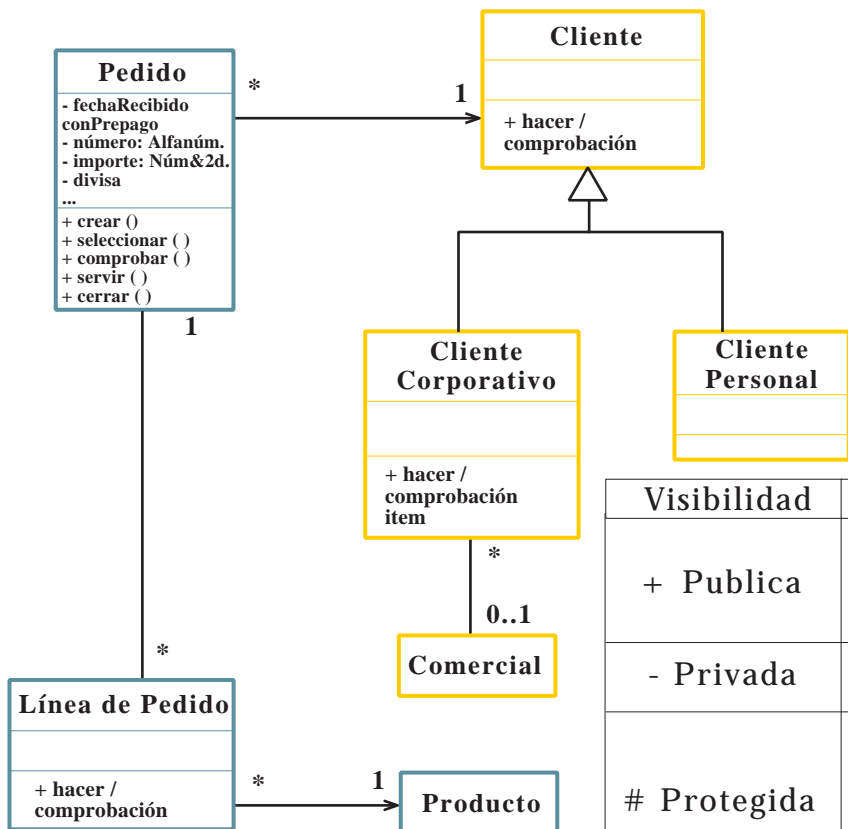
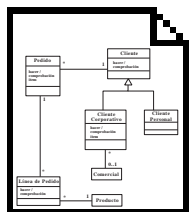


# Visibilidad

- Toda Clase encapsula unos elementos (atributos y operaciones) que disponen de ciertos criterios de visibilidad y manipulación para otras Clases.
- Los elementos públicos pueden ser usados por cualquier otra Clase.
- Los elementos privados pueden ser usados sólo por la Clase propietaria.
- Cada plataforma de desarrollo (C++, Smalltalk, Java) desarrolla sus propias reglas con respecto a la visibilidad y manipulación de atributos y operaciones.
- La notación UML especifica que todo atributo y operación de una Clase ha de disponer de un indicador de visibilidad.



Visibilidad	C++	Smalltalk	Java
+ Pública	Un elemento siempre es visible en cualquier parte del programa y puede ser llamado y modificado por cualquier objeto del sistema.	Todas las operaciones son públicas por defecto.	Un elemento siempre es visible en cualquier parte del programa y puede ser llamado y modificado por cualquier objeto del sistema.
- Privada	Un elemento sólo puede ser usado por la Clase que lo define.	Todas las variables instanciadas son privadas.	Un elemento sólo puede ser usado por la Clase que lo define.
# Protegida	Un elemento sólo puede ser usado por la Clase que lo define, o por las subclases de dicha Clase.		Un elemento puede ser usado por subclases y también por cualquier otra Clase en el mismo Package como la Clase propietaria. Esto implica que en Java, el concepto de visibilidad protegida es más público que package.
Package			Un elemento sólo puede ser usado por otras Clases que compartan el mismo Package.



Clases

## Ejemplos

Consideremos la instancia de **CLIENTE PERSONAL**, <<Miquel M.>>, este objeto puede acceder a cualquier elemento de <<Josep V.>>, que ha sido definido también como una instancia de **CLIENTE PERSONAL**, sea público, privado o protegido. <<Miquel M.>>, a su vez también puede acceder a cualquier elemento privado, protegido o público de <<Josep V.>>

Consideremos la Clase **CLIENTE** que dispone de una subclase **CLIENTE PERSONAL**. Consideremos también que el objeto <<Josep V.>>, es una instancia de **CLIENTE PERSONAL**.

- <<Josep V.>>
- Puede usar todo elemento público de cualquier objeto del sistema.
  - Puede usar también todo elemento privado de la Clase **CLIENTE PERSONAL**.
  - No puede usar los elementos privados definidos en la Clase **CLIENTE**.
  - Puede usar los elementos protegidos definidos para **CLIENTE** y **CLIENTE PERSONAL**.

En C++, podemos acceder a elementos de otros objetos de la propia Clase, de la misma manera que podemos acceder a los propios elementos de un objeto.

<<Josep V.>>, puede acceder a cualquier variable instanciada dentro de su propio objeto, si dicha variable ha sido definida dentro de **CLIENTE** o **CLIENTE PERSONAL**. De esta manera, el concepto de visibilidad privada en Smalltalk es parecido al concepto de visibilidad protegida en C++.

En Smalltalk no hay diferencia con respecto a que un objeto sea de la misma Clase o no. Podemos acceder sólo a los elementos públicos de otro objeto.

En Smalltalk, <<Miquel M.>>, no puede acceder a las variables instanciadas privadas de <<Josep V.>>, sólo a sus operaciones públicas.

Java permite marcar también las Clases como públicas o packages. Los elementos de una Clase pública pueden ser usados por cualquier Clase que importe el package a la que pertenece la Clase. Los elementos de una Clase package sólo pueden ser usados por otras Clases del mismo package.

