

# AWAKE LAB

## #programmingbootcamp



# HERENCIA Y POLIMORFISMO



# HERENCIA Y POLIMORFISMO



## ¿QUÉ VAMOS A VER?

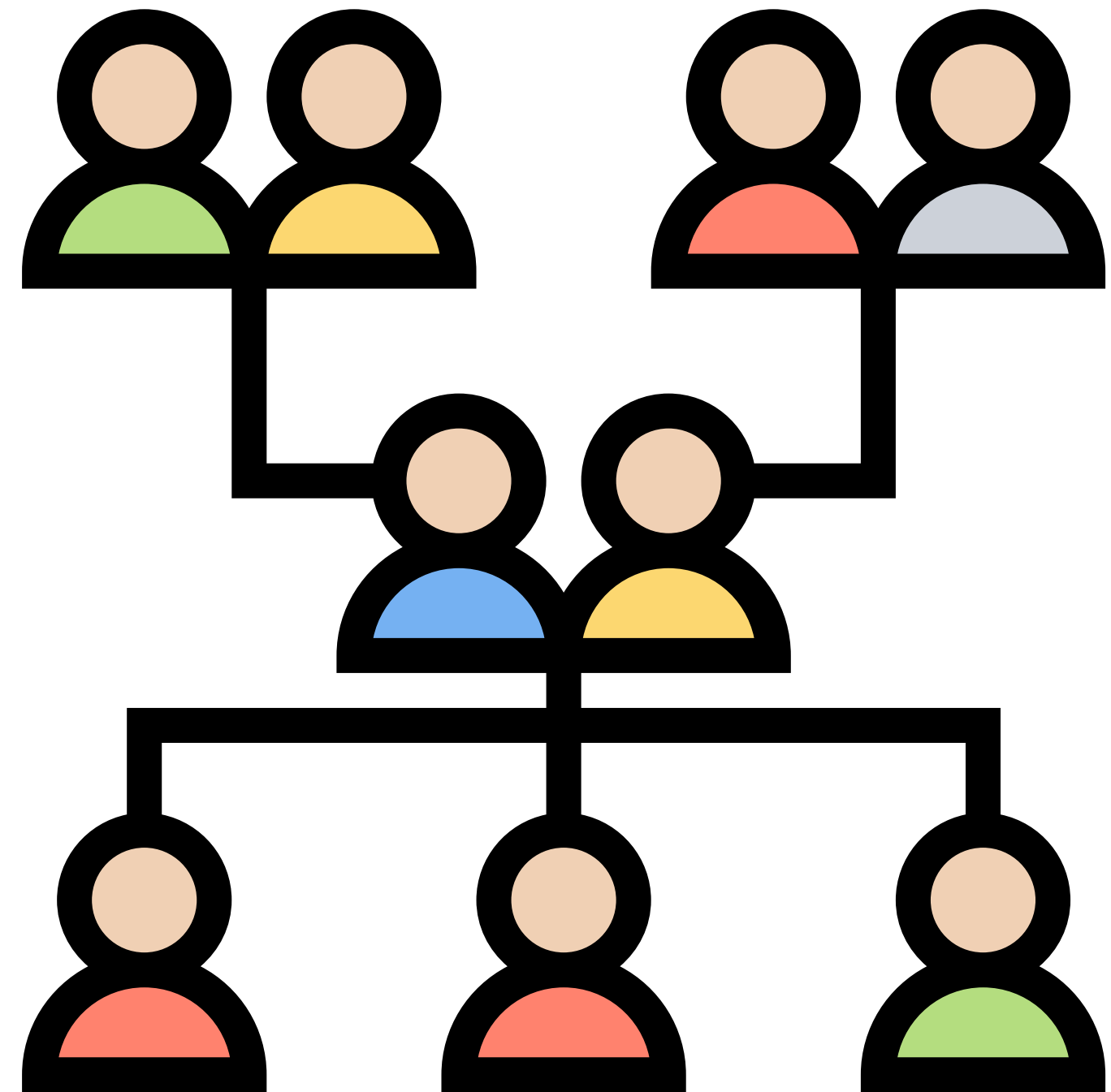
- Herencia.
- Interfaces.
- Polimorfismo.
- Clases abstractas.



