Visual Programming Lab

CSL-313

Lab Journal 4

Student Name: M. Anas Baig
Enrolment No.: 01-134152-037
Class and Section: BS(CS)-5A

Department of Computer Science
BAHRIA UNIVERSITY, ISLAMABAD

M. Anas Baig (01-134152-037)

Tea Shop

There is a tea stall where a cup of tea worth 40 Rupees each is sold along with other refreshments e.g muffins, pastries and biscuits etc. There are different types of teas e.g Pink tea, Green tea and English tea etc.

a) You have to make a tea and a refreshment class with valid attributes.

b) A sale class which should keep track of all sales. It should have functions
NewSale() to take new sale data from user
PrintInvoice() to generate receipt and display it on screen,
Before generating Invoice it should calculate total bill and check if the payment given by user is not less than the total bill.

c) A SaleFileHandling class which should have functions
GenerateInvoice() to generate receipt and store in a single file,
ViewAllSales() to view all the data stored in the file.

Invoice/Receipt should contain customer name, tea flavor, purchasing DateTime, refreshment name, refreshment cost, total cost, cash received, and cash returned. Use properties for getter and setter functions.

Procedure/Program:

Program.cs File:

```
1.  using System;
2.  using System.Collections.Generic;
3.  using System.Linq;
4.  using System.Text;
5.  using System.IO;
6.  using System.Collections;
7.
8.  namespace smjConsoleApplication1
9.  {
10.     class Program
11.     {
12.         static void Main(string[] args)
13.         {
14.             Console.BackgroundColor = ConsoleColor.Gray;
15.             Console.Clear();
16.             Console.ForegroundColor = ConsoleColor.Black;
17.             Console.WriteLine("================================================================
    ============");
18.             Console.WriteLine("      B A H R I A - U N I V E R S I T Y - T E A - S H O P -
     S Y S T E M");
19.             Console.WriteLine("================================================================
    ============");
20.
```

```
            menu m = new menu();
            m.consoleMenu();

            Console.ReadKey();
        }
    }
}
```

menu.cs File:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.IO;
using System.Collections;

namespace smjConsoleApplication1
{
    class menu
    {
        public void consoleMenu()
        {
            sale s = new sale();
            Console.WriteLine("Enter Desired operation:");
            Console.WriteLine("1. New Sale.");
            Console.WriteLine("2. Print Invoices.");

            int option1 = int.Parse(Console.ReadLine());

            if (option1 == 1)
            {
                s.newSale();
            }
            else if (option1 == 2)
            {
                ArrayList list = s.getAllSales();
                for (int i = 0; i < list.Count; i++)
                {
                    sale obj = list[i] as sale;
                    obj.printInvoice();
                }
            }
            else
            {
                Console.WriteLine("Invalid Input");
            }
        }
    }
}
```

teaClass.cs File:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace smjConsoleApplication1
{
    class teaClass
    {
        protected string teaFlavour;

        public string teaFlavourProperty
        {
            get { return teaFlavour; }
            set { teaFlavour = value; }
```

```
        }


    }
}
```

refreshment.cs File:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace smjConsoleApplication1
{
    class refreshment
    {
        protected string refreshmentName;

