

Relación de Ejercicios de Clases y Objetos (6): Clínica Veterinaria, Animales

En esta relación, implementaremos la “base de datos” de una clínica veterinaria. Se compondrá principalmente de una lista de animales. Como los animales tendrán diferentes características dependiendo de su especie, crearemos varias clases para las diferentes categorías que heredarán de la clase principal **Animal**.

Clases y Enumerados que nos hacen falta:

- **Animal** (clase abstracta)
 - Atributos (todos *protected*, para poder acceder desde los hijos):
 - **nombre**, de tipo *string*
 - **fechaNacimiento**, de tipo *DateTime*
 - **peso**, de tipo *double*
 - **comentarios**, de tipo *string*
 - Propiedades (todas *public* para poder acceder desde fuera):
 - Cuatro propiedades para acceder a los cuatro atributos, de las cuales sólo el **Peso** y los **Comentarios** serán de lectura y escritura.
 - Método abstracto:
 - **ToString()**, para obligar a que las clases que heredan de **Animal** implementen obligatoriamente el **ToString()**. La definición es:
`public override abstract string ToString();`
- **Perro** (hereda de **Animal**)
 - Atributos adicionales:
 - **raza**, del tipo enumerado **RazaPerro** que podrá tener los valores: PastorAleman, Husky, FoxTerrier, Dalmata, SanBernardo
 - **microchip**, de tipo *string*
 - Propiedades de sólo lectura para los dos atributos adicionales.
 - Constructor al que le pasas los datos: **nombre**, **raza**, **fechaNacimiento**, **peso**, **microchip** (los comentarios ya se añaden luego).
 - Método **ToString()** que sobrescribe al abstracto que servirá para mostrar por pantalla una fichita con todos los atributos de perro (incluyendo los de animal).
Ej.:
Ficha de Perro
Nombre: Luna
Raza: Pastor alemán
Fecha de Nacimiento: 12/02/2003
Peso: 12,4 kg
Microchip: 38479382749DF39
Comentarios: Está pachucho.

- **Gato** (hereda de Animal)
 - Atributos adicionales:
 - **raza**, del tipo enumerado **RazaGato** que podrá tener los valores: Comun, Siames, Persa, Angora, ScottishFold.
 - **Microchip**, de tipo string
 - Propiedades, constructor y método igual que Perro (adaptados a los atributos de Gato).
- **Pajaro** (hereda de Animal)
 - Atributos adicionales:
 - **especie**, del tipo enumerado **EspeciePajaro** que podrá tener los valores: Canario, Periquito, Agapornis.
 - **cantor**, de tipo *bool* que nos dirá si da mucho la lata por las mañanas o no.
 - Propiedades, constructor y método igual que Perro (adaptados a los atributos de Pajaro).
- **Reptil** (hereda de Animal)
 - Atributos adicionales:
 - **especie**, del tipo enumerado **EspecieReptil** que podrá tener los valores: Tortuga, Iguana, DragonDeComodo.
 - **venenoso**, que será de tipo *bool* y nos dirá lo divertido que es que te pique un bicho de éstos.
 - Propiedades, constructor y método igual que Perro (adaptados a los atributos de Reptil).
- **ClinicaVeterinaria**
 - Atributos:
 - **listaAnimales** será una *List* de Animal en la que meteremos todos los animales, independientemente de que sean Perros, Gatos, Pajaros o Reptiles (funciona).
 - Constructores:
 - Uno básico para inicializar la lista.
 - Métodos:
 - **InsertaAnimal** al que le pasamos un Animal y lo añade a la lista.
 - **BuscaAnimal** al que le pasamos el nombre del bicho y nos devuelve el Animal en cuestión.
 - **ModificaComentarioAnimal** al que le pasamos el nombre del animal y el nuevo comentario y nos modifica el comentario en la ficha del animal que está en la lista.
 - **ToString()** sobreescrito, como siempre, que nos muestra la lista de todas las fichas de los animales.

Hacer un menú que te permita añadir animales de los diferentes tipos y nos permita modificar los comentarios y mostrar por pantalla el tema.