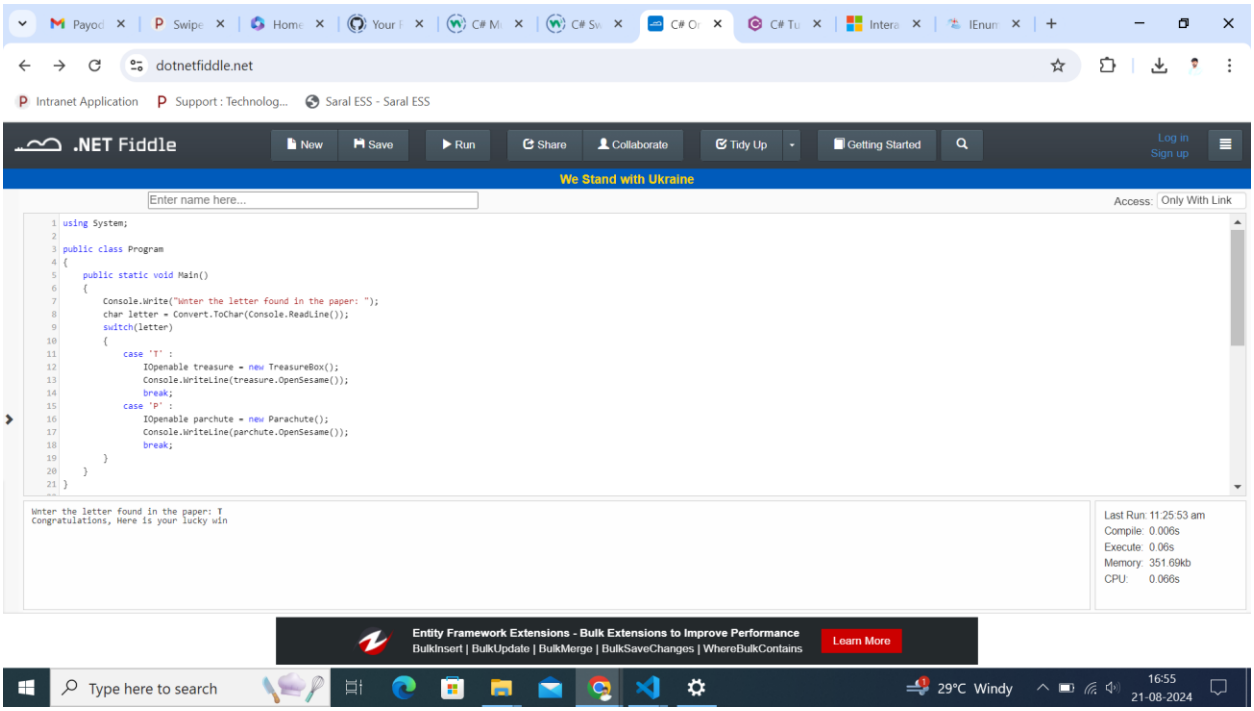
3 – DAY – TASK (21-08-2024)

1. **OpenableInterface Program Code:**

```
using System;

public class Program
{
public static void Main()
{
Console.Write("Enter the letter found in the paper: ");
char letter = Convert.ToChar(Console.ReadLine());
switch(letter)
{
case 'T' :
IOpenable treasure = new TreasureBox();
Console.WriteLine(treasure.OpenSesame());
break;
case 'P' :
IOpenable parchute = new Parachute();
Console.WriteLine(parchute.OpenSesame());
break;
}
}
}

interface IOpenable
{
string OpenSesame();
}

class TreasureBox : IOpenable
{
public string OpenSesame()
{
return "Congratulations, Here is your lucky win";
}
```

```
}

class Parachute : IOpenable
{
public string OpenSesame()
{
return "Have a thrilling experience flying in air";
}
}
```

Output:



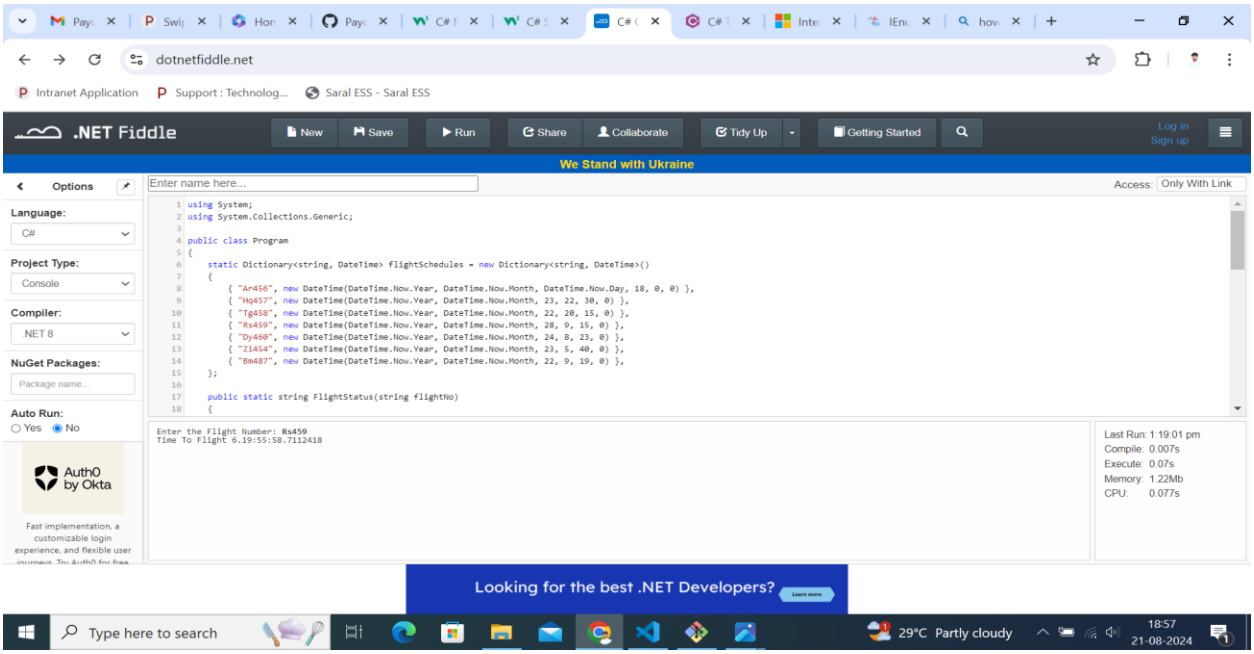
2. **FlightStatus Program Code:**

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

public class Program
{
   static Dictionary<string, DateTime> flightSchedules = new Dictionary<string,
DateTime>()
```

```
  {
    { "Ar456", new DateTime(DateTime.Now.Year, DateTime.Now.Month,
DateTime.Now.Day, 18, 0, 0) },
    { "Hq457", new DateTime(DateTime.Now.Year, DateTime.Now.Month, 23, 22,
30, 0) },
    { "Tg458", new DateTime(DateTime.Now.Year, DateTime.Now.Month, 22, 20, 15,
0) },
    { "Rs459", new DateTime(DateTime.Now.Year, DateTime.Now.Month, 28, 9, 15,
0) },
    { "Dy460", new DateTime(DateTime.Now.Year, DateTime.Now.Month, 24, 8, 23,
0) },
    { "Zi454", new DateTime(DateTime.Now.Year, DateTime.Now.Month, 23, 5, 40,
0) },
    { "Bm487", new DateTime(DateTime.Now.Year, DateTime.Now.Month, 22, 9, 19,
0) },
  };

  public static string FlightStatus(string flightNo)
  {
    if (flightSchedules.TryGetValue(flightNo, out DateTime departureTime))
    {
      TimeSpan timeLeft = departureTime - DateTime.Now;
      if (timeLeft.TotalSeconds > 0)
      {
        int days = timeLeft.Days;
        TimeSpan time = timeLeft - TimeSpan.FromDays(days);
        //return $"Time To Flight {days} days {time.ToString(@"hh\:mm\:ss\.ffffff")}";
        //Output
        //Enter the Flight Number: Rs459
        //Time To Flight 6 days 19:58:48.952218
        return $"Time To Flight {timeLeft}";
        //Output
        //Enter the Flight Number: Rs459
        //Time To Flight 6.19:55:58.7112418
      }
      else
      {
        return "Flight Already Left";
```

```
        }
      }
      else
      {
        return "Flight Not Found";
      }
    }

    public static void Main()
    {
      Console.Write("Enter the Flight Number: ");
      string flightNo = Console.ReadLine();
      Console.WriteLine(FlightStatus(flightNo));
    }
}
```

Output:



3. **ProductDetails Program Code:**

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

```
public class Program
{
public static void Main()
{
List<Product> list = new List<Product>();
list.Add(new Product("Apple", "Prod1234", DateTime.Now, 200));
list.Add(new Product("Hair Trimmer", "Prod1276", DateTime.Now, 1200));
list.Add(new Product("Steel Box", "Prod1209", DateTime.Now, 400));
list.Add(new Product("Rope", "Prod1213", DateTime.Now, 99));
list.Add(new Product("Chair", "Prod1342", DateTime.Now, 309));

Console.WriteLine(String.Format("{0,-15}{1,-15}{2,-25}{3,-15}", "Product Name",
"Serial Number","Purchase Date", "Purchase Cost"));
list.ForEach( a => Console.WriteLine(a));




}
}

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()
{
```

```
return String.Format("{0,-15}{1,-15}{2,-25}{3,-15}", _productName, _serialNumber,
_purchaseDate.ToString(@"yyyy:mm:dd"), _cost);
}

}
```

Output: