

Solución:

Clase Program

```
Program.cs
Visitor
Visitor.Program
Main(string[] args)

11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40

Referencias
static void Main(string[] args)
{
    int Healt_marine = 0;
    int Healt_marauder = 0;
    Ivisitor _Bullet;

    Console.Clear();
    ConsoleKeyInfo op;

    do
    {
        //Limpiar la pantalla
        Console.WriteLine("");
        Console.WriteLine("Menú - Patrón Visitor");
        Console.WriteLine("");
        Console.WriteLine("");
        Console.WriteLine("Que desea realizar.");
        Console.ForegroundColor = ConsoleColor.Red;
        Console.WriteLine("[A] - Ataque Marine");
        Console.WriteLine("[E] - Ataque Marauder ");
        Console.WriteLine("[Esc] Salir");
        Console.ForegroundColor = ConsoleColor.White;
        Console.WriteLine("Seleccione opcion...");
        op = Console.ReadKey(true);

        switch (op.Key)
        {
```

```
Program.cs
Visitor
Visitor.Program
Main(string[] args)

39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69

switch (op.Key)
{
    case ConsoleKey.A:
        Console.WriteLine("Ud seleccionó la opción [Ataque Marine]");
        Console.WriteLine("Presione una tecla para continuar...");
        Console.ReadKey();

        _Bullet = new TankBullet();
        IlightUnit _marine = new Marine();
        if(Healt_marine != 0) { _marine.Health = Healt_marine; }
        _marine.Accept(_Bullet);
        Console.WriteLine("El ataque fue de " + variables.Ataque_Marine + " puntos negativos | Marine queda con un total de " + _marine.Health + " puntos");
        Healt_marine = _marine.Health;

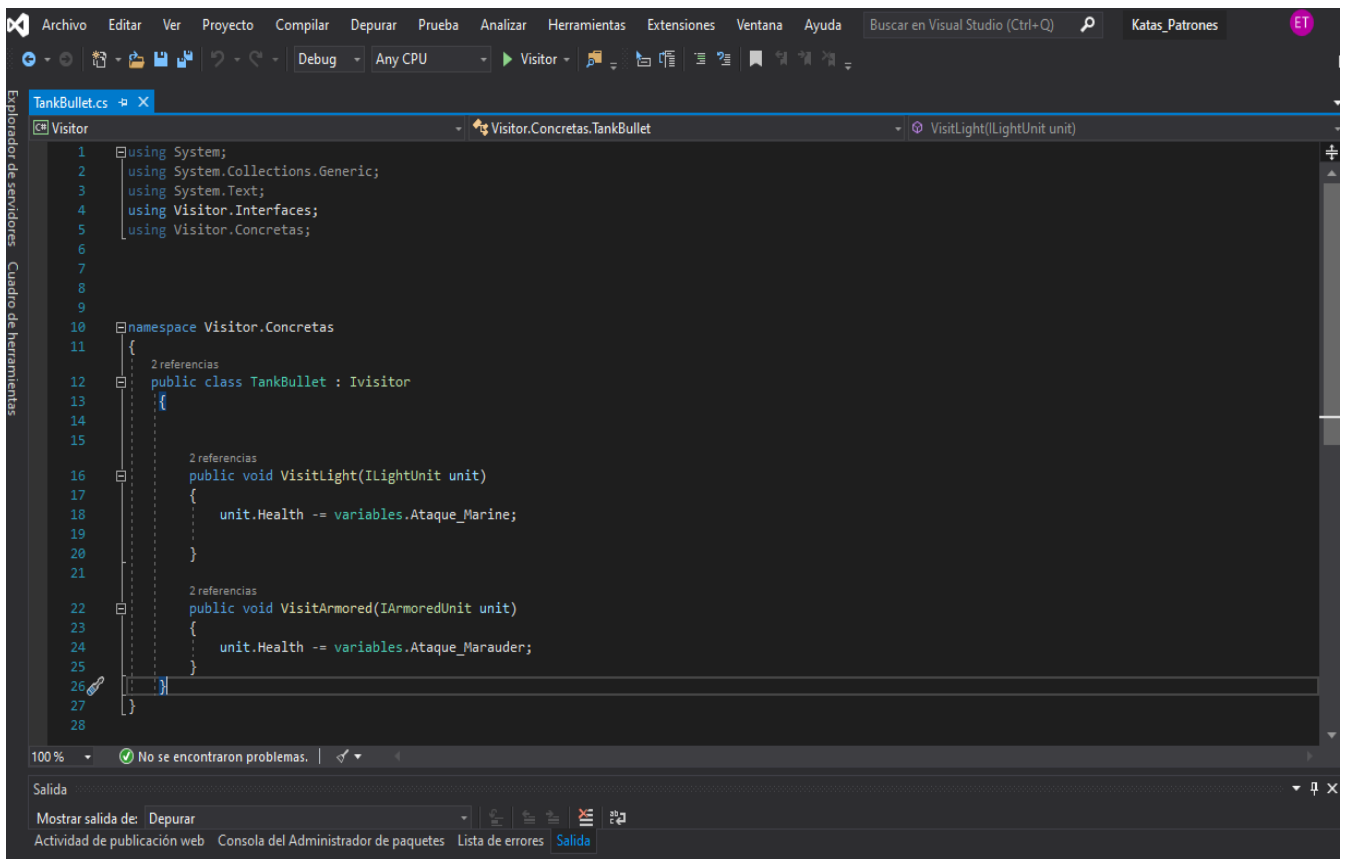
        break;

    case ConsoleKey.E:
        Console.WriteLine("Ud seleccionó la opción [Ataque Marauder]");
        Console.WriteLine("Presione una tecla para continuar...");
        Console.ReadKey();

        _Bullet = new TankBullet();
        IArmoredUnit _marauder = new Marauder();
        if(Healt_marauder != 0) { _marauder.Health = Healt_marauder; }
        _marauder.Accept(_Bullet);
        Console.WriteLine("El ataque fue de " + variables.Ataque_Marauder + " puntos negativos | Marauder queda con un total de " + _marauder.Health + " puntos");
        Healt_marauder = _marauder.Health;

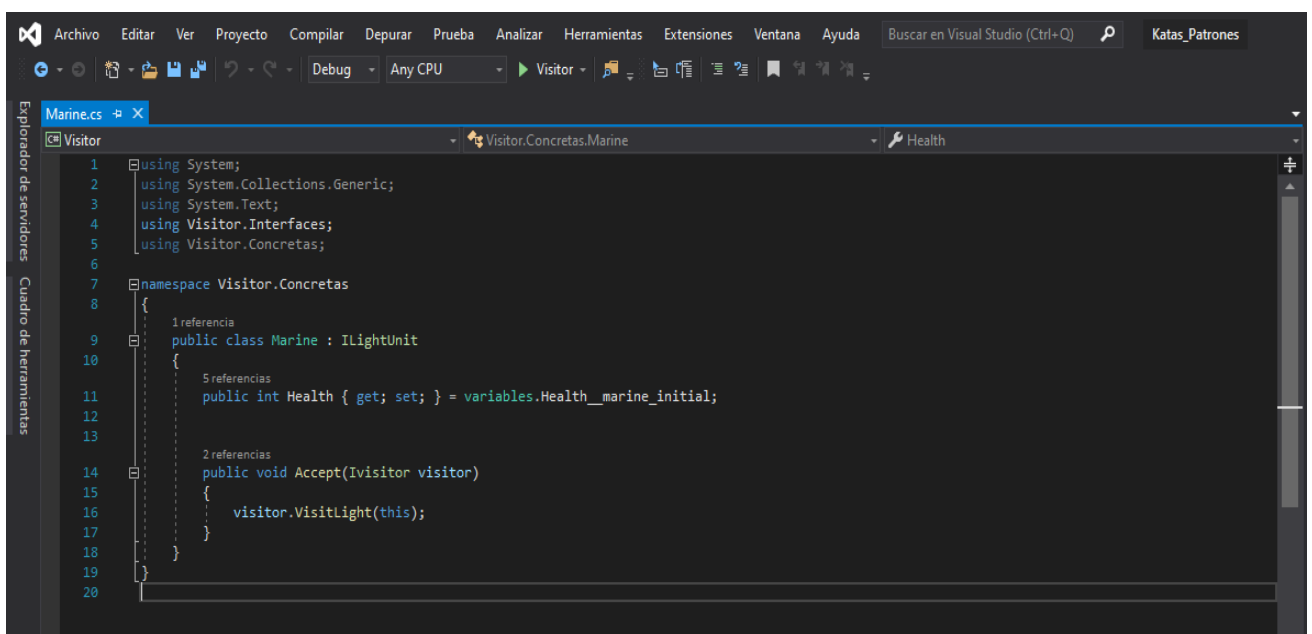
        break;
```

