ASSIGNMENT-4

Submitted by-DAS SUKHDEV

**CONSTRUCTOR CHAINING PROGRAM**

using System;

namespace constructor\_chaining

{

class Student

{

public int sid;

public Student(int id)

{

sid=id;

Console.WriteLine("Hey I'm inside base class");

Console.WriteLine("I'm also in base class");

}

}

class Example:Student

{

public Example(int id):base (id)

{

Console.WriteLine("Hey I'm inside derived class");

Console.WriteLine(id);

}

}

class Program

{

static void Main(string[] args)

{

Example obj=new Example(102);

}

}

}

**ABSTRACT CLASS FIRST PROGRAM**

using System;

namespace abstract\_1

{

abstract class absproject

{

public void Add(int x, int y)

{

Console.WriteLine("Addition of numbers is " + (x + y));

}

public void Sub(int x, int y)

{

Console.WriteLine("Substraction of numbers is " + (x - y));

}

public abstract void mul(int x, int y);

public abstract void div(int x, int y);

}

class abschild : absproject

{

public override void mul(int i, int j)

{

Console.WriteLine("Multiplication of numbers is "+(i \* j));

}

public override void div(int i, int j)

{

Console.WriteLine("Division of numbers is "+ (i / j));

}

}

class Program

{

static void Main(string[] args)

{

var obj =new abschild();

obj.Add(10,20);

obj.mul(10,20);

obj.Sub(20,10);

obj.div(20,5);

}

}

}

**ABSTRACT CLASS SECOND PROGRAM**

using System;

namespace abstract\_2

{

abstract class person

{

public string name;

public int age;

public abstract void connection();

}

class student : person

{

public override void connection()

{

string name = this.name;

int age = this.age;

Console.WriteLine("Name of student is " + name);

Console.WriteLine("Age of student is " + age);

}

}

class faculty : person

{

public override void connection()

{

string name = this.name;

int age = this.age;

Console.WriteLine("Name of Faculty is " + name);

Console.WriteLine("Age of Faculty is " + age);

}

}

class Program

{

static void Main(string[] args)

{

var fac = new faculty();

var stu = new student();

stu.age = 25;

stu.name = "Sukhdev";

fac.age = 45;

fac.name = "Sandeep Sir";

stu.connection();

fac.connection();

}

}

}

**INTERFACE FIRST PROGRAM**

using System;

namespace interface\_1

{

interface Ifirst

{

void add(int a,int b);

}

interface Isecond:Ifirst

{

void sub(int a,int b);

}

class test:Ifirst, Isecond

{

public void add(int a,int b)

{

Console.WriteLine("Addition of two numbers is "+(a+b));

}

public void sub(int a,int b)

{

Console.WriteLine("substraction of two numbers is "+(a-b));

}

}

class interfaces

{

static void Main(string[] args)

{

var obj=new test();

obj.add(10,20);

obj.sub(70,50);

}

}

}

**INTERFACE SECOND PROGRAM**

using System;

namespace interface\_2

{

interface Istudent

{

public void branch();

}

class college:Istudent

{

public void branch()

{

Console.WriteLine("I'm from computer science branch");

}

}

class Program

{

static void Main(string[] args)

{

var obj=new college();

obj.branch();

}

}

}

**INTERFACE THIRD PROGRAM**

using System;

namespace interface\_3

{

interface Icustomer

{

public void print1();

}

interface Ishopkeeper

{

public void print2();

}

public class test:Icustomer

{

public void print1()

{

System.Console.WriteLine("Hey I'm defining the funtion inside I customer interface");

}

}

class Program

{

static void Main(string[] args)

{

var obj=new test();

obj.print1();

}

}

}

**PROGRAM DEMONSTRATING STRUCTURE**

using System;

namespace struct\_demo

{

struct Mystructure

{

//public int i;

//int i=10; it will give error in structures but it will run in the case of classes

public string name;

public int id;

public Mystructure(string n, int id)

{

name=n;

this.id=id;

}

public void display()

{

System.Console.WriteLine(name);

System.Console.WriteLine(id);

}

}

class Program

{

static void Main(string[] args)

{

Mystructure m1=new Mystructure("sukhdev",100);

m1.display();

//m1.i=10;

//if we are not initializing value of i then it will give error when we are not using the new keyword then need to initialize data member explicitly

}

}

}

**PROGRAM DEMONSTRATING ENUM (USER DEFINED DATA TYPE)**

using System;

namespace enum\_demo

{

public enum days //enum is user defined data type

{

Monday,

Tuesday,

Wednesday,

Thursday,

Friday

}

class test

{

static void Main()

{

for(int i=0;i<=4;i++)

{

days d=(days)i;

Console.WriteLine(d);

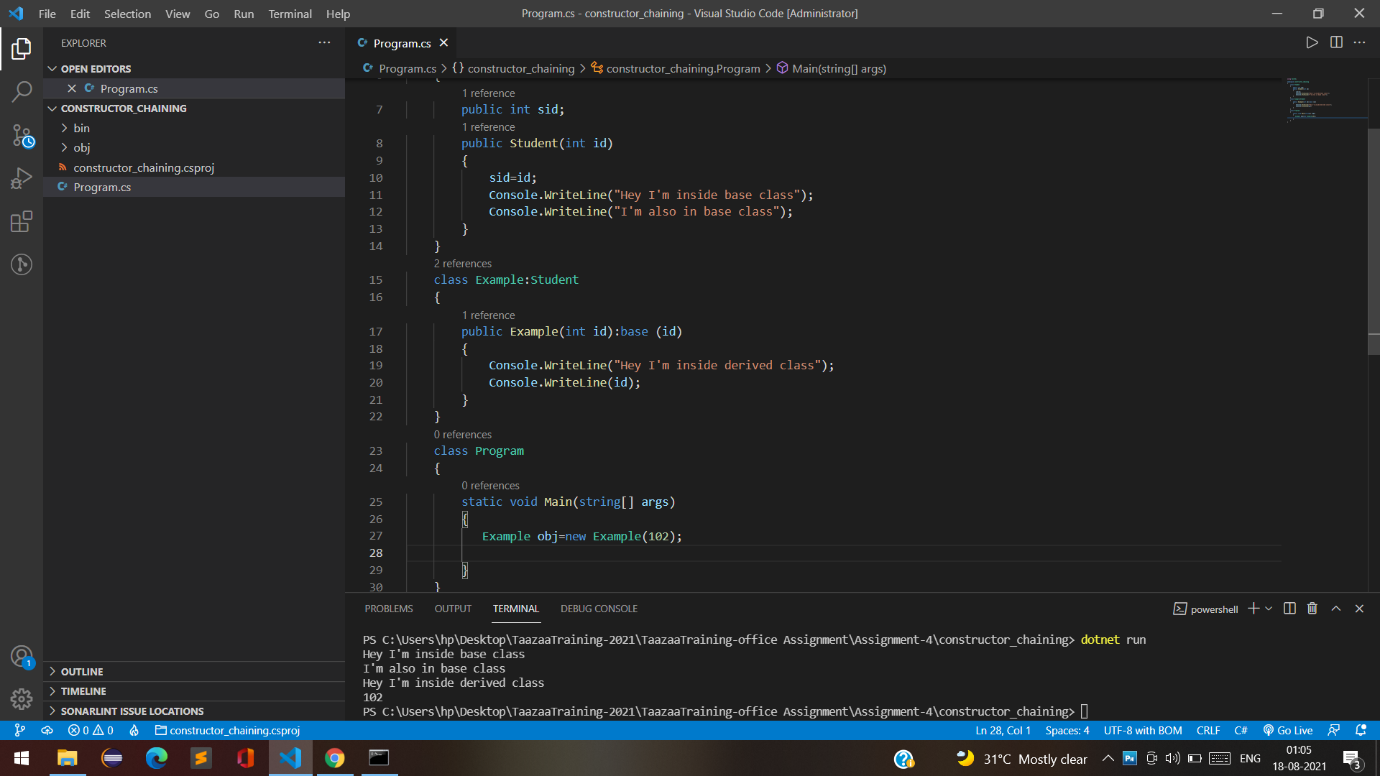
}

}

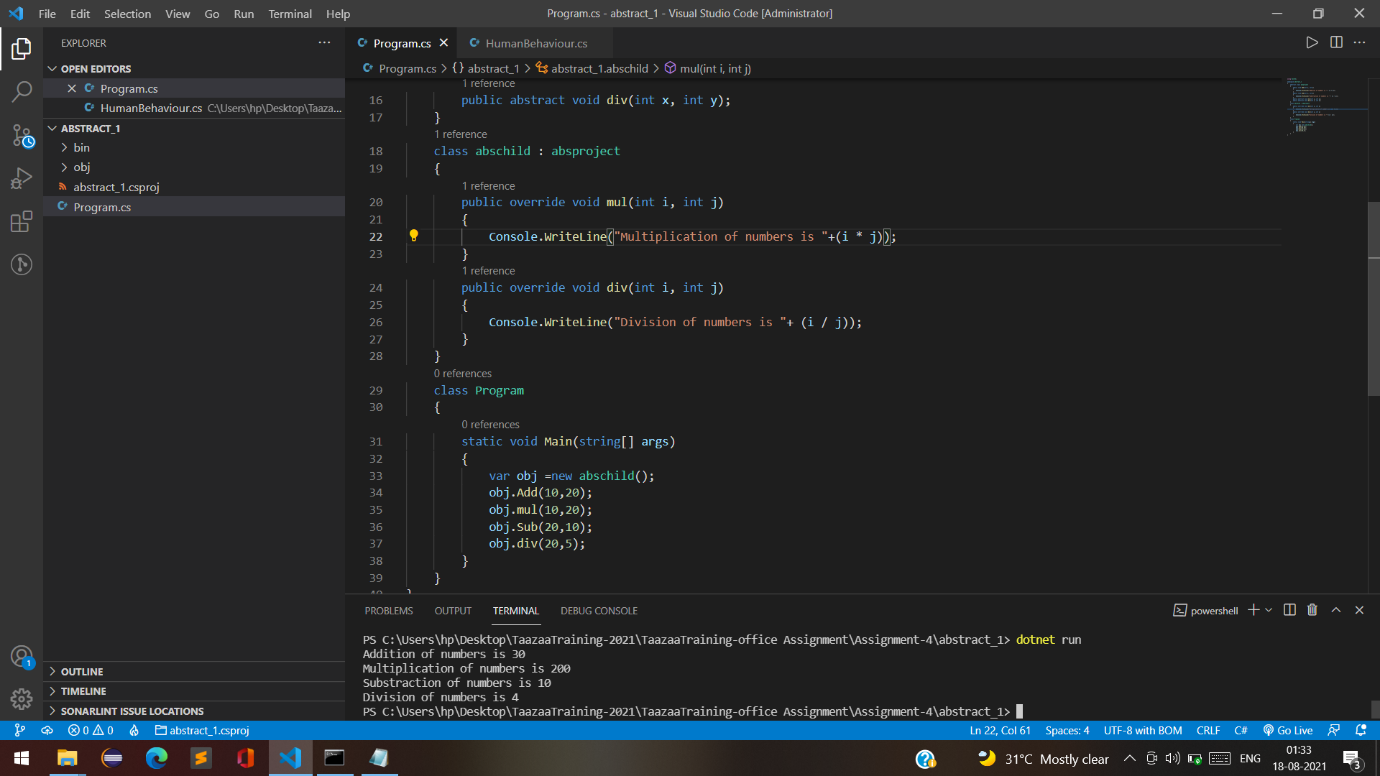
}

}

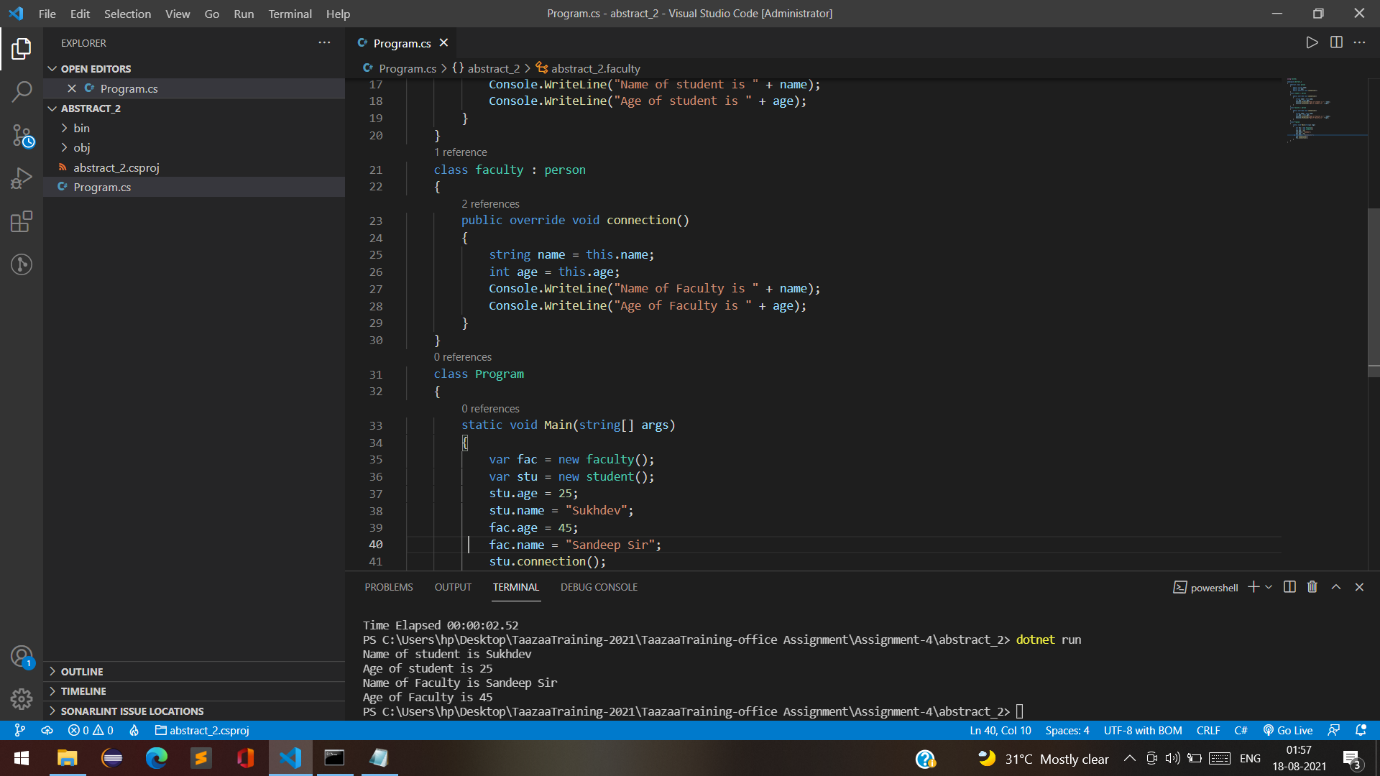
**SCREENSHOTS**

****

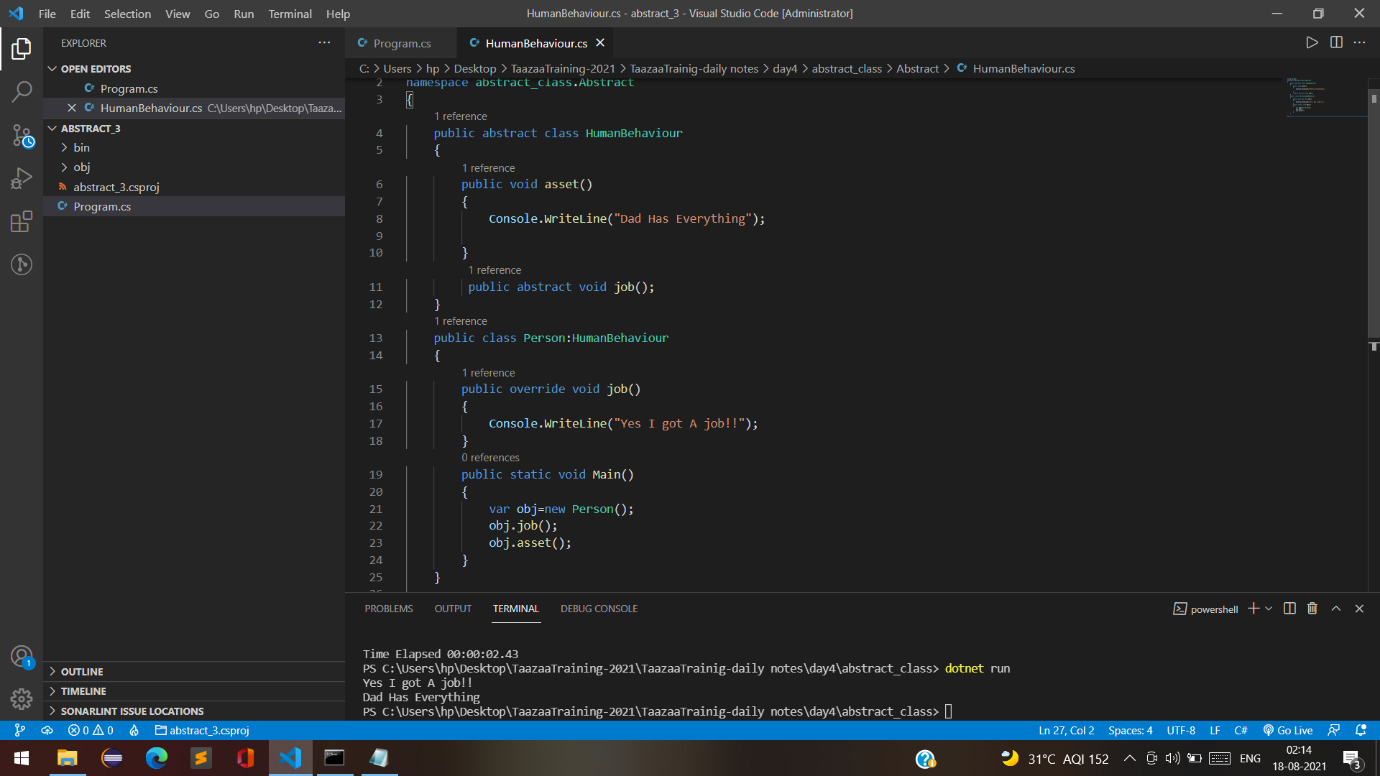
**CONSTRUCTOR CHAINING**

****

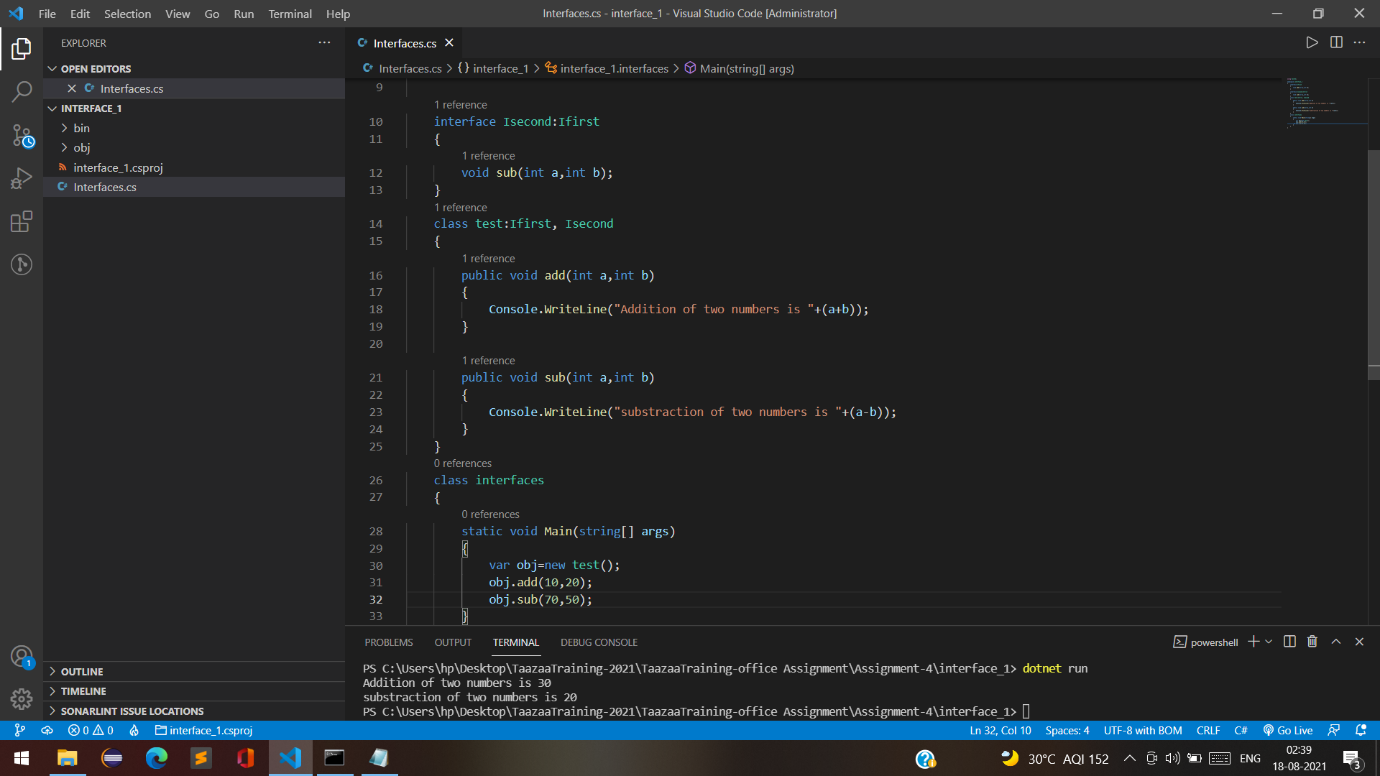
**ABSTRACT\_1**

****

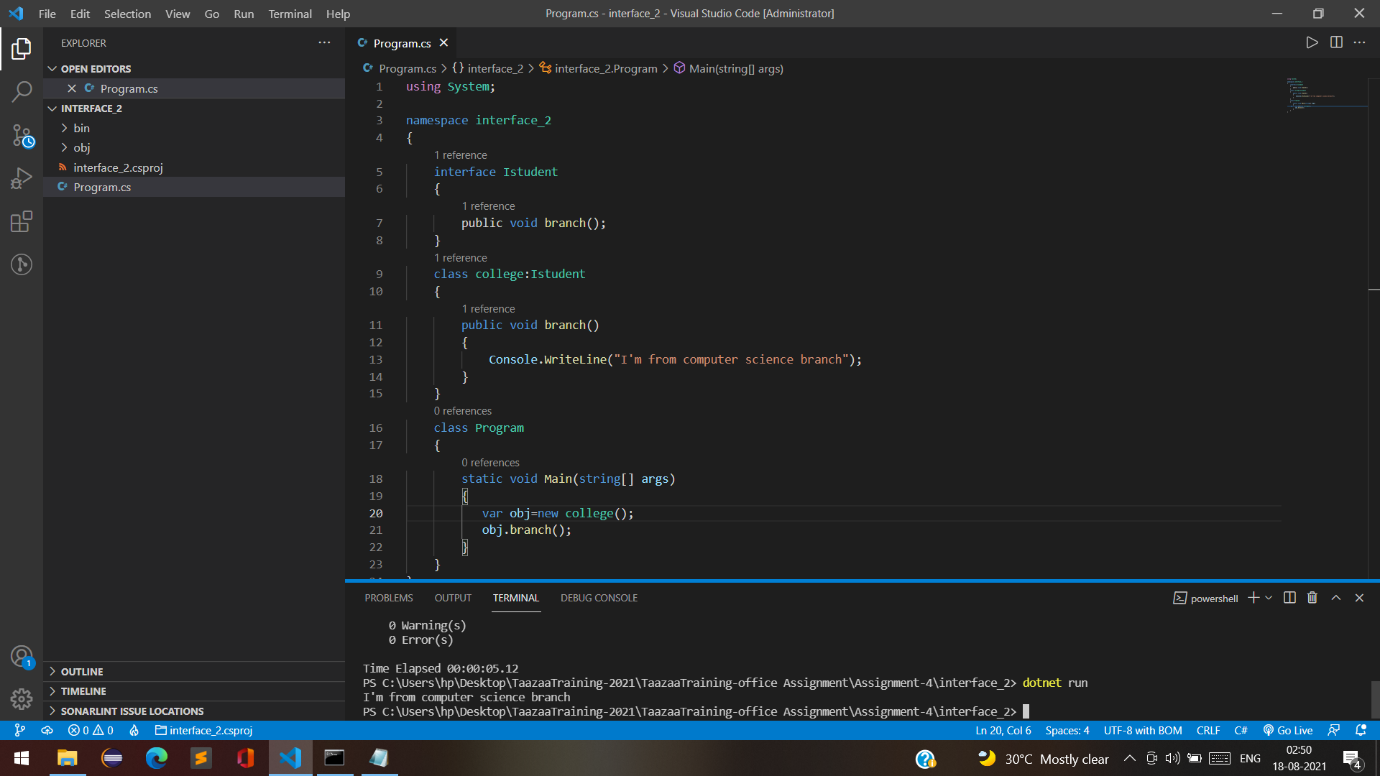
**ABSTRACT\_2**

****

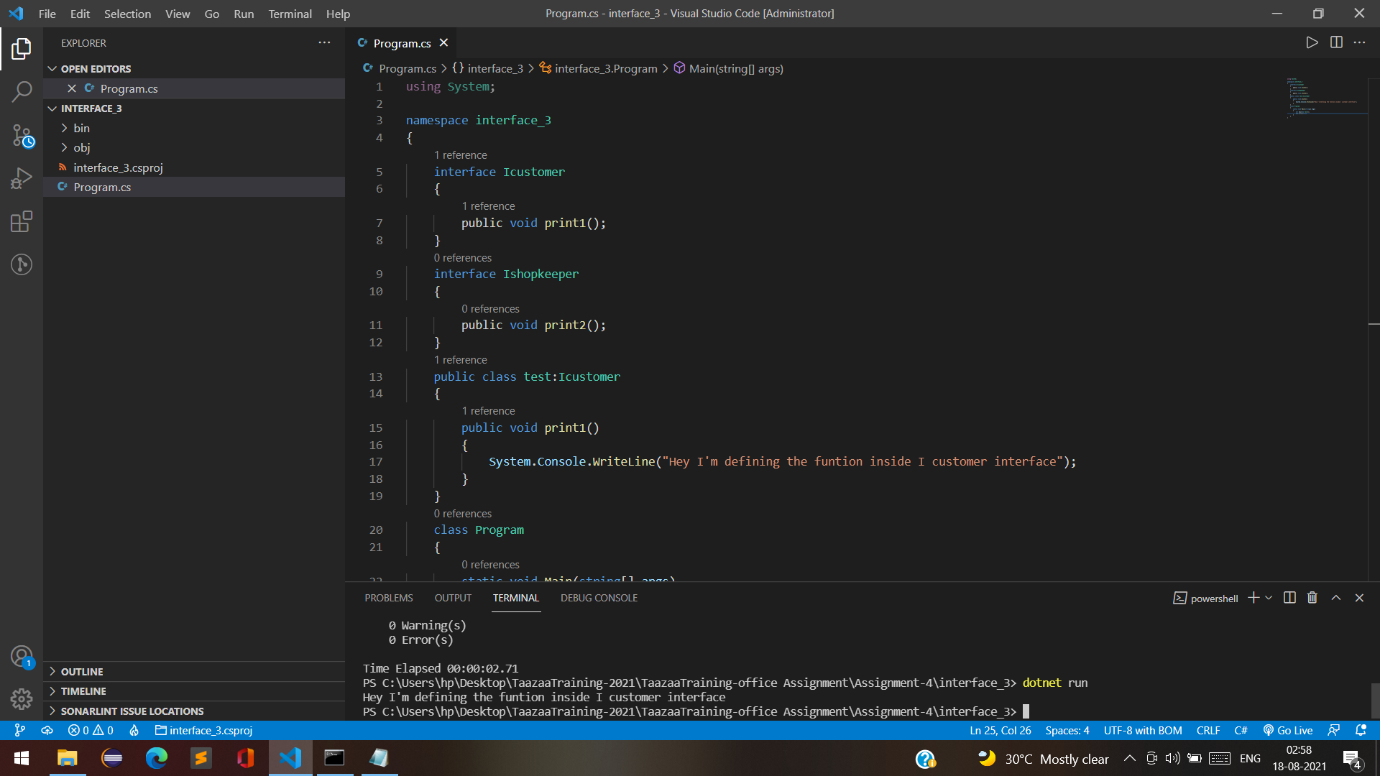
**ABSTRACT\_3**

****

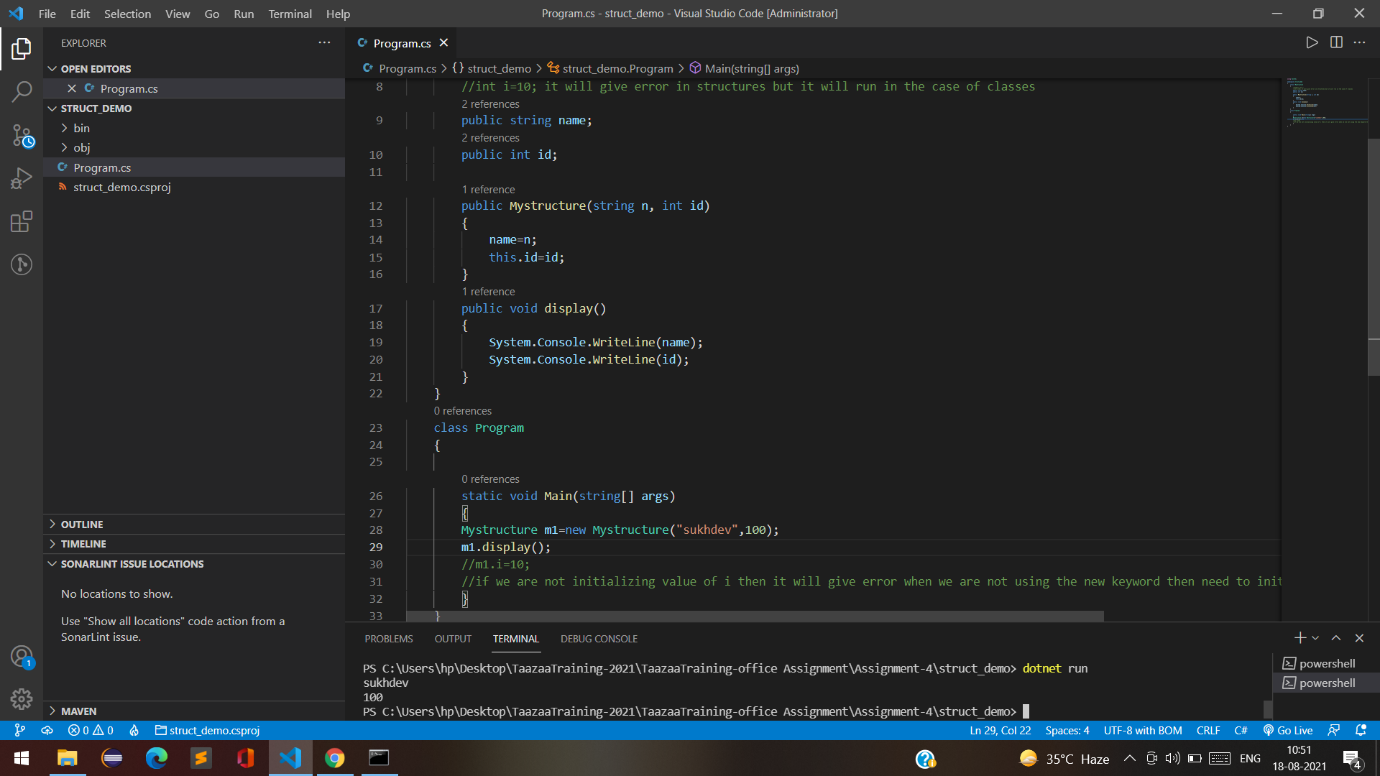
**INTERFACE\_1**

****

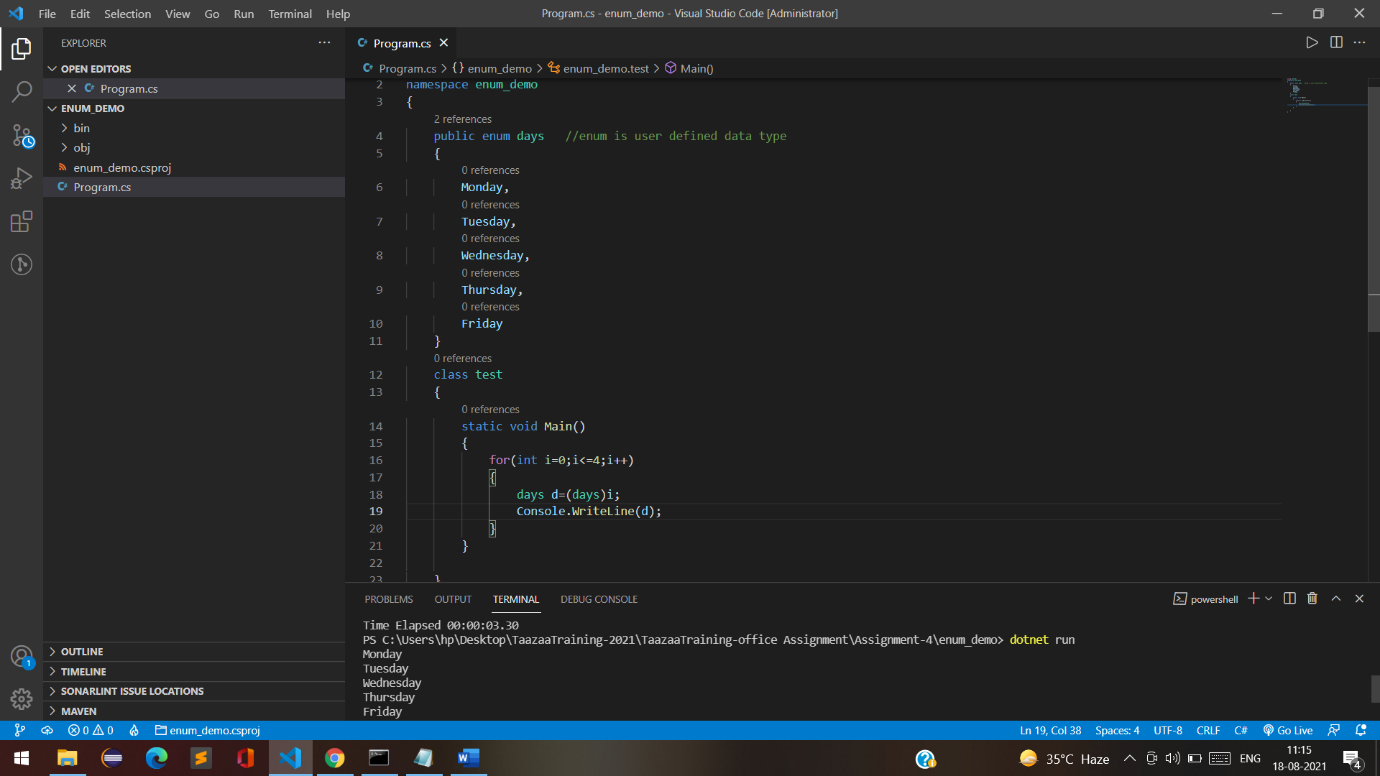
**INTERFACE\_2**

****

**INTERFACE\_3**

****

**STRUCT\_DEMO**

****

**ENUM\_DEMO**