

Résumé des fonctions standards de python

Une petite note pour aider les dev python à avoir en mains un résumé de leurs fonctions les plus utilisées

FONCTIONS BUILT-IN (disponibles sans import)

Ce sont celles que tu peux utiliser directement :

Fonction	Description
<code>print()</code>	Affiche un texte
<code>len()</code>	Longueur d'un objet
<code>type()</code>	Type d'un objet
<code>int(), float(), str()</code>	Conversion de types
<code>range()</code>	Génère une séquence de nombres
<code>input()</code>	Lit une entrée utilisateur
<code>sum()</code>	Somme d'une liste

<code>max(), min()</code>	Valeur max / min
<code>abs()</code>	Valeur absolue
<code>sorted()</code>	Trie une séquence
<code>reversed()</code>	Inverse une séquence
<code>enumerate()</code>	Donne index + valeur dans une boucle
<code>zip()</code>	Regroupe plusieurs listes
<code>map(), filter(), reduce()</code>	Programmation fonctionnelle
<code>any(), all()</code>	Teste des booléens dans un itérable
<code>dir()</code>	Liste les attributs et méthodes d'un objet
<code>help()</code>	Affiche l'aide intégrée
<code>eval()</code>	Exécute une expression Python

math — Mathématiques

```
import math
```

Fonction	Description
<code>math.sqrt(x)</code>	Racine carrée
<code>math.pow(x, y)</code>	Puissance
<code>math.floor(x)</code>	Arrondi vers le bas
<code>math.ceil(x)</code>	Arrondi vers le haut
<code>math.pi, math.e</code>	Constantes célèbres
<code>math.sin(), math.cos()</code>	Trigonométrie

random — Aléatoire

```
import random
```

Fonction	Description
<code>random.random()</code>	Float entre 0 et 1
<code>random.randint(a, b)</code>	Entier aléatoire entre a et b

<code>random.choice(liste)</code>	Choix aléatoire dans une séquence
<code>random.shuffle(liste)</code>	Mélange une liste

functools — Fonctions avancées

```
from functools import cache, lru_cache, reduce
```

Fonction	Description
<code>@cache</code>	Mémoïsation automatique (depuis Python 3.9)
<code>@lru_cache</code>	Version + ancienne, avec limite de taille
<code>reduce(f, iterable)</code>	Applique une fonction cumulée

itertools — Itérateurs puissants

```
import itertools
```

Fonction	Description
<code>itertools.count()</code>	Compteur infini
<code>itertools.product()</code>	Produit cartésien (utile pour QCM, combinaisons)
<code>itertools.permutations()</code>	Permutations d'une liste
<code>itertools.combinations()</code>	Combinaisons d'une liste

statistics — Statistiques de base

python

CopyEdit

```
import statistics
```

Fonction	Description
<code>mean()</code>	Moyenne
<code>median()</code>	Médiane

<code>stdev()</code>	Écart type
----------------------	------------

heapq — File de priorité (tas)

```
import heapq
```

Fonction	Description
<code>heapify()</code>	Transforme une liste en min-heap
<code>heappush()</code>	Ajoute un élément
<code>heappop()</code>	Retire le plus petit élément

collections — Structures utiles

```
from collections import Counter, defaultdict, deque, namedtuple
```

Élément	Description
---------	-------------

<code>Counter()</code>	Compte les occurrences
<code>defaultdict()</code>	Dictionnaire avec valeur par défaut
<code>deque()</code>	File double
<code>namedtuple()</code>	Tuple nommé

os — Système d'exploitation

```
import os
```

Fonction	Description
<code>os.getcwd()</code>	Dossier courant
<code>os.listdir()</code>	Liste les fichiers d'un dossier
<code>os.path.join()</code>	Joint des chemins

pathlib — Manipulation de fichiers (moderne)

```
from pathlib import Path
```

Fonction / classe	Description
<code>Path()</code>	Représente un chemin
<code>.exists()</code>	Vérifie l'existence
<code>.read_text()</code>	Lire un fichier texte
<code>.write_text()</code>	Écrire dans un fichier

datetime — Dates et heures

```
import datetime
```

Fonction / classe	Description
<code>datetime.now()</code>	Date et heure actuelles
<code>timedelta()</code>	Durée entre deux dates
<code>date()</code>	Créer une date