Payoda-Phase2 – Day5(04-08-23)

C#

Task 1: Create a C# program that models a simple banking system using classes and objects. Design a class called "BankAccount".

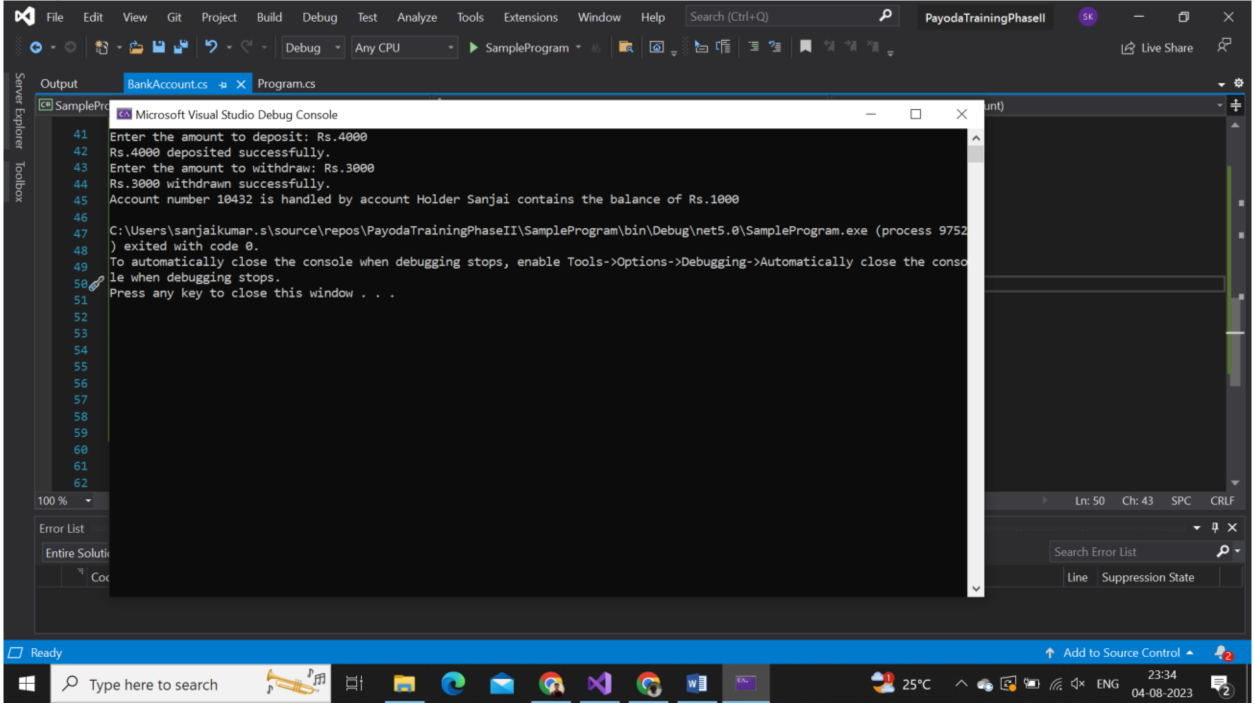
BankAccount.cs:

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

namespace SampleProgram
{
    class BankAccount
    {
        private readonly int _account_number;
        private string accountholdername;
        private int accountbalance = 0;

        public BankAccount(int _account_number, string account_holdername)
        {
            this._account_number = _account_number;
            Account_holdername = account_holdername;
        }

        public int Account_number => _account_number;

        public string Account_holdername { get => accountholdername; set =>
accountholdername = value; }
        public int Account_balance { get => accountbalance; set => accountbalance =
value; }

        public int amountDeposit(int amount)
        {
            if (amount <= 0)
            {
                Console.WriteLine("Invalid deposit amount. Deposit amount must be greater
than zero.");
                return -1;
            }
            else
            {
                Account_balance += amount;
                return Account_balance;
            }
        }


        public int amountWithdraw(int amount)
        {
            if (amount <= 0)
            {
                Console.WriteLine("Invalid withdrawal amount. Withdrawal amount must be 
greater than zero.");
                return -1;
            }
            else if (amount > Account_balance)
            {
                Console.WriteLine("Insufficient balance.");
                return -1;
            }
            else
            {
                Account_balance -= amount;
                return amount;
            }
        }

