

## Example (7) Class Relationships

1. Write a C# windows Form Application code to create a Class Named *Student* that contains the following :
  - ✓ **Fields :**
    - name of type string
    - tch is an object of class **Teacher**
  - ✓ **Methods:**
    - Default and parametrized constructor .
    - Set and get Methods .
    - Display details method.

### Solution :

```
1. using System;
2. using System.Collections.Generic;
3. using System.Linq;
4. using System.Text;
5.
6. namespace Teacher_Student
7. {
8.     class Teacher
9.     {
10.         private string name;
11.         public Teacher()
12.         {
13.             name = "Dina Riadh";
14.         }
15.     }
16.     public Teacher(string n)
17.     {
18.         bool c = checkName(n);
19.         if (c)
20.         {
21.             name = n;
22.         }
23.         else
24.             name = "Dina Riadh";
25.     }
26.     public void setName(string name){this.name =name ;}
27.     public string getName() { return name; }
28.     private bool checkName(string n)
29.     {
30.         bool c = false;
31.         for (int i = 0; i < n.Length; i++)
32.         {
33.             if(char.IsLetter(n[i]))
```

```
34.         {
35.             c=true ;
36.         }
37.         else
38.         {
39.             c=false ;
40.             break ;
41.         }
42.
43.     }
44.     return c;
45. }
46. public string Display() { return "Teacher Name : " + getName(); }
47. }
48. }
```

```
1. using System;
2. using System.Collections.Generic;
3. using System.Linq;
4. using System.Text;
5.
6. namespace Teacher_Student
7. {
8.     class Student
9.     {
10.         private string name;
11.         private Teacher tch;
12.
13.         public Student()
14.         {
15.             name = "Jullian Moore";
16.             tch = new Teacher();
17.         }
18.
19.         public Student(string name, Teacher tch)
20.         {
21.             this.name = name;
22.             this.tch = tch;
23.         }
24.
25.         public string getStName() { return name; }
26.         public void setName(string name) { this.name = name; }
27.         public void setTch(string n) { tch.setName(n); }
28.         public string print()
29.         {
30.             string r = "";
31.             r += "Student Name : " + getStName() + Environment.NewLine;
32.             r += tch.Display();
33.             return r;
34.         }
35.     }
36. }
```

```

1. private void button1_Click(object sender, EventArgs e)
2.     {
3.         Teacher tch = new Teacher("Dina Riadh");
4.         Student st = new Student("Christian Bale ", tch);
5.         textBox1.Text = st.print();
6.     }

```

## Example (8) Class Aggregation

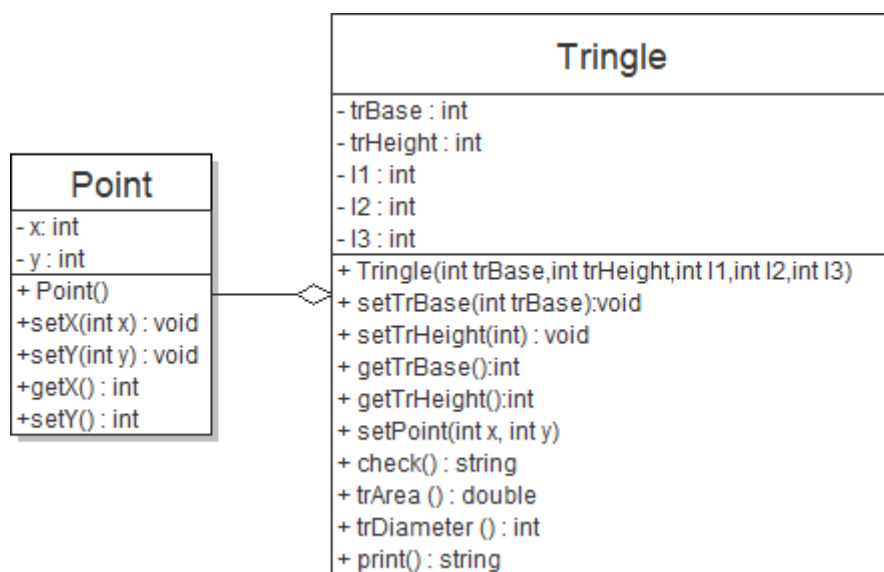
2. Write a C# windows Form Application code to create a Class named *Tringle* that contains the following :

✓ **Fields :**

- **trBase** and **trHeight** of type int
- **I1, I2, I3** of type int
- **center** is an object of class **Point** (Contain x and y fields )

✓ **Methods:**

- Default and parametrized constructor .
- Set and get Methods .
- check method to check weather the triangle is an equal legs triangle or not
- trArea () and trDiameter()
- Display details method.



