

## Exercices

### Exercice 1 \*

1. Écrire la classe **Livre** qui possède les attributs ci-après. Ces attributs seront initialisés par trois paramètres passés au moment de l’instanciation.

Livre
titre (str)
auteur (str)
prix (float)

2. Ajouter la méthode **afficher** qui renvoie une description du livre sous forme d’une chaîne de caractères.
3. Créer une instance de la classe **Livre** avec les arguments suivants:
  - Le guide du voyageur galactique
  - Douglas Adams
  - 6.42
4. Afficher la description du livre instancié.

### Exercice 2 \*

1. Créer une classe **Menu** qui permettra de calculer le prix de l’addition dans un restaurant. Il devra posséder les attributs **entree**, **plat**, **accompagnement**, **dessert** initialisés à **None**.
2. Créer les méthodes **prendre\_entree**, **prendre\_plat**, **prendre\_accompagnement**, **prendre\_dessert** qui auront toutes un paramètre de type **str**.
3. Créer la méthode **calculer\_note** qui additionne les prix des mets s’ils ont été choisis:
  - entrée: 8€
  - plat: 10€
  - accompagnement: 3€
  - dessert: 8€

La méthode renverra la somme à payer.

### Exercice 3 \*\*

1. Définir la classe **Rectangle** avec les attributs **longueur** et **largeur**. Ces attributs seront initialisés par deux paramètres passés au moment de l’instanciation.
2. Ajouter les méthodes:
  - **perimetre**: renvoie le périmètre du rectangle (nombre flottant),
  - **aire**: renvoie l’aire du rectangle (nombre flottant),
  - **est\_carre**: renvoie **True** si le rectangle est un carré.
3. Écrire un programme qui:
  - crée un rectangle de dimensions 5.3 et 2.8,
  - affiche l’aire et le périmètre du rectangle,
  - vérifie si c’est un carré.

**Exercice 4 \*\*** Chaque combattant de Fortnite est extrêmement personnalisable. C’est une des caractéristiques qui fait le succès du jeu. Dans cet exercice, on considère qu’il est possible de choisir les éléments:

- tenue,
- parachute,
- traînée.

De plus, chaque personnage possède un niveau d’énergie et un bouclier, tous les deux initialisés à 100. Enfin, il possède un sac à dos qui peut contenir trois éléments maximum.

Un combattant peut ramasser un élément et le mettre dans son sac à dos, si ce dernier n’est pas plein. On représentera les éléments ramassés par des chaînes de caractères (“Gun”, “shield”,...).

Il peut également subir une attaque d’une certaine valeur (entière). Le bouclier subit l’attaque en premier puis c’est le niveau d’énergie qui baisse si le bouclier atteint zéro.

1. Modéliser un combattant de Fortnite.



Figure 1: skin fortnite

2. Implémenter en Python, le modèle créé.

**Exercice 5 \*\*** On définit une classe **Date** pour représenter une date avec trois nombres entiers pour attributs : **jour**, **mois**, **annee**.

1. Écrire son constructeur.
2. Écrire la méthode **afficher** qui renvoie une chaîne de la forme *8 mai 1945*.
3. Écrire la méthode **est\_avant** qui prend une **Date** pour paramètre et renvoie **True** si la date en cours est plus ancienne que celle passée en paramètre.
4. Écrire un programme qui teste cette classe.

**Exercice 6 \*\*** Pour créer une application de gestion de liste de courses, on se propose de modéliser les objets utiles, sachant que:

- l'application pourra contenir plusieurs listes,
  - chaque liste contiendra tous les produits à acheter,
  - chaque produit sera défini par son nom et la quantité à acheter.
1. Modéliser le projet présenté, par plusieurs objets. La liste devra, entre autres, pouvoir afficher l'ensemble des produits à acheter.
  2. Implémenter le projet en Python.