* Containment

Containment means the use of an object of a class as a member of another class

Containment or Has-A relationship or Association can be divided into aggregation and composition.

A screenshot of a computer

Description automatically generated with low confidence

A composition reference class cannot exist if the container class is destroyed.

A screenshot of a computer code

Description automatically generated with medium confidence

A picture containing text, screenshot, font, algebra

Description automatically generated



A picture containing text, screenshot, font, document

Description automatically generated

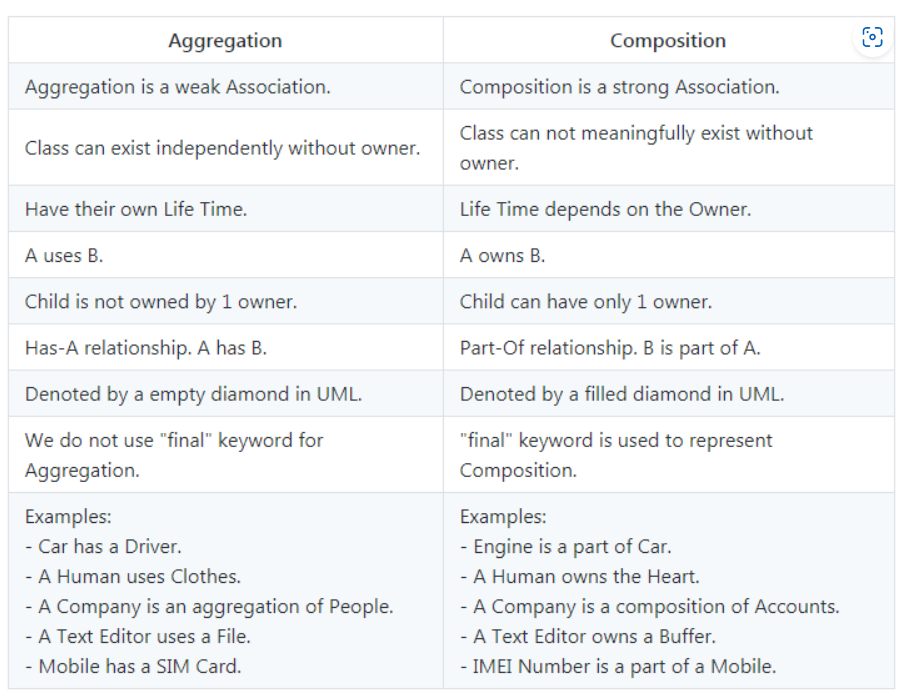
A picture containing text, line, font, screenshot

Description automatically generated

A screenshot of a computer program

Description automatically generated with medium confidence



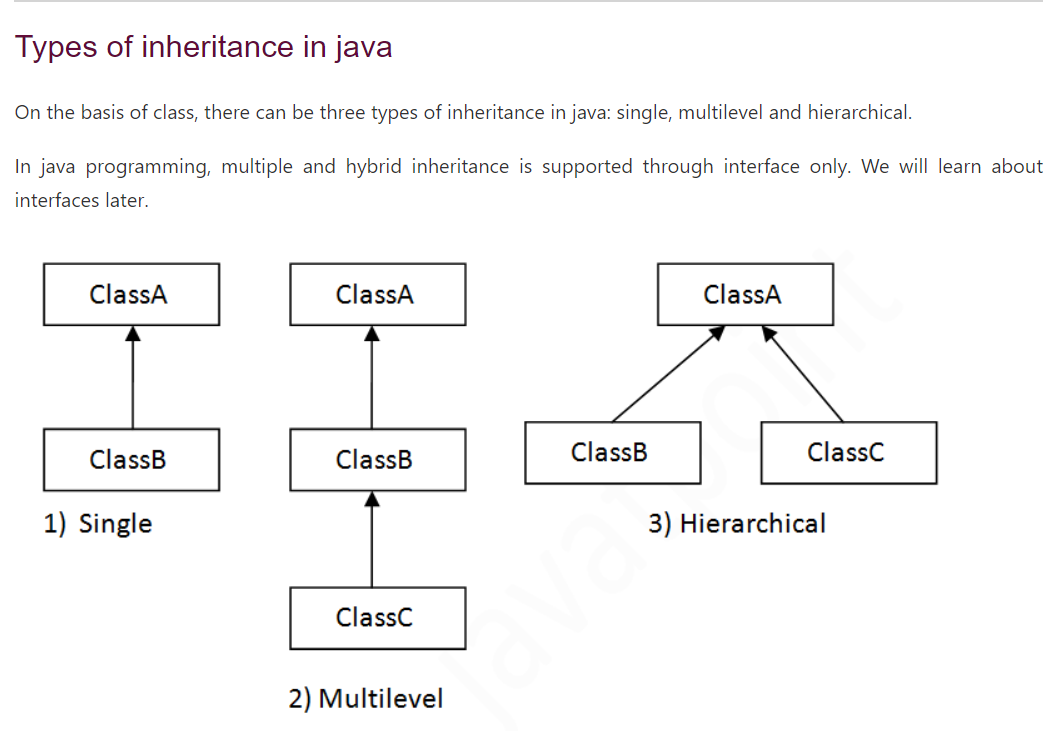


[Understanding Association, Aggregation, and Composition in Java - Coding Ninjas CodeStudio](https://www.codingninjas.com/codestudio/library/understanding-association-aggregation-and-composition-in-java)Inheritance

In Java, it is possible to inherit attributes and methods from one class to another. We group the "inheritance concept" into two categories:

* subclass (child) - the class that inherits from another class
* superclass (parent) - the class being inherited from

To inherit from a class, use the extends keyword.



* Super

The super keyword refers to superclass (parent) objects.

It is used to call superclass methods, and to access the superclass constructor.

The most common use of the super keyword is to eliminate the confusion between super classes and subclasses that have methods with the same name.

* Protected keyword

The protected keyword in Java refers to one of its access modifiers.

The methods or data members declared as protected can be accessed from

* Within the same class.
* Subclasses of the same packages.
* Different classes of the same packages.
* Subclasses of different packages.
* Method Overriding

If subclass (child class) has the same method as declared in the parent class, it is known as **method overriding in Java**.

Rules for Java Method Overriding

* The method must have the same name as in the parent class
* The method must have the same parameter as in the parent class.
* There must be an IS-A relationship (inheritance).

