INTERFACE - 1 :

CODE :

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

namespace stallproblem
{
    interface Stall
    {
        void display();
    }
    class GoldStall : Stall
    {
        public string StallName { get; private set; }
        public int Cost { get;private set; }
        public string OwnerName { get; private set; }
        public int TvSet { get; private set; }
         public GoldStall(string stallName, int cost, string ownerName, int
tvSet)
        {
            this.StallName = stallName;
            this.Cost = cost;
            this.OwnerName = ownerName;
            this.TvSet = tvSet;
        }
        public void display()
        {
            Console.WriteLine($"StallName : {StallName}");
            Console.WriteLine($"Cost : {Cost}");
            Console.WriteLine($"OwnerName : {OwnerName}");
            Console.WriteLine($"Number of TV Set : {TvSet}");
        }

    }

    class PremiumStall : Stall
    {
        public string StallName { get; private set; }
        public int Cost { get;private set; }
        public string OwnerName { get; private set; }
        public int Projector { get; private set; }

        public PremiumStall(string stallName, int cost, string ownerName, int
projector)
        {
            this.StallName = stallName;
            this.Cost = cost;
            this.OwnerName = ownerName;
            this.Projector = projector;
        }
        public void display()
        {
            Console.WriteLine($"StallName : {StallName}");
            Console.WriteLine($"Cost : {Cost}");
            Console.WriteLine($"OwnerName : {OwnerName}");
            Console.WriteLine($"Number of Projectors : {Projector}");
```

```csharp
            }
        }

    class ExecutiveStall : Stall
    {
        public string StallName { get; private set; }
        public int Cost { get; private set; }
        public string OwnerName { get; private set; }
        public int Screen { get; private set; }
        public ExecutiveStall(string stallName, int cost, string ownerName, int
screen)
        {
            this.StallName = stallName;
            this.Cost = cost;
            this.OwnerName = ownerName;
            this.Screen = screen;
        }
        public void display()
        {
            Console.WriteLine($"StallName : {StallName}");
            Console.WriteLine($"Cost : {Cost}");
            Console.WriteLine($"OwnerName : {OwnerName}");
            Console.WriteLine($"Number of Screens : {Screen}");
        }
    }

    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("ChooseStall Type");
            Console.WriteLine("1)Gold Stall");
            Console.WriteLine("2)Premium Stall");
            Console.WriteLine("3)Executive Stall");
            int option;
            option = Convert.ToInt32(Console.ReadLine());
            switch (option)
            {
                case 1:
                    GoldStall g = new GoldStall("The Mechanic", 120000,
"Johnson", 10);
                    g.display();
                    break;
                case 2:
                    PremiumStall p = new PremiumStall("Knitting plaza", 300000,
"Zain", 20);
                    p.display();
                    break;
                default:
                    ExecutiveStall e = new ExecutiveStall("Fruits Hunt", 10000,
"Uber", 7);
                    e.display();
                    break;
            }
            Console.ReadLine();
        }
    }
}
```

OUTPUT :

```
D:\.NET Framework\ConsoleApp\Interface\stallproblem\stallproblem\bin\Debug\stallproblem.exe
ChooseStall Type
1)Gold Stall
2)Premium Stall
3)Executive Stall
1
StallName : The Mechanic
Cost : 120000
OwnerName : Johnson
Number of TV Set : 10
```

```
D:\.NET Framework\ConsoleApp\Interface\stallproblem\stallproblem\bin\Debug\stallproblem.exe
ChooseStall Type
1)Gold Stall
2)Premium Stall
3)Executive Stall
2
StallName : Knitting plaza
Cost : 300000
OwnerName : Zain
Number of Projectors : 20
```

