

Python: Le hasard - Exercices

Voici 15 exercices pour apprendre à utiliser les fonctions `randint` et `choice` du module `random`. Ces exercices sont progressifs et adaptés à des élèves de secondaire.

Exercices de base : Générer des nombres aléatoires

1. Nombre aléatoire simple

Écris un programme qui génère un nombre entier aléatoire entre 1 et 10 et l'affiche.

2. Pair ou impair ?

Génère un nombre aléatoire entre 1 et 100, puis indique s'il est pair ou impair.

3. Jeu du devin

Génère un nombre aléatoire entre 1 et 20. L'utilisateur doit deviner ce nombre. Indique s'il est trop haut ou trop bas après chaque tentative.

4. Évaluation de mathématiques

Génère deux nombres aléatoires entre 1 et 10 et demande à l'utilisateur de calculer leur produit. Vérifie la réponse.

5. Moyenne de nombres aléatoires

Génère 5 nombres aléatoires entre 1 et 50, affiche-les, puis calcule leur moyenne.

Exercices intermédiaires : Manipuler des listes avec `choice`

6. Tirage au sort simple

Crée une liste de 5 prénoms (au choix) et utilise `random.choice` pour afficher un prénom tiré au hasard.

7. Tirage de plusieurs éléments

Crée une liste de 10 objets (par exemple : "stylo", "livre", "clavier", etc.). Affiche 3 objets choisis au hasard (sans gestion de doublons pour l'instant).

8. Mot aléatoire

Crée une liste de mots (exemple : ["chien", "chat", "oiseau", "poisson"]) et génère une phrase en ajoutant un mot aléatoire à "J'ai vu un(e) ...".

9. Générateur de menu

Crée trois listes : une pour les entrées, une pour les plats principaux et une pour les desserts. Choisis aléatoirement un élément dans chaque liste pour créer un menu.

10. Roulette des couleurs

Crée une liste contenant des couleurs (rouge, bleu, vert, jaune, etc.). Simule une roulette qui affiche une couleur au hasard à chaque tour.

Exercices avancés : Combiner `randint` et `choice`

11. Quiz avec questions aléatoires

Crée une liste contenant des questions simples (exemple : "Quelle est la capitale de la France ?", "Combien font 7 + 5 ?"). Utilise `choice` pour poser une question aléatoire.

12. Simulation de dé

Simule un lancer de dé à 6 faces avec `randint`. Répète l'expérience 10 fois et affiche le nombre de fois que chaque face apparaît.

13. Numéro de loto

Génère 6 nombres aléatoires entre 1 et 49 pour simuler un tirage de loto. Affiche les résultats triés.

14. Bataille de dés

Simule un jeu entre deux joueurs lançant chacun un dé (valeurs entre 1 et 6). Celui qui a la valeur la plus haute gagne. En cas d'égalité, recommence.

15. Code couleur aléatoire

Crée un générateur de couleurs en format hexadécimal. Pour cela, génère 6 valeurs aléatoires entre 0 et 15 (représentées par des chiffres ou des lettres de A à F).

Exercice : Création d'un cadavre exquis aléatoire

L'objectif de cet exercice est de créer des phrases aléatoires et amusantes en utilisant le concept des cadavres exquis. Chaque phrase sera composée de quatre parties : un sujet, un verbe, un complément et une circonstance, choisis aléatoirement dans des listes pré-définies.

Listes pré-remplies

Liste des sujets

- Le chat
- Un dinosaure
- Une sorcière
- Mon voisin
- Une étoile
- Un robot
- La lune
- Un ninja

Liste des verbes

- casse
- mange
- crie
- peint
- saute
- écrit
- découvre
- détruit
- fabrique

Liste des compléments

- un gâteau
- une fusée
- un arc-en-ciel
- des bonbons
- un château
- un trésor

- une forêt
- un secret

Liste des circonstances

- dans le salon
 - sur la Lune
 - en chantant
 - avec un crocodile
 - sous la pluie
 - à minuit
 - devant la classe
 - en pyjama
-

Énoncé

1. Crée un programme qui génère des phrases aléatoires en utilisant les listes ci-dessus.

À chaque exécution, le programme doit choisir :

- un sujet dans la liste des sujets,
- un verbe dans la liste des verbes,
- un complément dans la liste des compléments,
- et une circonstance dans la liste des circonstances.

Assemble les éléments pour former une phrase complète. Par exemple :

"Un dinosaure peint un arc-en-ciel sous la pluie."

2. Affiche cinq phrases différentes en exécutant une boucle.

Ces exercices permettent de travailler à la fois sur la compréhension des fonctions de `random`, sur les structures conditionnelles, les boucles et les manipulations de listes.