

Agregación



Una agregación se da cuando una clase es una colección o un contenedor de otras clases, pero no depende de la existencia de esas otras clases. **Si la clase que contiene a la colección desaparece, no afecta a las clases contenidas.**

En el siguiente ejemplo, la clase Departamento es un contenedor de clases Profesor:

```
public class Departamento{
    private String nombreDepartamento;
    private ArrayList<Profesor> listadoProfesores;

    //Los profesores ya existen y recibo un listado por parámetro
    public Departamento(ArrayList<Profesor> listadoProfesores){
        this.listadoProfesores = listadoProfesores;
    }
}
```

Los Profesores no desaparecen porque desaparezca el objeto Departamento.

Composición



La composición de clases es una relación aun más estricta que la agregación. Se da cuando una clase contiene referencias a otra clase, pero la vida de las instancias contenidas está ligada a la vida de la instancia contenedora. O sea, si **la clase compuesta desaparece también lo harán las clases contenidas.**

```

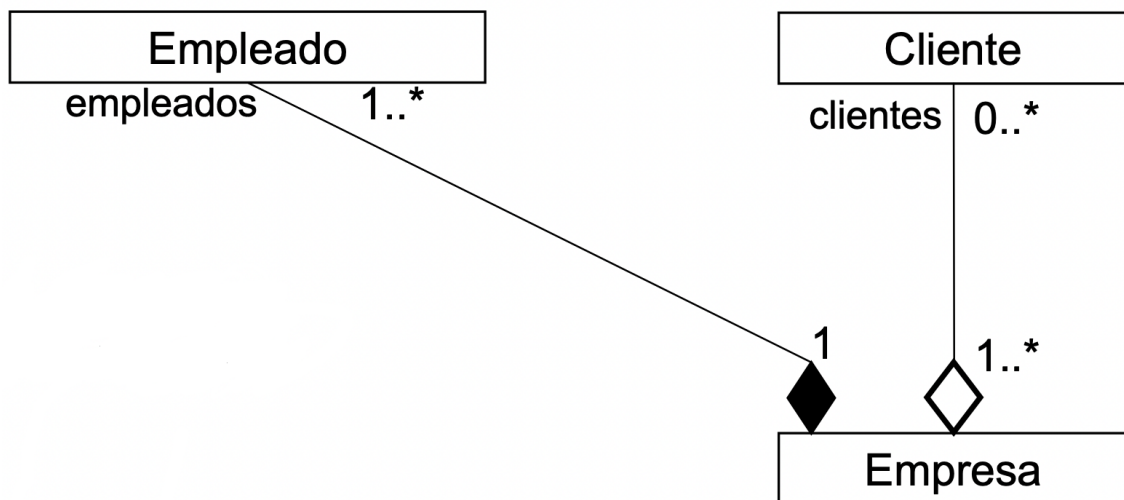
public class Universidad{
    private String nombreUniversidad;
    private ArrayList<Departamento> listadoDepartamentos;

    public Universidad(){
        this.listadoDepartamentos = new ArrayList<>();
    }

    public void nuevoDepartamento(String codigo, String nombreDepartamento){
        Departamento nuevoDepartamento = new Departamento(codigo, nombreDepartamento);
        listadoDepartamentos.add(nuevoDepartamento);
    }
}

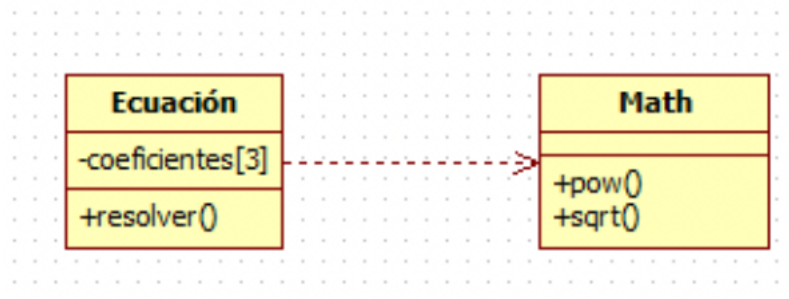
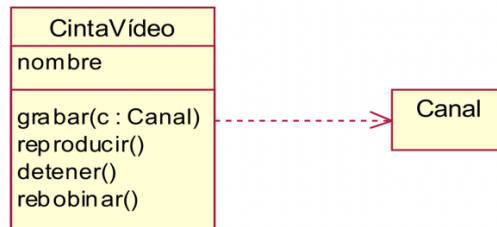
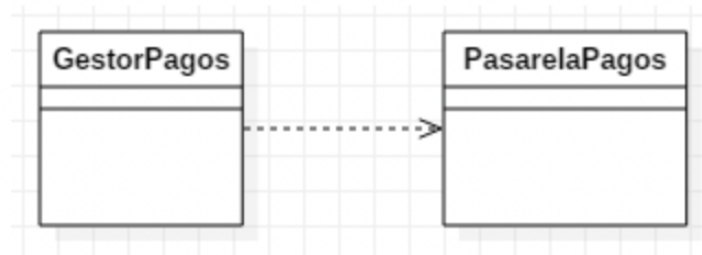
```

