

07 mai 2025

1-) Orienté-objet / classe-objet python.

2-) Concept de données.

3-

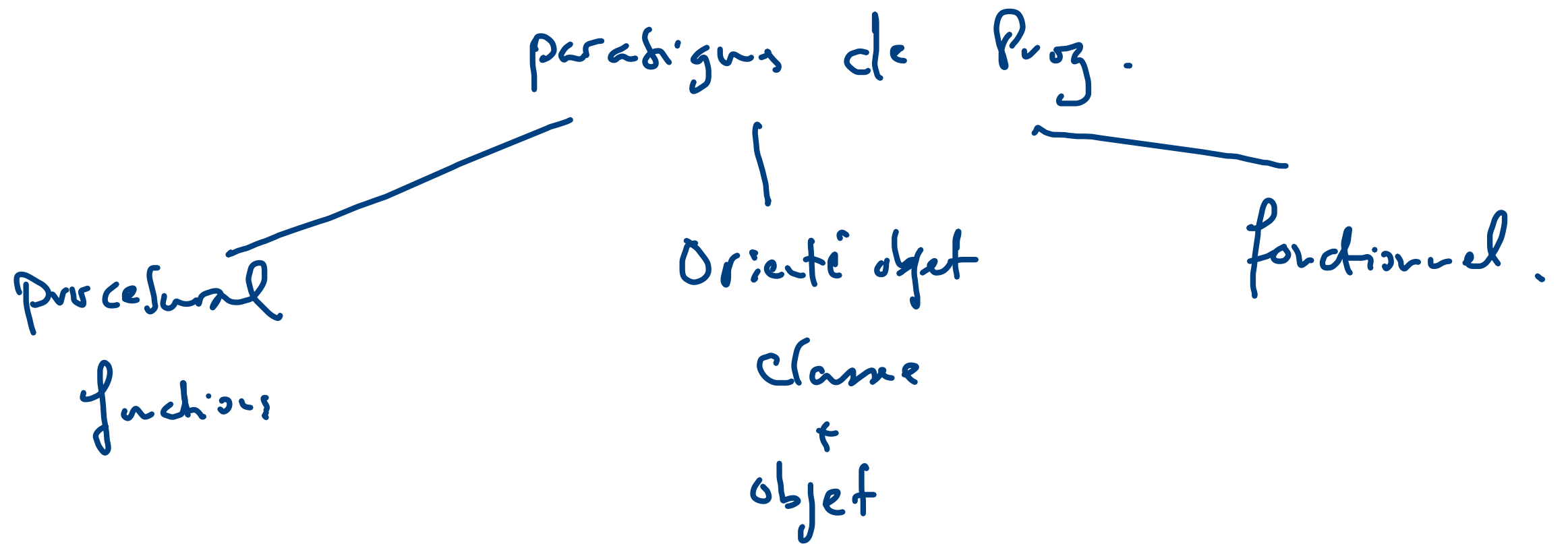
3-) Introduction à la collecte par fichiers

Programm : Syntaxe de Base.

- Variable.
- Ecriture & Lecture.
- Test
- Boucle.

Sequences: list, tuple, set, dict.

fonctions.



Data: Collecte
- fonctions

Modules pre - définir; fonctions
 classe/objet

1. Objectif :
- Acquérir des compétences en Python
 - Transmettre des compétences en Python.

2. Identification

- Etudiant
- Professeur

Thinking in objects

3. Modélisation

- Attributs
- Méthodes (fct's)

Att.

Méth.

Etudiant	
nom prenom <i>note.Final</i>	0.0 ✓
connecter_br()	5
ecouter()	10
faire_ex()	20

Professeur	
nom prenom <i>indiscipliné</i>	0.0 ✓
connecter_br()	
enseigner()	10
donner_ex()	30

