**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; class program

{

static int computeSalary(int basic, char grade)

{

double allowance; double hra, da, pf;

hra = 0.2 \* basic; da = 0.5 \* basic; pf = 0.11 \* basic;

if (grade == 'A')

{

allowance = 1700.0;

}

else if (grade == 'B')

{

}

else

{

}

allowance = 1500.0;

allowance = 1300.0;

double gross;

Console.WriteLine("HRA : {0}", hra);

Console.WriteLine("DA : {0}", da);

gross = Math.Round(basic + hra + da + allowance - pf); return (int)gross;

}

public static void Main(String[] args)

{

int basic = 20000; char grade = 'A';

Console.WriteLine("\nEmployee 1 : ");

Console.WriteLine("Total Pay : {0}", computeSalary(basic, grade)); basic = 8000;

grade = 'B'; Console.WriteLine("\nEmployee 2 : ");

Console.WriteLine("Total Pay : {0}", computeSalary(basic, grade)); basic = 9000;

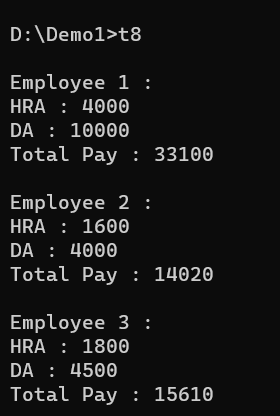
grade = 'A'; Console.WriteLine("\nEmployee 3 : ");

Console.WriteLine("Total Pay : {0}", computeSalary(basic, grade)); Console.ReadLine();

}

}

**OUTPUT:**



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

using System; class Shape

{

public double Width, 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(Style);

}

}

class Driver

{

public static void Main(string[] args)

{

Triangle t1 = new Triangle(); Triangle t2 = new Triangle(); t1.Width = 4.0;

t1.Height = 4.0;

t1.Style = "Triangle is isosceles"; t2.Width = 8.0;

t2.Height = 12.0;

t2.Style = "Triangle is 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();

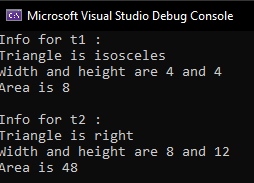
t2.ShowDim();

Console.WriteLine("Area is " + t2.Area());

}

}

**OUTPUT:**



**3.** **.Draw a real picture for single level inheritance. Perform following tasks.**

**Task 1: Create a class**

**Task 2: Add few data members as private, protected and public**

**Task 3: Add few methods as public to work on defined data members**

**Task 4: Create another applicable class which inherits members from above class Task 5: Add few data members as private, protected and public into second class**

**Task 6: Add few methods as public to work on defined data members into second class Task 7: Create a Demo class with main method.**

**Task 8: Create at least two objects of a second class defined in Task 4 into main method and call all methods using that object.**

**Task 9: Write comment for each important portion of code like data members’ declaration, methods, some important logic etc.**

**Task 10: Summarize above solution in your own few words to visualize the solution to the end user.**

**Code:**

using System; class A1

{

public Double width, height;

public void get\_data(double width, double height)

{

this.width = width; this.height = height;

}

}

class B1 : A1

{

public void area()

{

Console.WriteLine("Area of Rectangle is : {0}", (width \* height));

}

}

class program

{

public static void Main(string[] args)

{

B1 obj1 = new B1(); Console.Write("\nEnter width : ");

obj1.width = Convert.ToDouble(Console.ReadLine()); Console.Write("Enter height : ");

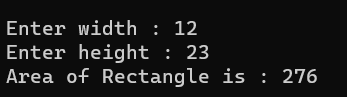
obj1.height = Convert.ToDouble(Console.ReadLine()); obj1.area();

Console.ReadLine();

}

}

**Output:**

****

**4.**

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

**Code**

using System;

namespace Tutorial4

{

class StaticVar

{

public static int num; public void counting()

{

num++;

}

public int getNum()

{

return num;

}

}

class StaticTester

{

public static void Main(string[] args)

