

---

# Introducción a UML.

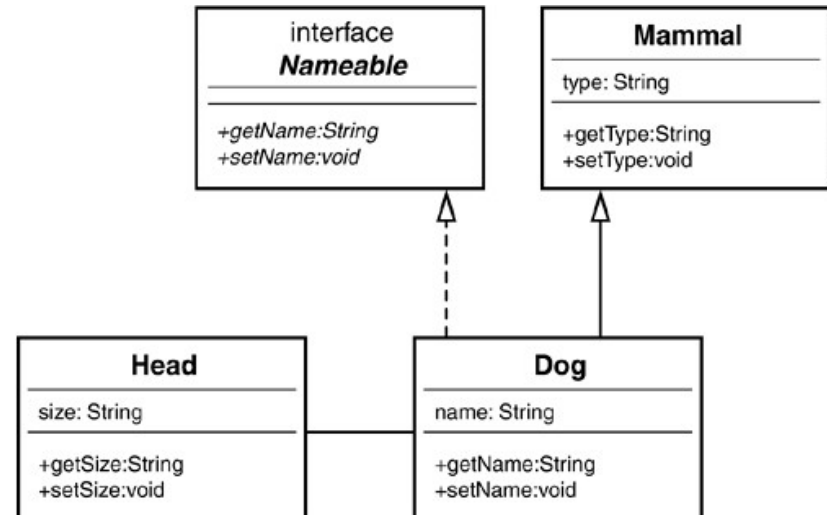
# Diagramas de clases

Entornos de desarrollo



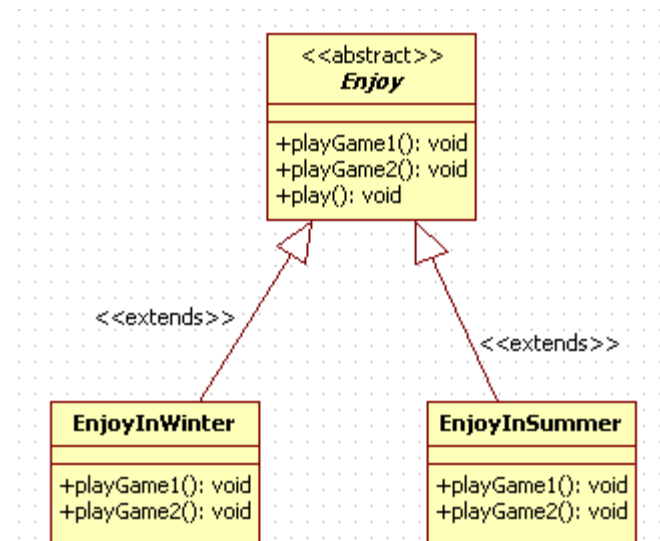
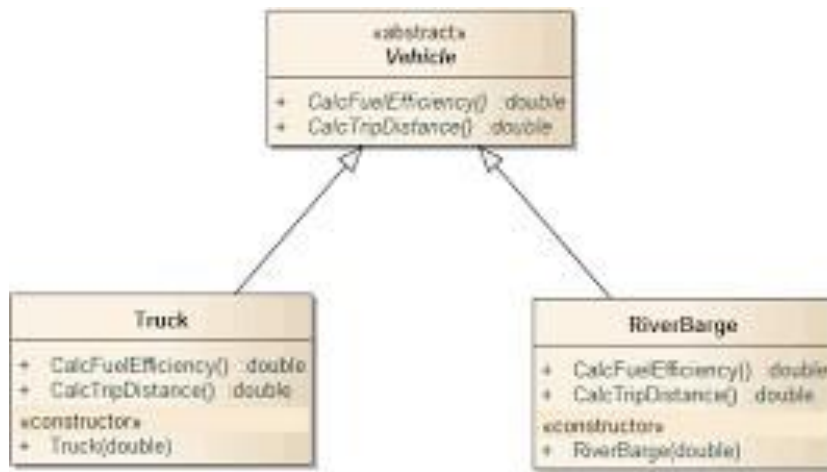
# Interfaz

- Son como **contratos** para las clases que les obliga a definir los métodos establecidos en cláusulas de funcionalidad.
- Las clases que se comprometen y cumplen con estos contratos **tienen implementada esa funcionalidad** y, por tanto, son capaces de llevarla a cabo. El “premio” es que pueden instanciar objetos.
- Las clases que se comprometen con estos contratos pero no los cumplen no son funcionales. El “castigo” es que son **abstractas** y no pueden instanciar objetos.



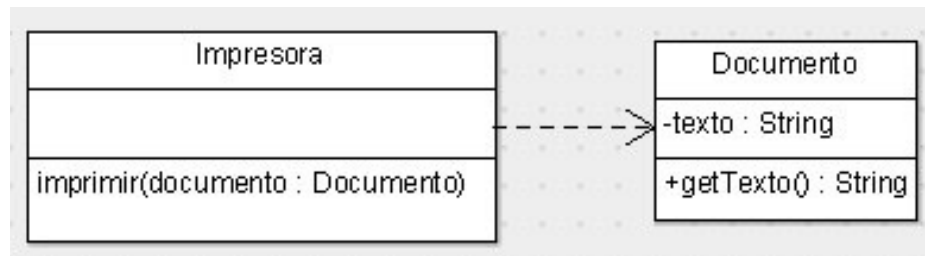
# Clase Abstracta

Son aquellas que por sí mismas no se pueden identificar con algo 'concreto', pero sí poseen determinadas características que son comunes en otras clases que pueden ser creadas a partir de ellas. (De ellas heredamos las características y los comportamientos pero no el modo de implementarlos.)



# Relaciones entre clases: Dependencia

- Es una relación de uso entre dos clases (una usa a la otra).
- Cualquier modificación en la clase de la que se depende (flecha), puede afectar en el funcionamiento de la clase dependiente.
- La clase de la que se depende, no tiene que conocer nada sobre la clase dependiente.



# Paquetes

---

- Un paquete es un mecanismo para agrupar clases u otros elementos de otro tipo de diagramas en modelos más grandes, donde dichos elementos pueden ser enlazados.
- Los paquetes o subsistemas se representan gráficamente por medio de un rectángulo grande junto con otro más pequeño situado en la esquina superior izquierda del rectángulo mayor. Si los contenidos de los paquetes no se muestran, entonces el nombre del paquete debe ir dentro del rectángulo grande; en caso contrario dentro del rectángulo pequeño.

# Paquetes

---

