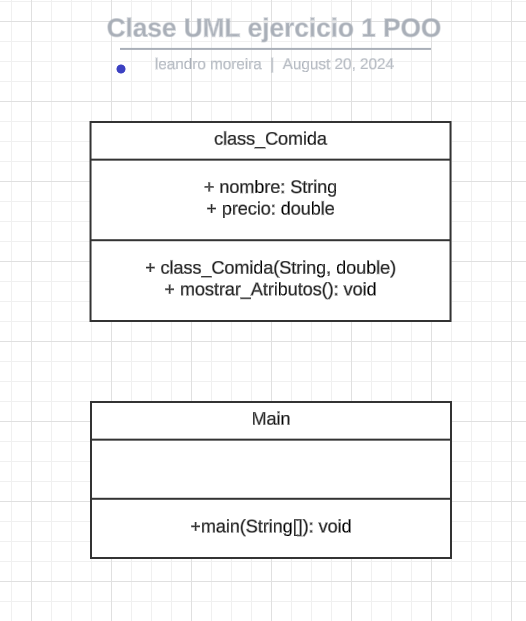
# ACTIVIDAD SOBRE CLASES Y OBJETOS EN POO

## Pregunta 1: Definición de Clases y Objetos

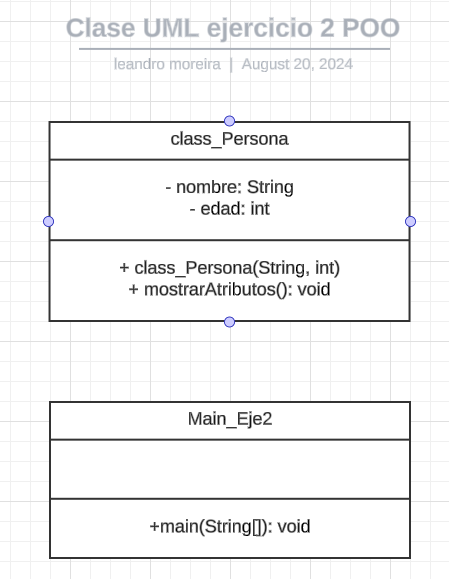
CLASE: es un modelo o plantilla que define un conjunto de comportamientos o acciones que posee el objeto que pertenecerá a esta clase.

OBJETO: es una instancia concreta de una clase, es todo lo que vemos en el mundo real y cada objeto tiene una característica única y especifica. Estos objetos pueden interactuar de forma síncrona con otros objetos.



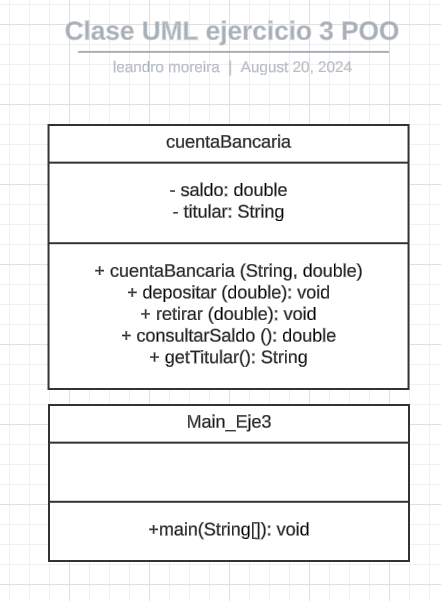
## Pregunta 2: Constructores

CONSTRUCTOR: es un tipo especial de método dentro de una clase que se utiliza para inicializar los objetos de esa clase. Tiene como propósito inicial asignar valores iniciales a los atributos del objeto en cuestión. En el caso de java tiene el mismo nombre que la clase y no tiene un tipo de retorno (void)



