Task 11: Interface/abstract class, and Single Inheritance, static variable
1. Create a 'Customer' class as mentioned above task.
2. Create an class 'Account' that includes the following attributes. Generate account number using
static variable.
• Account Number (a unique identifier).
• Account Type (e.g., Savings, Current)
• Account Balance
• Customer (the customer who owns the account)
• lastAccNo

```
namespace task11.entity
{
    public class Customer
    {
        private static int customerCounter = 1001;
        public int CustomerID { get; private set; }
        public string FirstName { get; set; }
        public string LastName { get; set; }

        public Customer(string firstName, string lastName)
        {
            this.CustomerID = customerCounter++;
            this.FirstName = firstName;
            this.LastName = lastName;
        }

        public void PrintCustomer()
        {
            Console.WriteLine($"Customer ID: {CustomerID}, Name:
{FirstName} {LastName}");
        }
    }
}
```

------ACCOUNTS.CS

```
namespace task11.entity
{
    public class Account
    {
        private static long accountCounter = 1001;
        public long AccountNumber { get; private set; }
        public string AccountType { get; set; }
        public double Balance { get; private set; }

        public Account(string accountType, double balance)
        {
            this.AccountNumber = accountCounter++;
            this.AccountType = accountType;
            this.Balance = balance;
        }

        public void Deposit(double amount)
        {
            Balance += amount;
        }

        public void Withdraw(double amount)
        {
            if (amount <= Balance)
                Balance -= amount;
```

```
            else
                Console.WriteLine("Insufficient Balance!");
        }

        public void PrintAccount()
        {
            Console.WriteLine($"Account No: {AccountNumber}, Type:
{AccountType}, Balance: {Balance}");
        }
    }
}


using task11.service;

class MainModule
{
    static void Main(string[] args)
    {
        Bank bank = new Bank();
        bool running = true;

        while (running)
        {
            Console.WriteLine("\n===== HM BANK MENU =====");
            Console.WriteLine("1. Add Customer and Account");
            Console.WriteLine("2. Exit");
            Console.Write("Enter your choice: ");
            string choice = Console.ReadLine();

            switch (choice)
            {
                case "1":
                    Console.Write("Enter First Name: ");
                    string first = Console.ReadLine();
                    Console.Write("Enter Last Name: ");
                    string last = Console.ReadLine();
                    Console.Write("Enter Account Type (Savings/Current):
");
                    string type = Console.ReadLine();
                    Console.Write("Enter Initial Balance: ");
                    double balance = Convert.ToDouble(Console.ReadLine());

                    bank.CreateAccount(first, last, type, balance);
                    break;

                case "2":
                    running = false;
                    Console.WriteLine("Thank you for using HM Bank!");
                    break;

                default:
                    Console.WriteLine("Invalid choice. Try again.");
                    break;
            }
        }
    }
}
```