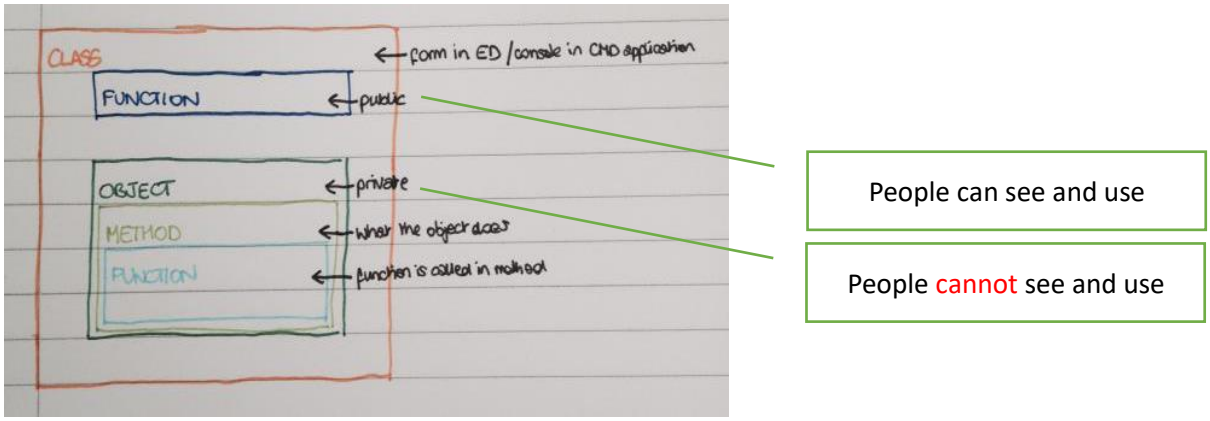
1756755 Kuanyow Chen

object oriented programming and its limitations

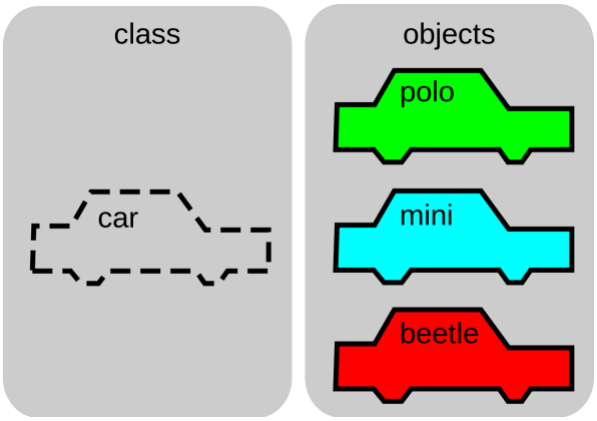
Object oriented programming (or it is called OOP) is one of language model in programming. OOP provides encapsulation, inheritance, and polymorphism. This is the reason why this is the most popular method.

Encapsulation is the first benefit if using OOP in programming. This advantage is that encapsulation provides simple methods with users. They cannot or might not know the content but all of them could use these methods. For example, all of us know how to use the television, but how many people know the structure of it? We can just use it and do not need to understand it. In programming, the engineers finished the class and allow other engineers could use these public methods and attributes. Moreover, encapsulation can be an outstand protection of codes. This means the codes in the class can be prevent others from stealing. About private methods and attributes cannot be used. This is also counting a kind of protection.



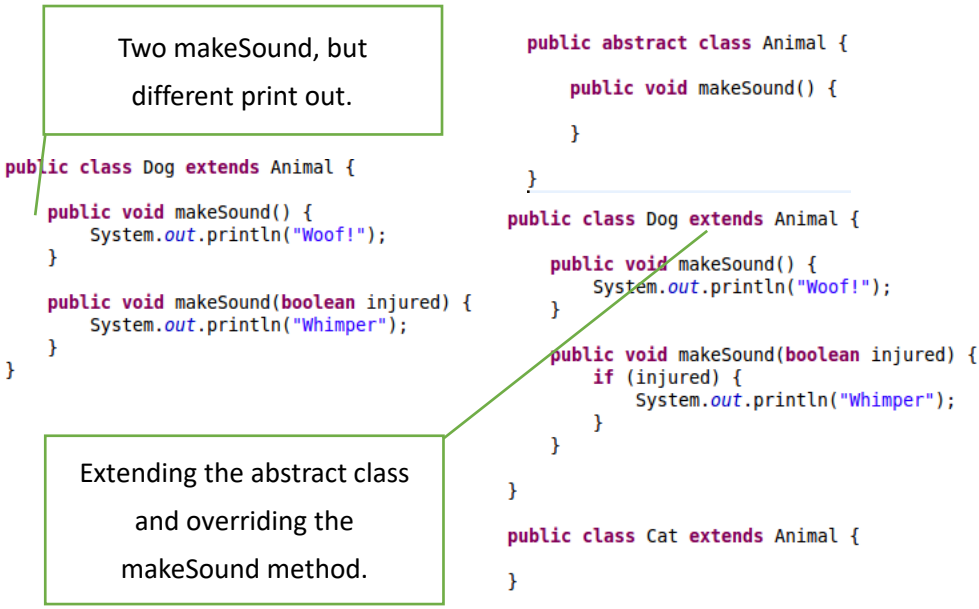
(source: http://www.vbforums.com/showthread.php?797697-Object-Oriented-Programming-Software-Structures)

Furthermore, inheritance is the second advantage when using OOP in programming. The subclass, or called derived class, could inherit the class, or called base class. Even the subclass could override the method from class, but it is unnecessary. This depends on the base class is declared as virtual or abstract. The functions from virtual class could be overridden by the subclass. However, the functions from abstract class must be overridden by the subclass. Specially, sealed functions and class cannot be inherited and overridden. This could simplify the programming because the similar functions can be written into a class. For instance, all of cars, including polo, mini, and beetle, can run. So car class is a base class, and others will be subclass to inherit the fun function from car class.



(source: https://brilliant.org/wiki/classes-oop/)

In addition of encapsulation and inheritance, polymorphism is the last main benefits if using OOP in programming. This means even the same name of function, it could provide different function with different condition. Polymorphism includes overloading and overriding. Overloading means multi-functions have the same name, but they could run different result that depends on different inputs. Additionally, overriding means subclass can override the method from the base class.



Overloading and overriding example
(source: https://www.java-made-easy.com/polymorphism-in-java.html)

Although OOP provides these three benefits, there are several questions if using the OOP. For example, it might take much more time to design the programming. Therefore, this might increase the time cost. Furthermore, a difficult problem would be more difficult when using encapsulation too much. Because the difficult problem might be divided into several method. Therefore, how to use OOP suitably is critical task while programming.