OOPS

(Object Oriented Programming System)

Programming Paradigms

Procedural   Object Oriented   Functional   -----
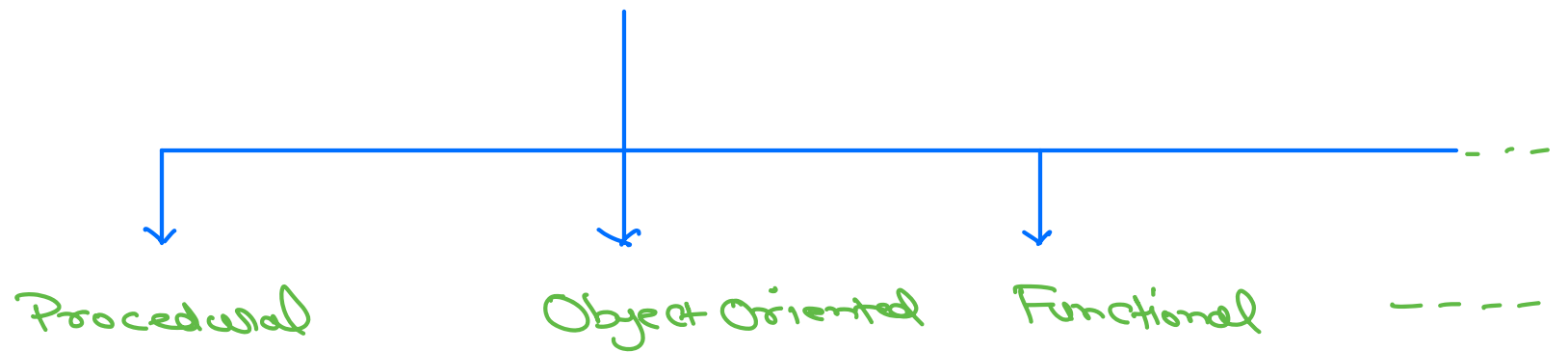
* Functions   * Class   *
* Data →

                          ↓
                    Objects/ instances

Class   Bank Account   (Blueprint)

Data / Properties   Behaviour / Functionalities

⤷ Acc-id   ⤷ check-balance
⤷ balance   ⤷ debit (withdraw
⤷ bank-name   ⤷ credit (deposit)
      "   "

4 pillars of OOP

* Encapsulation: Hiding data
  - All data must be stored in the instance Itself
  - Modifications can be done using methods

We can use Access specifier
  - Public (default)
  - Protected
  - Private

Python uses name-mangling:
  __Classname__private_var
to protect accidental updates to private var

* Polymorphism: Same entity multiple behaviour

  Ex ⟹ +

* Abstraction

* Inheritance

Initializer

* Initialize instance variable with Values

* It automatically gets invoked/called when Instance is created

Constructor ! = Initializer

Constructor	Initializer
⊙ Allocates memory	⊙ initialize Vars
⊙ initialize Vars	

Q) What is self

⊙ The very first Parameter stores the Object itself as Value