

## Ejercicios de Programación.

### Ejercicio9

Realizar un programa utilizando POO que contenga las siguientes clases y verifique su funcionamiento.

1. Clase Familia
  - Atributos  
Nombre integrante
  - Métodos  
Constructor sobrecargado  
Getters y Setters  
respirar  
comer
2. Clase Perro
  - Atributos  
Nombre integrante  
Apodo
  - Métodos  
Constructor por parámetro  
Getters y Setters  
respirar  
comer  
ladrar
3. Clase Gato
  - Atributos  
Nombre integrante
  - Métodos  
Constructor por defecto  
Getters y Setters  
respirar  
comer  
maullar
4. Clase Humano
  - Atributos  
Nombre integrante
  - Métodos  
Constructor por parámetro  
Getters y Setters  
respirar  
comer  
hablar

```

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace ClasesEjercicio8
8  {
9      internal class Program
10     {
11         static void Main(string[] args)
12         {
13             //Creamos instancias
14             Familia Nuevo = new Familia("ninguno");//Llama al constructor por defecto
15             Gato gatoSimpson = new Gato();//Llama al constructor por defecto
16
17             Perro perroSimpson = new Perro("Ayudante de Santa");//Llaman al segundo constructor
18             Humano hijoSimpson = new Humano("Bart");
19             Humano hijaSimpson = new Humano("Lisa");
20
21             //Utilizo las instancias
22             perroSimpson.ladrar();
23             gatoSimpson.mauallar();
24             gatoSimpson.respirar();
25             Nuevo.respirar();
26             Console.WriteLine(Nuevo.getInfo());
27             hijaSimpson.hablar();
28
29             Console.WriteLine("");
30             Console.WriteLine(gatoSimpson.getInfo());
31             gatoSimpson.setInfo("Bola de Nieve");
32             Console.WriteLine(gatoSimpson.getInfo());
33             Console.WriteLine("");
34
35             Console.WriteLine(perroSimpson.getInfo());
36             perroSimpson.setApodoPerro("huesos");
37             Console.WriteLine(perroSimpson.getApodoPerro());
38
39             Console.Read();
40         }
41     }
42
43     class Familia
44     {
45         //Campos de clases o variables
46         private string nombreIntegrante;
47
48         //Constructores
49         public Familia()
50         {
51             this.nombreIntegrante = "nada";
52         }
53         public Familia(string nombreIntegrante)
54         {
55             this.nombreIntegrante = nombreIntegrante;
56         }
57
58         //Getters y Setters
59         public string getInfo()
60         {
61             return(nombreIntegrante);
62         }
63         public void setInfo( string nombreIntegrante)
64         {
65             this.nombreIntegrante = nombreIntegrante;
66         }
67
68         //Metodos de la clase
69         public void respirar()
70         {
71             Console.WriteLine("Puedo respirar");
72         }
73         public void comer()
74         {
75             Console.WriteLine("Puedo comer");
76         }
77     }
78

```

```

79 class Gato
80 {
81     //Campos de clases o variables
82     private string nombreIntegrante;
83
84     //Constructores
85     public Gato()
86     {
87         this.nombreIntegrante = "nada";
88     }
89     public Gato(string nombreIntegrante)
90     {
91         this.nombreIntegrante = nombreIntegrante;
92     }
93
94     //Getters y Setters
95     public string getInfo()
96     {
97         return(nombreIntegrante);
98     }
99     public void setInfo( string nombreIntegrante)
100    {
101        this.nombreIntegrante = nombreIntegrante;
102    }
103
104    //Metodos de la clase
105    public void respirar()
106    {
107        Console.WriteLine("Puedo respirar");
108    }
109    public void comer()
110    {
111        Console.WriteLine("Puedo comer");
112    }
113    public void maullar()
114    {
115        Console.WriteLine("miau");
116    }
117 }
118

```

```
119 class Perro
120 {
121     //Campos de clases o variables
122     private string nombreIntegrante;
123     private string apodo;
124
125     //Constructores
126     public Perro()
127     {
128         this.nombreIntegrante = "nada";
129     }
130     public Perro(string nombreIntegrante)
131     {
132         this.nombreIntegrante = nombreIntegrante;
133     }
134
135     //Getters y Setters
136     public string getInfo()
137     {
138         return(nombreIntegrante);
139     }
140     public void setInfo( string nombreIntegrante)
141     {
142         this.nombreIntegrante = nombreIntegrante;
143     }
144     public void setApodoPerro(string apodo)
145     {
146         this.apodo = apodo;
147     }
148     public string getApodoPerro()
149     {
150         return (this.apodo);
151     }
152
153     //Metodos de la clase
154     public void respirar()
155     {
156         Console.WriteLine("Puedo respirar");
157     }
158     public void comer()
159     {
160         Console.WriteLine("Puedo comer");
161     }
162     public void ladrar()
163     {
164         Console.WriteLine("gua");
165     }
166 }
167
```

```

168 class Humano
169 {
170     //Campos de clases o variables
171     private string nombreIntegrante;
172
173     //Constructores
174     public Humano()
175     {
176         this.nombreIntegrante = "nada";
177     }
178     public Humano(string nombreIntegrante)
179     {
180         this.nombreIntegrante = nombreIntegrante;
181     }
182
183     //Getters y Setters
184     public string getInfo()
185     {
186         return(nombreIntegrante);
187     }
188     public void setInfo( string nombreIntegrante)
189     {
190         this.nombreIntegrante = nombreIntegrante;
191     }
192
193     //Metodos de la clase
194     public void respirar()
195     {
196         Console.WriteLine("Puedo respirar");
197     }
198     public void comer()
199     {
200         Console.WriteLine("Puedo comer");
201     }
202     public void hablar()
203     {
204         Console.WriteLine("soy una persona");
205     }
206 }
207
208

```