

Assignment no-06

1)EmployeeDelegate. (UniCast Delegate)

Create a MultiCast Delegate by adding address of another method to the above delegate to list the details of Marketing Executive.

Code:

```
using System;
```

```
using System.Runtime.Serialization;
```

```
using System.IO;
```

```
using System.Runtime.Serialization.Formatters.Binary;
```

```
namespace csharpAssignmentno8
```

```
{
```

```
    [Serializable]
```

```
    class Manager
```

```
    {
```

```
        public string Name { get; set; }
```

```
        public int Id { get; set; }
```

```
        public int Salary { get; set; }
```

```
    }
```

```
    class demo
```

```
    {
```

```
        public static void Main(string[] args)
```

```
        {
```

```
            Manager m1 = new Manager();
```

```
m1.Name = "Neha";
```

```
m1.Id = 11;
```

```
m1.Salary = 5000;
```

```
IFormatter formatter = new BinaryFormatter();
```

```
Stream stream = new FileStream(@"C:\file2.txt", FileMode.Create, FileAccess.Write);
```

```
//serialize
```

```
formatter.Serialize(stream, m1);
```

```
stream.Close();
```

```
//deserialize
```

```
stream = new FileStream(@"C:\file2.txt", FileMode.Open, FileAccess.Read);
```

```
Manager m2 = (Manager)formatter.Deserialize(stream);
```

```
Console.WriteLine("After Deserialize object:" + m1.Name);
```

```
Console.WriteLine("After Deserialize object:" + m1.Id);
```


```
Console.WriteLine("After Deserialize object:" + m1.Salary);
```

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

```
}
```

```
}  
}
```

Output:

 C:\Users\neha\Source\Repos\Bank\Bank\bin\Debug\Bank.exe

```
After Desirialize object:Neha  
After Desirialize object:11  
After Desirialize object:5000
```

2)Write separate HDFCBank class with EventHandlers for UnderBalance and BalanceZero Events.

EventHandler should allow account holder to withdraw money till balance goes to Rs - 1000;below this show message "Transaction cannot be continued below specified limit of Rs.-1000".

using System;

```
namespace Bank  
{  
    class HDFCBANK  
    {  
  
        public event EventHandler<bool> Transaction;  
  
        private double balance;  
  
        public HDFCBANK(double balance)  
        {  
  
            this.balance = balance;  
  
        }  
  
        public void Withdrawn(double amount)  
        {  
  
            double sum = 0;
```

```

        if (balance < 1000 && balance != 0)
        {
            Transaction?.Invoke(this, true);
        }
        else if (amount < balance && balance >= 1000)
        {

            sum = balance - amount;
            if (sum < 1000)
            {
                Transaction?.Invoke(this, true);
            }
            else
            {
                Console.WriteLine("remaining balance:" + sum);
            }
        }

        else if (amount > balance && balance != 0)
        {
            Transaction?.Invoke(this, true);

        }
        else if (balance == 0)
        {
            Transaction?.Invoke(this, false);
        }

    }

}

class Program1
{

    static void Main()
    {

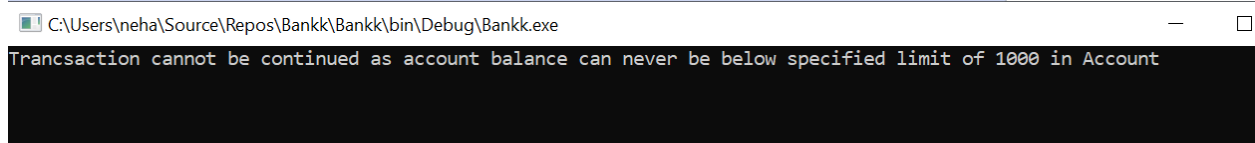
        HDFCBANK lc = new HDFCBANK(1000);
        lc.Transaction += lc_Trancsation;
        lc.Withdrawn(300);
        Console.ReadLine();
    }
}

```

```
private static void Ic_Trancsation(Object sender, bool b)
{
    Console.WriteLine("Trancsaction cannot be continued as account balance " + (b ? "can
never be below specified limit of 1000" : "is Zero") + " in Account");
}
}
```

Code:

Output:

A screenshot of a Windows console window. The title bar shows the file path "C:\Users\neha\Source\Repos\Bankk\Bankk\bin\Debug\Bankk.exe". The console output displays the message: "Trancsaction cannot be continued as account balance can never be below specified limit of 1000 in Account".

```
C:\Users\neha\Source\Repos\Bankk\Bankk\bin\Debug\Bankk.exe
Trancsaction cannot be continued as account balance can never be below specified limit of 1000 in Account
```