



Sesión 2

P00: Agregación y composición.

W W W . M A K A I A . O R G

Carrera 43 A # 34 - 155. Almacentro. Torre Norte. Oficina 701
Medellín (Antioquia), Colombia

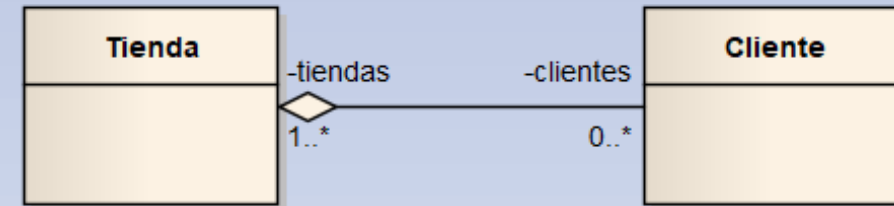


Contenido

1. Agregación
2. Composición
3. Cardinalidad.

¿Que es agregacion?

La agregación es un tipo de relación que indica que un objeto forma parte o le pertenece a otro objeto, es prácticamente una asociación, pero se diferencian por la notación que se utiliza en UML y su funcionalidad dentro del código





Para que se de la agregacion se debe de tener un constructor vacio. Ya que en este caso una tienda puede existir sin ningun cliente para ser creada.

Código Java

```
1 public class Tienda {
2     private Cliente[] clientes = new Clientes[3];
3     private int numClientes = 0;
4
5     public Tienda(){}
6
7     public void addCliente(Cliente cliente){
8         clientes[numClientes] = cliente;
9         numClientes++;
10    }
11 }
```

Código Java

```
1 public class ProgramaMain {
2     public static void main(String[] args){
3         Tienda tienda = new Tienda();
4         tienda.addCliente(new Cliente("Mauricio"));
5         tienda.addCliente(new Cliente("Frey"));
6     }
7 }
```