Moule

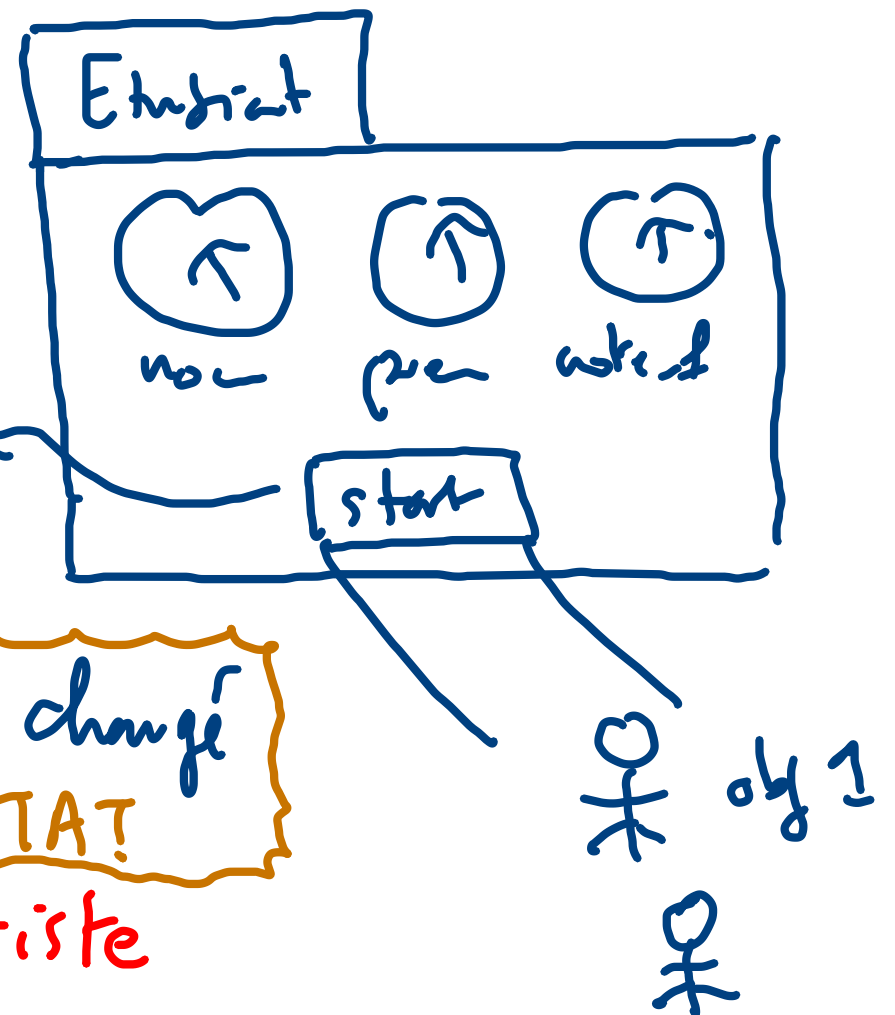
Classe Domaine

4. Instantiation (Création des objets)

obj E = Eudiant("Flouclier",
"Alain",
0.0)

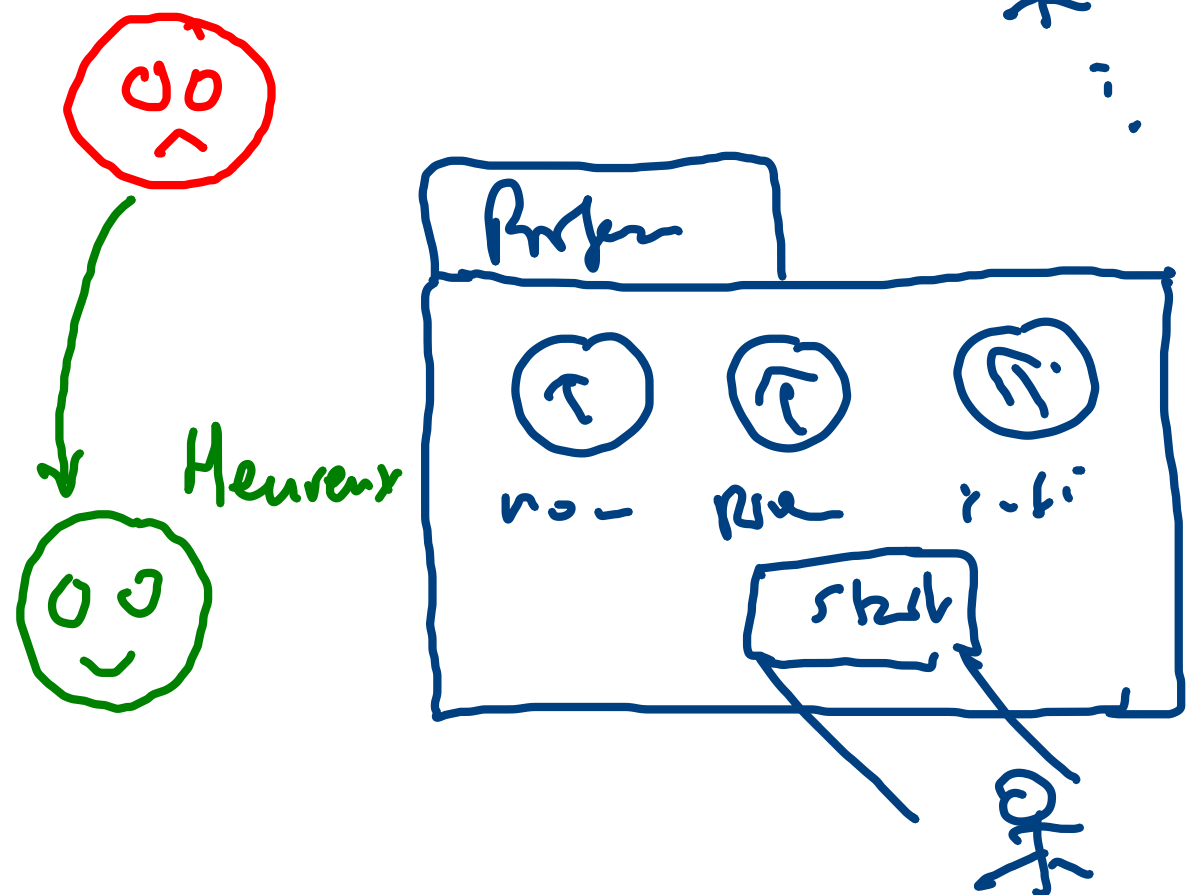
obj P = Professeur("Flouclier",
"Annie",
0.0)

