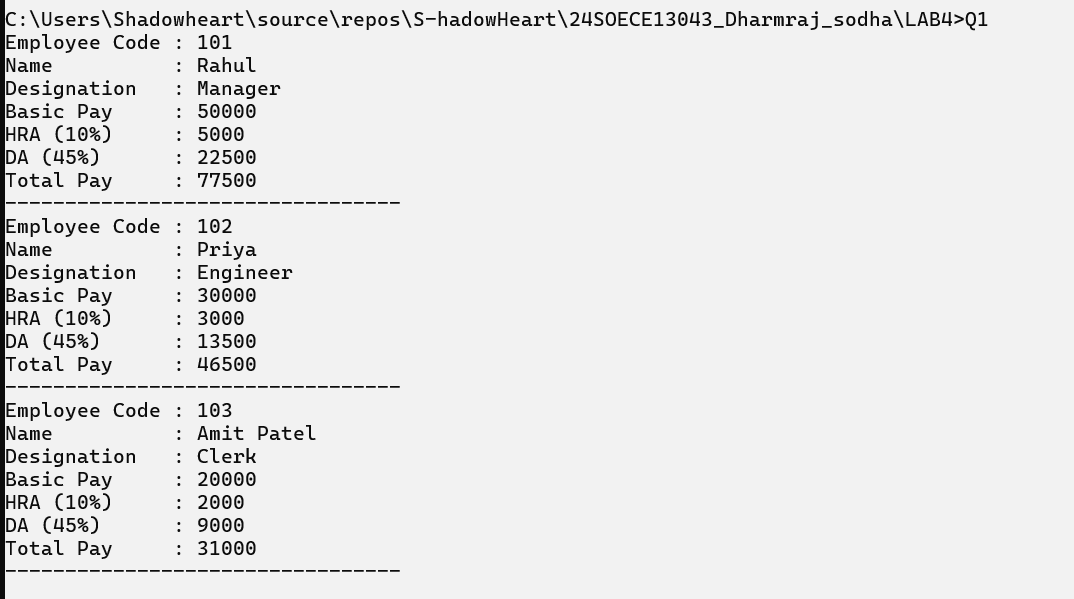
**Tutorial – 4**

**1 . The employee list for a company contains employee code, name, designation and basic pay. The employee is given a house rent allowance (HRA) of 10% of the basic pay and dearness allowance (DA) of 45% of the basic pay. The total pay of the employee is calculated as Basic Pay + HRA + DA. Write a class to define the details of the employee. Write a constructor to assign the required initial values. Add a method to calculate HRA, DA and total pay and print them. Write another class with main method. Create objects for three different employees and calculate HRA, DA and total pay.**

| **using System; namespace \_24SOECE13043\_Dharmraj\_sodha.LAB4 {  class Employee  {  public int EmpCode { get; set; }  public string Name { get; set; }  public string Designation { get; set; }  public double BasicPay { get; set; }   public Employee(int empCode, string name, string designation, double basicPay)  {  EmpCode = empCode;  Name = name;  Designation = designation;  BasicPay = basicPay;  }   public void CalculatePay()  {  double hra = 0.10 \* BasicPay;   double da = 0.45 \* BasicPay;   double totalPay = BasicPay + hra + da;   Console.WriteLine("Employee Code : " + EmpCode);  Console.WriteLine("Name : " + Name);  Console.WriteLine("Designation : " + Designation);  Console.WriteLine("Basic Pay : " + BasicPay);  Console.WriteLine("HRA (10%) : " + hra);  Console.WriteLine("DA (45%) : " + da);  Console.WriteLine("Total Pay : " + totalPay);  Console.WriteLine("---------------------------------");  }  }  class Q1  {  static void Main(string[] args)  {  Employee e1 = new Employee(101, "Rahul", "Manager", 50000);  Employee e2 = new Employee(102, "Priya", "Engineer", 30000);  Employee e3 = new Employee(103, "Amit Patel", "Clerk", 20000);  e1.CalculatePay();  e2.CalculatePay();  e3.CalculatePay();  }  } }** |
| --- |