Clase TankBullet



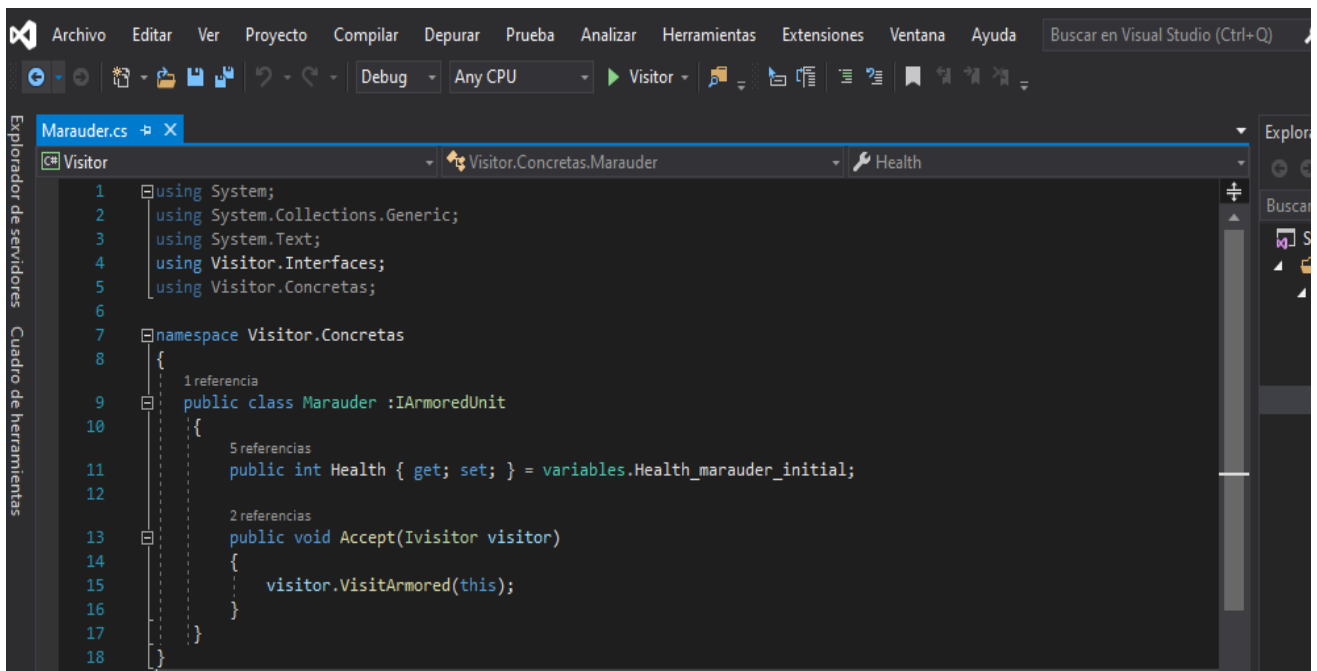
```
1 using System;
2 using System.Collections.Generic;
3 using System.Text;
4 using Visitor.Interfaces;
5 using Visitor.Concretas;
6
7
8
9
10 namespace Visitor.Concretas
11 {
12     2 referencias
13     public class TankBullet : Ivisitor
14     {
15
16         2 referencias
17         public void VisitLight(ILightUnit unit)
18         {
19             unit.Health -= variables.Ataque_Marine;
20         }
21
22         2 referencias
23         public void VisitArmored(IArmoredUnit unit)
24         {
25             unit.Health -= variables.Ataque_Marauder;
26         }
27     }
28 }
```

Clase Marine



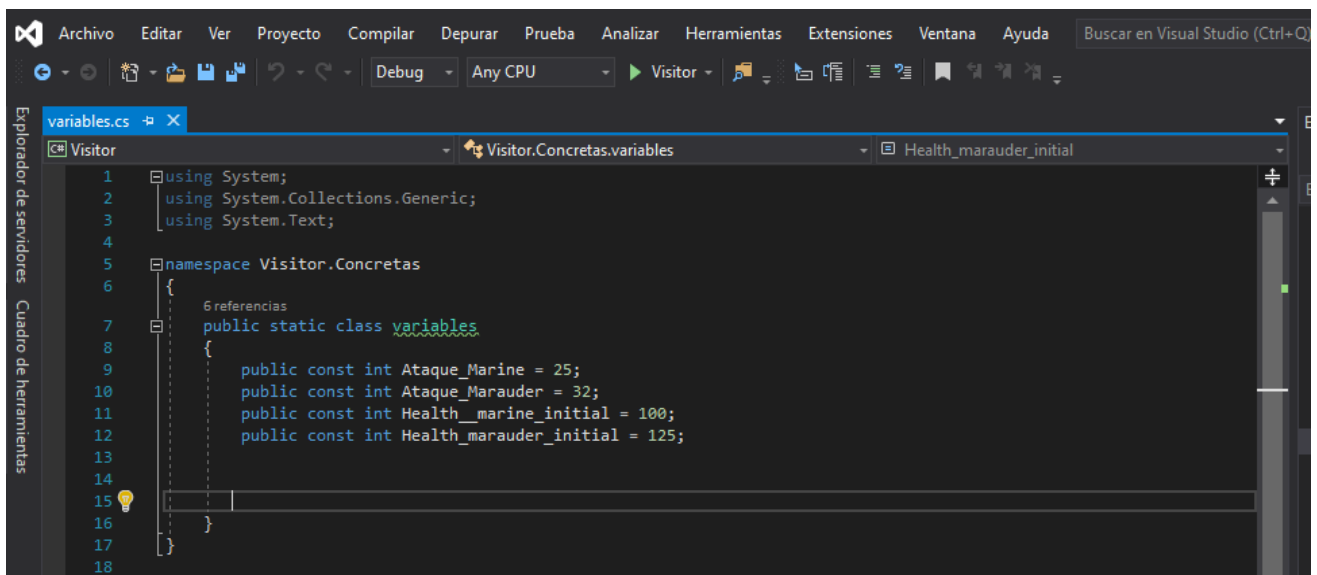
```
1 using System;
2 using System.Collections.Generic;
3 using System.Text;
4 using Visitor.Interfaces;
5 using Visitor.Concretas;
6
7 namespace Visitor.Concretas
8 {
9     1 referencia
10     public class Marine : ILightUnit
11     {
12         5 referencias
13         public int Health { get; set; } = variables.Health_marine_initial;
14
15         2 referencias
16         public void Accept(Ivisitor visitor)
17         {
18             visitor.VisitLight(this);
19         }
20     }
21 }
```

Clase Marauder



```
1 using System;
2 using System.Collections.Generic;
3 using System.Text;
4 using Visitor.Interfaces;
5 using Visitor.Concretas;
6
7 namespace Visitor.Concretas
8 {
9     1 referencia
10     public class Marauder : IArmoredUnit
11     {
12         5 referencias
13         public int Health { get; set; } = variables.Health_marauder_initial;
14
15         2 referencias
16         public void Accept(Ivisitor visitor)
17         {
18             visitor.VisitArmored(this);
19         }
20     }
21 }
```

Clase Variables



```
1 using System;
2 using System.Collections.Generic;
3 using System.Text;
4
5 namespace Visitor.Concretas
6 {
7     6 referencias
8     public static class variables
9     {
10         public const int Ataque_Marine = 25;
11         public const int Ataque_Marauder = 32;
12         public const int Health_marine_initial = 100;
13         public const int Health_marauder_initial = 125;
14
15     }
16 }
17
18
```

Ejecución de pruebas

7kyu

PatternCraft - Visitor

☆ 45 🌟 20 🔄 82% of 111 🕒 233 of 672 👤 brunolm ⓘ 4 Issues Reported

Instructions

Output

Time: 2359ms Passed: 2 Failed: 0

Test Results:

KataTestf

VisitArmored

✔ Test Passed

Completed in 24.7390ms

VisitLight

✔ Test Passed

Completed in 0.7440ms

Completed in 39.6950ms

You have passed all of the tests! :)

C#

7.3

VIM EMACS

Solution:

```
39     visitor.VisitArmored(this);  
40   }  
41 }  
42  
43 public class TankBullet : IVisitor  
44 {  
45  
46  
47  
48     public void VisitLight(ILightUnit unit)  
49     {  
50         unit.Health -= 21;  
51     }  
52 }
```

Sample Tests:

```
21     IVisitor bullet = new TankBullet();  
22     IArmoredUnit armored = new Marauder();  
23  
24     armored.Accept(bullet);  
25  
26     Assert.AreEqual(125 - 32, armored.Health);  
27 }  
28 }
```

SKIP

VIEW SOLUTIONS

DISCUSS (12)

RESET

TEST

ATTEMPT

Escribe aquí para buscar

10:40 AM
23/03/2020