

Exercices

PYTHON

Premier BreakPoint

Regardez les vidéos suivantes

[Les bases](#)
[les variables](#)
[Les conditions](#)

1. Affectez les variables temps et distance par les valeurs 6.892 et 19.7. Calculez et affichez la valeur de la vitesse.
Améliorez l'affichage en imposant un chiffre après le point décimal. RDV à l'exercice 2
2. Saisir un nom et un âge en utilisant l'instruction input(). Les afficher.
Enfin, utilisez la « bonne pratique » : recommencez l'exercice en *transtypant* les saisies effectuées pour les forcer à être du bon type (nom sera un str et age un entier) RDV à l'exercice 3
3. Saisissez un flottant. S'il est positif ou nul, affichez sa racine, sinon affichez un message d'erreur. RDV à l'exercice 4
4. L'ordre *lexicographique* est celui du dictionnaire.
Saisir deux mots, comparez-les pour trouver le « plus petit » et affichez le résultat.
RDV à l'exercice 5
5. On désire sécuriser une enceinte pressurisée.
On se fixe une pression seuil et un volume seuil : $p_{\text{Seuil}} = 2.3$, $v_{\text{Seuil}} = 7.41$.
On demande de saisir la pression et le volume courant de l'enceinte et d'écrire un script qui simule le comportement suivant :
 - si le volume et la pression sont supérieurs aux seuils : arrêt immédiat ;
 - si seule la pression est supérieure à la pression seuil : demander d'augmenter le volume de l'enceinte ;
 - si seul le volume est supérieur au volume seuil : demander de diminuer le volume de l'enceinte ;
 - sinon déclarer que « tout va bien ».Ce comportement sera implémenté par une alternative multiple.

A cette étape, faire valider vos exercices par un des coachs !
--

Deuxième BreakPoint

Regardez les vidéos suivantes :

[les boucles](#)

6. Demandez à l'utilisateur de saisir une chaîne de caractères à l'écran, puis vous vérifierez si cette chaîne possède un @ et si cette chaîne termine par .com, si c'est le cas, vous indiquerez que c'est un email valide

A cette étape, vous consulterez la ressource vidéo suivante

<https://www.youtube.com/watch?v=eW7bUNyXMR>
E

7. Faire une boucle pour faire afficher 10 fois un message à l'écran
8. Faire une boucle pour afficher les lettres d'un mot lettre par lettre.

9. Initialisez deux entiers : $a = 0$ et $b = 10$.
Écrire une boucle affichant et incrémentant la valeur de a tant qu'elle reste inférieure à celle de b . Si tu n'as eu aucune difficulté, RDV à l'exercice 8, sinon exercice 7.
10. Écrire une autre boucle décrémentant la valeur de b et affichant sa valeur si elle est impaire. Boucler tant que b n'est pas nul. RDV à l'exercice 8
11. Écrire une *saisie filtrée* d'un entier dans l'intervalle 1 à 10, bornes comprises. Affichez la saisie. Si tu n'as eu aucune difficulté, RDV à l'exercice 9, sinon exercice 10
12. Affichez chaque caractère d'une chaîne en utilisant une boucle for.
Affichez chaque élément d'une liste en utilisant une boucle for. RDV à l'exercice 10.
Affichez les entiers de 0 à 15 non compris, de trois en trois, en utilisant une boucle for et l'instruction range(). Si tu n'as eu aucune difficulté, RDV à l'exercice 11, sinon RDV à l'exercice 12.
13. Utiliser une boucle while puis une boucle for pour afficher les n premiers nombres pairs.
RDV à l'exercice 12

A cette étape, faire valider vos exercices par un des coaches !

Troisième BreakPoint

Regardez les vidéos suivantes :

[Les listes](#)

15. définir la liste : `liste = [17, 38, 10, 25, 72]`, puis effectuez les actions suivantes :
- trie et affichez la liste ;
 - ajoutez l'élément 12 à la liste et affichez la liste ;
 - renversez et affichez la liste ;
 - affichez l'indice de l'élément 17 ;
 - enlevez l'élément 38 et affichez la liste ;
 - affichez la sous-liste du 2^e au 3^e élément ;
 - affichez la sous-liste du début au 2^e élément ;
 - affichez la sous-liste du 3^e élément à la fin de la liste ;
 - affichez la sous-liste complète de la liste ;
- RDV à l'exercice 13.
16. Ecrire l'inverse d'une chaîne. RDV à l'exercice 14
17. Vérifier si une chaîne est un palindrome. Si tu as des difficultés, passe à l'exercice 15 d'abord.
18. Vérifier si une chaîne est un email en regardant la présence de @ et de . puis de max 3 caractères après le . Une fois fini, reviens à l'exercice 14, si tu as des difficultés à revenir au 14, vas à l'exercice 16.
19. Initialisez `truc` comme une liste vide, et `machin` comme une liste de cinq flottants nuls. Affichez ces listes. RDV à l'exercice 17.
20. Utilisez la fonction range() pour afficher :
- les entiers de 0 à 3 ;
 - les entiers de 4 à 7 ;

– les entiers de 2 à 8 par pas de 2.

Définir chose comme une liste des entiers de 0 à 5 et testez l'appartenance des éléments 3 et 6 à chose. RDV à l'exercice 18

A cette étape, faire valider vos exercices par un des coaches !

Quatrième BreakPoint

Regardez les vidéos suivantes :

[Les fonctions](#)

21. Ecrire du code qui demande à l'utilisateur un nombre appelé x, le programme demandera alors x chaînes de caractères que le programme enregistrera dans un fichier de nom data.txt (une chaîne par ligne)
22. Ecrire du code qui lit le contenu du fichier data.txt et vérifie pour chaque ligne si c'est un email (présence de @ et de .com à la fin)
23. Ecrire une fonction compterMots ayant un argument (une chaîne de caractères) et qui renvoie un *dictionnaire* qui contient la fréquence de tous les mots de la chaîne entrée. RDV à l'exercice 19
24. Écrire une fonction volumeSphere qui calcule le volume d'une sphère de rayon r fourni en argument et qui utilise la fonction cube.
Tester la fonction volumeSphere par un appel dans le programme principal. RDV à l'exercice 20
25. Écrire une autre fonction somme avec trois arguments, et qui renvoie leur somme.
Dans le programme principal, définir un tuple de trois nombres, puis utilisez la syntaxe d'appel à la fonction qui *décompresse* le tuple. Affichez le résultat.

A cette étape, faire valider vos exercices par un des coaches !