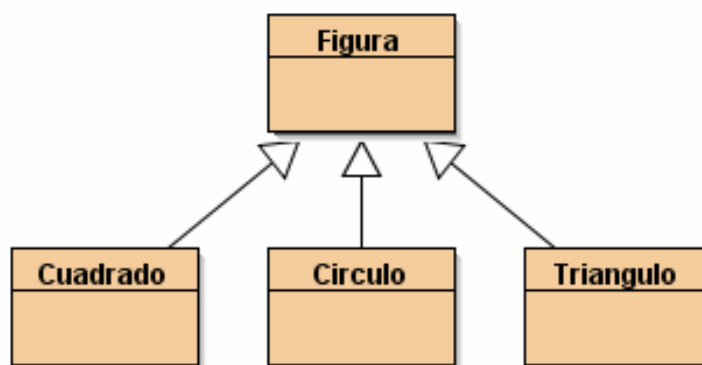


UD8 POO (II) – Ejercicios D

Clases Abastractas

Ejercicio D1. Figuras

Partiendo de tu código del ejercicio B1 (figuras con herencia) modifica el código de modo que **Figura sea una clase abstracta y su método area() sea abstracto también**. Así tiene más sentido ya que no será posible instanciar objetos Figura (solo podrán existir figuras de algún tipo concreto) y tampoco será necesario programar el método Figura.area(), bastará con declararlo.

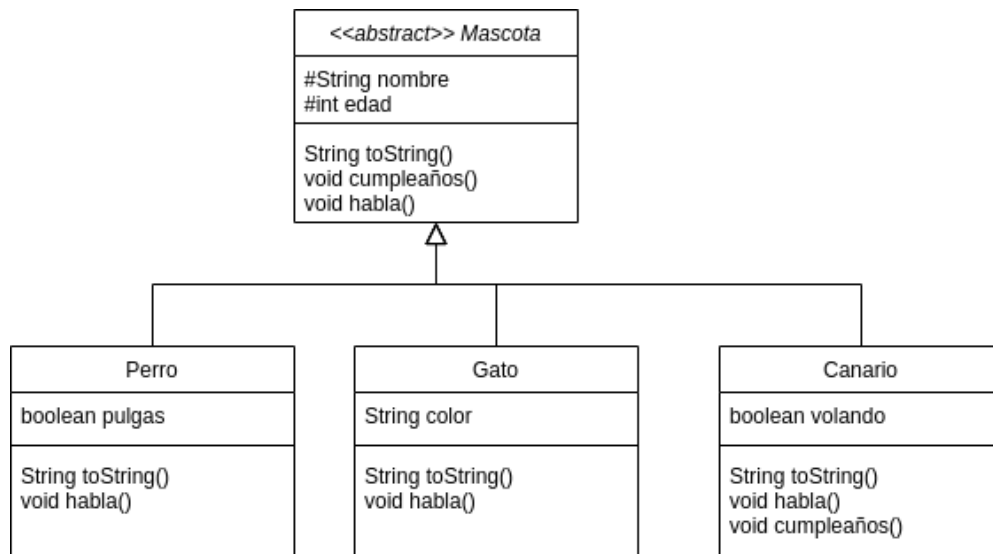


Añade **una subclase más llamada Rectangulo**, que herede de Figura, y añade a ProgramaFiguras la línea indicada en negrita. Comprueba que funciona correctamente.

```
public class ProgramaFiguras {  
    public static void main(String[] args) {  
        ArrayList<Figura> figuras = new ArrayList<Figura>();  
        figuras.add(new Circulo(10)); // Radio=10  
        figuras.add(new Cuadrado(10)); // Lado=10  
        figuras.add(new Triangulo(10,5)); // Base=10, Altura=5;  
        figuras.add(new Rectangulo(10, 5)); // Base=10, Altura=5;  
        for (Figura f: figuras)  
            System.out.println("Área: " + f.area());  
    }  
}
```

Ejercicio D2. Mascotas

Implementa las clases indicadas en el diagrama teniendo en cuenta también las indicaciones de abajo. Puedes añadir algún atributo o método si lo ves necesario.



- No pueden existir mascotas genéricas, tienen que ser de tipo perro, gato o canario.
- Todas las mascotas tienen nombre y edad. Los perros pueden tener pulgas. Los gatos deben tener un color. Los canarios pueden estar volando.
- En la clase Mascota, solo los métodos `toString()` y `habla()` son abstractos.
- Todas las mascotas tienen el método `toString()`, que devuelve la información de todos sus atributos.
- Todas las mascotas tienen el método `habla()` que imprime por pantalla la onomatopeya del animal en cuestión. Los perros dicen “Guau Guau”, los gatos “Miau Miau” y los canarios dicen “Pio Pio”.
- Todas las mascotas tienen el método `cumpleaños()` que aumenta la edad de la mascota. Los perros y gatos aumentan su edad en 1 cada cumpleaños, pero los canarios la aumentan en 2.

Crea una clase ProgramaMascotas con función ‘main’ para hacer algunas pruebas:

1. Intenta crear un objeto Mascota para añadirlo al ArrayList (no debería ser posible).
2. Crea un ArrayList con dos perros, dos gatos y dos canarios.
3. Muestra por pantalla la información de todas las mascotas (con un bucle for).
4. Haz que todas las mascotas cumplan años 5 veces (con un doble bucle for).
5. Muestra de nuevo la información de todas las mascotas. Comprueba su edad.
6. Haz que todas las mascotas hablen. Comprueba su sonido.

Ejercicio D3. Sistema Solar

Implementa un programa para registrar la información de un sistema solar y sus astros (planetas y satélites). Ten en cuenta lo siguiente:

- A) Un sistema solar tiene un nombre y puede tener varios planetas. Tiene que ser posible añadir y eliminar planetas, además de mostrar la información de todos sus planetas.
- B) Existen dos tipos de astros, los planetas y los satélites (lunas). No pueden existir astros genéricos, todos serán de uno de los dos tipos.
- C) De todos los astros necesitamos saber su nombre, masa y temperatura media. De los planetas, además, distancia al sol. De los satélites, además, distancia al planeta.
- D) Un planeta puede tener varios satélites. Tiene que ser posible añadir y eliminar satélites, además de mostrar la información de todos sus satélites.

Identifica las clases que vas a necesitar, si hay herencia, si alguna es abstracta, sus atributos, métodos y haz el diagrama de clases antes de programarlo.

Utiliza la función main del programa para hacer pruebas. Comprueba que es posible crear un sistema solar con varios planetas que a su vez tengan satélites, hacer algunos cambios (eliminar y añadir) y mostrar su información por pantalla.