Constructeur



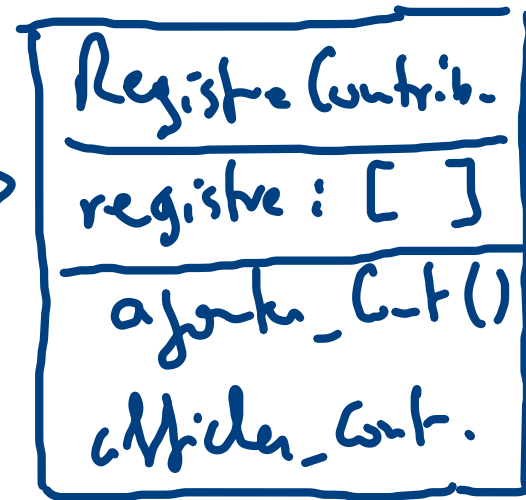
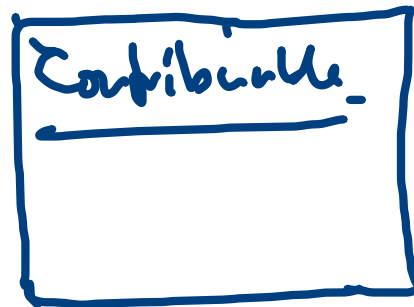
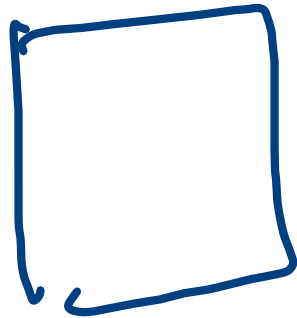
5-) COLLABORATION

obj P. connect_bon("Host")
obj E. connecteur("Eudiant")
obj P. enseigner() &
obj E. ecounter()
obj P. donner_examen()
obj E. faire_examen()

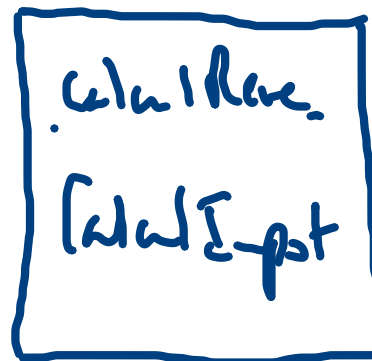


Pratigone Contribution.

main



utils
module



* * * * *

1. Calculer I-pot

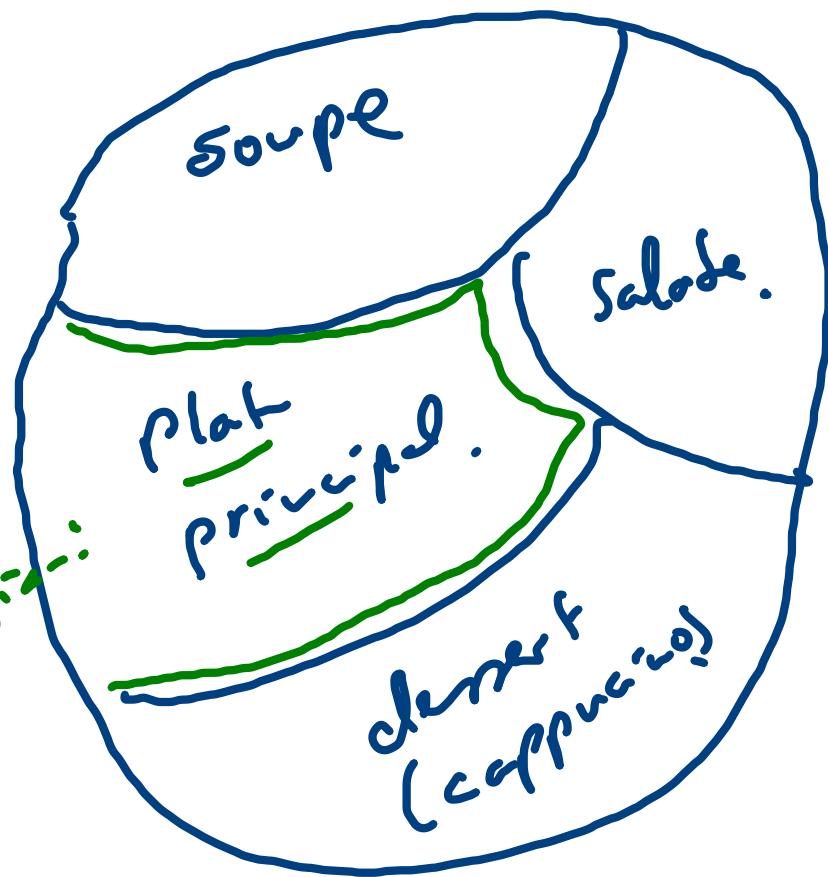
2. Afficher Contribution

3. Quitter

... -

Souper.

(Exemple
pour illustrer)



Client A

objet
Responsabilité

(Exemple
pour illustrer)



Client B