

EXERCICES DE PROGRAMMATION EN PYTHON VI

Exercice 1 :

- Ecrire une fonction **remplir_liste(n)** qui renvoie une liste de **n** nombres entrés par l'utilisateur qui correspondent à des poids en tonnes.
- Remplir une liste en appelant la fonction **remplir_liste**.
- Déterminer s'il existe une charge supérieure à 20 tonnes dans une liste et informer l'utilisateur.
- Calculer la charge moyenne de la liste de poids.

Exercice 2 :

- Ecrire une fonction **dimensions** qui renvoie trois valeurs entrées par l'utilisateur, correspondant à la longueur, la largeur et la hauteur d'une dalle rectangulaire.
- Calculer le volume de béton nécessaire pour une dalle rectangulaire.
- Si le mètre cube de béton coûte 100 €, calculer le coût total de béton nécessaire pour une dalle.
- Calculer l'énergie potentielle.

Exercice 3 :

- Ecrire une fonction qui demande à l'utilisateur d'entrer la masse, la vitesse, le rayon et l'accélération d'un objet.
- Calculer l'énergie cinétique de cet objet.
- Calculer le moment d'inertie de l'objet.
- Trouver la force de l'objet.

Exercice 4 :

- Ecrire une fonction qui demande à l'utilisateur d'entrer une résistance et une intensité.
- Calculer la tension aux bornes de la résistance.
- Calculer la puissance active.
- Un appareil de 1000 W fonctionne pendant 3 heures. Calcule l'énergie consommée en kWh.

Exercice 5 :

- Ecrire une fonction qui convertit un nombre en binaire. *On peut utiliser les opérandes div et mod ainsi que la concaténation des chaînes de caractères.*
- Convertir un nombre binaire en décimal. *On peut entrer le nombre sous la forme d'une chaîne de caractères.*
- Un serveur a une capacité de 120 requêtes/s et en reçoit 90. Calculer son facteur de charge.
- Calculer le produit scalaire de $A = [2, 3, 4]$ et $B = [1, 0, -1]$.

Exercice 6 :

- Écris un programme qui choisit un nombre aléatoire entre 1 et 100.
- L'utilisateur doit deviner ce nombre.
- À chaque essai, le programme indique si le nombre proposé est **trop petit**, **trop grand** ou **correct**.
- Le jeu continue jusqu'à ce que l'utilisateur trouve la bonne réponse.
- À la fin, affiche le nombre total de tentatives.

import random

random.randint(1, 100)

On pourra demander à l'utilisateur de recommencer le jeu et faire des niveaux de difficulté différents en fonction des bornes qui comprennent le nombre.