***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**

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.

1. You have to make a tea and a refreshment class with valid attributes.
2. 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.

1. 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;
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("===============================================================================");
21. menu m = **new** menu();
22. m.consoleMenu();
24. Console.ReadKey();
25. }
26. }
27. }

**menu.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;
8. **namespace** smjConsoleApplication1
9. {
10. **class** menu
11. {
12. **public** **void** consoleMenu()
13. {
14. sale s = **new** sale();
15. Console.WriteLine("Enter Desired operation:");
16. Console.WriteLine("1. New Sale.");
17. Console.WriteLine("2. Print Invoices.");
19. **int** option1 = **int**.Parse(Console.ReadLine());
21. **if** (option1 == 1)
22. {
23. s.newSale();
24. }
25. **else** **if** (option1 == 2)
26. {
27. ArrayList list = s.getAllSales();
28. **for** (**int** i = 0; i < list.Count; i++)
29. {
30. sale obj = list[i] **as** sale;
31. obj.printInvoice();
32. }
33. }
34. **else**
35. {
36. Console.WriteLine("Invalid Input");
37. }
38. }
39. }
40. }

**teaClass.cs File:**

1. **using** System;
2. **using** System.Collections.Generic;
3. **using** System.Linq;
4. **using** System.Text;
6. **namespace** smjConsoleApplication1
7. {
8. **class** teaClass
9. {
10. **protected** **string** teaFlavour;
12. **public** **string** teaFlavourProperty
13. {
14. **get** { **return** teaFlavour; }
15. **set** { teaFlavour = value; }
16. }

19. }
20. }

**refreshment.cs File:**

1. **using** System;
2. **using** System.Collections.Generic;
3. **using** System.Linq;
4. **using** System.Text;
6. **namespace** smjConsoleApplication1
7. {
8. **class** refreshment
9. {
10. **protected** **string** refreshmentName;
12. **public** **string** refreshmentNameProperty
13. {
14. **get** { **return** refreshmentName; }
15. **set** { refreshmentName = value; }
16. }
17. }
18. }

