

```
using Microsoft.Win32;
using System.IO;
using System.Text;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Data;
using System.Windows.Media;

namespace WpfApp1
{
    public partial class MainWindow : Window
    {
        Dictionary<string, int> drinks = new Dictionary<string, int>();
        Dictionary<string, int> orders = new Dictionary<string, int>();
        string takeout = "";

        public MainWindow()
        {
            /*Dictionary<string, int> drinks = new Dictionary<string, int>
            {
                { "紅茶大杯", 60 },
                { "紅茶小杯", 40 },
                { "綠茶大杯", 60 },
                { "綠茶小杯", 40 },
                { "可樂大杯", 50 },
                { "可樂小杯", 30 },
                { "咖啡大杯", 70 },
            };*/

            InitializeComponent();

            AddNewDrink(drinks);

            DisplayDrinkMenu(drinks);
        }

        private void AddNewDrink(Dictionary<string, int> drinks)
        {
```

```

        OpenFileDialog openFileDialog = new OpenFileDialog();
        openFileDialog.Filter = "CSV檔案|*.csv|文字檔案|*.txt|所有檔案|*.*";
        openFileDialog.Title = "選擇飲料菜單檔案";
        if (openFileDialog.ShowDialog() == true)
        {
            string fileName = openFileDialog.FileName;
            ReadDrinksFromFile(fileName, drinks);
        }
    }

    private void ReadDrinksFromFile(string fileName, Dictionary<string, int>
drinks)
    {
        try
        {
            string[] lines = File.ReadAllLines(fileName, Encoding.UTF8);
            foreach (var line in lines)
            {
                string[] tokens = line.Split(',');
                string drinkName = tokens[0];
                int price = Convert.ToInt32(tokens[1]);
                drinks.Add(drinkName, price);
            }
        }
        catch (InvalidOperationException ex)
        {
            MessageBox.Show($"讀取檔案時發生錯誤: {ex.Message}");
        }
    }

    private void DisplayDrinkMenu(Dictionary<string, int> drinks)
    {
        stackpanel_DrinkMenu.Children.Clear();
        stackpanel_DrinkMenu.Height = drinks.Count * 40;
        foreach (var drink in drinks)
        {
            var sp = new StackPanel
            {

```

```
Orientation = Orientation.Horizontal,
Margin = new Thickness(2),
Height = 35,
VerticalAlignment = VerticalAlignment.Center,
Background = Brushes.AliceBlue
};

var cb = new CheckBox
{
    Content = $"{drink.Key} {drink.Value}元",
    FontFamily = new FontFamily("微軟正黑體"),
    FontSize = 18,
    Foreground = Brushes.Blue,
    Margin = new Thickness(10, 0, 40, 0),
    VerticalContentAlignment = VerticalAlignment.Center
};

var sl = new Slider
{
    Width = 150,
    Value = 0,
    Minimum = 0,
    Maximum = 10,
    IsSnapToTickEnabled = true,
    VerticalAlignment = VerticalAlignment.Center,
};

var lb = new Label
{
    Width = 30,
    Content = "0",
    FontFamily = new FontFamily("微軟正黑體"),
    FontSize = 18,
};

Binding myBinding = new Binding("Value")
{
    Source = sl,
```

```

        Mode = BindingMode.OneWay
    };
    lb.SetBinding(ContentProperty, myBinding);

    sp.Children.Add(cb);
    sp.Children.Add(sl);
    sp.Children.Add(lb);

    stackpanel_DrinkMenu.Children.Add(sp);
}
}

private void RadioButton_Checked(object sender, RoutedEventArgs e)
{
    var rb = sender as RadioButton;
    if ((rb.IsChecked == true))
    {
        takeout = rb.Content.ToString();
    }
}

private void OrderButton_Click(object sender, RoutedEventArgs e)
{
    ResultTextBlock.Text = "";
    string discountMessage = "";
    orders.Clear();
    for (int i = 0; i < stackpanel_DrinkMenu.Children.Count; i++)
    {
        var sp = stackpanel_DrinkMenu.Children[i] as StackPanel;
        var cb = sp.Children[0] as CheckBox;
        var sl = sp.Children[1] as Slider;
        var lb = sp.Children[2] as Label;

        if (cb.IsChecked == true && sl.Value > 0)
        {
            string drinkName = cb.Content.ToString().Split(' ')[0];
            orders.Add(drinkName, int.Parse(lb.Content.ToString()));
        }
    }
}