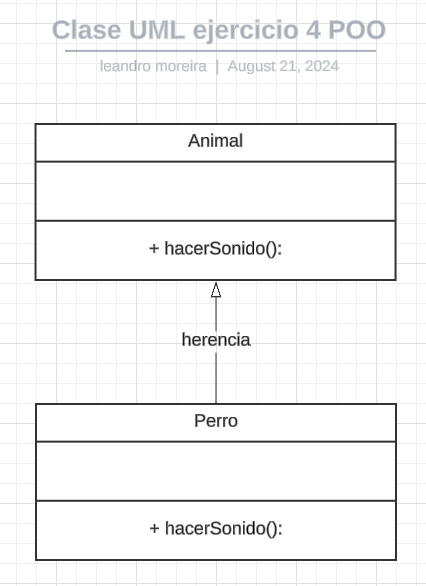
## Pregunta 3: Encapsulamiento

ENCAPSULAMIENTO: uno de los pilares fundamentales de POO, consiste en restringir el acceso a los atributos de una clase desde el exterior. Permitiendo que se controlen desde métodos públicos llamados **getters, setters**. Esto permite ocultar los detalles internos de una clase y como se pueden controlar los atributos.



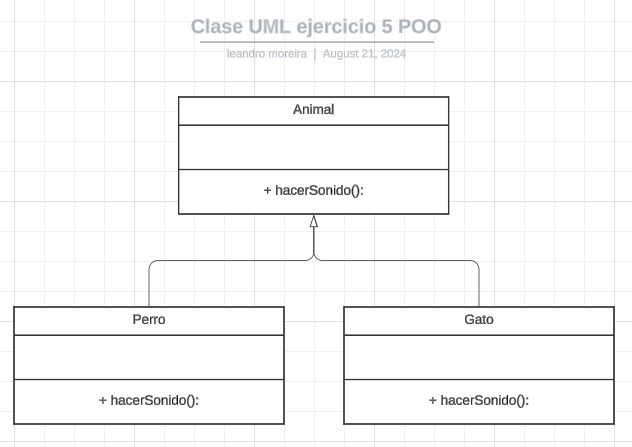
## Pregunta 4: Herencia

HERENCIA pilar fundamental en la POO que permite a una subclase derivada de una superclase, herede los atributos y métodos. Esto facilita reutilizar códigos y la creación de relaciones jerárquicas entre clases



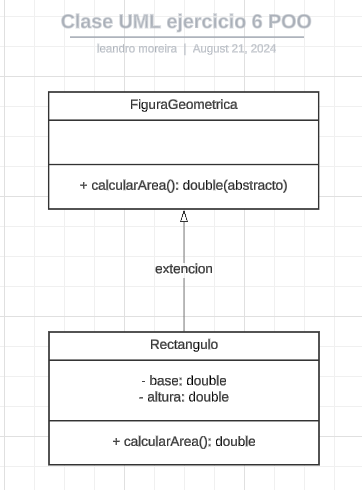
## Pregunta 5: Polimorfismo

POLIFORMISMO: 3 pilar fundamental en POO que mencionamos que permite que los objetos de diferentes clases sean tratados como si fueran de una misma clase base, mientras que cada uno puede tener un comportamiento diferente, implica que una sola interfaz puede ser utilizada para representar diferentes tipos de objetos.



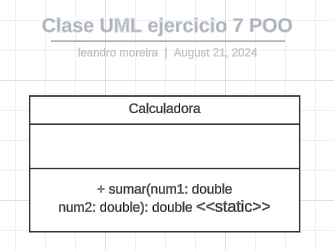
## Pregunta 6: Abstracción

ABSTRACCION: último pilar fundamental de POO, consiste en que un objeto puede tener muchos atributos y métodos pero debemos seleccionar solo los necesarios para poder realizar la actividad en específico. Es decir esconder todos los detalles y dejar a la vista solo lo fundamental. CLASE ABSTRACTA es una clase que no puede ser instanciada, está diseñada para solo ser heredada



## Pregunta 7: Métodos Estáticos

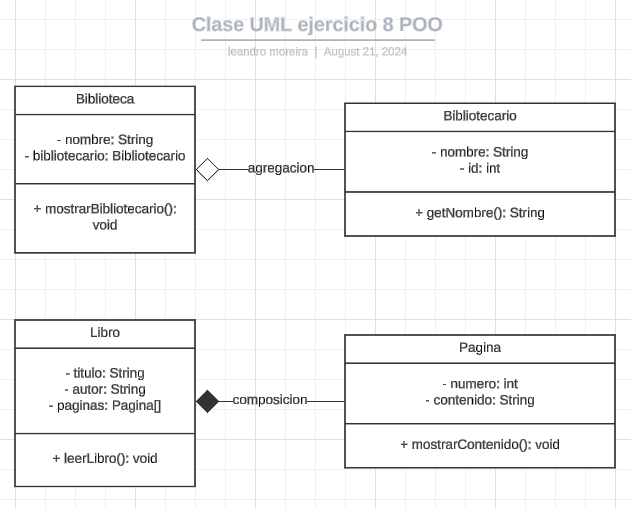
METODO ESTATICO: pertenece a la clase en sí, en lugar de pertenecer a una instancia individual de la clase. Significa que un método estático puede ser llamado sin crear un objeto de la clase. Sirve para aquellas funcionalidades que no depende de un objeto en sí, sino de ser resuelto directamente.



## Pregunta 8: Relaciones entre Clases

RELACIONES: para hablar de relaciones primero debemos entender la COMPOSICION y la AGREGACION

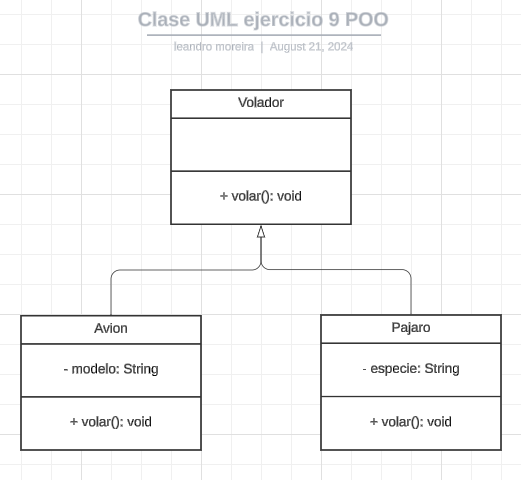
* Composición es una relación fuerte, la vida útil de la parte depende completamente de u objeto contenedor (todo). Eso quiere decir si el objeto contenedor se destruye las partes también lo harán, sería una relación de propiedad, es decir que el objeto contenedor es responsable de crear y destruir las partes. Como lo puede ser un libro y sus hojas, si el libro se destruye las hojas también
* Agregación es una relación débil, porque las partes pueden existir independientemente del todo. Por ejemplo un objeto contenedor tiene referencia a las partes, pero las partes no dependen de la vida útil del contenedor. Como lo sería una biblioteca y los libros. Ambos pueden existir con o sin el otro.



## Pregunta 9: Interfaces

INTERFACES: en POO es una especie de contrato que define un conjunto de métodos sin proporcionar una implementación concreta. Es decir, que nos permite definir el comportamiento implementado por cualquier clase, independiente de su lugar en la jerarquía de clases.

* Método abstracto: la interfaz puede declarar métodos, pero no proporciona su implementación.
* Implementación múltiple: una clase puede implementar múltiples interfaces, lo que nos permite es que una clase adquiera comportamientos de diferentes fuentes.
* No contiene estado: la interfaces no pueden contener variables de instancia. Solo pueden ser constantes (“public static final” por defecto) es decir que son globales y no asociadas a una instancia especifica
  + CONTEXTO “variable de instancia” son variables que pertenecen a una instancia específica de una clase, se definen dentro de una clase y cada instancia de la clase tiene su propio conjunto de estas variables. Estos pueden ser de cualquier tipo y sus estados pueden variar entre diferentes instancias de la clase.



## Pregunta 10: Sobrecarga de Métodos

SOBRECARGA DE METODOS: es una característica, permite definir múltiples métodos en una clase con el mismo nombre pero con diferentes parámetros. Se basa en el concepto de que los métodos se diferencian por su lista de parámetros, no por su tipo de retorno.

* Nombre del Método: los métodos sobrecargados deber tener el mismo nombre
* Lista de Parámetros: los métodos deber diferenciarse por el número, tipo o el orden de los parámetros
* Tipo de Retorno: la sobrecarga no depende del tipo de retorno; es decir, no puedes sobrecargar métodos basándote solo en el tipo de retorno.

