



UNIVERSIDAD
Popular del Cesar

Ingeniería de Sistemas

Patrones de Diseño de Software


Profesor: Deivis Martínez Acosta






Diagramas de Clases

Los diagramas de clases están directamente relacionados con el modelado de la programación orientada a objetos, en este diagrama definiremos las clases que vamos a usar cuando estemos en la fase de codificación y cómo las clases se encuentran relacionadas entre sí. El diagrama de clases es un diagrama estático.

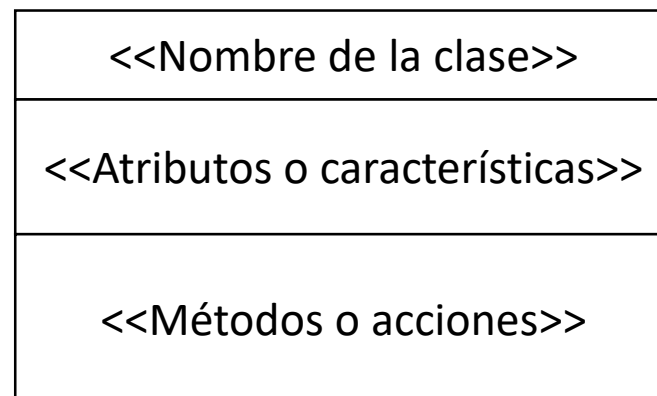


Elementos de los diagramas de clases:

- ✓ La Clase
 - ✓ Las Relaciones
 - ✓ La Multiplicidad
- 

Diagramas de Clases

El elemento principal en este tipo de diagramas es la “clase”, que se representa con un rectángulo y tiene tres partes verticales diferenciadas, donde van: el nombre, los atributos y los métodos.




Diagramas de Clases

Estas tres secciones son importantes y en el caso de atributos y métodos debe aparecer cada uno en una línea (es lo estandarizado).

Empleado
<ul style="list-style-type: none">- nombre: string- numeroidentificacion: string- salario: long- telefono: string- cargo: string
<ul style="list-style-type: none">+ modificarSalario(long salario):void+ getNombre():String+ setNombre(String):void



Modificadores de acceso

- Privado (-)
 - Protegido (#)
 - Público (+)
 - Paquete (~)
 - Estático (subrayado)
- 

Relaciones

En el diagrama de clases la forma de comunicarse las clases es por medio de relaciones, las relaciones denotan el uso que le damos a una clase con respecto a otra.

La idea general de esta relación es que solo una de las clases sepa de la relación.

Tipos de Relaciones

- Asociación _____
- Herencia/Generalización _____>
- Agregación _____◇
- Composición _____◆
- Uso/Dependencia - - - - ->