    }
}
```

Program.cs:

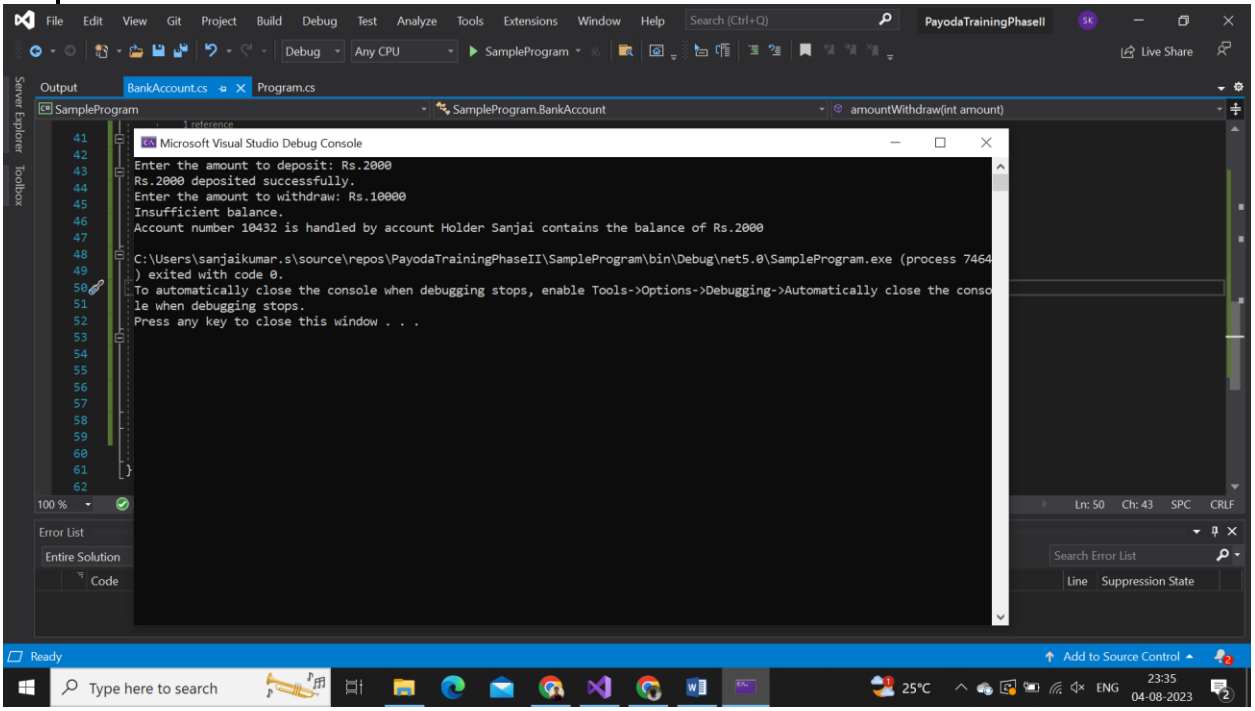
```
using System;

namespace SampleProgram
{
    class Program
    {
        static void Main(string[] args)
        {
            BankAccount bankaccount = new BankAccount(10432, "Sanjai");

            // Deposit
            Console.Write("Enter the amount to deposit: Rs.");
            int depositAmount = Convert.ToInt32(Console.ReadLine());
            int depositedAmount = bankaccount.amountDeposit(depositAmount);
            if (depositedAmount > 0)
            {
                Console.WriteLine($"Rs.{depositedAmount} deposited successfully.");
            }

            // Withdraw
            Console.Write("Enter the amount to withdraw: Rs.");
            int withdrawAmount = Convert.ToInt32(Console.ReadLine());
            int withdrawnAmount = bankaccount.amountWithdraw(withdrawAmount);
            if (withdrawnAmount > 0)
            {
                Console.WriteLine($"Rs.{withdrawnAmount} withdrawn successfully.");
            }

            // Display final account details
            Console.WriteLine($"Account number {bankaccount.Account_number} is handled by 
account Holder {bankaccount.Account_holdername} contains the balance of 
Rs.{bankaccount.Account_balance}");
        }
    }
}
```

Output:



```
Enter the amount to deposit: Rs.4000
Rs.4000 deposited successfully.
Enter the amount to withdraw: Rs.3000
Rs.3000 withdrawn successfully.
Account number 10432 is handled by account Holder Sanjai contains the balance of Rs.1000