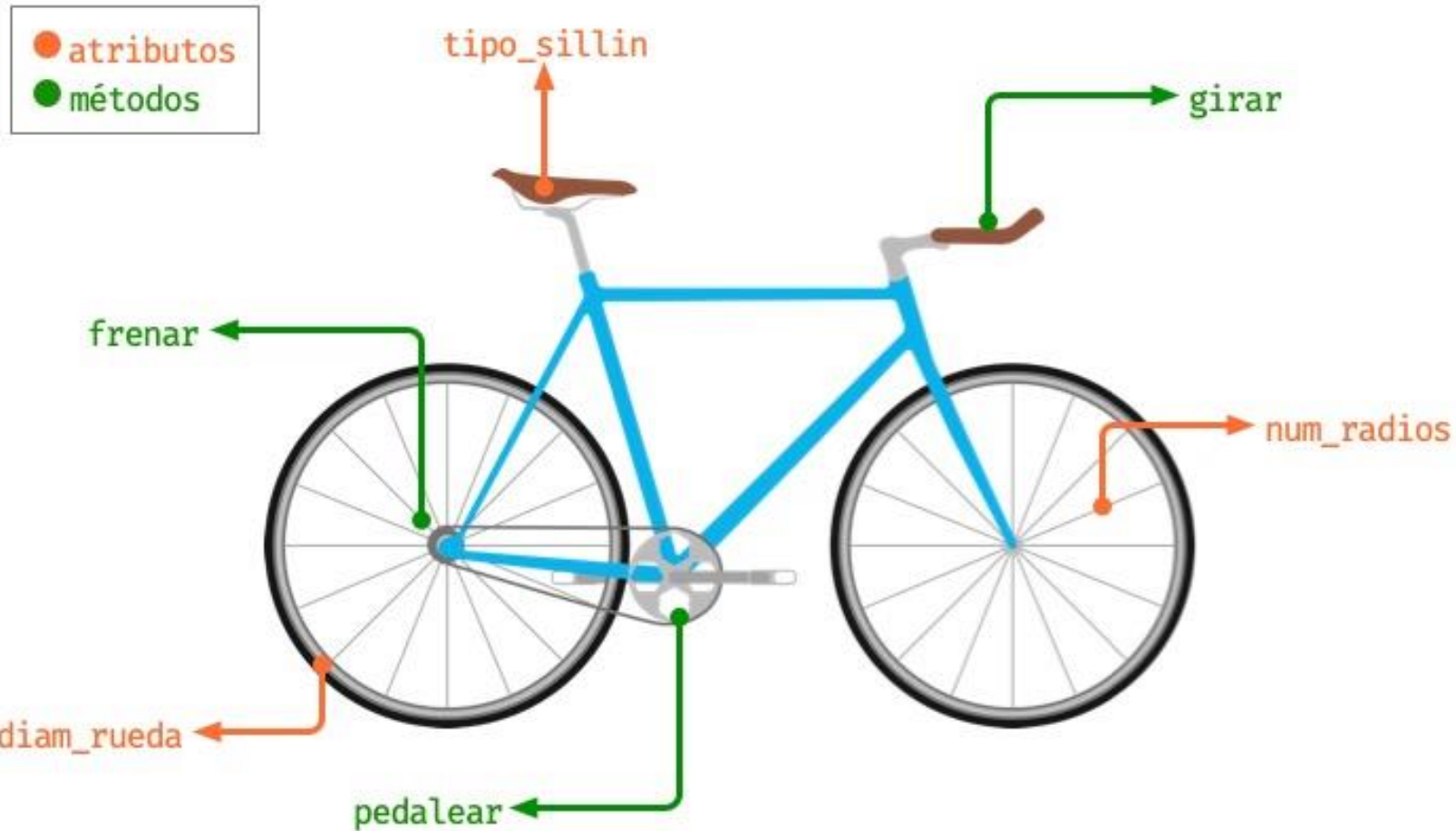
W W W . M A K A I A . O R G

Carrera 43 A # 34 - 155. Almacentro. Torre Norte. Oficina 701
Medellín (Antioquia), Colombia



W W W . M A K A I A . O R G

¿Que es la composicion.?



W W W . M A K A I A . O R G

Carrera 43 A # 34 - 155. Almacentro. Torre Norte. Oficina 701
Medellín (Antioquia), Colombia

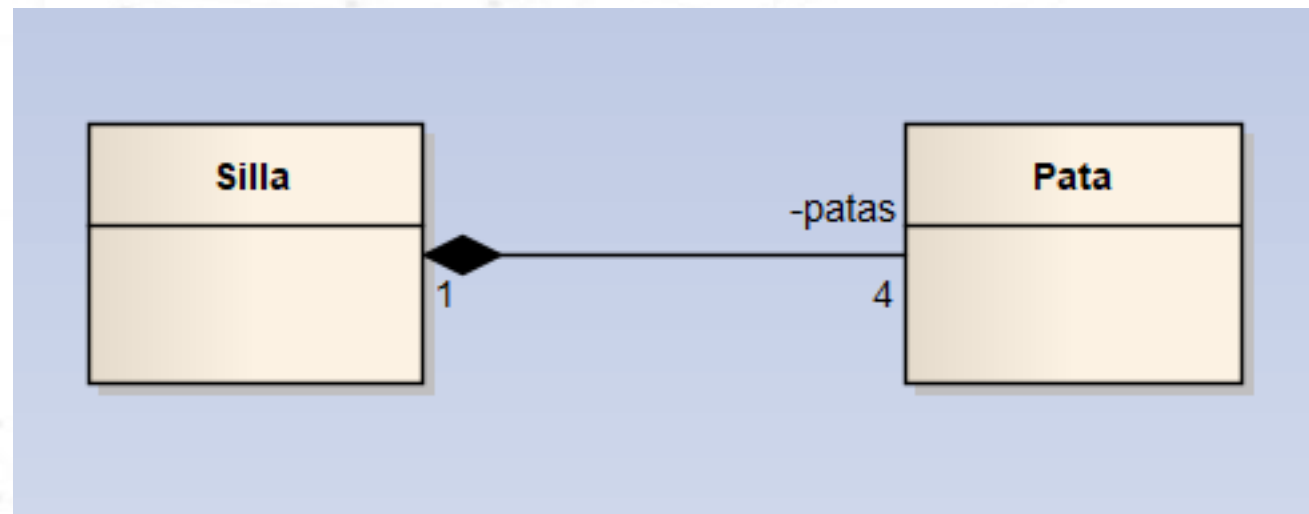


W W W . M A K A I A . O R G



composición

En este ejemplo podemos ver, que una silla para poder existir y ser creada necesita tener patas, y en este caso 4 patas.



composición

Código Java

```
1 public class Silla {  
2     private Pata[] patas;  
3     private int numPatas = 0;  
4  
5     public Silla(){  
6         patas = new Pata[4];  
7     }  
8  
9     public void agregarPata(String color, float peso){  
10         patas[numPatas] = new Pata(color, peso);  
11         numPatas++;  
12     }  
13 }
```

Código Java

```
1 public class Pata {  
2     private String color;  
3     private float peso;  
4  
5     public Pata(String color, float peso){  
6         this.color = color;  
7         this.peso = peso;  
8     }  
9 }
```

W W W . M A K A I A . O R G

Carrera 43 A # 34 - 155. Almacentro. Torre Norte. Oficina 701
Medellín (Antioquia), Colombia



W W W . M A K A I A . O R G

composición

Código Java

```
1 public class Silla {  
2     private Pata[] patas;  
3     private int numPatas = 0;  
4  
5     public Silla(){  
6         patas = new Pata[4];  
7     }  
8  
9     public void agregarPata(String color, float peso){  
10         patas[numPatas] = new Pata(color, peso);  
11         numPatas++;  
12     }  
13 }
```

Código Java

```
1 public class Pata {  
2     private String color;  
3     private float peso;  
4  
5     public Pata(String color, float peso){  
6         this.color = color;  
7         this.peso = peso;  
8     }  
9 }
```

W W W . M A K A I A . O R G

Carrera 43 A # 34 - 155. Almacentro. Torre Norte. Oficina 701
Medellín (Antioquia), Colombia



W W W . M A K A I A . O R G

Cardinalidad.