ASOCIACIÓN

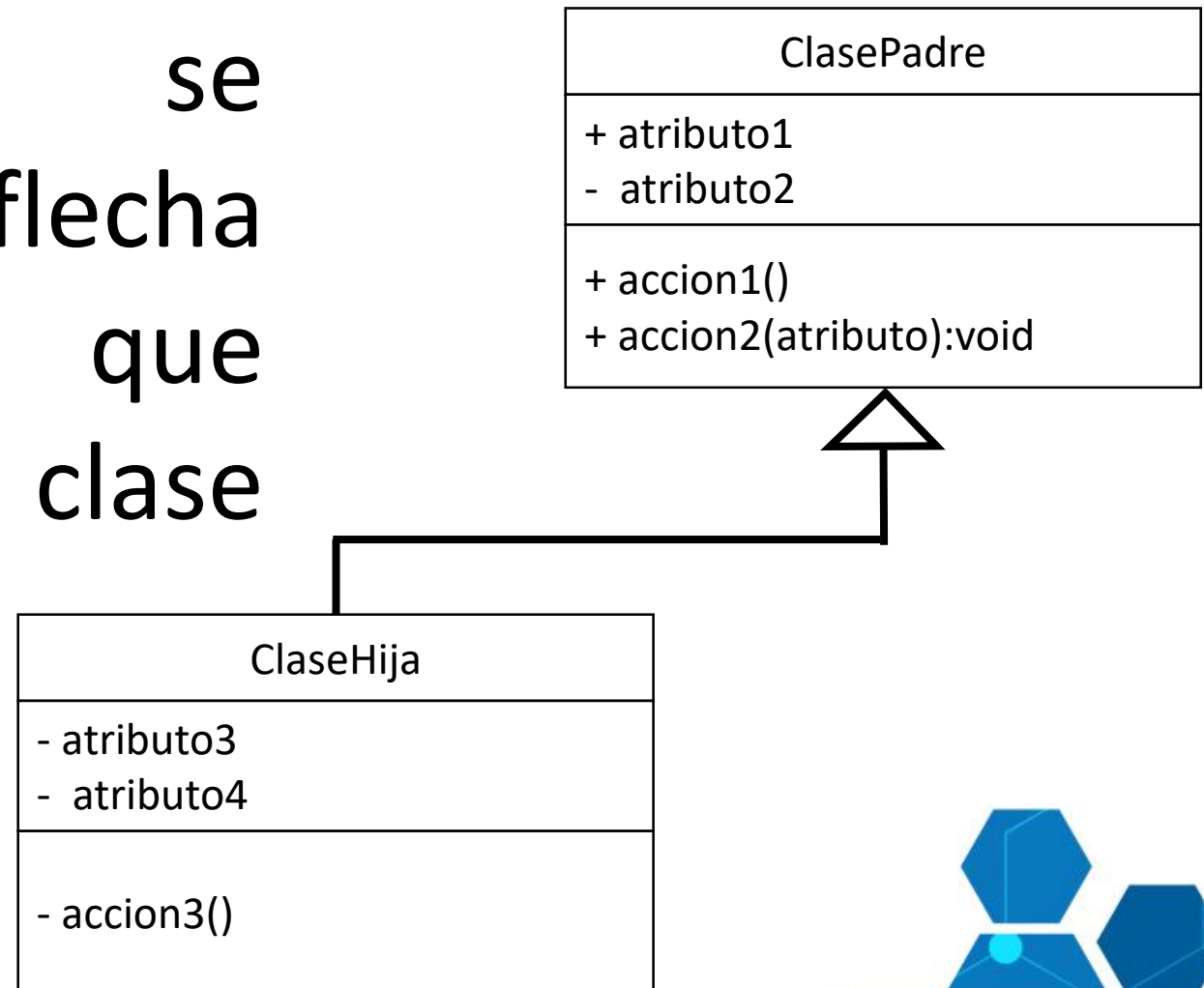
Es la forma básica de relacionarse las clases dentro de un diagrama. Una misma asociación puede relacionar varias clases, como es el caso de la asociación ternaria en donde se encuentran relacionadas tres clases. Podemos dar indicaciones o hacer anotaciones sobre las clases de un diagrama.

HERENCIA

Indica que una clase (clase hija) hereda los métodos y atributos de otra clase (clase padre), por lo que la clase hija tiene atributos y métodos propios accediendo adicionalmente a las características y acciones de la clase padre (los protegidos y públicos).

HERENCIA

Por lo general se representa por una flecha cerrada y vacía que apunta hacia la clase padre.