**Solution :**

```
49. using System;
50. using System.Collections.Generic;
51. using System.Linq;
52. using System.Text;
53.
54. namespace TriangleExam
55. {
56.     class Point
57.     {
58.         private int x, y;
59.         public Point()
60.         {
61.             x = 5;
62.             y = 8;
63.         }
64.         public Point(int x, int y)
65.         {
66.             this.x = x;
67.             this.y = y;
68.         }
69.         public int getX()
70.         {
71.             return x;
72.         }
73.         public void setX(int x)
74.         {
75.             this.x = x;
76.         }
77.         public int getY()
78.         {
79.             return y;
80.         }
81.         public void setY(int y)
82.         {
83.             this.y = y;
84.         }
85.     }
86. }
87. }
```

```
1. using System;
2. using System.Collections.Generic;
3. using System.Linq;
4. using System.Text;
5.
6. namespace TriangleExam
7. {
8.     class Triangle
9.     {
```

```
10.     private int trBase, trHeight, l1, l2, l3;
11.     private Point center;
12.     public Triangle(int trBase,int trHeight,int l1,int l2,int l3,Point center)
13.     {
14.
15.         this.trBase = trBase;
16.         this.trHeight = trHeight;
17.         this.l1 = l1;
18.         this.l2 = l2;
19.         this.l3 = l3;
20.         this.center = center;
21.
22.     }
23.
24.     public double trArea()
25.     {
26.         return 0.5 * trBase * trHeight;
27.     }
28.
29.     public int trDiameter()
30.     {
31.         return l1 + l2 + l3;
32.     }
33.
34.     public void setBase(int trBase)
35.     {
36.         this.trBase = trBase;
37.     }
38.
39.     public void setHeight(int trHeight)
40.     {
41.         this.trHeight = trHeight;
42.     }
43.
44.     private string checkL()
45.     {
46.         string eq="";
47.         if ((l1 == l2) && (l1 == l3) && (l2 == l3))
48.             eq = "Its an equal Lengle leg Triangle";
49.         else
50.             eq = "Its not an equal Lengle leg Triangle";
51.         return eq;
52.     }
53.
54.
55.     public void setPoint(int x, int y)
56.     {
57.         center.setX(x);
58.         center.setY(y);
59.     }
60.
61.
62.     public string print()
63.     {
```

```

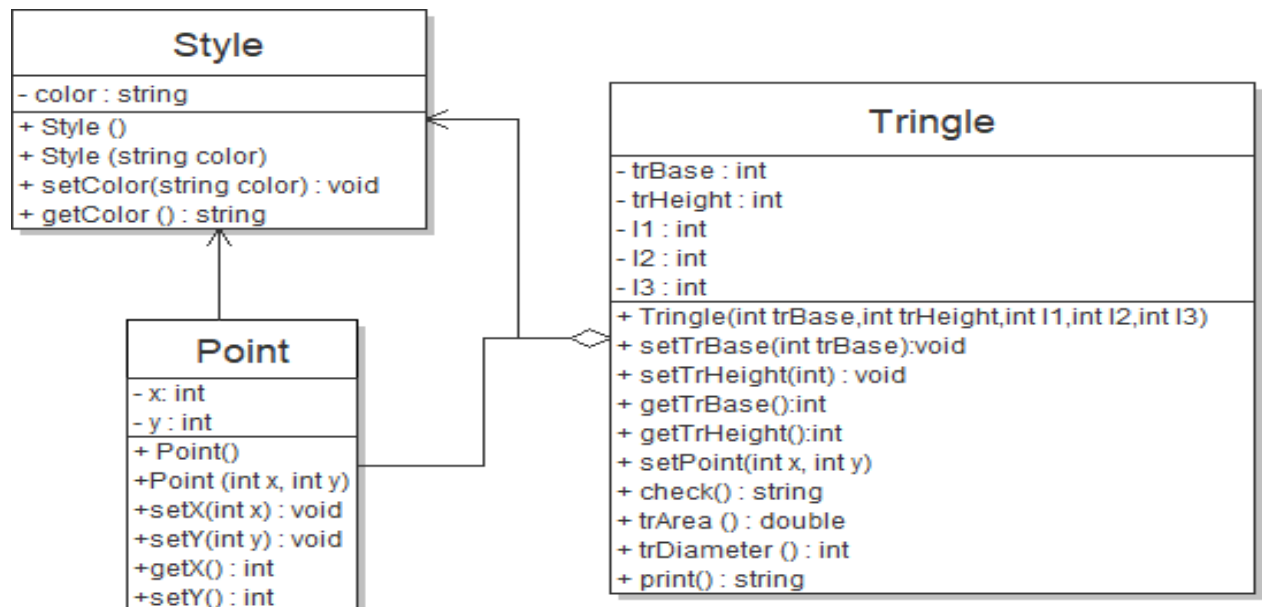
64.
65.     string e = checkL();
66.     string r="";
67.     r += " Tringle Area= " + trArea()+ Environment.NewLine;
68.     r += " Tringle Diameter = " + trDiameter();
69.     r += Environment.NewLine +e+Environment.NewLine ;
70.     r += " X center Coordinate = " ;
71.     r+= "Y center Coordinate =" +center .getY ();
72.     return r;
73. }
74. }
75. }
76.
77.
    
```

```

1. private void button1_Click(object sender, EventArgs e)
2. {
3.     Point p = new Point();
4.     Triangle tr = new Triangle(12, 4, 5, 3, 3,p);
5.     textBox1.Text = tr.print();
6. }
    
```

