

DATE : 21/08/2024

### Assignment -3

#### 1. OpenableInterface

##### CODE:

```
using System;
internal interface IOpenable{
    string OpenSesame();
}
public class TreasureBox:IOpenable{
    public string OpenSesame(){
        return("Congratulations,Here is your lucky win");
    }
}
public class Parachute:IOpenable{
    public string OpenSesame(){
        return ("Have a thrilling experience flying in air");
    }
}
public class Program
{
    public static void Main(string[] args)
    {
        Console.WriteLine ("Enter the letter found in the paper");
        string ch=Console.ReadLine();
        if(ch.ToLower().Equals("t")){
            TreasureBox tb=new TreasureBox();
            Console.WriteLine(tb.OpenSesame());
        }
        else if(ch.ToLower().Equals("p")){
            Parachute p=new Parachute();
            Console.WriteLine(p.OpenSesame());
        }
    }
}
```

## OUTPUT:

```
Output Clear  
mono /tmp/wnjKGmZHCs.exe  
Enter the letter found in the paper  
T  
Congratulations,Here is your lucky win  
=== Code Execution Successful ===
```

30°C Haze 16:44 21-08-2024

```
Output Clear  
mono /tmp/11pTotcubo.exe  
Enter the letter found in the paper  
P  
Have a thrilling experience flying in air  
=== Code Execution Successful ===
```

30°C Haze 16:45 21-08-2024

## 2.Flight Status

### CODE:

```
using System;
using System.Collections.Generic;
public class Program
{

    public static string flightStatus(string flightNo){
        Dictionary<string,DateTime> flight = new Dictionary<string,DateTime>();
        flight.Add("AR456",Convert.ToDateTime("6:30:30"));
        flight.Add("ZW346",Convert.ToDateTime("18:20:40"));
        flight.Add("BR267",Convert.ToDateTime("22:40:20"));

        if(flight.ContainsKey(flightNo)){
            DateTime start_time=flight[flightNo];
            DateTime current=DateTime.Now;
            if(start_time<current){
                return "Flight Already Left";
            }
            else{

            }
        }
        else{
            return("Invalid Flight No");
        }
    }

    public static void Main(string[] args)
    {
        Console.WriteLine("Enter the Flight Number :");
        string fno=Console.ReadLine();
        string result=flightStatus(fno);
        Console.WriteLine(result);
    }
}
```

## OUTPUT:

```
Output Clear  
mono /tmp/DrEVrTEHIH.exe  
Enter the Flight Number :  
BR267  
Time to flight: 06:12:47.6504600  
  
=== Code Execution Successful ===
```

26°C Windy 19:17 21-08-2024

```
Output Clear  
mono /tmp/SrLZ1ercp4.exe  
Enter the Flight Number :  
AR456  
Flight Already Left  
  
=== Code Execution Successful ===
```

26°C Windy 19:17 21-08-2024

### 3. Product Details

#### CODE:

```
using System;
using System.Collections.Generic;
public class Product
{
    string _productName;
    string _serialNumber;
    DateTime _purchaseDate;
    double _cost;
    public Product(string productName, string serialNumber, DateTime purchaseDate,
double cost)
    {
        _productName = productName;
        _serialNumber = serialNumber;
        _purchaseDate = purchaseDate;
        _cost = cost;
    }

    public override string ToString()
    {
        string Date = _purchaseDate.ToString("dd-MM-yyyy");
        return $"{_productName,-15}{_serialNumber,-15}{Date,-15}{_cost,-15}";
    }
}

public class Program
{
    public static void Main()
    {
        List<Product> products = new List<Product>();
        products.Add(new Product("HairTrimmer", "HT123", new DateTime(2017, 2,
10), 800));
        products.Add(new Product("Steel Box", "SB231", new DateTime(2018, 4, 11),
250));
        products.Add(new Product("Rope", "RP240", new DateTime(2019, 5, 13), 100));
        Console.WriteLine(String.Format("{0,-15}{1,-15}{2,-15}{3,-15}", "Product
Name", "Serial Number", "Purchase Date", "Purchase Cost"));

        foreach (Product p in products)
        {
            Console.WriteLine(p.ToString());
        }
    }
}
```

OUTPUT:

Output

Clear

```
mono /tmp/0ySv3q2DoK.exe
Product Name  Serial Number  Purchase Date  Purchase Cost
HairTrimmer   HT123          10-02-2017     800
Steel Box     SB231          11-04-2018     250
Rope          RP240          13-05-2019     100

=== Code Execution Successful ===
```

26°C Windy

19:30

ENG

21-08-2024