Programmation orientée objet avec Python

Mars 2021

académie Versailles
GRETA
DES HAUTS-DE-SEINE
FORMEZ-VOUS À DEMAIN
RÉGION ACADÉMIQUE ÎLE-DE-FRANCE
LUDIKSCIENCES

Exercice Nombre premiers

Trouver les nombres premiers inférieur à N

On rappelle un nombre est premier s'il a deux diviseurs : 1 et lui-même.

Selon cette définition, 1 n'est pas premier (un seul diviseur : 1)

La méthode du crible d'Eratosthène consiste à éliminer de l'espace de recherche les nombres qui sont multiples des nombres premiers déjà trouvés.

Afficher les nombres premiers trouvés dans l'ordre croissant

LUDIKSCIENCES

Exercice PGCD itératif

Ecrire la fonction PGCD en évitant l'algorithme récursif déjà visité et en utilisant une boucle

LUDIKSCIENCES

Exercice Vecteurs

Lire un vecteur de dimension N

Afficher un vecteur

Faire la somme de 2 vecteurs

Multiplication d'un vecteur par un scalaire (un nombre flottant)

Faire le produit scalaire de 2 vecteurs

LUDIKSCIENCES

Les mots-clés ajoutés

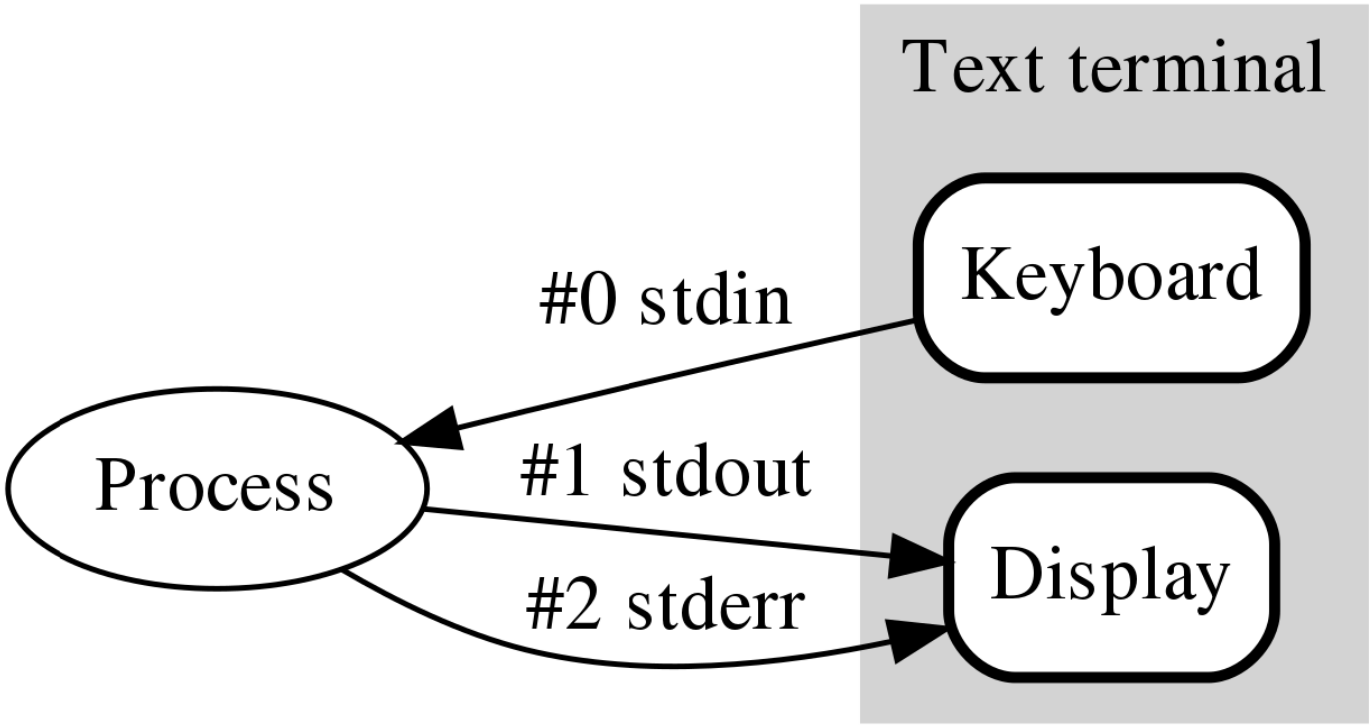
False	None	True	and	as	assert
break	class	continue	def	del	elif
else	except	finally	for	from	global
if	import	in	is	lambda	nonlocal
not	or	pass	raise	return	try
while	with	yield			

