



Un mini jeu de rôle du genre donjons et dragons



Réalise en P00 un script de combat entre un héros et un ou plusieurs monstres. Les résultats ressembleront à ceux présentés ci-dessous. Équipe de 2 ou 3 maximums !!



Exemple de résultat d'une partie

DONJONS ET DRAGONS MINI PROJET P00

CREATION DU MONSTRE

Point de vie : 5
Force d'attaque : 3

CREATION DU HERO

Point de vie : 15
Force d'attaque : 5



Attaque par le héros : 6
Attaque du monstre : 3
!!! Le monstre est touché
Point de vie du Hero : 15
Point de vie du monstre : 0

MONSTRE TUE CREATION D'UN NOUVEAU MONSTRE

CREATION DU MONSTRE

Point de vie : 10
Force d'attaque : 3

Attaque par le héros : 6
Attaque du monstre : 4
!!! Le monstre est touché
Point de vie du Hero : 15
Point de vie du monstre : 5

Attaque par le héros : 10
Attaque du monstre : 5
!!! Le monstre est touché
Point de vie du Hero : 15
Point de vie du monstre : 0



MONSTRE TUE CREATION D'UN NOUVEAU MONSTRE

CREATION DU MONSTRE

Point de vie : 10
Force d'attaque : 3

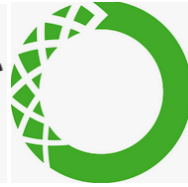
Attaque par le héros : 1
Attaque du monstre : 7
Aie!!! Ouille!!! Le héros est touché
Point de vie du Hero : 0
Point de vie du monstre : 10



FIN DU COMBAT

VICTOIRE DU MONSTRE





Conseils de réalisation

1.1 Générer du hasard

Les programmes informatiques réalisent les calculs toujours de la même manière, ils sont donc prévisibles. La difficulté pour un générateur de nombre aléatoire est de le réaliser pour qu'il soit non-déterministe. Ceci est particulièrement vrai pour des applications comme les jeux par exemple. Si je relance plusieurs fois le jeu je ne dois pas avoir toujours le même comportement du joueur artificiel.

Étudions le hasard avec python¹.



Génération de hasard avec des nombres : fonctions seed(), random(), randint()



```
# Essai des valeurs aléatoires avec python

from random import *

# choix du germe de la séquence pseudo-aléatoire
# valeur 666, python 3
seed(666,3)

# La fonction random renvoi un nombre flottant compris entre 0 et 1
print("Tirage d'un nombre éléatoire")
print(random(),"\n")

# 100 lancer de dés
# La fonction randint(a,b) renvoi un nombre compris entre a et b
print("100 lançés de dés")
for x in range(100):
    print(randint(1,6), end=" ")
```

Résultat : Tirage d'un nombre éléatoire
0.45611964897696833

```
100 lançés de dés
4 4 3 5 1 5 5 6 3 1 1 2 4 6 4 5 2 1 1 3 1 3 6 2 2 3 6 5 5 4 3 6 3 5 1 3
2 5 1 2 6 1 5 1 1 6 1 2 1 3 2 3 4 4 2 4 1 4 4 3 5 3 4 1 6 4 5 3 4 4 2 1
4 6 6 5 2 4 4 4 4 3 5 4 4 6 4 3 2 3 2 5 3 1 5 6 2 5 5 3
```

1.2 Création des personnages

Deux personnages sont créés au départ un monstre et un héro. Les monstres et les héros ont deux caractéristiques : les points de vie et la force d'attaque.

Exemple création de la classe monstre



```
class Honstre():
    def __init__(self,FA,PV):
        self.pointDeVie = PV
        self.forceAttaque = FA
        print("CREATION DU MONSTRE")
        print("Point de vie : ",self.pointDeVie)
        print("Force d'attaque : ",self.forceAttaque)

    def modificationPointDeVie(self,Valeur):
        self.pointDeVie = self.pointDeVie - Valeur

    def getForceAttaque(self):
        return self.forceAttaque

    def getPointDeVie(self):
        return self.pointDeVie
```

¹ Site Python3 objectif jeux <https://www.apprendre-en-ligne.net/pj/hasard/index.html> consulté le 2 mars 2017.