

## INHERITENCE

### \*Inheritences Mode :

Public : Public and Protected Members Will save the same aspect

Protected : Both Protected and Public will become Protected and can be directly accessed in the derived class

Private : Both Protected and Public will become Private

Note====> Private attributes can not be directly accessed in the derived class

### \*Constructors

By Default

Creating child object with or without parameter ( means using default or parametered constructor instead) will automatically generate base class default constructor

we can edit a little bit this default behavior

by calling base class param constructor inside the child **def** or **param** const

in all cases : base class cons will be executed before :

big note : in case there is an attribute in a class with a type of any other class (base or whatever):

the constructor of type(attributes) will be executed before the main class if there is not inheritance

:====> in case of inheritance : we adopt this sequences of events :

- 1) Calling of def or param base class constructor
- 2) Calling the attributes constructor (if they already exists ofc)
- 3) Executing the main object const(child constructor)

====> THIS sequences is always true whatever there is inheritences or not