# HERENCIA



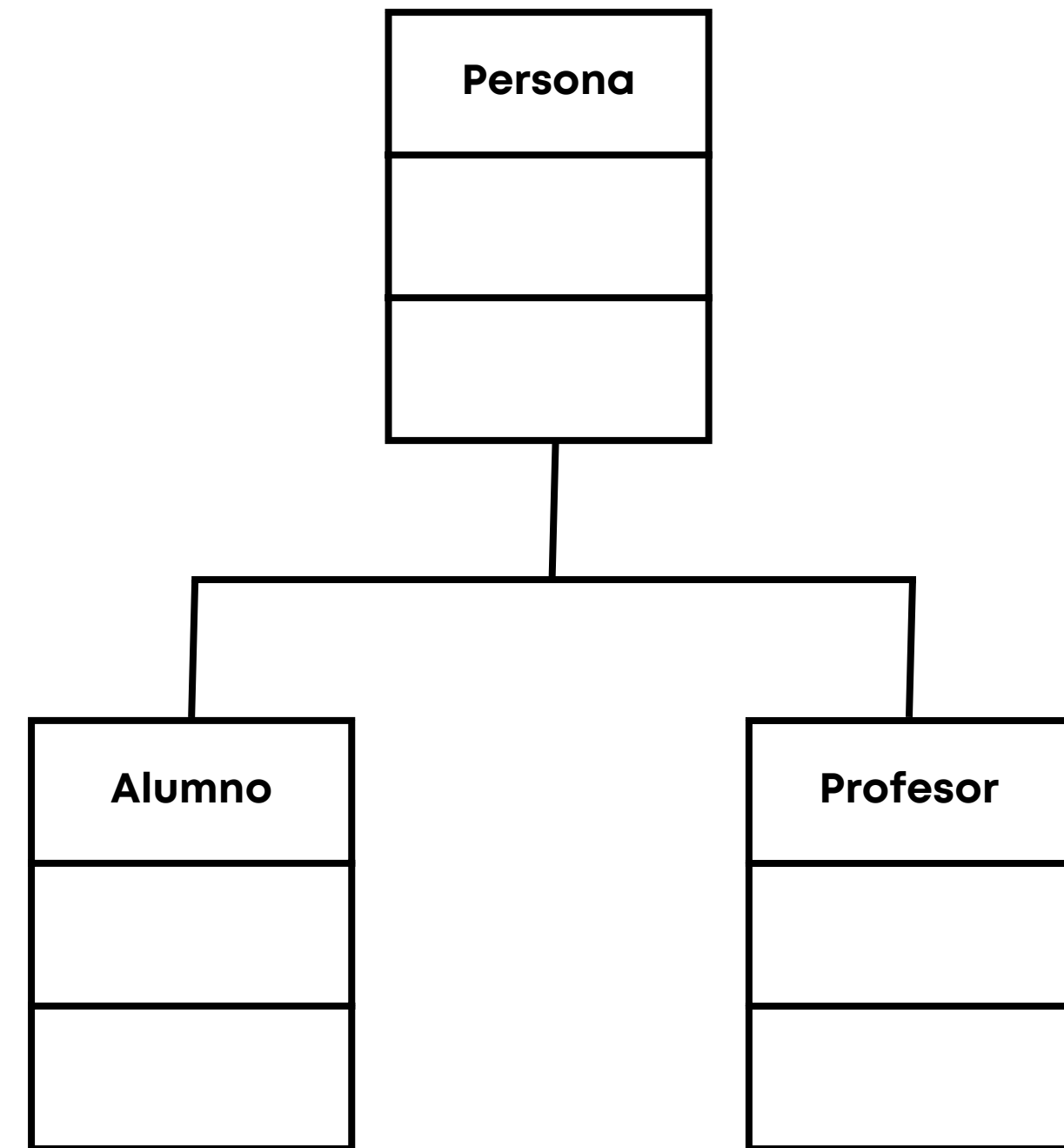
- Herencia es un concepto clave que permite a una clase adquirir los atributos y métodos de otra.
- Su funcionamiento va en pro de reutilizar el código.
- La nueva clase creada se denomina subclase o clase hija. Su principal cualidad es que hereda atributos y métodos ya existentes, pero puede también tener los suyos propios.
- La clase original es conocida como super clase o clase padre.



# HERENCIA DE CLASES



- **Persona:** Es una clase genérica que almacena los datos comunes de las personas.
- **Alumno:** Hereda los atributos de la clase Persona, incluyendo además los suyos propios como, por ejemplo, idAlumno, materia, etc.
- **Profesor:** Otra clase hija de Persona, que incluye todos los atributos de su clase padre, además del identificador de profesor.



# INTERFACES



- Básicamente se trata de un conjunto de métodos que una clase puede implementar.
- Una interfaz declara los nombres de los métodos y los parámetros que aceptan (En caso de ser necesarios).
- Es una colección de métodos con propiedades constantes.
- Solamente puede extender a otras interfaces.
- Solo pueden tener métodos de acceso público.



# POLIMORFISMO



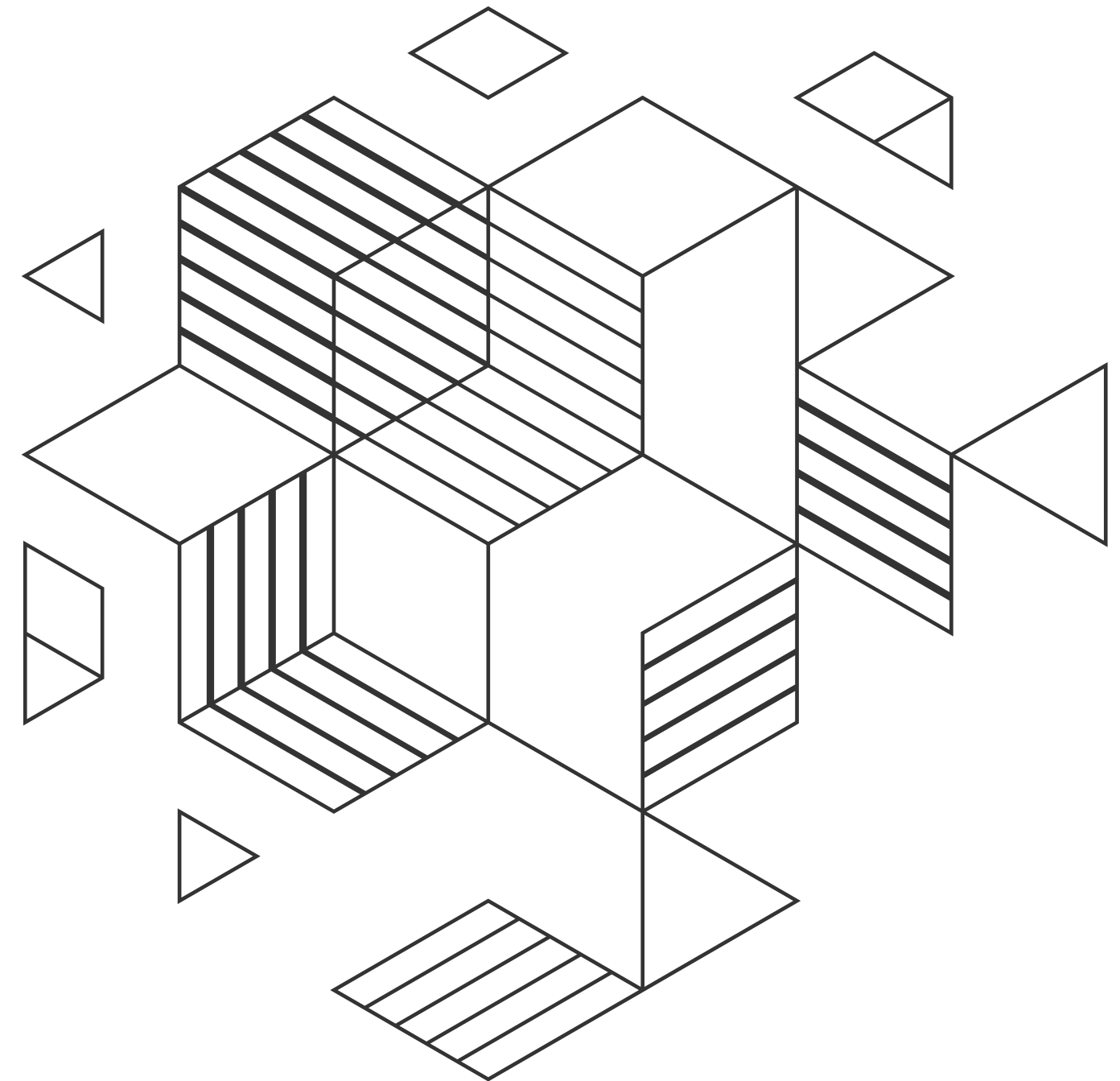
- Se refiere a la capacidad que tiene un objeto de tomar muchas formas diferentes. Es decir, un objeto puede ser tratado como si fuera de varios tipos.
- Supongamos que tenemos una clase padre llamada “**Animal**”, de ella derivan dos clases hijas, “**Perro**” y “**Gato**”.
- El polimorfismo nos va a permitir tratar a un perro o un gato como un animal genérico. Por ejemplo, si tienes un parámetro de tipo “Animal”, le puedes pasar tanto un objeto perro como uno gato. Esto gracias a que ambos son clases hijas de Animal.



# CLASES ABSTRACTAS



- Son clases que no pueden ser instanciadas.
- Se utiliza como clase base o plantilla para otras clases relacionadas.
- Su principal característica es que contiene *métodos abstractos*, es decir, aquellos que no tienen nada más que el nombre declarado.
- Sus métodos llevan la palabra reservada **abstract**.
- Sí o sí debe implementar un método abstracto para considerarse clase abstracta.







# **AWAKELAB**

#programmingbootcamp

[nodovirtual.awakelab.cl](https://nodovirtual.awakelab.cl)

 **jELON** futurejob <sup>by</sup>  **adalid** <sup>Chile</sup>