LUDIKSCIENCES

Entrées/sorties

print

format



https://fr.wikipedia.org/wiki/Flux_standard

LUDIKSCIENCES

Print

En Python3, print devient une fonction

```
print (a, b, c, sep=",\t", end=".\n")
```

sep et end sont des chaînes de caractères

sep

- sépare l'impression de deux arguments consécutifs
- par défaut, chaîne vide

end

- s'ajoute à la fin de la chaîne
- par défaut, fin de ligne/retour

LUDIKSCIENCES

Format

Formatage historique

```
>>> nom = 'Georges'; metier = 'professeur'
>>> 'Salut %s le %s' % (nom,metier)
'Salut Georges le professeur'
```

A partir de Python 2.6

str.format()

```
print ('Salut {}'.format('Pierre'))

print ('Salut {1}'.format('Pierre','Jean'))

print ('Salut {nom}'.format(nom='Pierre'))
```

A partir de Python 3.6

chaine introduite par f ou F

f-string

```
>>> nom = 'Georges'
>>> f'Salut {nom}'
'Salut Georges'
```

%5.3f

format pour un nombre flottant à 5 chiffres, dont 3 après le point

```
>>> import math
>>> f'Pi vaut {math.pi:.5f}'
```

LUDIKSCIENCES

Fichiers

Ouverture

- Mode r,w,a,x : désigne l'ouverture
- +: mixer lecture/écriture
- Type de fichier : t=texte , b=binaire

```
f = open ('myfile')
data = f.read()
f.close()
```

Garde

- Fermeture auto à la fin du with

```
with open ('myfile') as f:
    data = f.read()
…
```

Lecture des fichiers texte

```
line = f.readline()
```

- ligne par ligne

```
for line in f:
…
```

- toutes les lignes dans une liste

```
lines = f.readlines()
```

Lecture des fichiers binaires

```
f.seek(offset, pos)
l = f.read(nb_bytes)
```

LUDIKSCIENCES

Modes

r Read

w Write

a Append

fichier texte - lignes qui se terminent par \n (UNIX) ou \r\n (Windows) (Ascii 13 CR – Ascii 10 LF)

b Fichier binaire - pas de notion prédéterminée d'enregistrement.

En général, un programme qui s'arrête sans fermer un fichier ouvert en écriture fait perdre le contenu de ce fichier.

- Autant que possible ouvrir en "r" ou en "a" protège le contenu des fichiers.

LUDIKSCIENCES

LUDIKSCIENCES

LUDIKSCIENCES

Exception

Les exceptions prédéfinies ont des noms qui se terminent par Error

```
# Quelques exemples choisis
ArithmeticError
ImportError
NameError
OverflowError
RecursionError
ZeroDivisionError
```