### Exercise:

#### 1. Convert the following UML diagram into C# code



2. Try string for x and y coordinate and add method check to make sure to use digit inputs.
3. Make two objects of Triangle and compare their area if Triangle 1 has a bigger area make a note for that else make a note that Triangle 2 is bigger.
4. Make five objects of Triangle .

### Example (9)

5. Write a C# windows Form Application code to create a Class named *BankAccount* that contains the following :

✓ **Fields :**

- **accNumber** and **balance** of type int
- **owner** is an object of class **Owner** (Contain name and address as fields )

✓ **Methods:**

- Default and parametrized constructor for both classes .
- Set and get Methods for both classes .
- check method in **Owner** class to check name and address is it contain letters.
- withdraw(int amt) and deposite(int amt) in *BankAccount*
- Display details method.

```
1. using System;
2. using System.Collections.Generic;
3. using System.Linq;
4. using System.Text;
5.
6. namespace BankAccount_and_Owner
7. {
8.     class Owner
9.     {
10.         private string name, address;
11.         public Owner()
12.         {
13.             name = "Dina Riadh";
14.             address = "Baghdad";
15.         }
16.
17.         public Owner(string n, string a)
18.         {
19.             if (check(n))
20.                 name = n;
```

```
21.     else
22.         name = " Colin Firth";
23.     if (check(a ))
24.         address = a;
25.     else
26.         address = " Basra";
27.     }
28.
29.     public string getName() { return name; }
30.     public void setName(string name) { this.name = name; }
31.     public string getAddress() { return address; }
32.     public void setAddress(string address) { this.address = address ; }
33.     private bool check(string na)
34.     {
35.         bool c = false;
36.         for (int i = 0; i < na.Length; i++)
37.             if (char.IsLetter(na[i]))
38.                 c = true;
39.             else
40.                 {
41.                     c = false;
42.                     break;
43.                 }
44.         return c;
45.     }
46. }
47. }
```

```
1. using System;
2. using System.Collections.Generic;
3. using System.Linq;
4. using System.Text;
5. using System.Windows.Forms;
6.
7. namespace BankAccount_and_Owner
8. {
9.     class BankAccount
10.    {
11.
12.        private Owner owr;
13.        private double balance;
14.        private int accNumber;
15.
16.        public BankAccount(Owner owr, int accNumber ,double balance)
17.        {
18.
19.            this.owr = owr;
20.            this.accNumber = accNumber;
21.            this.balance = balance;
22.        }
23.
24.
25.
```



```
26. public double getBalance()
27. {
28.     return balance;
29. }
30. public void setBalance(double b)
31. {
32.     balance = b;
33. }
34.
35. public int getAccNumber()
36. {
37.     return accNumber;
38. }
39.
40. public void setAccNumber(int ac)
41. {
42.     accNumber = ac;
43. }
44.
45. public void Withdraw(double amount)
46. {
47.     if (amount <= balance)
48.         balance -= amount;
49.     else
50.         MessageBox.Show("Not Enough Balance");
51. }
52. }
53.
54. public void Deposit(double amount)
55. {
56.     balance += amount;
57. }
58.
59. public void AddInterests()
60. {
61.     balance += balance * 0.04;
62. }
63.
64. public string print()
65. {
66.     string r = "";
67.     r += " Owner Information : " + Environment.NewLine ;
68.     r += " Owner Name : " + owr.getName()+Environment.NewLine;
69.     r += " Owner Address : " + owr.getAddress() + Environment.NewLine;
70.     r += " Account Number = " + getAccNumber() + nvironment.NewLine;
71.     r+= " Balance = "+balance + " Dinars";
72.     return r;
73. }
74.
75. }
76. }
```

1. private void button1\_Click(object sender, EventArgs e)

```
2.      {  
3.      Owner o = new Owner("Dina", "Baghdad3");  
4.  
5.      BankAccount ba = new BankAccount(o,100, 1000);  
6.      ba.AddInterests();  
7.      ba.Deposit(1000);  
8.      ba.Withdraw(2000);  
9.      textBox1.Text = ba.print();  
10.     }
```