```

```
}  
  
double total = 0.0;  
double sellPrice = 0.0;  
  
string orderMessage = "";  
DateTime now = DateTime.Now;  
orderMessage += $"訂購時間: {now.ToString("yyyy/MM/dd HH:mm:ss")}\n";  
orderMessage += $"取餐方式: {takeout}\n"; ;  
  
int num = 1;  
  
foreach (var item in orders)  
{  
    string drinkName = item.Key;  
    int quantity = item.Value;  
    int price = drinks[drinkName];  
  
    int subTotal = price * quantity;  
    total += subTotal;  
    orderMessage += $"{num}. {drinkName} x {quantity}杯，共{subTotal}元\n";  
    num++;  
}  
  
if (total >= 500)  
{  
    discoutMessage = "滿500元8折";  
    sellPrice = total * 0.8;  
}  
else if (total >= 300)  
{  
    discoutMessage = "滿300元9折";  
    sellPrice = total * 0.9;  
}  
else  
{  
    discoutMessage = "無折扣";  
    sellPrice = total;  
}
```

```

        orderMessage += $"總金額: {total}元\n";
        orderMessage += $"{discountMessage}，應付金額: {sellPrice}元\n";
        ResultTextBlock.Text = orderMessage;
        SaveOrder(orderMessage);
    }

    private void SaveOrder(string orderMessage)
    {
        SaveFileDialog saveFileDialog = new SaveFileDialog();
        saveFileDialog.Filter = "文字檔案|*.txt|所有檔案|*.*";
        saveFileDialog.Title = "儲存訂單";
        if (saveFileDialog.ShowDialog() == true)
        {
            string fileName = saveFileDialog.FileName;

            try
            {
                using (StreamWriter sw = new StreamWriter(fileName))
                {
                    sw.Write(orderMessage);
                }
                MessageBox.Show("訂單已成功儲存。");
            }
            catch (IOException ex)
            {
                MessageBox.Show($"儲存檔案時發生錯誤: {ex.Message}");
            }
        }
    }
}

```

MainWindow

內用/外帶

內用

外帶

☒紅茶大杯 60元

3

☐紅茶小杯 40元

0

☐綠茶大杯 50元

0

☐綠茶小杯 30元

0

☐可樂大杯 50元

0

☐可樂小杯 30元

0

☒心痛的感覺小杯 800元

1

☒心痛的感覺大杯 1000元

1

確定訂購

訂購時間: 2024/10/22 17:14:36

取餐方式: 內用

1. 紅茶大杯 x 3杯, 共180元

2. 心痛的感覺小杯 x 1杯, 共800元

3. 心痛的感覺大杯 x 1杯, 共1000元

總金額: 1980元

滿500元8折, 應付金額: 1584元

名稱	修改日期	類型	大小
.vs	2024/10/22 下午 02:47	檔案資料夾	
bin	2024/10/22 下午 02:47	檔案資料夾	
obj	2024/		
App.xaml	2024/		
App.xaml.cs	2024/		
AssemblyInfo.cs	2024/		
MainWindow.xaml	2024/		
MainWindow.xaml.cs	2024/		
test.csv	2024/		
TEST.txt	2024/		
WpfApp1.csproj	2024/		
WpfApp1.csproj.user	2024/		
WpfApp1.sln	2024/		

TEST.txt - 記事本

檔案(F) 編輯(E) 格式(O) 檢視(V) 說明

訂購時間: 2024/10/22 17:17:09

取餐方式: 內用

1. 紅茶大杯 x 3杯, 共180元

2. 心痛的感覺小杯 x 1杯, 共800元

3. 心痛的感覺大杯 x 1杯, 共1000元

總金額: 1980元

滿500元8折, 應付金額: 1584元

(S:) 建置