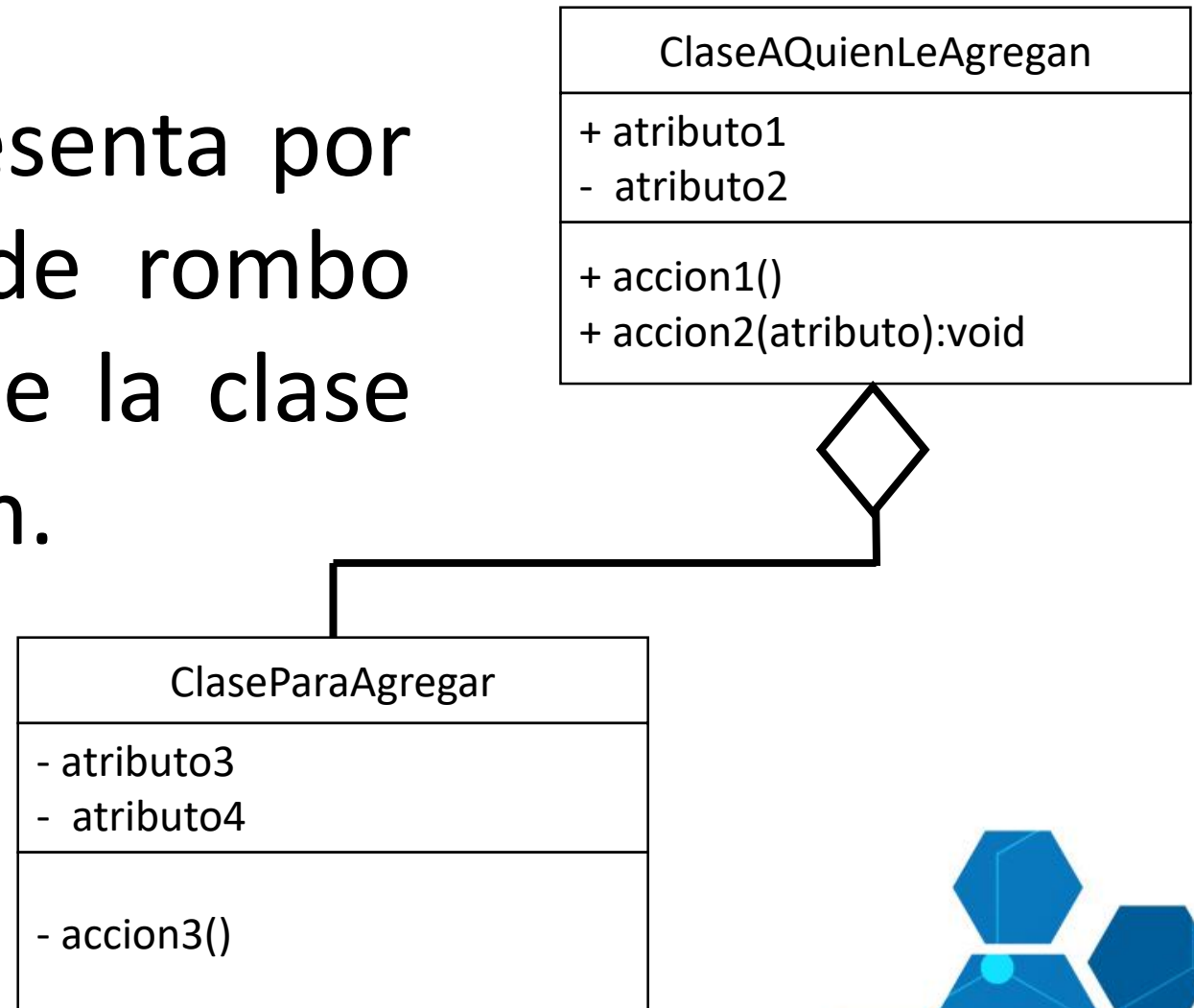


AGREGACIÓN

Con la agregación relacionamos las clases en la que una instancia hace parte de la otra. La agregada hace parte de la base, pero debemos tener en cuenta que las agregadas pueden existir y tener funcionalidad por si solas.

AGREGACIÓN


Por lo general se representa por una línea con punta de rombo vacío, en el extremo de la clase que recibe la agregación.