**Output :**

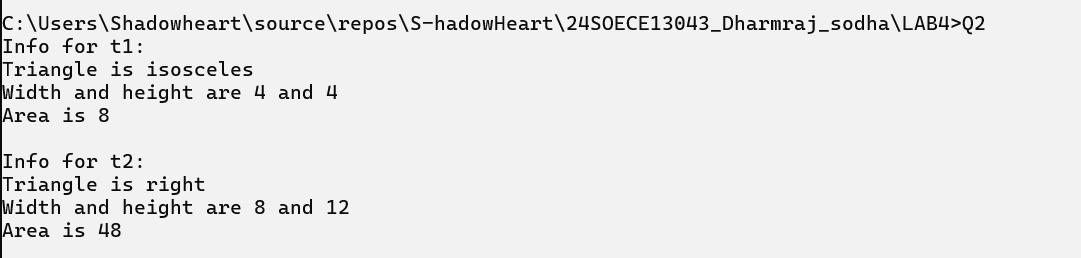
****

**2 : From the following code and given output, complete missing statements and find out error code and correct it.**

**using System;**

| **namespace \_24SOECE13043\_Dharmraj\_sodha.LAB4 {  class Shape  {  public double Width;  public double Height;   public void ShowDim()  {  Console.WriteLine("Width and height are " + Width + " and " + Height);  }  }   class Triangle : Shape  {  public string Style;   public double Area()  {  return Width \* Height / 2;  }   public void ShowStyle()  {  Console.WriteLine("Triangle is " + Style);  }  }   class Q2  {  static void Main()  {  Triangle t1 = new Triangle();  Triangle t2 = new Triangle();   t1.Width = 4.0;  t1.Height = 4.0;  t1.Style = "isosceles";   t2.Width = 8.0;  t2.Height = 12.0;  t2.Style = "right";   Console.WriteLine("Info for t1: ");  t1.ShowStyle();  t1.ShowDim();  Console.WriteLine("Area is " + t1.Area());   Console.WriteLine();   Console.WriteLine("Info for t2: ");  t2.ShowStyle();  t2.ShowDim();  Console.WriteLine("Area is " + t2.Area());  }  } }** |
| --- |

