


# Procesos involucrados en el modelado de software




# DIAGRAMAS DE CLASE



El diagrama de clases es describir las clases que conforman un modelo determinado de un sistema/software

Existen tres perspectivas diferentes para abordar y utilizar los diagramas de clase los cuales son los siguientes



**Conceptual:** manejo de conceptos basados en el problema que se esta abordando con el proyecto

**Especificación:** Nombra las clases y tipos de datos.

**Implementación:** representa tal cual como aparecen las clases en el software de desarrollo.

# Elementos de un diagrama de clases

**CLASES:** Descripción de un conjunto de objetos que tienen unos atributos, unas operaciones, métodos, relaciones y semántica y las cuales serán representadas gráficamente en el diagrama de clases.





Dependiendo del detalle del diagrama de clase, la notación utilizada debe seguir un modelo para indicar el nombre, su tipo, si tiene un valor de inicio, su visibilidad.

Visibilidad nombre: tipo=valor

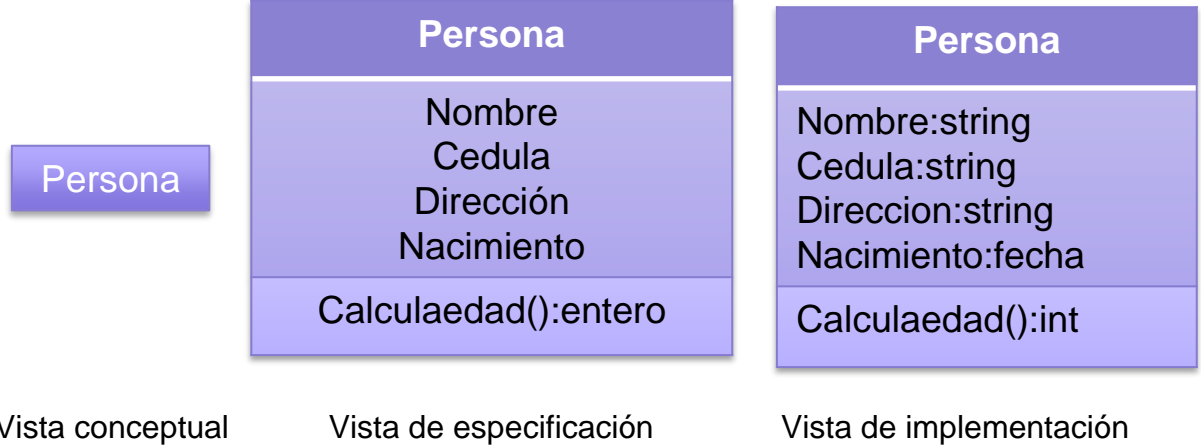
Tipo--- tipo de dato

Valor-opcional si se inicia con un valor

Visibilidad –Publica-Protégida-Privada



# Representación grafica de un Diagrama de clases según como se desee representar



# Representación grafica de un Diagrama de clases según como se desee representar



En el área superior figura el nombre de la clase que utilizamos como ejemplo, en la central están sus atributos y en la inferior las acciones que ella realiza. Note que las acciones llevan paréntesis al final del nombre dado que las mismas son funciones y por lo tanto devuelven un valor.

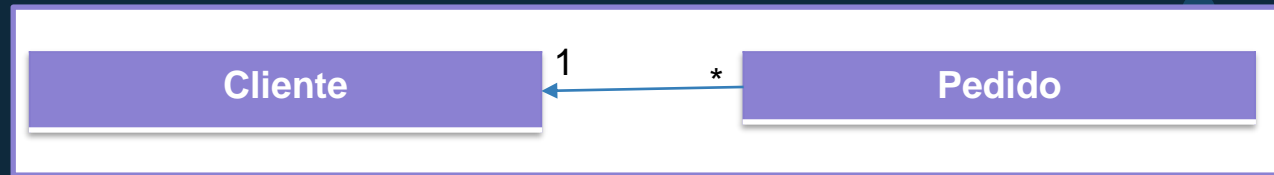


# Asociaciones

Las asociaciones representan las relaciones más generales entre clases, es decir, las relaciones con menor contenido semántico. Para UML una asociación va a describir un conjunto de vínculos entre las instancias de las clases.