        public string refreshmentNameProperty
        {
            get { return refreshmentName; }
            set { refreshmentName = value; }
        }
    }
}
```

sale.cs File:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Collections; //for arraylist

namespace smjConsoleApplication1
{
    class sale
    {
        string customerName;
        DateTime localDate;
        int refreshmentCost;
        int totalCost;
        int cashReceived;
        int cashReturned;
        teaClass tea = new teaClass();
        refreshment refresh = new refreshment();

        public string customerNameProperty
        {
            get { return customerName; }
            set { customerName = value; }
        }

        public string flavourProperty
        {
            get{    return tea.teaFlavourProperty;  }
            set{    tea.teaFlavourProperty = value; }
        }

        public DateTime localDateProperty
        {
            get{    return localDate;    }
            set{    localDate = value;   }
        }

        public string refreshmentProperty
        {
```

```
40.             get { return refresh.refreshmentNameProperty; }
41.             set { refresh.refreshmentNameProperty = value; }
42.         }
43.
44.         public int refreshmentCostProperty
45.         {
46.             get{    return refreshmentCost;    }
47.             set{    refreshmentCost = value;   }
48.         }
49.
50.         public int totalCostProperty
51.         {
52.             get{    return totalCost;    }
53.             set{    totalCost = value;   }
54.         }
55.
56.         public int cashReceivedProperty
57.         {
58.             get{    return cashReceived;    }
59.             set{    cashReceived = value;   }
60.         }
61.
62.         public int cashReturnedProperty
63.         {
64.             get{    return cashReturned;    }
65.             set{    cashReturned = value;   }
66.         }
67.
68.         public void newSale()
69.         {
70.             Console.WriteLine("Enter Tea Flavour:");
71.             Console.WriteLine("1. Pink Tea.");
72.             Console.WriteLine("2. Green Tea.");
73.             Console.WriteLine("3. English Tea.");
74.             int option1 = int.Parse(Console.ReadLine());
75.
76.             if (option1 == 1)
77.             {
78.                 flavourProperty = "Pink Tea";
79.             }
80.             else if (option1 == 2)
81.             {
82.                 flavourProperty = "Green Tea";
83.             }
84.             else if (option1 == 3)
85.             {
86.                 flavourProperty = "English Tea";
87.             }
88.             else
89.             {
90.                 Console.WriteLine("Invalid Input");
91.             }
92.
93.             Console.WriteLine("Enter Refreshment Name:");
94.             Console.WriteLine("1. Muffins.");
95.             Console.WriteLine("2. Pastries.");
96.             Console.WriteLine("3. Biscuits.");
97.             int option2 = int.Parse(Console.ReadLine());
98.
99.             if (option2 == 1)
100.                {
101.                    flavourProperty = "Muffins";
102.                }
103.                else if (option2 == 2)
104.                {
105.                    refreshmentProperty = "Pastries";
106.                }
107.                else if (option2 == 3)
108.                {
109.                    refreshmentProperty = "Biscuits";
110.                }
```

```
                else
                {
                    Console.WriteLine("Invalid Input");
                }

                Console.WriteLine("Enter Customer Name:");
                customerNameProperty = Console.ReadLine();

                localDate = DateTime.Now;

                Console.WriteLine("Enter Cash Returned:");
                cashReturnedProperty = int.Parse(Console.ReadLine());

                totalCostProperty = (40 + 50);

                bool check = true;
                while (check)
                {
                    Console.WriteLine("Enter Cash Received:");
                    cashReceivedProperty = int.Parse(Console.ReadLine());
                    if (cashReceived < totalCost)
                    {
                        Console.WriteLine("ERROR!!! Enter Correct Amount");
                    }
                    else
                    {
                        check = false;
                    }
                }

                cashReturnedProperty = (cashReceived - totalCost);

                printInvoice();
                saleFileHandling sale =  new saleFileHandling();
                sale.saveNewSale(this);
            }

            public void printInvoice()
            {
                Console.WriteLine("----------------------INVOICE----------------------");
                Console.WriteLine("Customer Name: " + this.customerNameProperty);
                Console.WriteLine("Date and Time: " + this.localDateProperty);
                Console.WriteLine("Tea Flavour: " + this.flavourProperty);
                Console.WriteLine("Refreshment Name: " + this.refreshmentProperty);
                Console.WriteLine("Total Cost: " + this.totalCostProperty);
                Console.WriteLine("Cash Received: " + this.cashReceivedProperty);
                Console.WriteLine("Cash Returned: " + this.cashReturnedProperty);
            }

            public ArrayList getAllSales()
            {
                saleFileHandling obj = new saleFileHandling();
                return obj.collectAllSale();
            }
        }
    }
```