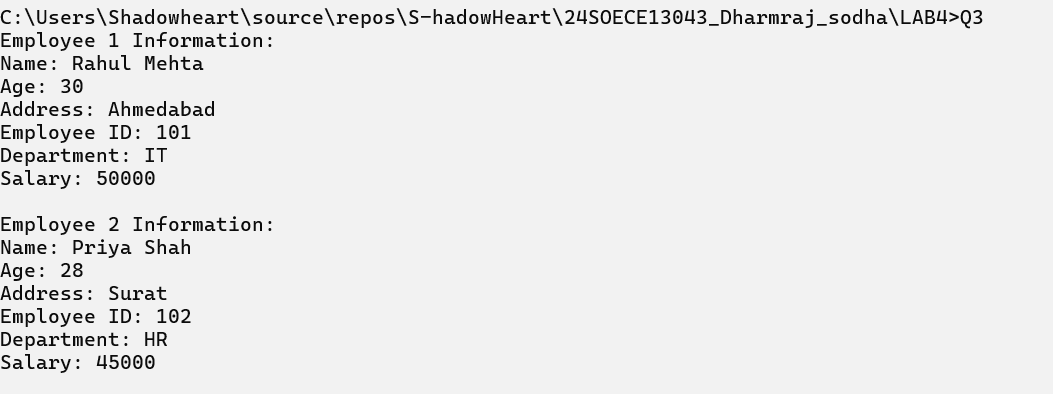
**Output:**

****

**3: Draw a real picture for single level inheritance.**

| **using System;  namespace \_24SOECE13043\_Dharmraj\_sodha.LAB4 {  class Person  {  private int age;  protected string address;  public string name;   public void SetDetails(string n, int a, string addr)  {  name = n;  age = a;  address = addr;  }   public void ShowDetails()  {  Console.WriteLine("Name: " + name);  Console.WriteLine("Age: " + age);  Console.WriteLine("Address: " + address);  }  }   class Employee : Person  {  private double salary;  protected string department;  public int empId;   public void SetEmployee(int id, double sal, string dept)  {  empId = id;  salary = sal;  department = dept;  }   public void ShowEmployee()  {  Console.WriteLine("Employee ID: " + empId);  Console.WriteLine("Department: " + department);  Console.WriteLine("Salary: " + salary);  }  }   class Q3  {  static void Main()  {  Employee e1 = new Employee();  Employee e2 = new Employee();   e1.SetDetails("Rahul Mehta", 30, "Ahmedabad");  e1.SetEmployee(101, 50000, "IT");   e2.SetDetails("Priya Shah", 28, "Surat");  e2.SetEmployee(102, 45000, "HR");   Console.WriteLine("Employee 1 Information:");  e1.ShowDetails();  e1.ShowEmployee();   Console.WriteLine();   Console.WriteLine("Employee 2 Information:");  e2.ShowDetails();  e2.ShowEmployee();  }  } }** |
| --- |

**Output:**

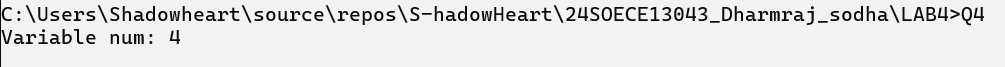
****

**4. From the following code and given output complete missing statements and find out error code and correct it.**

**using System;**

| **namespace \_24SOECE13043\_Dharmraj\_sodha.LAB4 {  class StaticVar  {  public static int num;   public StaticVar()  {  num++;  }   public void count()  {  num++;  }   public static int getNum()  {  return num;  }  }   class Q4  {  static void Main(string[] args)  {  StaticVar s = new StaticVar();   s.count();  s.count();  s.count();   Console.WriteLine("Variable num: {0}", StaticVar.getNum());   Console.ReadKey();  }  } }** |
| --- |

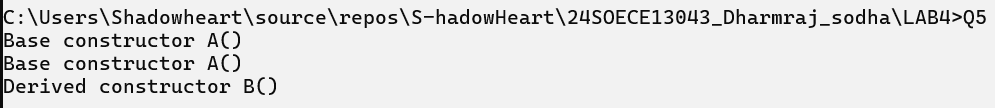
**Output :**

****

**Q5.Find out error code and correct it. Print appropriate output as desired.**

| **using System;  namespace \_24SOECE13043\_Dharmraj\_sodha.LAB4 {  public class A  {  public A(int value)  {  Console.WriteLine("Base constructor A()");  }  }   public class B : A  {  public B(int value) : base(value)  {  Console.WriteLine("Derived constructor B()");  }  }   class Q5  {  static void Main()  {  A a = new A(0);  B b = new B(1);  }  } }** |
| --- |

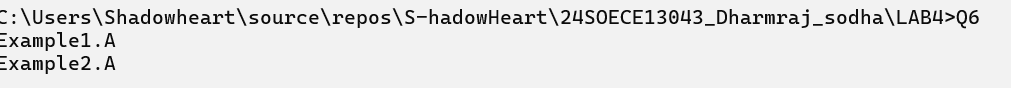
**Output :**

****

**6 . Find out error code and correct it. Print appropriate output as desired.**

| **using System;  namespace \_24SOECE13043\_Dharmraj\_sodha.LAB4 {  abstract class Test  {  protected int a;  public abstract void A();  }   class Example1 : Test  {  public override void A()  {  Console.WriteLine("Example1.A");  a++;  }  }   class Example2 : Test  {  public override void A()  {  Console.WriteLine("Example2.A");  a--;  }  }   class Q6  {  static void Main()  {  Test test1 = new Example1();  test1.A();   Test test2 = new Example2();  test2.A();  }  } }** |
| --- |

