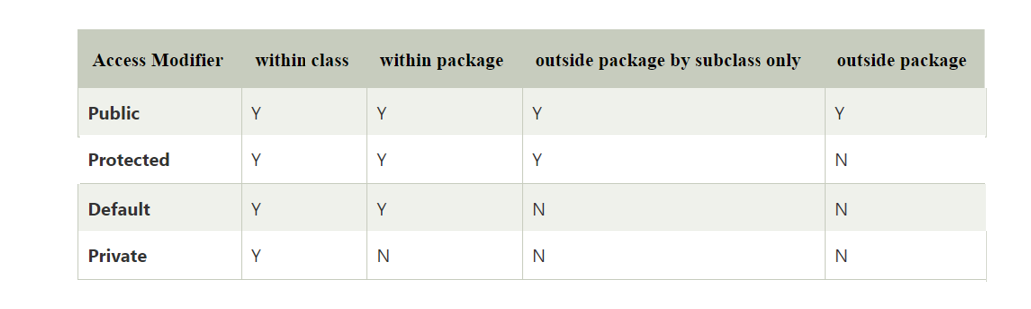
Access modifiers :

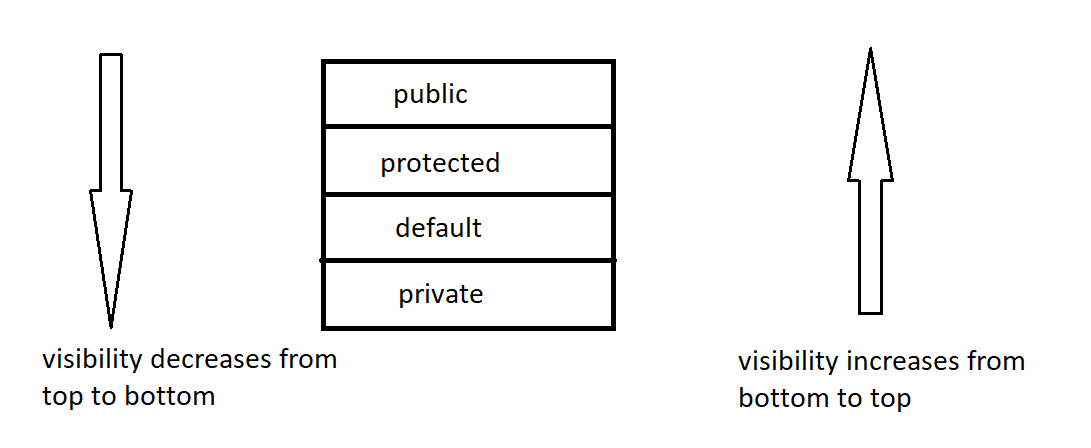
Access modifiers can be applied on class , variable ( can be used only on static variables , instance variables . Local variables cannot use access variables , since their scope only within their methods (or) block (or) loops (or) conditional statements ) constructor and method.

They cannot be applied on static block and java block.



Note : we cannot declare a top -level class as private (or ) protected .

// work on this after packages .



Note :

1. Methods which are inherited from parent , and child is using that methods without any change they are called as inherited methods.
2. Methods which are inherited from parent . and child is making some changes to that method they are called as overriding methods.
3. Methods which are present in the child class , but not present in parent class is called specialized methods.

Overloading : when two or more methods in same class having same name but different parameters is called overloading.

Overriding : when method signature (name and parameters) present in parent class and child class are same then it is called Overriding.

Note : overloading can be done in same class , it is achieved in inheritance also.

Overriding cannot be done in same class , it throws compile time error. It is achieved only during inheritance.

Eg: Inheritance\_Method\_Types

// go through the code

// practice uml for better understanding . draw diagram parent Object class and other child predefined classes .

// make a mini project on loan

Note : All the inbuilt classes have methods they have access specifiers public .

Rules to override method :

1. We cannot decrease the visibility of overridden method. But we can increase the visibility of overridden method.

Note : overriding method is present in parent class.

Overridden method is present in child class.

1. Return type of overriding method and overridden method must be same.

Note : For methods the return types can be class type also. if return type of method is class ,the method should return object of that return type class .

