Rapport Python

Les classes sont des objets

A chaque fois qu’on appelle une classe il va créer de nouvelle instance

Et il ya une relation entre la class et son instance : c’est une relation d’héritage

L’instance va hériter tous les attributs de la class

Une class a son propre espace de nommage

Une instance a son propre espace de nommage

La recherche d’attribut dans un espace de nommage s’appelé arbre de nommage

Accéder à l’espace de nommage avec \_\_dict\_\_

Vars () permet d’accéder au dictionnaire de l’espace de nommage

Les class et les instances sont des Object mutable

Méthode spéciale nous permet de créer nos propre class

On peut implémenter plusieurs opérations sur notre class

La méthode spéciale commence par \_\_ et se termine par \_\_

Def \_\_len\_\_(self) :

Pour vérifier un test d’appartenance o écrit

Def \_\_contrains\_\_(self, mots sur lequel on fait le tes)

Il va retourner un booléen vrai si mot se trouve dedans et false s’il ne se trouve pas

Arbre d’héritage

Une class peut hériter d’un autre class

Pour montrer qu’une class hérité un Aure class on le met () pendant la créations de la class

Isinstance () permet de vérifier si votre Object directement d’une class ou du superclasse

Quand on hérite une class on hérite aussi tous ses méthodes

Surcharge de méthodes