Diferencia entre ambas



Dependencia

Se utiliza este tipo de relación para representar que una clase requiere de otra para ofrecer sus funcionalidades. Es muy sencilla y se representa con una flecha discontinua que va desde la clase que necesita la utilidad de la otra flecha hasta esta misma.

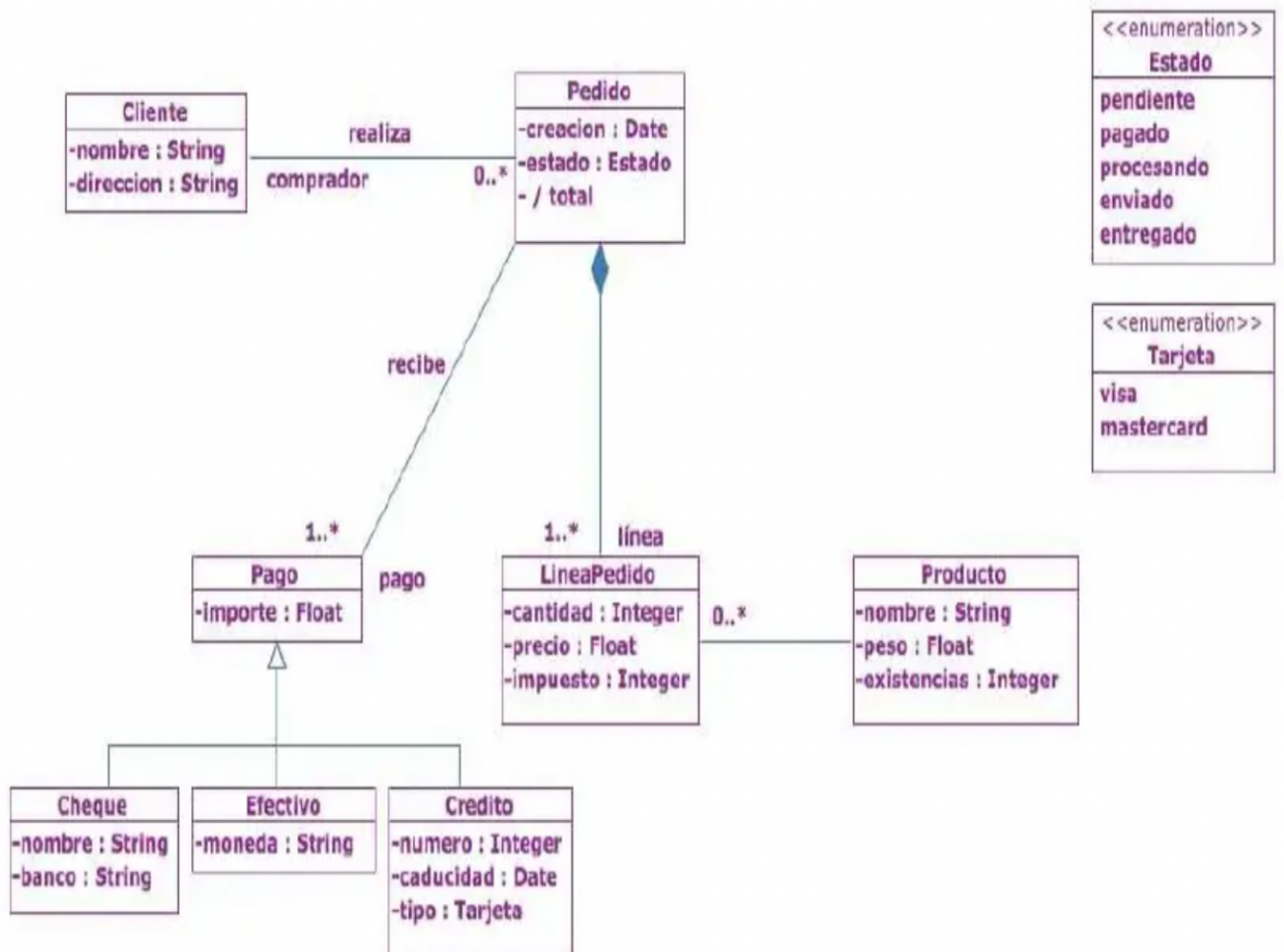


Multiplicidad

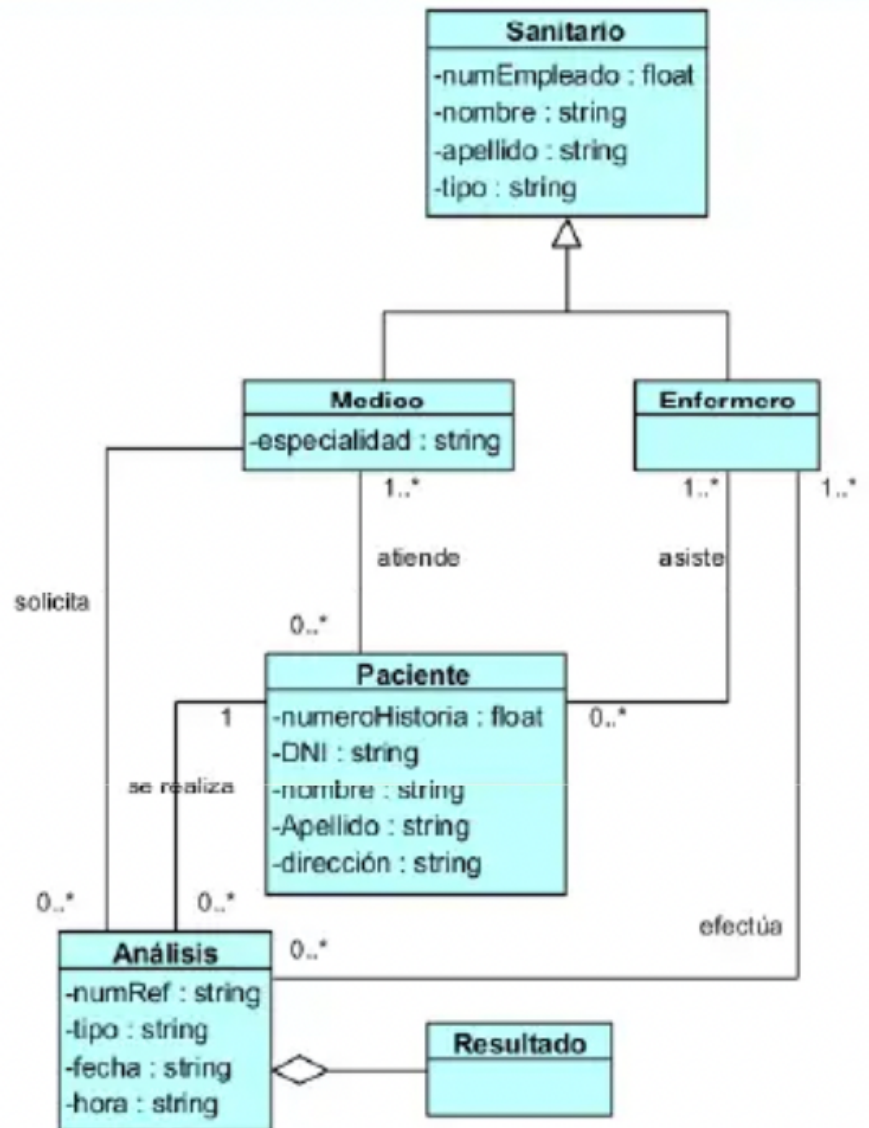
Multiplicidad	Significado
1	Uno y sólo uno
0..1	Cero o uno
N..M	Desde N hasta M
*	Cero o varios
0..*	Cero o varios
1..*	Uno o varios (al menos uno)