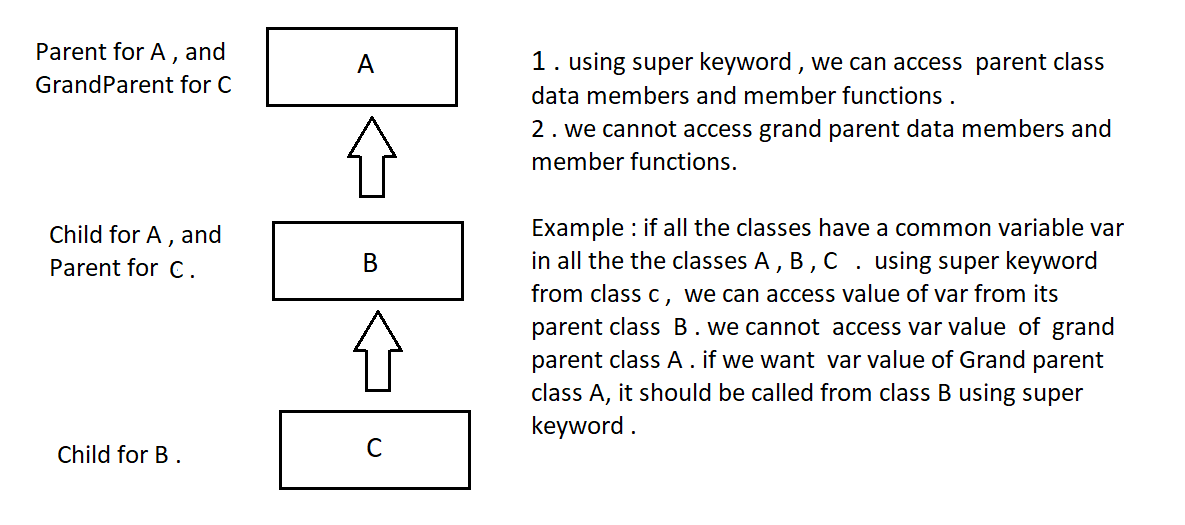
1. If there exist is-a relationship between returns types of overriding method and overridden method , then return types of overriding method and overridden method can be different. This is called co-variant return type.

Eg: Co\_Variant

1. Parameters of overridden method must be same as that of overriding method ,else it will treat as specialized method (considered as overloading ) .

Eg: Override\_Rule\_4

Note : super keyword is used to call parent class instance variable, if name of both variables is same.



Eg: Super\_Keyword\_Inheritance

Note: this keyword is used to access current class data members and member functions . while super keyword is used to access parent class data members and member functions .

final keyword

1. It can be applied to class , variable , method .
2. If a class is made inherited it cannot be inherited.
3. If a method is made final it can be inherited , but we cannot override it.
4. Final variable acts as constant we cannot change that value. it participates in inheritance ,but value can’t be changed.

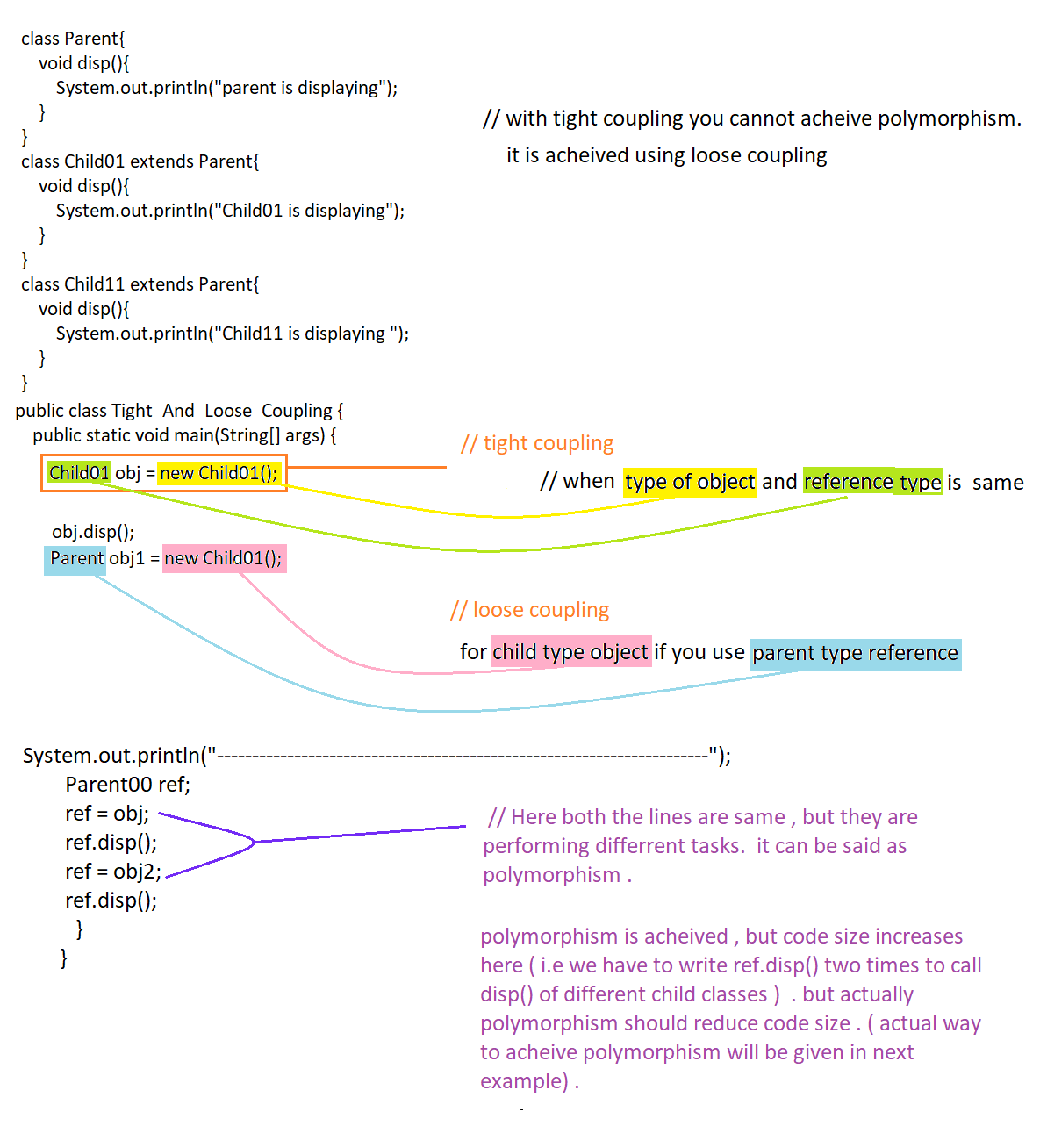
Polymorphism is divided into two types.

