.NET Assignment -1

Prajwal_77

Q1. Create a Class Employee with the following specifications
Properties
----
string Name -> no blank names should be allowed
int EmpNo -> must be greater than 0 decimal
Basic -> must be between some range short
DeptNo -> must be > 0

Methods
----
decimal GetNetSalary() -> returns calculated net salary (Use any formula to get net salary based on Basic salary)

Create constructors to accept initial values for Employee obj eg
Employee o1 = new Employee(1,"Amol",123465, 10);
Employee o2 = new Employee(1,"Amol",123465);
Employee o3 = new Employee(1,"Amol");
Employee o4 = new Employee(1);
Employee o5 = new Employee();

```
namespace Assignment1
{
    internal class Program
    {
        static void Main(string[] args)
        {
            // calling static property
            Employee.Name = "Prajwal";
            Console.WriteLine(Employee.Name);

            Employee e1 = new Employee();

            //calling non-static properties
            e1.EmpNo = 100;
            Console.WriteLine(e1.EmpNo);

            e1.Basic = 7500;
            Console.WriteLine(e1.Basic);

            e1.DeptNo = 10;
            Console.WriteLine(e1.DeptNo);

            //Calling Method
            Console.WriteLine(e1.GetNetSalary());

            //Calling Constructor
            Employee o1 = new Employee(1, "Amol", 123465, 10);
            Employee o2 = new Employee(1, "Amol", 123465);
            Employee o3 = new Employee(1, "Amol");
            Employee o4 = new Employee(1);
            Employee o5 = new Employee();
            o1.display();
            o2.display();
            o3.display();
            o4.display();
            o5.display();
        }
    }

    public class Employee
    {
        //fields
        private static string name;
        private int empNo;
        private decimal basic;
        private short deptno;
        private int id;
        private string empname;
        private int mob;
        private int no;
        private decimal netsalary;

        //Properties
        public static string Name
        {
            set
            {
                if (string.IsNullOrWhiteSpace(value))
                    Console.WriteLine("Empty String Not allowed");
                else
```

```
                name = value;
            }

            get
            {
                return name;
            }
        }

        //Properties
        public int EmpNo
        {
            set
            {
                if (value > 0)
                    empNo = value;
                else
                    Console.WriteLine("Employee no must be greater than  0");
            }

            get { return empNo; }
        }

        //Properties
        public decimal Basic
        {
            set
            {
                if ((value > 5000) && (value < 10000))
                    basic = value;
                else
                    Console.WriteLine("Value not in the given range");
            }

            get { return basic; }
        }

        //Properties
        public short DeptNo
        {
            set
            {
                if (value > 0)
                    deptno = value;
                else
                    Console.WriteLine("Department no must be > 0");
            }

            get { return deptno; }
        }

        //Method
        public decimal GetNetSalary()
        {
            netsalary = basic * 12 * (decimal)0.5;
            return netsalary;
        }

        //Create constructors to accept initial values for Employee obj
        public Employee(int id = 0, string empname = "", int mob = 0, int no =
0)
        {
            this.id = id;
```

```
            this.empname = empname;
            this.mob = mob;
            this.no = no;
        }

        public void display()
        {
            Console.WriteLine(id + " " + empname + " " + mob + " " + no);

        }

    }

}
```