

Les fonctions

Une fonction est un concept fondamental en programmation et en mathématiques. Elle représente une relation entre un ensemble de valeurs en entrée, appelées arguments, et un ensemble de valeurs en sortie, appelées résultats. Les fonctions permettent de regrouper un ensemble d'instructions ou de calculs qui peuvent être réutilisés à plusieurs endroits dans un programme, favorisant ainsi la modularité, la lisibilité et la réduction de la redondance de code.

En mathématiques, une fonction est définie comme une correspondance qui associe à chaque élément d'un ensemble, appelé domaine, un unique élément d'un autre ensemble, appelé codomaine ou ensemble d'arrivée. Les fonctions mathématiques sont généralement représentées par des expressions algébriques, des équations ou des graphiques.

En programmation, une fonction est une séquence d'instructions regroupées sous un nom spécifique et qui peut être invoquée (ou appelée) depuis d'autres parties du programme.

Les fonctions peuvent prendre des arguments en entrée, qui sont des valeurs fournies lors de l'appel de la fonction, et elles peuvent renvoyer un résultat en sortie. Les fonctions permettent d'encapsuler des portions de code dans des unités logiques et autonomes, facilitant ainsi la maintenance, la réutilisation et la compréhension du code. Une fonction est généralement définie par un en-tête qui spécifie son nom, ses paramètres d'entrée et son type de retour, ainsi qu'un corps qui contient les instructions à exécuter lorsque la fonction est appelée. Les paramètres d'entrée permettent de transmettre des valeurs à la fonction, et le type de retour indique le type de valeur renvoyé par la fonction.

Les fonctions peuvent être utilisées pour effectuer des opérations simples ou complexes, comme effectuer des calculs mathématiques, manipuler des chaînes de caractères, accéder à des données dans une base de données, interagir avec l'utilisateur, etc. Elles peuvent également être utilisées pour décomposer un problème complexe en sous-problèmes plus simples, en adoptant une approche de développement modulaire.

En plus de ces fonctionnalités de base, les fonctions peuvent avoir des caractéristiques supplémentaires, telles que la récursivité, où une fonction s'appelle elle-même, permettant ainsi de résoudre des problèmes itérativement, ou les fonctions anonymes, qui sont des fonctions sans nom pouvant être utilisées en tant qu'arguments ou retournées par d'autres fonctions. Les fonctions peuvent être organisées en modules ou en bibliothèques, ce qui permet de les regrouper selon leur utilisation ou leur domaine d'application, favorisant ainsi la modularité et la réutilisabilité du code.

Les langages de programmation offrent généralement un ensemble de fonctions prédéfinies, appelées bibliothèques standard, qui fournissent des fonctionnalités courantes telles que la manipulation de fichiers, les opérations mathématiques, la gestion de la mémoire, etc.

Une **définition de fonction** (aussi appelée **déclaration de fonction** ou **instruction de fonction**) est construite avec le mot-clé `def functionName()`, suivi par :

- Le nom de la fonction.
- Une liste d'arguments à passer à la fonction, entre parenthèses et séparés par des virgules.
- Les instructions Python définissant la fonction, entre deux points `:`.

En python vous avez la possibilité de stocker dans des variables des fonctions

Exercices :

Réalisez les exercices suivants (vous n'avez pas le droit d'utiliser les objets ou toute autre possibilité algorithmique de python hormis les variables, conditions et les boucles et les méthodes qui s'y rattachent).

- 1) Vous devez créer une fonction **“myPutStr”** cette fonction doit permettre de retourner un String. Si c'est un nombre retournez **“et pourquoi pas 42 ?”**.
- 2) Créer une fonction **“computeSurfaceM2”** qui calcule une surface en m² carré.

Type de résultat attendu : **“Votre surface est de 200 m2”**

- 3) Créer une fonction **“detectMyAgeByNight”** capable d'implémenter votre précédent script de la boîte de nuit et de faire entrer dans une boîte de nuit une personne ayant plus de 18 ans et refusant celles qui ont entre **0** et **17**.

Vous devez faire en sorte que **“Vous ne pouvez pas entrer vous n'êtes pas majeur vous avez {age}”** ou **“Vous pouvez entrer vous êtes majeur vous avez {age}”** s'affiche.

Cette fonction devra utiliser **“Input”** pour être déclenchée.

- 4) Faites en sorte de créer une fonction **“tableGenerator”** capable de générer un tableau matriciel avec des “|” et des “-” d’après une liste python à plusieurs dimensions.

Se tableau devra obligatoirement comprendre des titres pour chaque colonnes même vide et des valeurs même si vide.

Appuyez-vous sur la structure de votre liste pour reproduire votre tableau à l’identique dans votre console.

Resultat attendu :

	Test1	Test2	Test3
Data1	1	2	3.33
Data2	3	2	1
Data3	6.7	4	2

- 5) Vous devez réaliser une horloge numérique en affichant l’heure actuelle en temps réel. Cette horloge devra être actualisée toutes les secondes.

Resultat attendu :

23:00 1 seconde après -> **23:01** etc ...

- 6) Faites en sorte de créer une fonction **“is_palindrome”** qui vérifie si une chaîne de caractère est un palindrome elle devra renvoyer “True” ou “False”
- 7) Faites en sorte de créer une fonction **“validMyInternationalPhone”** qui contrôle un numéro de téléphone passé en argument et l’indicatif du pays concerné dans un second argument. Vous devrez vérifier et faire en sorte que les numéros Français et des Etats-Unis renvoient **“True”** si le numéro est compatible avec le pays du second argument et **“False”** dans le cas contraire.
- 8) Vous devez réaliser une fonction capable de calculer la suite de fibonacci à partir d’une taille limite fix en argument pour limiter la taille de la suite. Vous devrez réaliser obligatoirement une recursive.

Par exemple si je donne 5 en argument la resultat sera le suivant :

[1, 1, 2, 3, 5]

- 9) Écrivez un décorateur appelé `calculate_time` qui mesure le temps d'exécution d'une fonction et affiche le temps écoulé.