1. Compile time polymorphism achieved by method overloading .
2. Runtime time polymorphism (or ) true polymorphism achieved by method overriding.

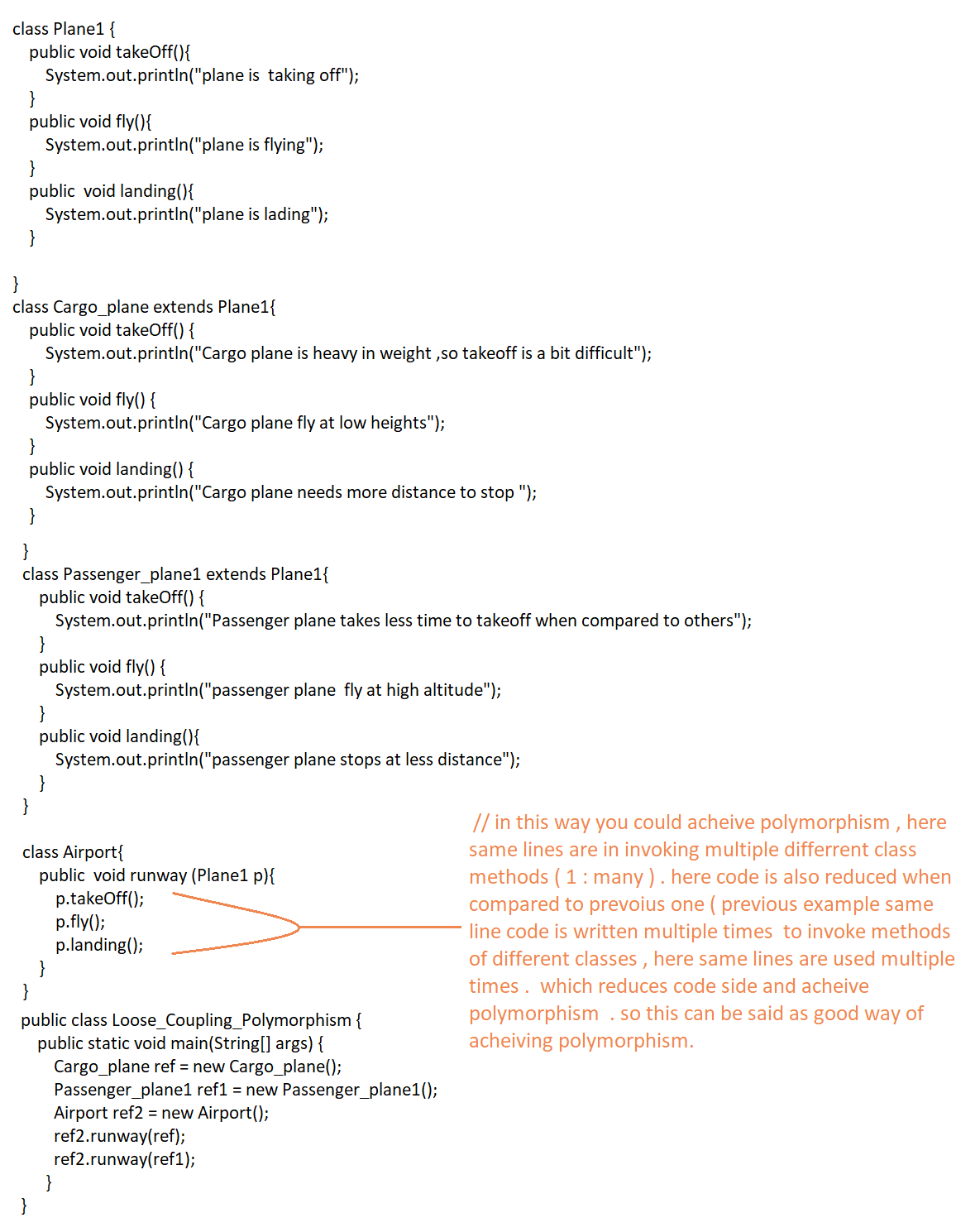
Advantages of runtime polymorphism :

1. Flexibility
2. Code size reduces

Eg: Tight\_And\_Loose\_Coupling



Eg: Loose\_Coupling\_Polymorphism



Eg: Downcasting\_Upcasting.