{

StaticVar s = new StaticVar(); s.counting();

s.counting();

s.counting();

s.counting();

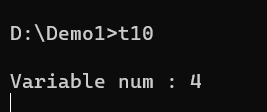
Console.WriteLine("\nVariable num : {0}", s.getNum()); Console.ReadLine();

}

}

}

Output:



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

using System;

namespace Tutorial4

{

class A

{

public A(int value)

{

Console.WriteLine("Base constructor A()");

}

}

class B : A

{

public B(int value) : base(value)

{

Console.WriteLine("Derived constructor B()");

}

}

class Program

{

public static void Main()

{

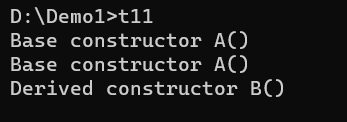
A a = new A(0); B b = new B(1);

}

}

}

Output:



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

**Code:**

using System; abstract class Test

{

public int a;

public abstract void A();

}

class Example1 : Test

{

public override void A()

{

Console.WriteLine("Example1.A"); base.a++;

}

}

class Example2 : Test

{

public override void A()

{

Console.WriteLine("Example2.A"); base.a--;

}

}

class Program

{

static void Main()

{

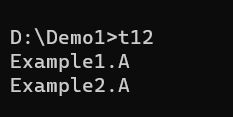
Test test1 = new Example1(); test1.A();

Test test2 = new Example2(); test2.A();

}

}

**Output:**



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

**Code:**

using System; class A

{

public int x; public int y;

}

sealed class B : A

{

public int z;

}

class SealedTest2

{

static void Main()

{

A sc = new A(); sc.x = 110;

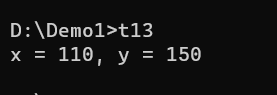
sc.y = 150;

Console.WriteLine("x = {0}, y = {1}", sc.x, sc.y);

}

}

Output:



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

**Code:**

using System; class X

{

public virtual void F() { Console.WriteLine("X.F"); } public virtual void F2() { Console.WriteLine("X.F2"); }

}

class Y : X

{

sealed public override void F() { Console.WriteLine("Y.F"); } public override void F2() { Console.WriteLine("Y.F2"); }

}

class Z : Y

{

//public override void F() { Console.WriteLine("Z.F"); } public override void F2() { Console.WriteLine("Z.F2"); }

}

class program

{

public static void Main()

{

* 1. Obj1 = new X(); Obj1.F();

Obj1.F2();

* 1. Obj2 = new Y(); Obj2.F();

Obj2.F2();

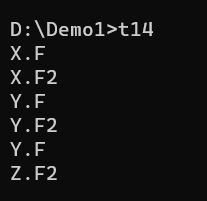
* 1. Obj3 = new Z(); Obj3.F();

Obj3.F2();

}

}

**Output:**

****

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

using System;

class program

{

public static void Main()

{

int x = 0;

int div = 0;

try

{

div = 100 / x;

Console.WriteLine("This line will be not executed.");

}

catch (DivideByZeroException)

{

Console.WriteLine("Exception occured.");

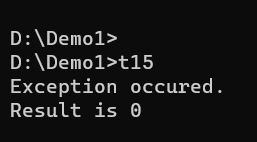
}

Console.WriteLine($"Result is {div}");

}

}

**Output:**

****

10.Arrange the code to get desirable output

using System;

class MyException : Exception

{

public MyException(string str)

{

Console.WriteLine("User defined exception");

}

}

class MyClient

{

public static void Main()

{

try

{

}

throw new MyException("AB");

catch (Exception e)

{

Console.WriteLine("Exception caught here " + e.ToString());

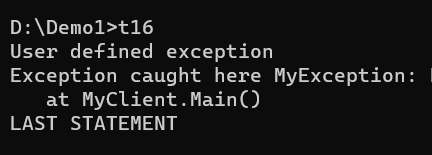
}

Console.WriteLine("LAST STATEMENT");

}

}

**Output:**

****