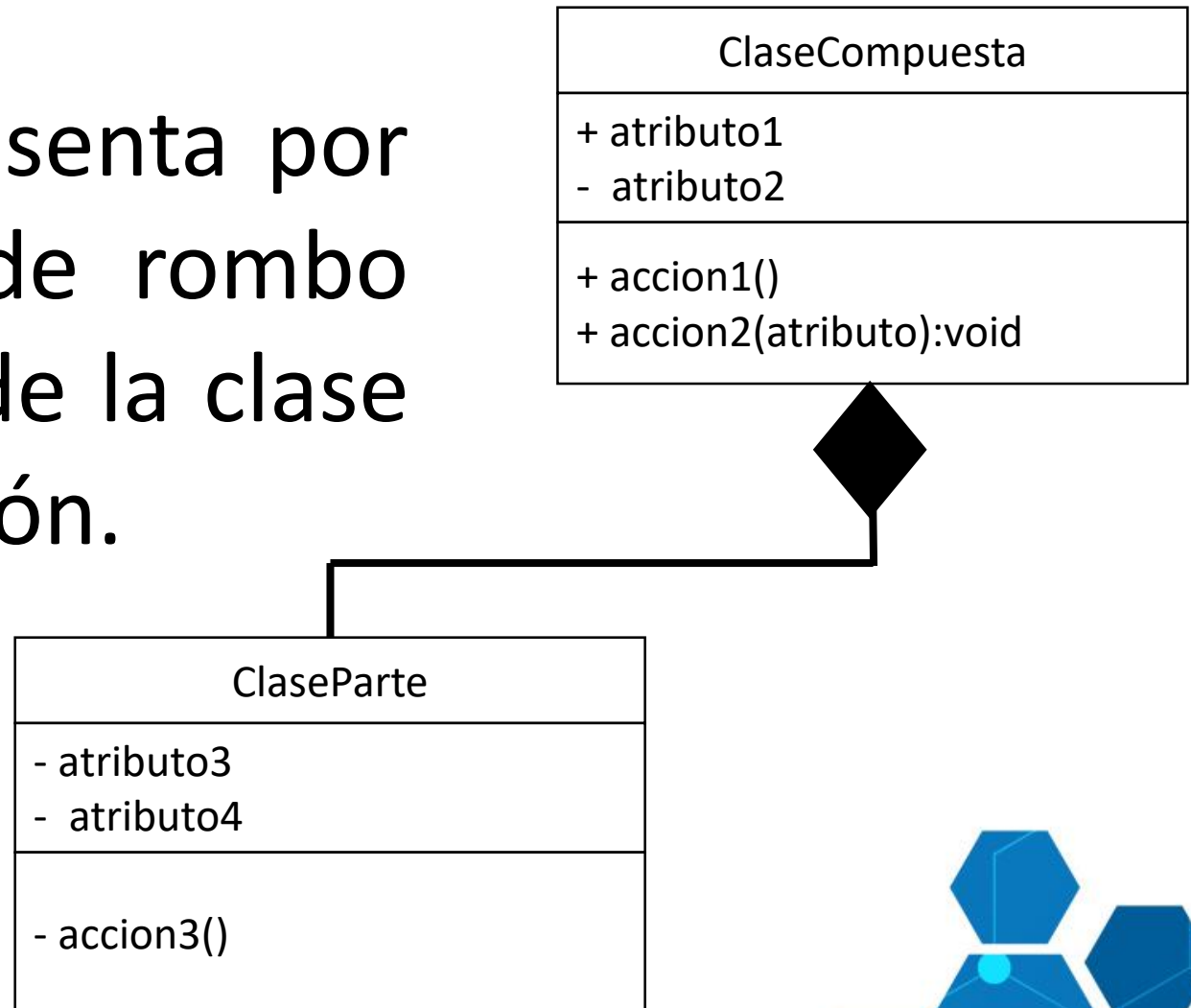
COMPOSICIÓN

Con la composición relacionamos las clases en la que una instancia hace parte de la otra y están ligadas con un lazo más fuerte que las de asociación, los elementos que forman parte no tienen sentido de existencia cuando el que los recibe no existe.



COMPOSICIÓN


Por lo general se representa por una línea con punta de rombo relleno en el extremo de la clase que recibe la composición.