Representa mediante un diagrama de clases la siguiente especificación relacionada con una aplicación de gestión de pedidos

1. Dicha aplicación debe guardar la información de los pedidos que realizan sus clientes y la fecha en que se formalizan.
2. Los pedidos constan habitualmente de varios productos.
3. El coste total del pedido se calcula a partir de los precios individuales de cada producto, así como de la cantidad e impuestos asociados a cada uno.
4. Es importante mantener información sobre las existencias de cada producto con el fin de poder informar al cliente si habrá retrasos en la entrega del pedido.
5. El pedido podrá pagarse de una vez o en varios pagos.
6. Las formas de pago posible son:
 - Tarjeta de crédito (fecha de caducidad, número, VISA o MASTERCARD)
 - Efectivo (moneda)
 - Cheque (nombre, entidad bancaria)
7. El pedido podrá estar en uno de los siguientes estados: pendiente, pagado, procesando, enviado y entregado.



- Construir un diagrama de clases para un hospital con un conjunto de pacientes y un conjunto de empleados sanitarios (médicos y enfermeros) siguiendo las siguientes especificaciones
 - ▣ Un paciente es atendido por uno o más médicos y es asistido por un grupo de enfermeros
 - ▣ Cada paciente se describe por su número de historia clínica, su nombre y dirección
 - ▣ En la base de datos se mantiene información del personal sanitario referente a su número de empleado, nombre y tipo. Para los médicos hay que indicar además su especialidad
 - ▣ Para cada paciente se mantiene un registro de los análisis realizados identificados por un número de referencia, además se indica el tipo de análisis, la fecha en la que se realizó, el médico que solicitó el análisis, el o los enfermeros que realizaron la prueba y los resultados que se obtuvieron en la misma



- Representa mediante un diagrama de clases la siguiente especificación sobre una clínica veterinaria
 - ▣ Un veterinario tiene como pacientes animales y como clientes familias.
 - ▣ Un cliente es un conjunto de personas que suele corresponderse con una familia. Cada cliente tiene un código, el primer apellido del cabeza de familia, un número de cuenta bancaria, una dirección, un teléfono y los nombres y DNI de las personas correspondientes. No existe límite en el número de personas asociadas a una entidad cliente. Además, una persona puede estar dada de alta en varios clientes (p.e.: un hombre que vive con su esposa tiene un gato y como tal pertenece a un cliente, pero también está dado de alta en el cliente asociado con el perro de sus padres)
 - ▣ Los clientes pueden tener varias mascotas, cada mascota tiene un código, un alias, una especie, una raza, color de pelo, fecha de nacimiento aproximada, peso medio del animal en las últimas 10 visitas y el peso actual del animal
 - ▣ Se guardará un historial médico de cada mascota con
 - Cada enfermedad que tuvo y la fecha en la que enfermó
 - Calendario de vacunación, en el que se registrará la fecha de cada vacuna, la enfermedad de la que se vacuna

