

Ejercicio 1 – PRODUCTOS

Supongamos una clase Producto con dos atributos:

- String nombre
- int cantidad

Implementa esta clase con un constructor (con parámetros) además de los getters y setters de sus dos atributos. No es necesario comprobar los datos introducidos.

A continuación, en el programa principal haz lo siguiente:

1. Crea 5 instancias de la Clase Producto.
2. Crea un ArrayList.
3. Añade las 5 instancias de Producto al ArrayList.
4. Visualiza el contenido de ArrayList utilizando Iterator.
5. Elimina dos elemento del ArrayList.
6. Inserta un nuevo objeto producto en medio de la lista.
7. Visualiza de nuevo el contenido de ArrayList utilizando Iterator.
8. Elimina todos los valores del ArrayList.

Ejercicio 2- MASCOTAS

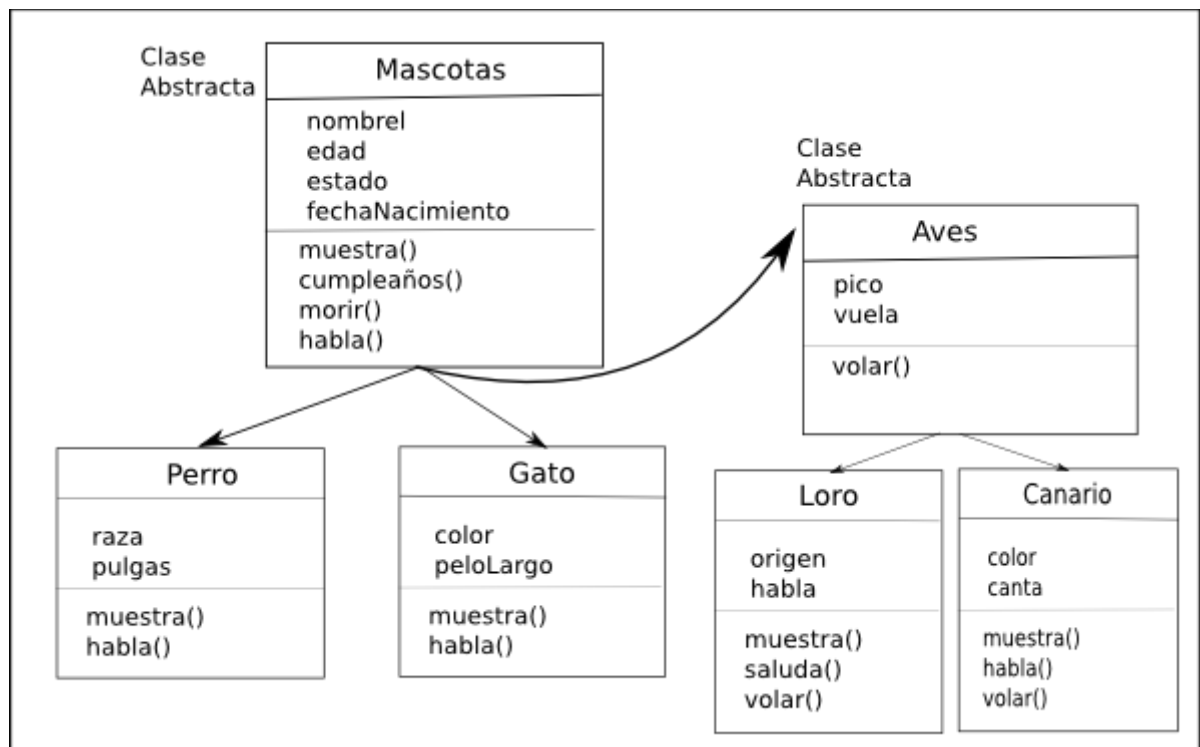
Implementa una clase llamada **Inventario** que utilizaremos para almacenar referencias a todos los animales existentes en una tienda de mascotas.

Esta clase debe cumplir con los siguientes requisitos:

- En la tienda existirán 4 tipos de animales: perros, gatos, loros y canarios.
- Los animales deben almacenarse en un ArrayList privado dentro de la clase **Inventario**.
- La clase debe permitir realizar las siguientes acciones:
 - Mostrar la lista de animales (solo tipo y nombre, 1 línea por animal).
 - Mostrar todos los datos de un animal concreto.
 - Mostrar todos los datos de todos los animales.
 - Insertar animales en el inventario.
 - Eliminar animales del inventario.
 - Vaciar el inventario.

Implementa las demás clases necesarias para la clase Inventario.

El diagrama UML sería:



Ejercicio 3- EMPRESA y EMPLEADOS

Vamos a implementar dos clases que permitan gestionar datos de empresas y sus empleados.

Los **empleados** tienen las siguientes características:

- Un empleado tiene nombre, DNI, sueldo bruto (mensual), edad, teléfono y dirección.
- El nombre y DNI de un empleado no pueden variar.
- Es obligatorio que todos los empleados tengan al menos definido su nombre, DNI y el sueldo bruto. Los demás datos no son obligatorios.
- Será necesario un método para imprimir por pantalla la información de un empleado.
- Será necesario un método para calcular el sueldo neto de un empleado. El sueldo neto se calcula descontando del sueldo bruto un porcentaje que depende del IRPF. El porcentaje del IRPF depende del sueldo bruto anual del empleado (sueldo bruto x 12 pagas).(*)

Sueldo bruto anual	IRPF
Inferior a 12.000 €	20%
De 12.000 a 25.000 €	30%
Más de 25.000 €	40%

Por ejemplo, un empleado con un sueldo bruto anual de 17.000 € tendrá un 30% de IRPF. Para calcular su sueldo neto mensual se descontará un 30% a su sueldo bruto mensual.

Las **empresas** tienen las siguientes características:

- Una empresa tiene nombre y CIF (datos que no pueden variar), además de teléfono, dirección y empleados. Cuando se crea una nueva empresa esta carece de empleados.
- Serán necesarios métodos para:
 - Añadir y eliminar empleados a la empresa.
 - Mostrar por pantalla la información de todos los empleados.
 - Mostrar por pantalla el DNI, sueldo bruto y neto de todos los empleados.
 - Calcular la suma total de sueldos brutos de todos los empleados.
 - Calcular la suma total de sueldos netos de todos los empleados.

Implementa las clases Empleado y Empresa con los atributos oportunos, un constructor, los getters/setters oportunos y los métodos indicados. Puedes añadir más

métodos si lo ves necesario. Estas clases no deben realizar ningún tipo de entrada por teclado.

Implementa también una clase Programa con una función main para realizar pruebas: Crear una o varias empresas, crear empleados, añadir y eliminar empleados a las empresas, listar todos los empleados, mostrar el total de sueldos brutos y netos, etc.

(*) El IRPF realmente es más complejo pero se ha simplificado para no complicar demasiado este ejercicio.