C:\Users\sanjaikumar.s\source\repos\PayodaTrainingPhaseII\SampleProgram\bin\Debug\net5.0\SampleProgram.exe (process 9752
) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the conso
le when debugging stops.
Press any key to close this window . . .
```

Output2:



```
Enter the amount to deposit: Rs.2000
Rs.2000 deposited successfully.
Enter the amount to withdraw: Rs.10000
Insufficient balance.
Account number 10432 is handled by account Holder Sanjai contains the balance of Rs.2000

C:\Users\sanjaikumar.s\source\repos\PayodaTrainingPhaseII\SampleProgram\bin\Debug\net5.0\SampleProgram.exe (process 7464
) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the conso
le when debugging stops.
Press any key to close this window . . .
```

Task2: Create a C# program to model a simple Library Management System using classes and objects. Design classes for "Book" and "Library".

Library.cs:

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

namespace SampleLibrary
{
    internal class Library
    {
        Book[] book = new Book[4];
        public Library(Book[] arr)
        {
            book = arr;
        }
        public void BorrowBook(string title)
        {
            int count = 0;
            for (int i = 0; i < book.Length; i++)
            {
                if (book[i].Title.Equals(title))
                {
                    book[i].IsAvailable = false;
                    Console.WriteLine("Borrowed");
                    count++;
                }
            }
            if (count == 0) { Console.WriteLine("Book not Available"); }
        }
        public void ReturnBook(string title)
        {
            for (int i = 0; i < book.Length; i++)
            {
                if (book[i].Title.Equals(title))
                {
                    book[i].IsAvailable = true;
                    Console.WriteLine("Returned");
                }
            }

        }
        public void DisplayBookDetails()
        {
            for (int i = 0; i < book.Length; i++)
            {
                Console.WriteLine("Title :" + book[i].Title + " Author :" +
book[i].Author + " Availablity " + book[i].IsAvailable);
            }
        }
    }
}
```

Book.cs:

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

namespace SampleProgram
{
    class Book
    {
        private readonly int bookId;
        private string title;
        private string author;
        private bool isAvailable;

        public Book(int bookId, string title, string author, bool isAvailable)
        {
            this.bookId = bookId;
            Title = title;
            Author = author;
            IsAvailable = isAvailable;
        }

        public string Title { get => title; set => title = value; }
        public string Author { get => author; set => author = value; }
        public bool IsAvailable { get => isAvailable; set => isAvailable = value; }
    }
}
```

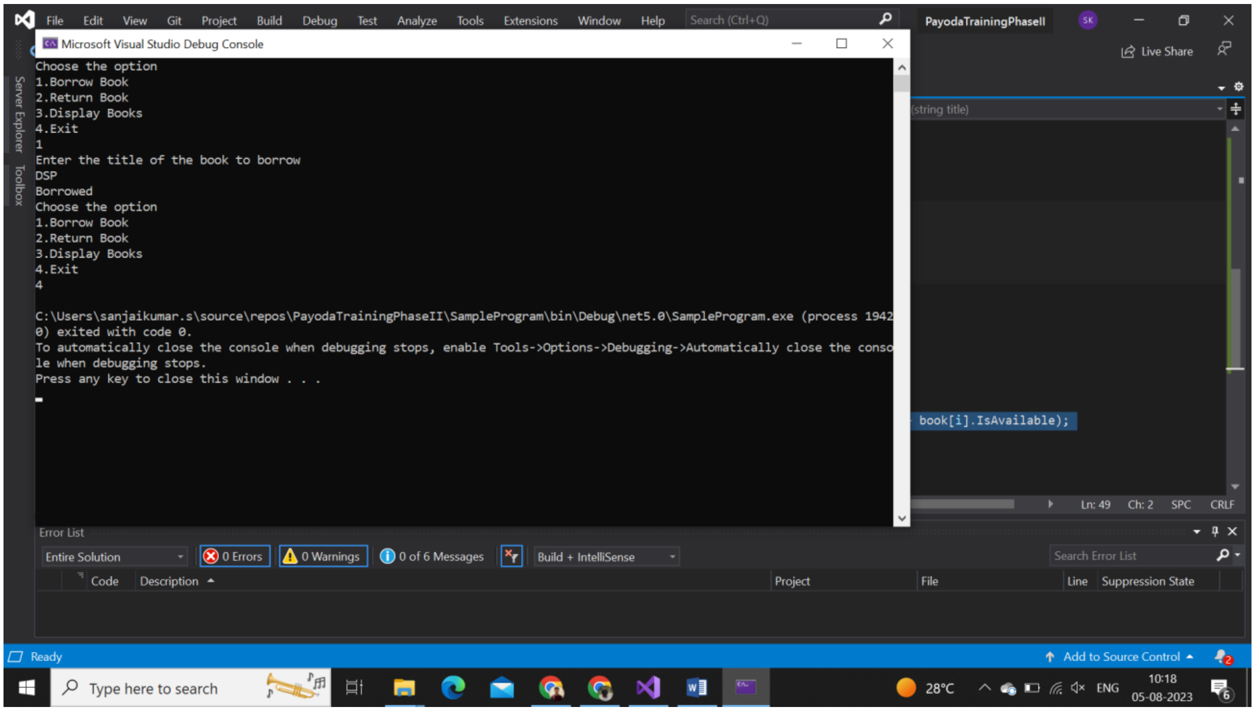
Program.cs:

```
using SampleLibrary;
using System;
using SampleProgram;


namespace SampleProgram
{
    class Program
    {
        static void Main(string[] args)
        {
            //int consumernumber = Convert.ToInt32(Console.ReadLine());
            //String consumername = Console.ReadLine();
            //int curreading = Convert.ToInt32(Console.ReadLine());
            //int prevreading = Convert.ToInt32(Console.ReadLine());
            //String consumertype = Console.ReadLine();

            //ElectricReading electricReading = new ElectricReading(1234, "Sanjai",
10000, 8000, "Commercial");
            //int billamt = electricReading.CalculateBill();

            //    Console.WriteLine($"Bill: " + $"{electricReading.consumernumber}" +
$"{electricReading.consumername}" + $"{billamt}");
```

```
            //// Create an instance of the BankAccount class
            //BankAccount bankaccount = new BankAccount(10432, "Sanjai");

            //// Deposit
            //Console.Write("Enter the amount to deposit: Rs.");
            //int depositAmount = Convert.ToInt32(Console.ReadLine());
            //int depositedAmount = bankaccount.amountDeposit(depositAmount);
            //if (depositedAmount > 0)
            //{
            //    Console.WriteLine($"Rs.{depositedAmount} deposited successfully.");
            //}

            //// Withdraw
            //Console.Write("Enter the amount to withdraw: Rs.");
            //int withdrawAmount = Convert.ToInt32(Console.ReadLine());
            //int withdrawnAmount = bankaccount.amountWithdraw(withdrawAmount);
            //if (withdrawnAmount > 0)
            //{
            //    Console.WriteLine($"Rs.{withdrawnAmount} withdrawn successfully.");
            //}

            //// Display final account details
            //Console.WriteLine($"Account number {bankaccount.Account_number} is handled
by account Holder {bankaccount.Account_holdername} contains the balance of
Rs.{bankaccount.Account_balance}");

            Book[] arr = { new Book(101, "Microprocessor", "Harsha", true), new Book(102,
"DSP", "Sanjai", true), new Book(103, "Java", "JK", true), new Book(104, "C#", "Yogi",
false) };
            Library library = new Library(arr);
            int choice = 0;
            while (choice != 4)
            {
                Console.WriteLine("Choose the option\n1.Borrow Book\n2.Return
Book\n3.Display Books\n4.Exit");
                choice = Convert.ToInt32(Console.ReadLine());
                if (choice == 1)
                {
                    Console.WriteLine("Enter the title of the book to borrow");
                    string title = Console.ReadLine();
                    library.BorrowBook(title);
                }
                else if (choice == 2)
                {
                    Console.WriteLine("Enter the title of the book to return");
                    string title = Console.ReadLine();
                    library.ReturnBook(title);
                }
                else if (choice == 3)
                {
                    library.DisplayBookDetails();
                }
                else if (choice == 4)
                {
                    break;
                }
            }
        }
```

```
    }
    }
```

Output:



```
Choose the option
1.Borrow Book
2.Return Book
3.Display Books
4.Exit
1
Enter the title of the book to borrow
DSP
Borrowed
Choose the option
1.Borrow Book
2.Return Book
3.Display Books
4.Exit
4

C:\Users\sanjaikumar.s\source\repos\PayodaTrainingPhaseII\SampleProgram\bin\Debug\net5.0\SampleProgram.exe (process 1942
0) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the conso
le when debugging stops.
Press any key to close this window . . .
```

```
book[i].IsAvailable);
```