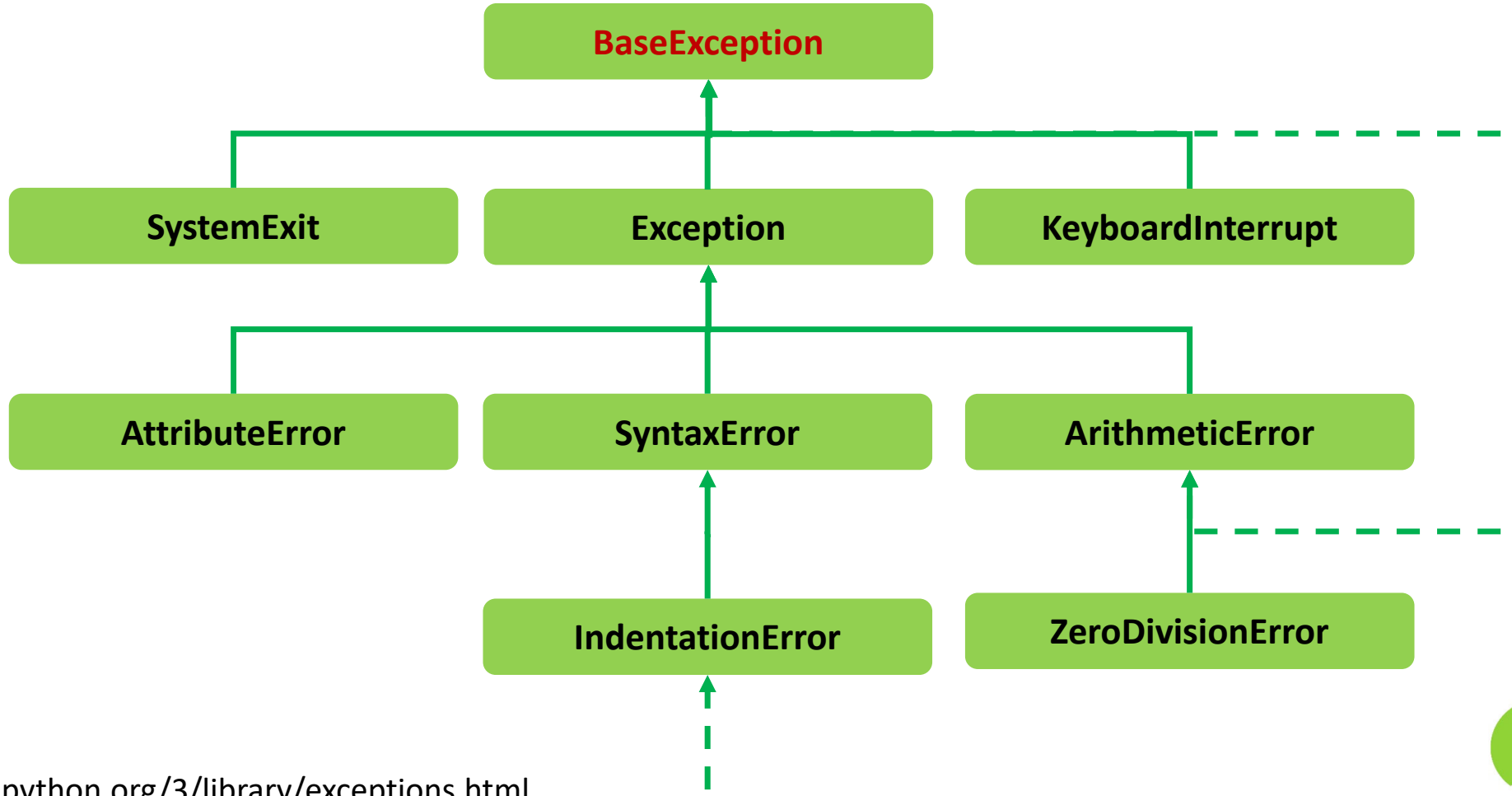
Imprévues : arrêt brutal avec l'affichage des fonctions traversées,

Prévues : comportement d'arrêt en douceur, voire possibilité de continuer selon ce que le programmeur a conçu.

programmation défensive

LUDIKSCIENCES

Exception



https://docs.python.org/3/library/exceptions.html

LUDIKSCIENCES

Try

Des erreurs se produisent à l'exécution

On protège l'exécution d'un segment de code sensible par la clause try (obligatoire pour gérer except)

La forme complète de try

as permet de collecter l'objet exception

- intéressant pour faire figurer les détails dans un fichier .log

```
#section de code à protéger  action
…
except ZeroDivisionError :
#ce que l'on fait,
#si l'erreur se produit  action
```

```
try:
…
except xxError as ex:
…
except yyError as ey:
…  else:
…# Ce que l'on fait si pas d'erreur
finally:
…#Ce que l'on fait à la fin
 # dans tous les cas
```

LUDIKSCIENCES

Raise

lever une exception

interrompre le cours du traitement

pour se rattraper éventuellement sur une clause except

```
if x <0:
raise Exception('Valeurs négatives
non permises. x={}'.format(x))
```

LUDIKSCIENCES

Assert

Vérifier qu'une condition est remplie

```
assert condition, message
```

si ce n'est pas le cas, une exception AssertionError est levée

le traitement s'arrête

sauf si l'on est dans un cadre try-except.

```
Try:
    #s'assurer que a==b
    #avant de continuer
    assert a==b, "les objets sont
    différents"
    #suite du traitement


except AssertionError as e:
    logging.log(e)
except Exception as e1:
    #traitement de toutes les autres
    #exceptions
```

selon les options de compilation (mode debug ou pas), les assert sont pris en compte ou ignorés

LUDIKSCIENCES

LUDIKSCIENCES

Exercice Dictionnaire

On dispose d'un texte écrit en Français (testinput.txt)

Substituer les marques de ponctuation par des espaces

Compter les mots

Compter le nombre de fois où chaque mot apparaît

Exporter l'histogramme au format CSV afin de l'exploiter avec un tableur

Prendre en compte l'échec possible à l'ouverture du fichier texte (un fichier de ce nom n'existe pas, ou bien on n'a pas le droit d'y accéder…) en gérant une exception

Donner les 10 mots les plus fréquents

Améliorer l'analyse en excluant les mots "vides" (peu de lettres, pronoms, prépositions, articles)

LUDIKSCIENCES