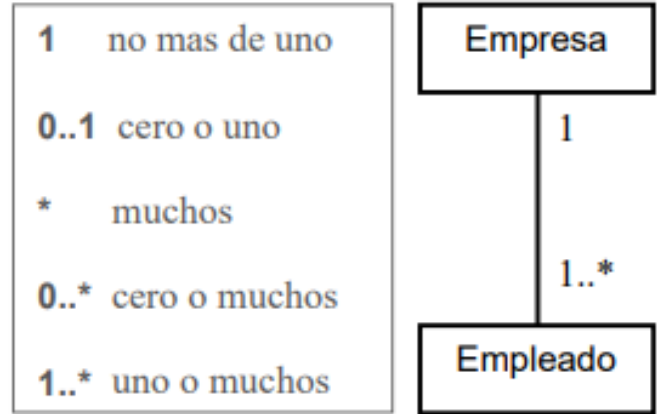
Ejemplo:

“Un Cliente puede tener asociados 0 o más Pedidos, y un Pedido tiene que tener asociado siempre un solo Cliente”.



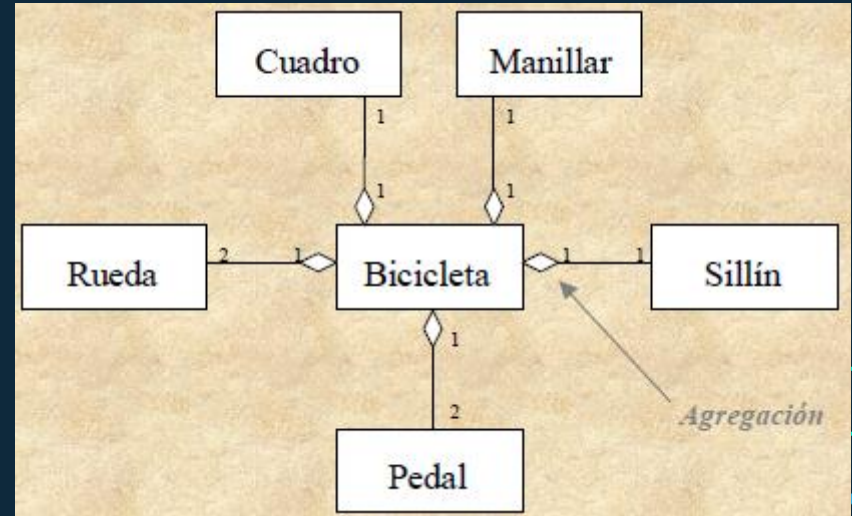
# Multiplicidad

Las notaciones utilizadas para señalar la multiplicidad se colocan cerca del final de una asociación. Estos símbolos indican el número de instancias de una clase vinculadas a una de las instancias de la otra clase. Por ejemplo, una empresa puede tener uno o más empleados, pero cada empleado trabaja para una sola empresa solamente.



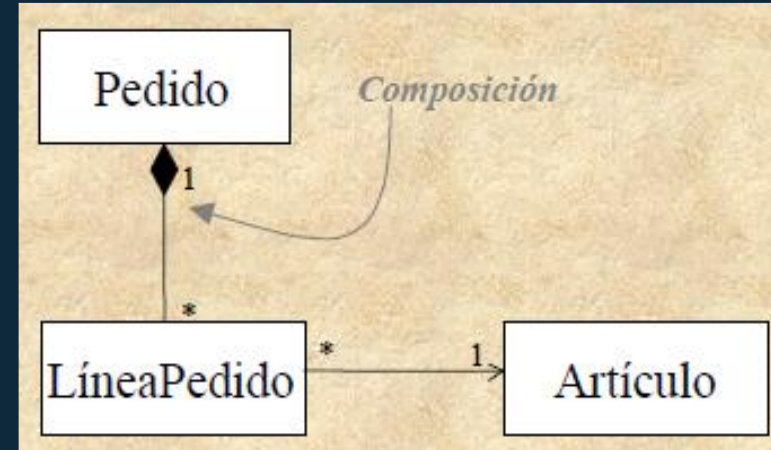
# Agregación

La agregación es una asociación con unas connotaciones semánticas más definidas: la agregación es la relación parte-de, que presenta a una entidad como un agregado de partes (en orientación a objeto, un objeto como agregado de otros objetos).



