***Methods*** *block of code that performs a task*

***Returning type*** *mayo or may no return anything*

***Method parameters***

***Method overloading***

***Object entity that has behaviour***

***Class***

***Object***

***Constructor***

***Modifiers (), in order to import packages click “alt + Enter”***

* ***Public ()***
* ***Private*** *(Can only be accessed only in the same class)*
* ***Protected*** *(Can be accessed in the same classes or sub classes)*
* ***Default*** *(can access only by classes within the same package)*

***Encapsulation*** *(in order to modify private data create getters and setters)*

***Inheritance*** *(Use extends keywork in the sub class and in the constructor use the super keyword in order to access the data from the parent class)*

***This keyword*** *(reference variable that represents the current instance of the class in which it is used)*

***Method overriding***  *(“@Override” followed by the method and change the implementation)[same method name, return type, parameter]*

***Polymorphism (****allows objects from different classes to be treated as object of a common parent class****)***

***Abstraction*** *(hide unnecessary details) [in the class declare it as abstract “public abstract class” or methods]*

* *You can not extend more the class (you need to extend the subclasses)*
* *Abstract methods do not define anybody [you force the class to implement the method]*

***Interfaces (****contract or a set of rules that a class has to follow****)****[you need to use “implements”]*

*Note: multiple inheritance in Java is achieved by interfaces*