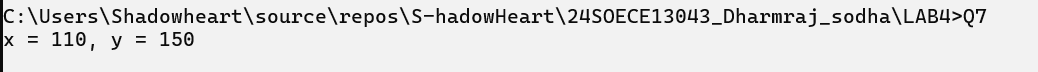
**Output:**

****

**7 . Refer given output and find out error code and correct it.**

| **using System;**  **namespace \_24SOECE13043\_Dharmraj\_sodha.LAB4**  **{**  **sealed class A**  **{**  **public int x;**  **public int y;**  **}**  **class SealedTest2**  **{**  **static void Main()**  **{**  **A sc = new A();**  **sc.x = 110;**  **sc.y = 150;**  **Console.WriteLine("x = {0}, y = {1}", sc.x, sc.y);**  **}**  **}**  **}** |
| --- |

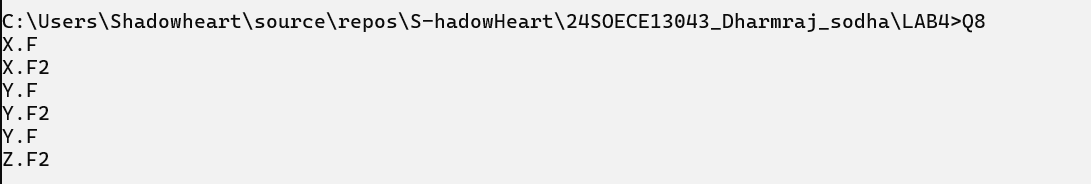
**Output:**

****

**8 : Find out error code and correct it. Print appropriate output as desired.**

| **using System;  namespace \_24SOECE13043\_Dharmraj\_sodha.LAB4 {  class X  {  public virtual void F() { Console.WriteLine("X.F"); }  public virtual void F2() { Console.WriteLine("X.F2"); }  }   class Y : X  {  public sealed override void F() { Console.WriteLine("Y.F"); }  public override void F2() { Console.WriteLine("Y.F2"); }  }   class Z : Y  {  // Cannot override F() because it is sealed in Y  public override void F2() { Console.WriteLine("Z.F2"); }  }   class SealedMethodTest  {  static void Main()  {  X Obj1 = new X();  Obj1.F();  Obj1.F2();   Y Obj2 = new Y();  Obj2.F();  Obj2.F2();   Z Obj3 = new Z();  Obj3.F(); // calls Y.F (sealed)  Obj3.F2(); // calls Z.F2  }  } }** |
| --- |

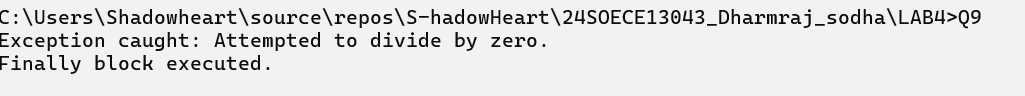
**Output :**

****

**9 : This program will throw an exception. Add try, catch and finally blocks to handle this exception**

| **using System;  namespace \_24SOECE13043\_Dharmraj\_sodha.LAB4 {  class MyClient  {  public static void Main()  {  int x = 0;  try  {  int div = 100 / x;  Console.WriteLine(div);  }  catch (DivideByZeroException e)  {  Console.WriteLine("Exception caught: " + e.Message);  }  finally  {  Console.WriteLine("Finally block executed.");  }  }  } }** |
| --- |

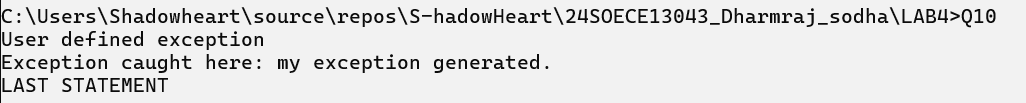
**Output :**

****

**10: Arrange the code to get desirable output**

| **using System;  namespace \_24SOECE13043\_Dharmraj\_sodha.LAB4 {  // User-defined exception  class MyException : Exception  {  public MyException(string str) : base(str)  {  Console.WriteLine("User defined exception");  }  }   class MyClient  {  public static void Main()  {  try  {  throw new MyException("my exception generated.");  }  catch (Exception e)  {  Console.WriteLine("Exception caught here: " + e.Message);  }  Console.WriteLine("LAST STATEMENT");  }  } }** |
| --- |

**Output:**

****