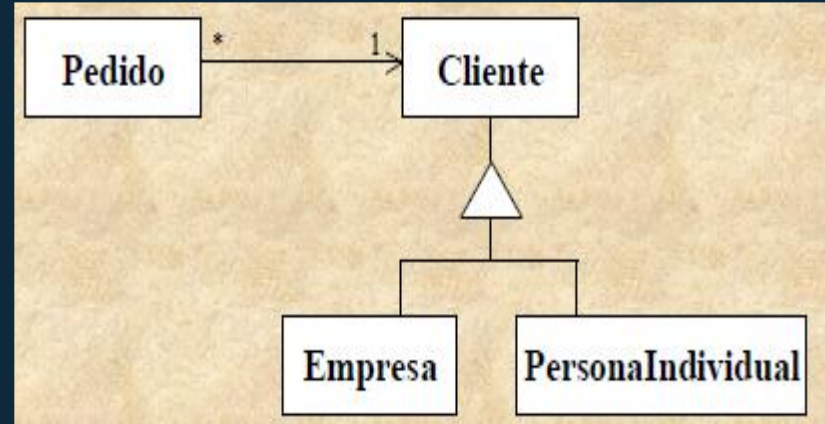
# Composición

La composición implica que los componentes de un objeto sólo pueden pertenecer a un solo objeto agregado, de forma que cuando el objeto agregado es eliminado todas sus partes son eliminadas también.

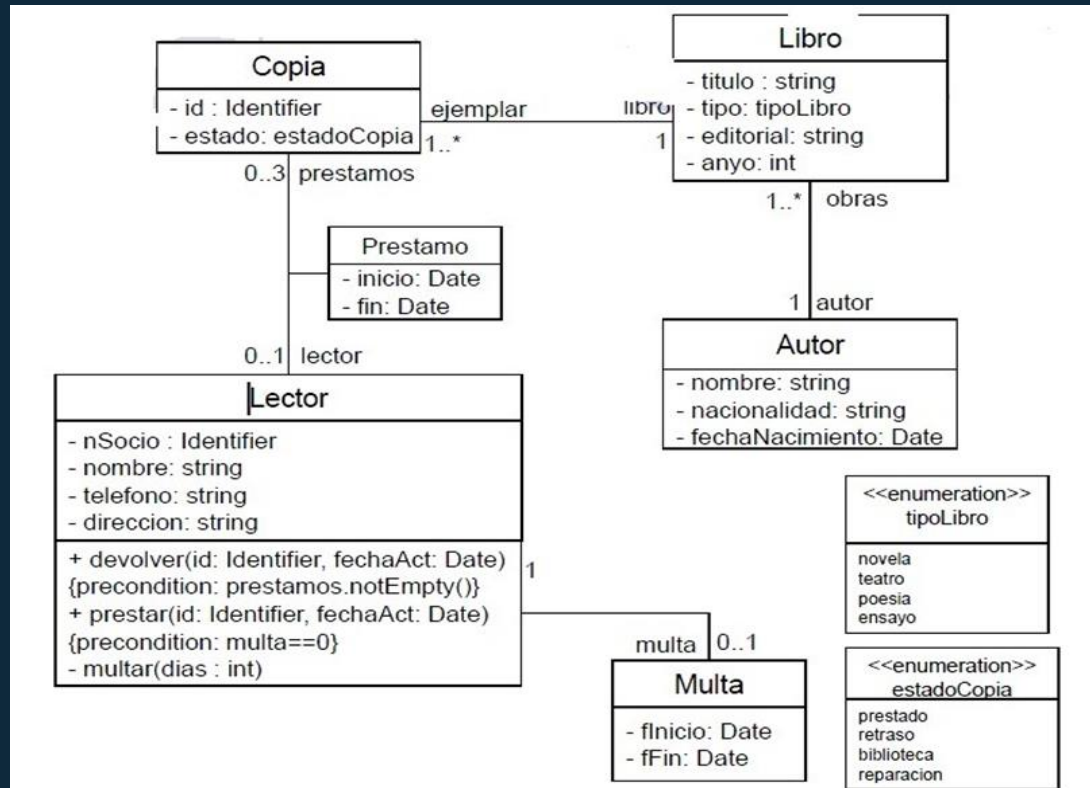


# Herencia

La herencia es la típica relación de generalización/especialización entre clases. En UML la herencia se representa mediante una flecha, cuya punta es un triángulo vacío. La flecha que representa a la herencia va orientada desde la subclase a la superclase



# Ejemplo global de diagrama de clases





# Avance diagrama de clases

Realizar actualización de la documentación realizando el diagrama de clase del sistema desarrollado. Sera la nota de parcial el desarrollo de este avance





# Thanks!

**Any questions?**