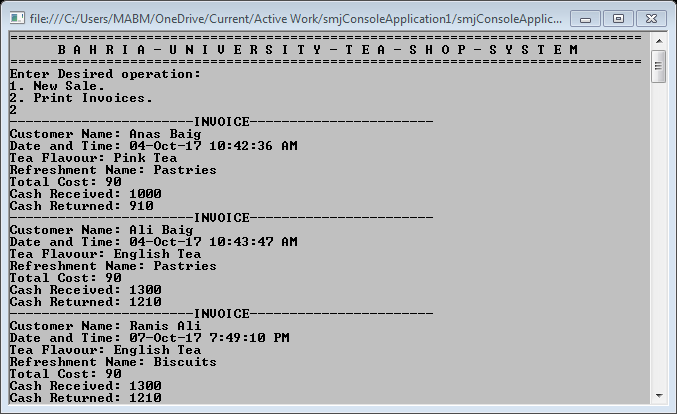
**sale.cs File:**

1. **using** System;
2. **using** System.Collections.Generic;
3. **using** System.Linq;
4. **using** System.Text;
5. **using** System.Collections; //for arraylist
7. **namespace** smjConsoleApplication1
8. {
9. **class** sale
10. {
11. **string** customerName;
12. DateTime localDate;
13. **int** refreshmentCost;
14. **int** totalCost;
15. **int** cashReceived;
16. **int** cashReturned;
17. teaClass tea = **new** teaClass();
18. refreshment refresh = **new** refreshment();
20. **public** **string** customerNameProperty
21. {
22. **get** { **return** customerName; }
23. **set** { customerName = value; }
24. }
26. **public** **string** flavourProperty
27. {
28. **get**{    **return** tea.teaFlavourProperty;  }
29. **set**{    tea.teaFlavourProperty = value; }
30. }
32. **public** DateTime localDateProperty
33. {
34. **get**{    **return** localDate;    }
35. **set**{    localDate = value;   }
36. }
38. **public** **string** refreshmentProperty
39. {
40. **get** { **return** refresh.refreshmentNameProperty; }
41. **set** { refresh.refreshmentNameProperty = value; }
42. }
44. **public** **int** refreshmentCostProperty
45. {
46. **get**{    **return** refreshmentCost;    }
47. **set**{    refreshmentCost = value;   }
48. }
50. **public** **int** totalCostProperty
51. {
52. **get**{    **return** totalCost;    }
53. **set**{    totalCost = value;   }
54. }
56. **public** **int** cashReceivedProperty
57. {
58. **get**{    **return** cashReceived;    }
59. **set**{    cashReceived = value;   }
60. }
62. **public** **int** cashReturnedProperty
63. {
64. **get**{    **return** cashReturned;    }
65. **set**{    cashReturned = value;   }
66. }
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());
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. }
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());
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. }
111. **else**
112. {
113. Console.WriteLine("Invalid Input");
114. }
116. Console.WriteLine("Enter Customer Name:");
117. customerNameProperty = Console.ReadLine();
119. localDate = DateTime.Now;
121. Console.WriteLine("Enter Cash Returned:");
122. cashReturnedProperty = **int**.Parse(Console.ReadLine());
124. totalCostProperty = (40 + 50);
126. **bool** check = **true**;
127. **while** (check)
128. {
129. Console.WriteLine("Enter Cash Received:");
130. cashReceivedProperty = **int**.Parse(Console.ReadLine());
131. **if** (cashReceived < totalCost)
132. {
133. Console.WriteLine("ERROR!!! Enter Correct Amount");
134. }
135. **else**
136. {
137. check = **false**;
138. }
139. }
141. cashReturnedProperty = (cashReceived - totalCost);
143. printInvoice();
144. saleFileHandling sale =  **new** saleFileHandling();
145. sale.saveNewSale(**this**);
146. }
148. **public** **void** printInvoice()
149. {
150. Console.WriteLine("-----------------------INVOICE-----------------------");
151. Console.WriteLine("Customer Name: " + **this**.customerNameProperty);
152. Console.WriteLine("Date and Time: " + **this**.localDateProperty);
153. Console.WriteLine("Tea Flavour: " + **this**.flavourProperty);
154. Console.WriteLine("Refreshment Name: " + **this**.refreshmentProperty);
155. Console.WriteLine("Total Cost: " + **this**.totalCostProperty);
156. Console.WriteLine("Cash Received: " + **this**.cashReceivedProperty);
157. Console.WriteLine("Cash Returned: " + **this**.cashReturnedProperty);
158. }
160. **public** ArrayList getAllSales()
161. {
162. saleFileHandling obj = **new** saleFileHandling();
163. **return** obj.collectAllSale();
164. }
165. }
166. }

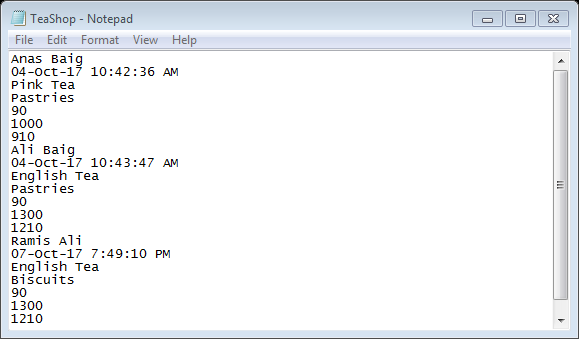
**saleFileHandling.cs File:**

1. **using** System;
2. **using** System.Collections.Generic;
3. **using** System.Linq;
4. **using** System.Text;
5. **using** System.IO;
6. **using** System.Collections;
8. **namespace** smjConsoleApplication1
9. {
10. **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. }
25. **public** ArrayList collectAllSale()
26. {
27. ArrayList list = **new** ArrayList();
28. StreamReader readFile = **new** StreamReader("TeaShop.txt");
29. sale s;
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:**

****