             // the implicit assignment operator is used below to "extract" the Port<int>
311             // instance so the int receiver can be registered
312             Arbiter.Activate(taskQueue,
313                 Arbiter.Receive<string>(true, portInt, item => Console.WriteLine(item))
314                 //Arbiter.Receive<int>(true, portInt, item => Console.WriteLine(item.ToString()))
315             );
316
317             portSet.Post(11); // 仍可輸入
318             portInt.Post("Hello");
319             //portInt.Post(15);
320         }
321     }
322 }

```