

LLAMADAS A MÉTODOS DENTRO DE LA MISMA CLASE O ENTRE CLASE PADRE Y CLASE HIJA.

Cuando quieres llamar a un método dentro de la misma clase en Java, se utiliza el nombre del método, seguido de paréntesis:

```
nombreDelMetodo();
```

Lo vemos con un ejemplo:

```
public class MiClase {  
    public void metodo1() {  
        System.out.println("Este es el método 1");  
        metodo2();  
    }  
  
    public void metodo2() {  
        System.out.println("Este es el método 2");  
    }  
}
```

En este ejemplo, podemos ver que el método `metodo1` llama al método `metodo2` dentro de la misma clase `MiClase`.

Si el método que quieres utilizar es heredado de otra clase, se puede llamar de la misma manera, utilizando el nombre del método y paréntesis:

```
public class Padre {  
    public void metodo1() {  
        System.out.println("Este es el método 1 de la clase Padre");  
    }  
}  
  
public class Hijo extends Padre {  
    public void metodo2() {  
        System.out.println("Este es el método 2 de la clase Hijo");  
        metodo1();  
    }  
}
```

En este ejemplo, podemos ver que la clase **Hijo** hereda del método **metodo1** de la clase **Padre**. El método **metodo2** de la clase **Hijo** llama al método **metodo1** de la clase **Padre** de la misma manera que se llamaría a un método dentro de la misma clase.

¿Qué representan las partículas **this** y **super**?

En Java, "**this**" se refiere a una referencia a la instancia actual de la clase. Se utiliza para hacer referencia a los miembros de la clase y para llamar a otros métodos de la misma clase.

Por otro lado, "**super**" es una palabra clave que se utiliza para hacer referencia a la clase padre. Se utiliza para llamar a métodos heredados de la clase padre, para acceder a los atributos de la clase padre, y para llamar al constructor de la clase padre.

En el ejemplo anterior, la llamada que hace el hijo a **metodo1()** del padre, podría haberse hecho también como **this.metodo1()** y como **super.metodo1()**, puesto que todo apunta al mismo método.