```
D:\.NET Framework\ConsoleApp\Interface\stallproblem\stallproblem\bin\Debug\stallproblem.exe
ChooseStall Type
1)Gold Stall
2)Premium Stall
3)Executive Stall
3
StallName : Fruits Hunt
Cost : 10000
OwnerName : Uber
Number of Screens : 7
```

INTERFACE -2 :

CODE :

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

namespace banksystem
{
    interface Bank
    {
        void InterestCalc(int a);
    }

    class Deposit : Bank
    {
        string Customer;
        double Balance;
        float Rate;

        public Deposit(string customer,double balance,float rate)
        {
            this.Customer = customer;
            this.Balance = balance;
            this.Rate = rate;
        }

        public void InterestCalc(int months)
        {
            float res = 0;
            if (Balance>0 && Balance<1000)
            {
                Console.WriteLine($"Interest rate : {res}");
            }
            else
            {
                res = months * Rate;
                Console.WriteLine($"Interest rate : {res}");
            }


        }



    }
    class Loan : Bank
    {
        string Customer;
        double Balance;
        float Rate;

        public Loan (string customer, double balance, float rate)
        {
            this.Customer = customer;
            this.Balance = balance;
            this.Rate = rate;
```

```csharp
        }

        public void InterestCalc(int months)
        {
            float res = 0;
            if ((Customer== "Individual")&&(months<3))
            {
                Console.WriteLine($"Interest rate : {res}");
            }
            else if ((Customer == "Company") && (months < 2))
            {
                Console.WriteLine($"Interest rate : {res}");
            }
            else
            {
                if (Customer == "Individual")
                {
                    months = months - 3;
                    res = months * Rate;
                    Console.WriteLine($"Interest rate : {res}");
                }
                else
                {
                    months = months - 2;
                    res = months * Rate;
                    Console.WriteLine($"Interest rate : {res}");

                }

            }


        }

}
class Mortage : Bank
{
    string Customer;
    double Balance;
    float Rate;
    public Mortage(string customer, double balance, float rate)
    {
        this.Customer = customer;
        this.Balance = balance;
        this.Rate = rate;
    }

    public void InterestCalc(int months)
    {
        float res = 0;
        if ((Customer == "Individual") && (months < 6))
        {
            Console.WriteLine($"Interest rate : {res}");
        }
        else if ((Customer == "Company") && (months < 12))
        {
            res = (months * Rate)/2;
            Console.WriteLine($"Interest rate : {res}");
        }
        else
        {
```

```csharp
                if (Customer == "Individual")
                {
                    months = months - 6;
                    res = months * Rate;
                    Console.WriteLine($"Interest rate : {res}");
                }
                else
                {
                    res = (12 * Rate) / 2;
                    months = months - 12;
                    res += months * Rate;
                    Console.WriteLine($"Interest rate : {res}");

                }

            }


        }


    }
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Choose Types Of Accounts :");
            Console.WriteLine("-----------------------------------------");
            Console.WriteLine("1. Deposit Account");
            Console.WriteLine("2. Loan Account");
            Console.WriteLine("3. Mortage Account");

            int accountType;
            accountType = Convert.ToInt32(Console.ReadLine());

            switch(accountType)
            {
                case 1:
                    double bal;
                    float ir;
                    int m;
                    Console.Write("Enter Balance :");
                    bal = Convert.ToDouble(Console.ReadLine());
                    Console.Write("Enter Interest Rate :");
                    ir = float.Parse(Console.ReadLine());
                    Deposit d = new Deposit("Individual", bal, ir);
                    Console.Write("Enter No.Of.Months :");
                    m = Convert.ToInt32(Console.ReadLine());
                    d.InterestCalc(m);
                    break;
                case 2:
                    double b;
                    float i;
                    string acc;
                    int m1;
                    Console.Write("Enter Balance :");
                    b = Convert.ToDouble(Console.ReadLine());
                    Console.Write("Enter Interest Rate :");
                    i = float.Parse(Console.ReadLine());
                    Console.WriteLine("Choose Account Type :");
                    Console.WriteLine("1.Individual");
                    Console.WriteLine("2.Company");
```

```csharp
                    int no;
                    no = Convert.ToInt32(Console.ReadLine());
                    if(no==1)
                    {
                        acc = "Individual";
                    }
                    else
                    {

                        acc = "Company";
                    }
                    Loan l = new Loan(acc, b, i);
                    Console.Write("Enter No.Of.Months :");
                    m1 = Convert.ToInt32(Console.ReadLine());
                    l.InterestCalc(m1);
                    break;
                case 3:
                    double b1;
                    float i1;
                    string acc1;
                    int m2;
                    Console.Write("Enter Balance :");
                    b1 = Convert.ToDouble(Console.ReadLine());
                    Console.Write("Enter Interest Rate :");
                    i1 = float.Parse(Console.ReadLine());
                    Console.WriteLine("Choose Account Type :");
                    Console.WriteLine("1.Individual");
                    Console.WriteLine("2.Company");
                    int no1;
                    no1 = Convert.ToInt32(Console.ReadLine());
                    if (no1 == 1)
                    {
                        acc1 = "Individual";
                    }
                    else
                    {

                        acc1 = "Company";
                    }
                    Mortage mo = new Mortage(acc1, b1, i1);
                    Console.Write("Enter No.Of.Months :");
                    m2 = Convert.ToInt32(Console.ReadLine());
                    mo.InterestCalc(m2);

                    break;


            }



            Console.ReadLine();

        }
    }
}
```

OUTPUT :

```
D:\.NET Framework\ConsoleApp\Interface\banksystem\banksystem\bin\Debug\banksystem.exe
Choose Types Of Accounts :
-------------------------------------------
1. Deposit Account
2. Loan Account
3. Mortage Account
1
Enter Balance :900
Enter Interest Rate :5
Enter No.Of.Months :5
Interest rate : 0
```

```
D:\.NET Framework\ConsoleApp\Interface\banksystem\banksystem\bin\Debug\banksystem.exe
Choose Types Of Accounts :
-------------------------------------------
1. Deposit Account
2. Loan Account
3. Mortage Account
2
Enter Balance :1100
Enter Interest Rate :5
Choose Account Type :
1.Individual
2.Company
1
Enter No.Of.Months :6
Interest rate : 15
```

```
D:\.NET Framework\ConsoleApp\Interface\banksystem\banksystem\bin\Debug\banksystem.exe
Choose Types Of Accounts :
-------------------------------------------
1. Deposit Account
2. Loan Account
3. Mortage Account
3
Enter Balance :2000
Enter Interest Rate :5
Choose Account Type :
1.Individual
2.Company
1
Enter No.Of.Months :7
Interest rate : 5
```