

```
1 using System.Text;
2 using System.Windows;
3 using System.Windows.Controls;
4 using System.Windows.Data;
5 using System.Windows.Documents;
6 using System.Windows.Input;
7 using System.Windows.Interop;
8 using System.Windows.Media;
9 using System.Windows.Media.Imaging;
10 using System.Windows.Navigation;
11 using System.Windows.Shapes;
12
13 namespace Drink
14 {
15     public partial class MainWindow : Window
16     {
17         Dictionary<string, int> drinks = new Dictionary<string, int>
18         {
19             {"紅茶大杯", 60},
20             {"紅茶小杯", 40},
21             {"綠茶大杯", 50},
22             {"綠茶小杯", 30},
23             {"可樂大杯", 50},
24             {"可樂小杯", 30},
25             {"咖啡大杯", 80},
26             {"咖啡小杯", 50},
27         };
28
29         Dictionary<string, int> orders = new Dictionary<string, int>();
30         string takeout = "";
31
32         public MainWindow()
33         {
34             InitializeComponent();
35             loadDrink();
36         }
37         private void loadDrink()
38         {
39             drinkList.Height = 46 * drinks.Count;
40             foreach (var drink in drinks)
41             {
42                 var cp = new StackPanel
43                 {
44                     Orientation = Orientation.Horizontal,
45                     Margin = new Thickness(3),
46                     Background = Brushes.LightPink,
47                     Height = 40
48                 };
49
50                 var cb = new CheckBox
51                 {
52                     Content = drink.Key,
53                     FontSize = 16,
```

```
54         FontWeight = FontWeights.Bold,
55         VerticalAlignment = VerticalAlignment.Center,
56         Margin = new Thickness(5)
57     };
58
59     var lb = new Label
60     {
61         Content = $"{drink.Value}元",
62         FontSize = 16,
63         FontWeight = FontWeights.Bold,
64         VerticalAlignment = VerticalAlignment.Center,
65         Margin = new Thickness(5)
66     };
67     var slider = new Slider
68     {
69         Width = 180,
70         Minimum = 0,
71         Maximum = 10,
72         Value = 0,
73         VerticalAlignment = VerticalAlignment.Center,
74         Margin = new Thickness(50, 5, 5, 5),
75         IsSnapToTickEnabled = true,
76         TickFrequency = 1
77     };
78
79     var num = new Label
80     {
81         Content = "0",
82         FontSize = 16,
83         FontWeight = FontWeights.Bold,
84         VerticalAlignment = VerticalAlignment.Center,
85         Margin = new Thickness(5)
86     };
87
88     num.SetBinding(Label.ContentProperty, new Binding("Value") { Source = ➤
89         slider });
90
91     cp.Children.Add(cb);
92     cp.Children.Add(lb);
93     cp.Children.Add(slider);
94     cp.Children.Add(num);
95     drinkList.Children.Add(cp);
96 }
97
98 private void RadioButton_Checked(object sender, RoutedEventArgs e)
99 {
100     var rb = sender as RadioButton;
101     if (rb.IsChecked == true)
102     {
103         takeout = rb.Content.ToString();
104     }
105 }
```

```
106
107     private void OrderButton_Click(object sender, RoutedEventArgs e)
108     {
109         for (int i = 0; i < drinkList.Children.Count; i++)
110         {
111             var cp = drinkList.Children[i] as StackPanel;
112             var cb = cp.Children[0] as CheckBox;
113             var slider = cp.Children[2] as Slider;
114             var amount = (int)slider.Value;
115
116             if (cb.IsChecked == true && slider.Value > 0)
117             {
118                 if (orders.ContainsKey(cb.Content.ToString()))
119                 {
120                     orders[cb.Content.ToString()] = amount;
121                 }
122                 else
123                 {
124                     orders.Add(cb.Content.ToString(), amount);
125                 }
126             }
127         }
128         GenerateOutputList();
129     }
130
131     private void GenerateOutputList()
132     {
133         int total = 0;
134         var result = new StringBuilder();
135         result.Append($"內用/外帶：{takeout}\n");
136         result.Append("訂購清單:\n");
137         result.Append("=====\n");
138
139         foreach (var order in orders)
140         {
141             result.Append($"{order.Key} x {order.Value}\n");
142             total += drinks[order.Key] * order.Value;
143         }
144         //增加一個分隔線
145         result.Append("=====\n");
146         result.Append($"小計：{total}元\n");
147         result.Append("優惠項目:\n");
148         result.Append("-----\n");
149         int sellPrice = total;
150         if (total >= 500)
151         {
152             sellPrice = (int)(total * 0.8);
153             result.Append("滿500元打8折\n");
154         }
155         else if (total >= 300)
156         {
157             sellPrice = (int)(total * 0.9);
158             result.Append("滿300元打9折\n");
```

```

159         }
160         else
161         {
162             result.Append("無折扣\n");
163         }
164         result.Append("-----\n");
165         result.Append($"應付:{sellPrice}\n");
166
167         ResultTextBlock.Text = result.ToString();
168     }
169 }
170 }

```

MainWindow

內用/外帶

☐ 內用 ☒ 外帶

買!

內用/外帶：外帶

訂購清單：

=====

綠茶大杯 x 10

紅茶大杯 x 10

紅茶小杯 x 10

綠茶小杯 x 10

可樂大杯 x 10

可樂小杯 x 10

咖啡大杯 x 10

咖啡小杯 x 10

=====

小計：3900元

優惠項目：

滿500元打8折

應付:3120

<input checked="" type="checkbox"/> 紅茶大杯	60元	<input type="text" value="10"/>	10
<input checked="" type="checkbox"/> 紅茶小杯	40元	<input type="text" value="10"/>	10
<input checked="" type="checkbox"/> 綠茶大杯	50元	<input type="text" value="10"/>	10
<input checked="" type="checkbox"/> 綠茶小杯	30元	<input type="text" value="10"/>	10
<input checked="" type="checkbox"/> 可樂大杯	50元	<input type="text" value="10"/>	10
<input checked="" type="checkbox"/> 可樂小杯	30元	<input type="text" value="10"/>	10
<input checked="" type="checkbox"/> 咖啡大杯	80元	<input type="text" value="10"/>	10
<input checked="" type="checkbox"/> 咖啡小杯	50元	<input type="text" value="10"/>	10