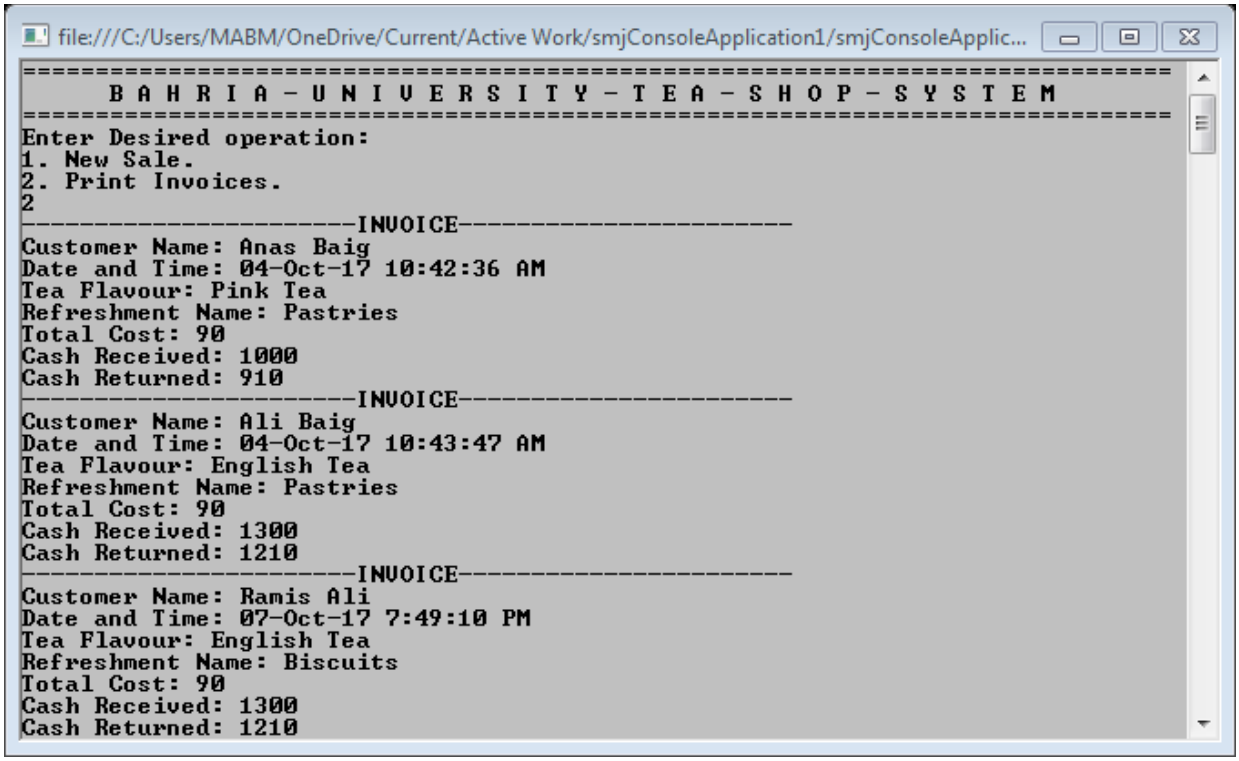
saleFileHandling.cs File:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.IO;
using System.Collections;

namespace smjConsoleApplication1
{
    class saleFileHandling
```

```
11.     {
12.         public void saveNewSale(sale s)
13.         {
14.             StreamWriter writeFile = new StreamWriter("TeaShop.txt", true);
15.             writeFile.WriteLine(s.customerNameProperty);
16.             writeFile.WriteLine(s.localDateProperty);
17.             writeFile.WriteLine(s.flavourProperty);
18.             writeFile.WriteLine(s.refreshmentProperty);
19.             writeFile.WriteLine(s.totalCostProperty);
20.             writeFile.WriteLine(s.cashReceivedProperty);
21.             writeFile.WriteLine(s.cashReturnedProperty);
22.             writeFile.Close();
23.         }
24.
25.       public ArrayList collectAllSale()
26.       {
27.           ArrayList list = new ArrayList();
28.           StreamReader readFile = new StreamReader("TeaShop.txt");
29.           sale s;
30.
31.           while (!readFile.EndOfStream)
32.           {
33.               s = new sale();
34.               s.customerNameProperty = readFile.ReadLine();
35.               s.localDateProperty = DateTime.Parse(readFile.ReadLine());
36.               s.flavourProperty = readFile.ReadLine();
37.               s.refreshmentProperty = readFile.ReadLine();
38.               s.totalCostProperty = int.Parse(readFile.ReadLine());
39.               s.cashReceivedProperty = int.Parse(readFile.ReadLine());
40.               s.cashReturnedProperty = int.Parse(readFile.ReadLine());
41.               list.Add(s);
42.           }
43.           readFile.Close();
44.           return list;
45.       }
46.     }
47. }
```

Display Console Output:



Text File Output: