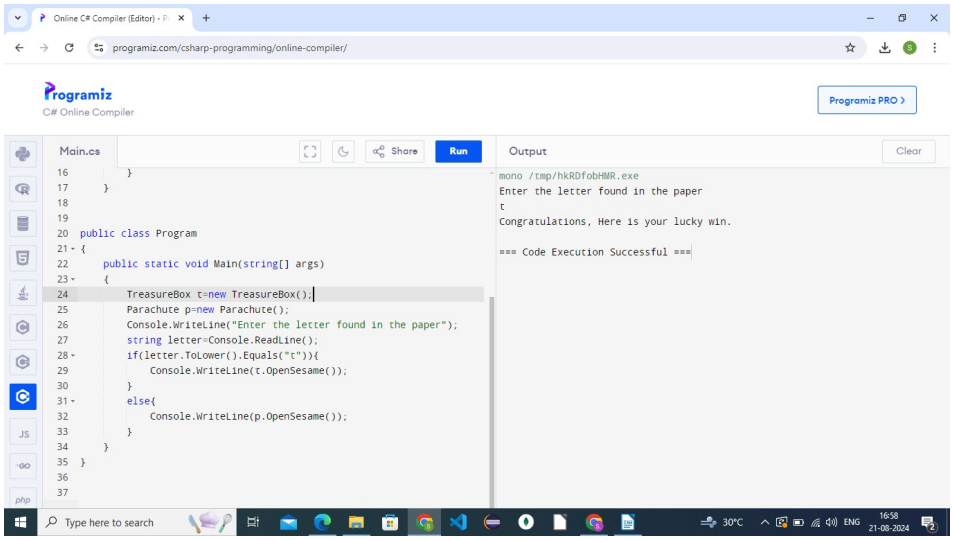
DAY 3
C#

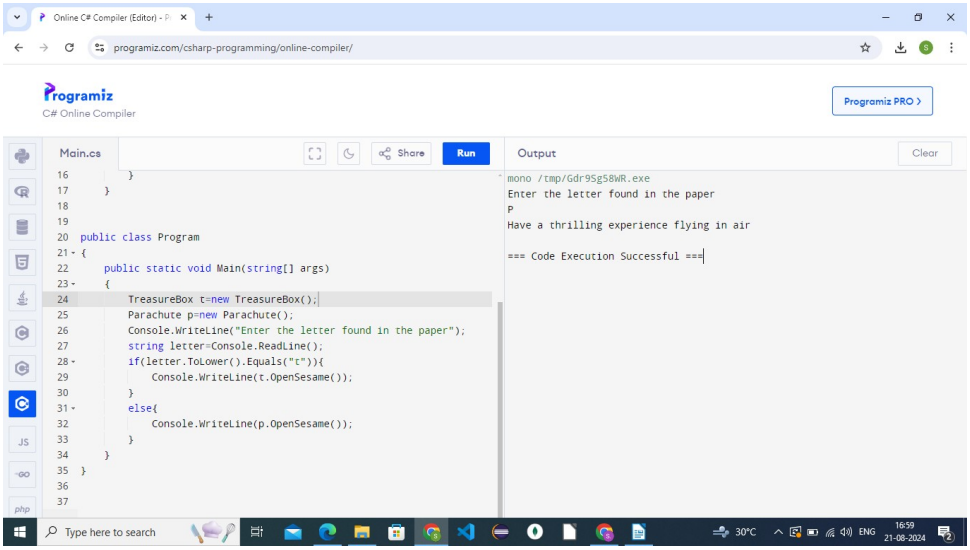
1. OPENABLEINTERFACE

CODE:

```
using System;
    public 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)
   {
      TreasureBox t=new TreasureBox();
      Parachute p=new Parachute();
      Console.WriteLine("Enter the letter found in the paper");
      string letter=Console.ReadLine();
      if(letter.ToLower().Equals("t")){
         Console.WriteLine(t.OpenSesame());
      }
      else{
         Console.WriteLine(p.OpenSesame());
      }
   }
}
```

OUTPUT:


Online C# Compiler (Editor)
programiz.com/csharp-programming/online-compiler/
Programiz
C# Online Compiler
Programiz PRO
Main.cs
Share
Run
Output
Clear
mono /tmp/hkRDfobHWR.exe
Enter the letter found in the paper
t
Congratulations, Here is your lucky win.
=== Code Execution Successful ===
Type here to search
30°C
ENG
16:58
21-08-2024

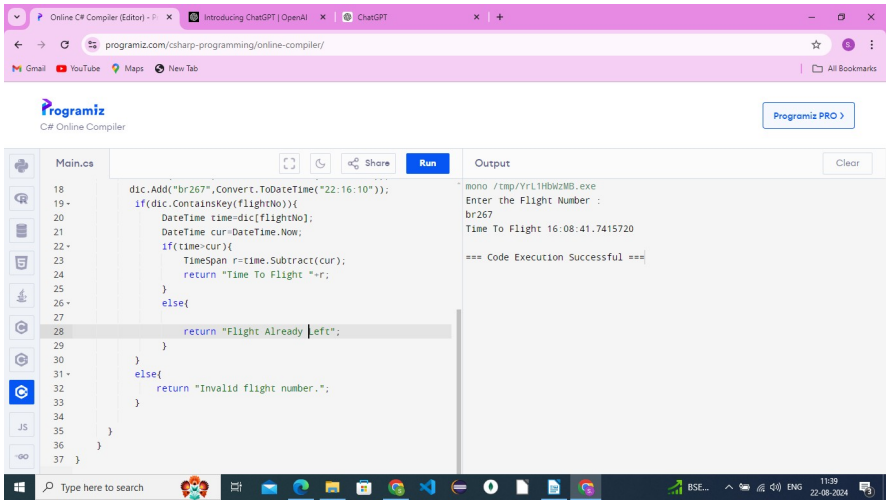

Online C# Compiler (Editor)
programiz.com/csharp-programming/online-compiler/
Programiz
C# Online Compiler
Programiz PRO
Main.cs
Share
Run
Output
Clear
mono /tmp/Gdr9Sg58WR.exe
Enter the letter found in the paper
P
Have a thrilling experience flying in air
=== Code Execution Successful ===
Type here to search
30°C
ENG
16:59
21-08-2024

2. FLIGHT STATUS

CODE:

```
using System;
using System.Collections.Generic;

public class HelloWorld
{
   public static void Main(string[] args)
   {
     Console.WriteLine("Enter the Flight Number : ");
     string flight_no=Console.ReadLine();

     Console.WriteLine(flightStatus(flight_no));



      static string flightStatus(string flightNo){
         Dictionary<string,DateTime>dic=new Dictionary<string,DateTime>();
        dic.Add("zw346",Convert.ToDateTime("11:15:50"));
        dic.Add("br267",Convert.ToDateTime("22:16:10"));
         if(dic.ContainsKey(flightNo)){
             DateTime time=dic[flightNo];
             DateTime cur=DateTime.Now;
             if(time>cur){
                TimeSpan r=time.Subtract(cur);
                return "Time To Flight "+r;
             }
             else{

                return "Flight Already Left";
             }
         }
         else{
            return "Invalid flight number.";
         }

     }
   }
}
```

OUTPUT:



3. PRODUCT DETAILS

CODE:

```
using System;
using System.Collections.Generic;
public class Program
{
   public static void Main(string[] args)
   {
       List<Product> p = new List<Product>();
      p.Add(new Product("HairTrimmer", "HT123", new DateTime(2017, 2, 10), 800));
      p.Add(new Product("Steel Box", "SB231", new DateTime(2018, 4, 11), 250));
      p.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 res in p)
      {
         Console.WriteLine(res.ToString());
      }
   }
}

public class Product{
   public string _productName{set; get;}
   public string _serialNumber{set; get;}
   public DateTime _purchaseDate{set; get;}
```

```
    public double _cost{set; get;}

    public Product(string _productName,string _serialNumber,DateTime
_purchaseDate,double _cost){
        this._productName=_productName;
        this._serialNumber=_serialNumber;
        this._purchaseDate=_purchaseDate;
        this._cost=_cost;

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

OUTPUT: