**Q)** **Write a c# program to calculate the simple Interests code and implement Nunit testing for the same.Create class library which will contain the class SimpleCalculation and the method for calculating simpleInterests.Formula is PNR/100.PRINCIPAL,RATE OF INTEREST,YEARS.Create Nunit test project and add the reference to above library created and test the method whether simpleinterest is calculated properly or not.**

**Soln:**

**Code:**

//dll for simpleInterestcal

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace simpleInterestcal

{

public class simpleInterestcal

{

public double simplecal(double pri,double rateofinterest, double y)

{

return (pri \* rateofinterest \* y) / 100;

}

}

}

//Testcases for simpleInterestcal

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using simpleInterestcal;

namespace simpleInterestcaltest

{

[TestFixture]

public class simpleInterestcaltest

{

[Test]

public void SimcalMethodTest()

{

siscal c = new siscal();

double r = c.simplecal(100, 30, 1);

Assert.That(r, Is.EqualTo(30));

}

[Test]

[TestCase(100, 5, 1,5)]

[TestCase(100, 9,1, 10)]

[TestCase(100, 13,1, 13)]

public void SimcalMethodTest(int n1, int n2, int n3,int e)

{

siscal c = new siscal();

double r = c.simplecal(n1, n2, n3 );

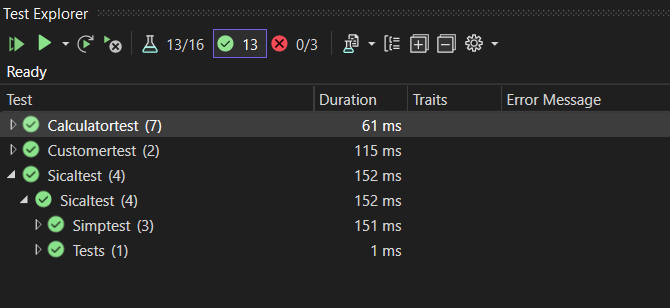
Assert.AreEqual(e, r);

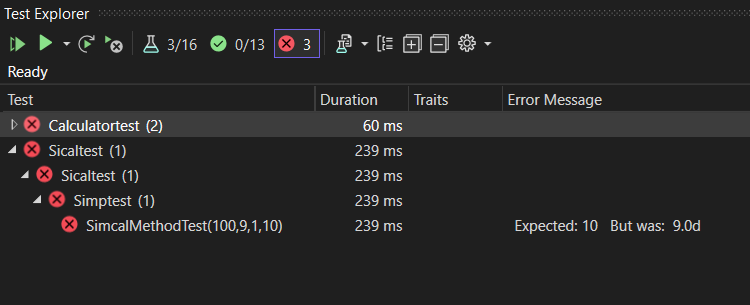
}

}

}

**Output:**





**Q) Write a program to find whether the number is odd or even number. Accept the number from user.Create the Nunit test project for above application and check whether the above logic of odd and even number is working properly or not.**

**Soln:**

**Code:**

//evenoddtest

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using Evenod;

using System.Threading.Tasks;

namespace Evenoddtest

{

[TestFixture]

public class Simptest

{

[Test]

public void evenodMethodTest()

{

evenod c = new evenod();

string r = c.iseven(8);

Assert.That(r, Is.EqualTo("Even"));

}

[Test]

[TestCase(100,"Even")]

[TestCase( 9, "Odd")]

[TestCase(13, "Even")]

public void evenodMethodTest(int n1,string r)

{

evenod c = new evenod();

string e = c.iseven(n1);

Assert.AreEqual(e, r);

}

}

}

//evenod dll

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Evenod

{

public class evenod

{

public string iseven(int x)

{

if (x % 2 == 0)

{

return ("Even");

}

else

{

return ("Odd");

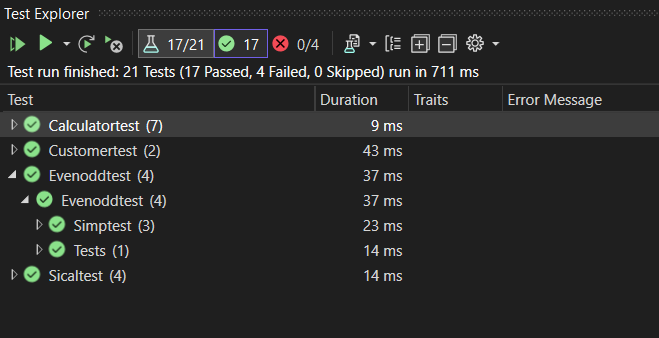
}

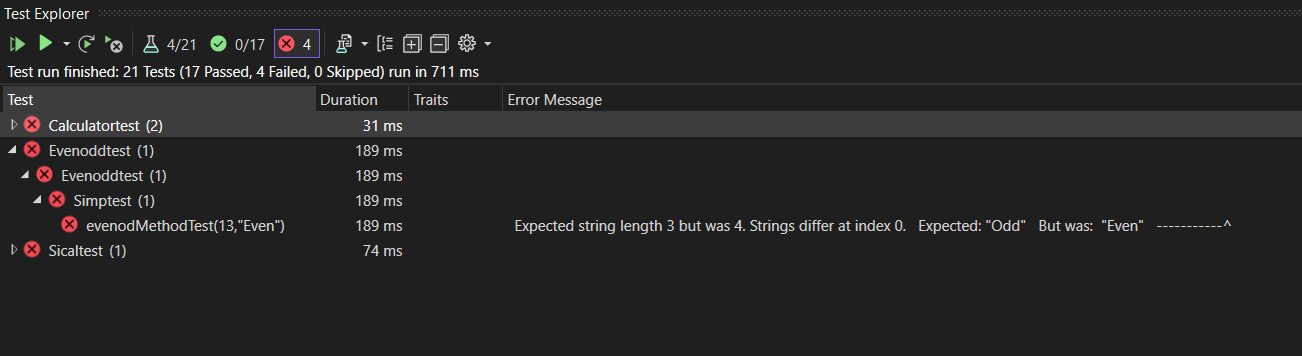
}

}

}

**Output:**





**Q) The project\_Manager class should inherit the Emp class and it will contain data member such as projectname and inherited method CalculateSalary().The Developer class should inherit the Emp class and it will contain data member such as SkillSetname (whether java or .NET) and inherited method CalculateSalary().the CalculateSalary() should display the polymorphism feature.**

**Soln:**

**Code:**

//polymorphism using CalculateSalary()

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Ass3d5

{

public class Emp

{

public int empno;

public string empName;

public double salary;

public Emp(int empno, string empName, double salary)

{

this.empno = empno;

this.empName = empName;

this.salary = salary;

}

public virtual void CalculateSalary()

{

Console.WriteLine("Emp salary="+salary);

}

}

public class Project\_Manager : Emp

{

public double s;

public string projectname;

public Project\_Manager(int empno, string empName, double salary) : base(empno, empName, salary)

{

s = salary;

}

public override void CalculateSalary()

{

Console.WriteLine("Project\_Manager salary="+s);

}

}

public class Developer : Emp

{

public string skillset;

public Developer(int empno, string empName, double salary) : base(empno, empName, salary)

{

}

public override void CalculateSalary()

{

double s =salary;

Console.WriteLine("Developer salary="+ s);

}

}

public class Ass3d5

{

public static void Main()

{

Emp e = new Emp(1, "Nikhil", 100000);

Project\_Manager p = new Project\_Manager(1, "Hulk", 2000);

Developer d = new Developer(3, "Arnold", 5000);

e.CalculateSalary();

p.CalculateSalary();

d.CalculateSalary();

}

}

}

**Output:**