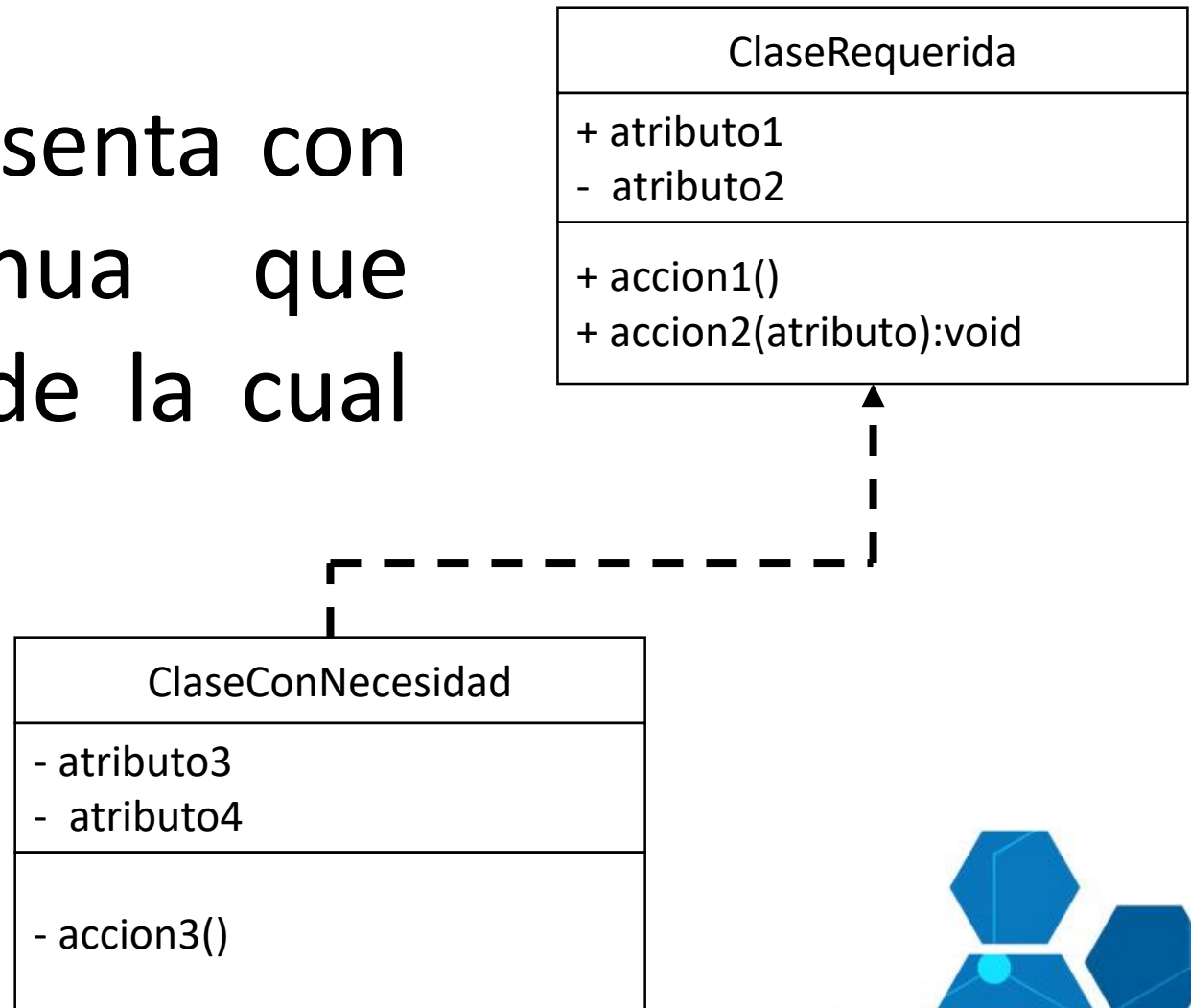
DEPENDENCIA

Este tipo de relación se establece cuando una clase requiere de otra para ofrecer sus funcionalidades.



DEPENDENCIA

Por lo general se representa con una flecha discontinua que apunta hacia la clase de la cual requerimos la utilidad.



Multiplicidad

Nomenclatura	Descripción
1	Uno y Solo Uno
0 .. 1	Cero o Uno
n .. m	Desde Varios hasta Varios
*	Cero o Varios
0 .. *	Cero o Varios
1 .. *	Uno o Varios (Por lo menos uno)

