

Jeu de dés

Le but de l'exercice est de créer un jeu de dés classique dans lesquels des joueurs joueront à tour de rôle en lançant un gobelet, qui contiendra des dés. Au bout d'un certain nombre de tour, On désigne le gagnant, celui-ci est le joueur qui a le score le plus élevé. Pour cela nous allons devoir définir plusieurs classes :

Classe De

La classe De est celle qui représente un De du gobelet.

Celle-ci a 1 attribut :

- **valeur** : un nombre représentant la valeur d'un lancé de dé

Un constructeur sans arguments :

- **constructor()** : initialise la valeur du dé à 0

Ainsi que 3 méthodes :

- **get_valeur()** : renvoie la valeur du dé
- **lancer()** : change la valeur du dé : cette valeur doit être un nombre généré aléatoirement entre 1 et 6

Classe Gobelet

La classe Gobelet représente le gobelet utilisé dans la partie :

Celui-ci a 2 attributs :

- **valeur** : nombre représentant la valeur d'un lancé du gobelet
- **dés** : tableau de dés qui contient un certain nombre de dés

Un constructeur avec un argument :

- **constructor(nb_dés)** : initialise la valeur du gobelet à 0, génère le nombre de dés nécessaires à la partie et les ajoute au tableau **dés**

Ainsi que 3 méthodes :

- **get_valeur()** : renvoie la valeur du gobelet
- **lancer()** : change la valeur des dés du gobelet ; met à jour la valeur du gobelet
- **afficher_score()** : affiche en console le score du dernier lancé de gobelet

Classe Joueur

La classe Joueur représente une personne participant à la partie :

Celui-ci à 2 attributs :

- **nom** : chaîne de caractères représentant le nom du joueur
- **score** : nombre représentant le score du joueur (à un instant t)

Un constructeur avec un argument :

- **constructor(nom)** : initialise la valeur du nom du joueur à partir du paramètre nom, initialise le score du joueur à 0

Ainsi que 3 méthodes :

- **get_nom()** : renvoie le nom du joueur
- **get_score()** : renvoie le score du joueur
- **jouer(gobelet)** : prend en paramètre le gobelet de la partie, lance le gobelet, met à jour le score du joueur en fonction du résultat du lancé
- **afficher_score()** : affiche en console le score du joueur

Classe Partie

La classe Partie représente une partie de dés :

Celui-ci à 2 attributs :

- **joueurs** : tableau de joueurs
- **nb_tours** : entier représentant le nombre de tours à jouer
- **gobelet** : gobelet de dés

Un constructeur avec 2 arguments :

- **constructor(nb_tours, nb_des)** : crée l'objet Partie en récupérant le nombre de tours et nombre de dés (définis par le programmeur).

Ainsi que 2 méthodes :

- **initialiser()** : permet d'inscrire des joueurs dans la partie
- **lancer()** : lance la partie : chaque joueur joue à tour de rôle pendant les n tours. Une fois la partie terminée, affiche le gagnant.
- **afficher_gagnant()** : compare les scores des joueurs et affiche le gagnant.

Indications pour l'affichage :

Exemple d'affichage console pour le déroulement de la partie

```
La partie commence ! Celle-ci comportera 5 tours.  
Les joueurs présents sont :  
Nicole  
Michel  
Jean  
Mireille  
Tour 1 :  
Nicole lance le gobelet  
Nicole fait 6  
Nicole a un total de 6 points  
Michel lance le gobelet  
Michel fait 12
```

```
Nicole a un total de 12 points
...
Tour 5 :
Nicole lance le gobelet
Nicole fait 13
Nicole a un total de 57 points
Michel lance le gobelet
Michel fait 4
Nicole a un total de 49 points
...
Fin de la partie
Le gagnant est Nicole
```

NB

Faites une version statique dans laquelle le programmeur s'occupe de paramétrer les joueurs (dans la méthode `initialiser()` de la classe `Partie`). On ne gèrera pas le cas de l'égalité au moment de l'affichage du gagnant.