

EXERCICE PYTHON

Apprentissage Au Langage Python



Auteur :

AGOH CHRIS (*Mister Py* )

- Exercice 1 : Calculatrice

Créez un programme qui simule les fonctionnalités d'une calculatrice.

- Exercice 2 : Générateur de mots de passe

Écrivez un programme qui génère des mots de passe aléatoires.

- Exercice 3 : Conversion de température

Écrivez une fonction qui convertit les températures entre Celsius et Fahrenheit.

- Exercice 4 : Jeu du pendu

Implémentez le jeu classique du pendu en utilisant Python.

- Exercice 5 : Analyse de texte

Écrivez un programme qui analyse un texte et compte le nombre de mots, de caractères, etc.

- Exercice 6 : Calculatrice de pourboire

Créez un programme qui calcule le montant d'un pourboire en fonction du pourcentage spécifié.

- **Exercice 7** : Compteur de lettres

Écrivez une fonction qui compte le nombre d'occurrences de chaque lettre dans une chaîne de caractères.

- **Exercice 8** : Recherche de mots

Écrivez une fonction qui recherche tous les mots d'une certaine longueur dans une liste de mots.

- **Exercice 9** : Tri de liste

Implémentez un algorithme de tri pour trier une liste de nombres.

- **Exercice 10** : Calculatrice de taxes

Créez un programme qui calcule le montant de taxes à partir d'un montant hors taxes et d'un taux de taxe.

- **Exercice 11** : Jeu de devinettes

Créez un jeu où l'ordinateur choisit un nombre aléatoire et le joueur doit deviner ce nombre.

- **Exercice 12** : Analyse de données CSV

Écrivez un programme qui lit des données à partir d'un fichier CSV et effectue des opérations statistiques.

- **Exercice 12** : Cryptographie

Implémentez un algorithme de chiffrement comme le chiffre de César ou le chiffre de Vigenère.

- **Exercice 13** : Générateur de mots croisés

Créez un programme qui génère des grilles de mots croisés aléatoires.

- **Exercice 14** : Simulation de particules

Écrivez un programme qui simule le mouvement et l'interaction de particules dans un espace donné.

- **Exercice 15** : Reconnaissance d'images

Utilisez une bibliothèque de vision par ordinateur pour reconnaître des objets ou des visages dans une image.

- **Exercice 16** : Jeu de blackjack

Implémentez le jeu de blackjack avec des règles de jeu complètes.

- **Exercice 17** : Analyse de sentiments

Écrivez un programme qui analyse le sentiment d'un texte en utilisant des techniques de traitement du langage naturel.

- **Exercice 18** : Système de recommandation

Créez un système de recommandation qui suggère des articles, des films ou des produits aux utilisateurs.

- **Exercice 19** : Intelligence artificielle pour le jeu du morpion

Implémentez une IA qui joue au jeu du morpion contre un joueur humain.

- **Exercice 20** : Diviseur de nombre

Écrire une fonction pour trouver tous les diviseurs d'un nombre.

- **Exercice 21** : Nombre Premier

Écrire une fonction pour vérifier si un nombre est premier.

- **Exercice 22** : Pangramme

Écrire une fonction pour vérifier si une phrase est un pangramme (contient toutes les lettres de l'alphabet).

- **Exercice 23** : Produit cartésien

Écrire une fonction pour calculer le produit cartésien de deux ensembles.

- **Exercice 24** : Anagramme

Écrire une fonction pour vérifier si deux chaînes de caractères sont des anagrammes.

- **Exercice 25** : Fichier

Écrire une fonction pour télécharger un fichier à partir d'une URL.

- **Exercice 26** : Conversion de base

Implémentez des fonctions pour convertir un nombre d'une base à une autre (par exemple, décimal en binaire).

- **Exercice 27** : Jeu du mot mystère

Créez un jeu où le joueur doit deviner un mot en proposant des lettres.

- **Exercice 28** : Compteur de mots fréquents

Écrivez un programme qui compte les mots les plus fréquents dans un texte.

- **Exercice 29** : Simulation de trafic

Écrivez un programme qui simule le trafic routier et calcule des statistiques sur les temps de trajet.

- **Exercice 30** : Système de recommandation de musique

Créez un système de recommandation de musique basé sur les préférences des utilisateurs.

- **Exercice 31** : Conversion de nombres romains

Écrivez une fonction qui convertit un nombre romain en nombre entier.

- **Exercice 32** : Validation de mot de passe

Créez une fonction qui vérifie si un mot de passe répond à certains critères (longueur, caractères spéciaux, etc.).

- **Exercice 33** : Jeu du Pierre-Papier-Ciseaux

Implémentez le jeu Pierre-Papier-Ciseaux entre un joueur et l'ordinateur.

- **Exercice 34** : Compression d'images

Implémentez un algorithme de compression d'images comme JPEG.

- **Exercice 35** : (Exo final => Projet d'application à faire en équipe.)

Projet : Gestionnaire de tâches et de projets

Description du projet :

Vous allez créer une application qui permet aux utilisateurs de gérer leurs tâches et leurs projets. L'application devrait offrir les fonctionnalités suivantes :

- Création de projets : Les utilisateurs peuvent créer de nouveaux projets et leur assigner un nom et une date d'échéance.
- Ajout de tâches : Les utilisateurs peuvent ajouter des tâches à leurs projets existants en spécifiant un nom, une description, une date d'échéance et une priorité.
- Affichage des tâches : Les utilisateurs peuvent afficher toutes les tâches associées à un projet particulier ou afficher toutes les tâches regroupées par priorité.
- Marquage des tâches terminées : Les utilisateurs peuvent marquer les tâches comme terminées une fois qu'elles ont été accomplies.
- Suppression des tâches et des projets : Les utilisateurs peuvent supprimer des tâches et des projets existants.

Voici quelques étapes pour vous guider dans la création de ce projet :

- Définissez une structure de données appropriée pour stocker les projets et les tâches. Vous pouvez utiliser des listes, des dictionnaires ou des classes selon votre préférence.
- Implémentez les fonctions pour créer de nouveaux projets, ajouter des tâches, afficher les tâches, marquer les tâches comme terminées et supprimer des tâches/projets.
- Créez une interface utilisateur conviviale en utilisant une bibliothèque telle que Tkinter, PyQt ou Kivy. Concevez des fenêtres, des boutons et d'autres éléments d'interface pour interagir avec l'application.
- Liez l'interface utilisateur aux fonctions de gestion des tâches et des projets. Par exemple, lorsque l'utilisateur clique sur un bouton "Ajouter une tâche", cela devrait déclencher la fonction correspondante pour ajouter une nouvelle tâche.
- Testez votre application en créant différents projets, en ajoutant des tâches, en les marquant comme terminées et en vérifiant si les fonctionnalités se comportent comme prévu.

Bonne Pratique *Pythoniale*