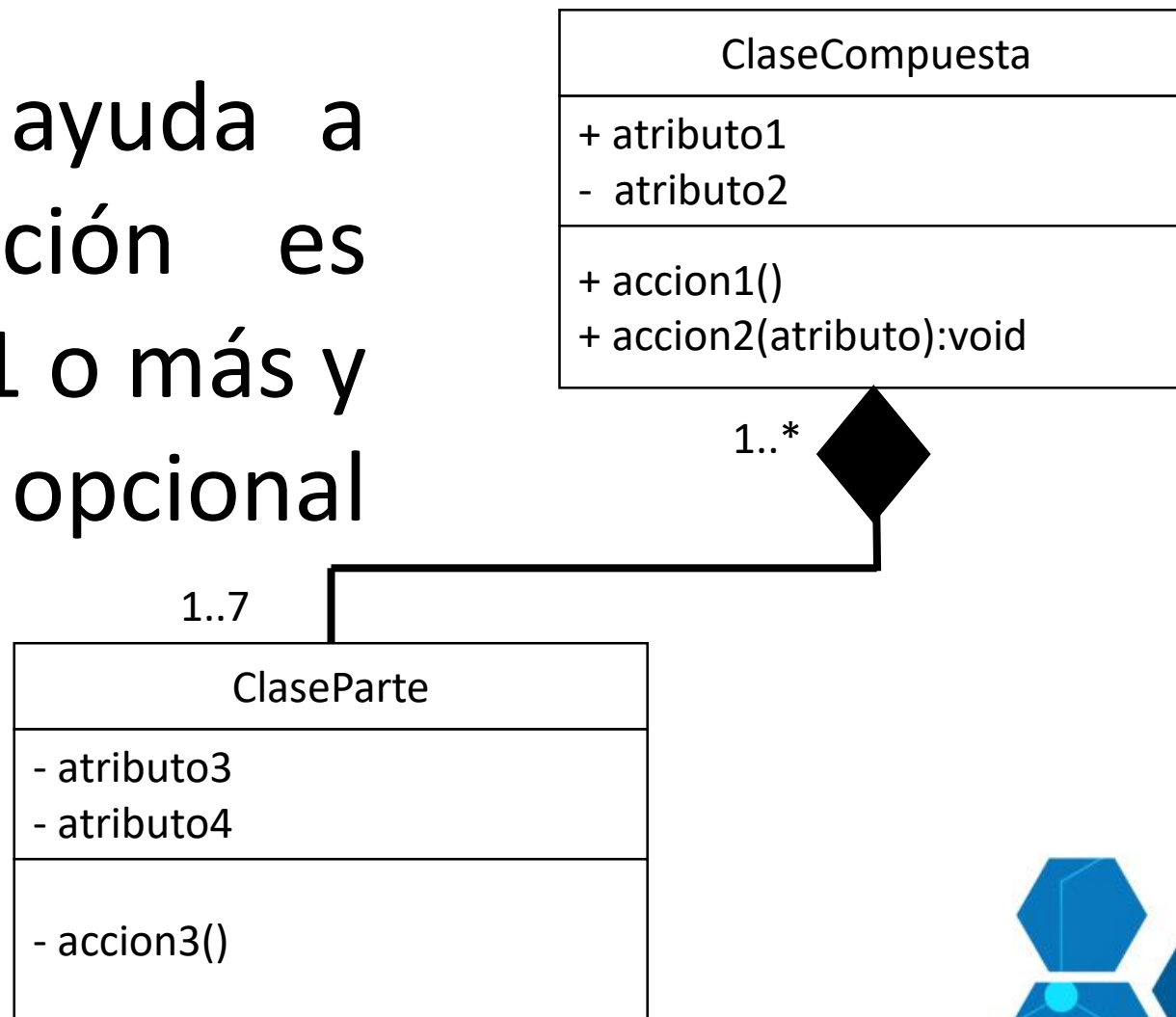
Multiplicidad

La multiplicidad en las asociaciones determinan cuantos objetos de cada tipo están presentes en la relación.

En las relaciones hablamos de dos extremos de la relación, entonces tendremos dos multiplicidades una para cada extremo que indica cuantos objetos de una clase se relacionarían con un objeto de la otra.

Multiplicidad

La multiplicidad nos ayuda a detectar si la relación es obligatoria cuando es 1 o más y es una relación opcional cuando es cero.



Vamos a la practica



UNIVERSIDAD
Popular del Cesar