Especificacion	Cardinalidad
0..1	Cero o una vez
1	Una unica vez.
*	De cero a varias veces.
1..*	De una a varias veces.

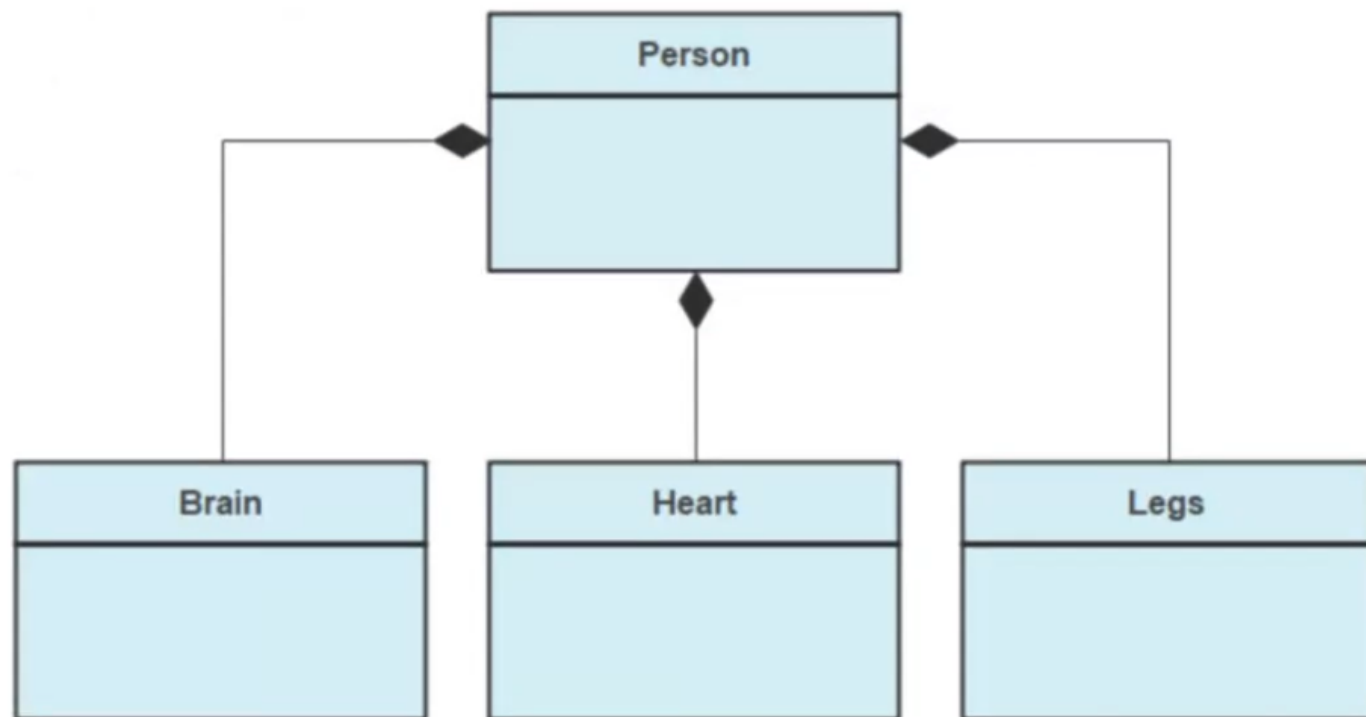


W W W . M A K A I A . O R G

Carrera 43 A # 34 - 155. Almacentro. Torre Norte. Oficina 701
Medellín (Antioquia), Colombia



W W W . M A K A I A . O R G





Fuentes

1. [https://www.seas.es/blog/informatica/agregacion-vs-composicion-en-diagramas-de-clases-uml/#:~:text=La%20agregaci%C3%B3n%20es%20un%20tipo,varias%20asociaciones%20de%20agregaci%C3%B3n%20distintas\).](https://www.seas.es/blog/informatica/agregacion-vs-composicion-en-diagramas-de-clases-uml/#:~:text=La%20agregaci%C3%B3n%20es%20un%20tipo,varias%20asociaciones%20de%20agregaci%C3%B3n%20distintas).)
2. <https://joanpaon.wordpress.com/2013/06/06/uml-diagramas-de-clases-relacion/>

W W W . M A K A I A . O R G

Carrera 43 A # 34 - 155. Almacentro. Torre Norte. Oficina 701
Medellín (Antioquia), Colombia



W W W . M A K A I A . O R G



■ WWW.MAKAIA.ORG

Info: comunicaciones@makaia.org

Corporación MAKAI A

Medellín, Colombia

Carrera 43A – 34-155. Almacentro

Torre Norte, Oficina 701

Teléfono: (+574) 448 03 74

Móvil: (+57) 320 761 01 76

