

Mini-projet programmation orientée objet

L'objectif du projet est de mettre en place des classes d'un jeu (type World Of Warcraft). La partie graphique sera abordée en fin de projet, mais les tests se feront en mode console. Le projet se fera en binôme : un élève sera chargé de la création des classes relatives aux armes, un autres des classes relatives aux personnages. La fin du projet sera codée en commun.

Élève 1

A) Les armes

On va créer un type d'arme, l'épée, que tous les personnages peuvent utiliser avec plus ou moins d'efficacité.

La classe Epee

- Toute épée possède un nom et un niveau d'attaque.
- On doit pouvoir initialiser ses 2 paramètres à la création de l'arme.
- On doit pouvoir connaître le niveau d'attaque et le nom de l'arme.
- On doit pouvoir visualiser (dans la console, par exemple) le nom et le niveau (redéfinir la méthode __str__).



Test (programme principal) :

Créer 2 épées, avec des paramètres différents : "Excalibur", 7 et "Durandal", 4.

Créer une liste dans laquelle mettre les 2 épées.

Afficher dans la console les caractéristiques de ces armes.

Élève 2

B) Les personnages

On va créer ici un type de personnage, le guerrier.

La classe Guerrier

- Tout guerrier possède un nom et un niveau de vie.
- On doit pouvoir initialiser ses 2 paramètres à la création du guerrier.
- On doit pouvoir connaître le niveau de vie et le nom du guerrier.
- On doit pouvoir visualiser (dans la console, par exemple) le nom et le niveau de vie (redéfinir la méthode __str__).



Test (programme principal) :

Créer 2 guerriers avec des paramètres différents : "Conan", 78 et "Lannister", 45.

Créer une liste dans laquelle mettre les 2 guerriers.

Afficher dans la console les caractéristiques de ces guerriers.

Mise en commun : Chaque élève donne la classe qu'il a codée (Guerrier ou Epee) à son binôme qui la copie dans son propre répertoire.

En commun

C) Gestion des armes par le guerrier

- Un guerrier peut posséder plusieurs armes (utiliser une liste comme attribut).
- On peut donc ajouter une arme à un guerrier avec un maximum de 5.
- On doit pouvoir choisir l'épée que le guerrier va utiliser parmi toutes celles possibles en spécifiant le nom de l'épée. Utiliser un attribut représentant cette épée (__armeEnMain) et ajouter son getter.
- Modifier la méthode __str__() de façon à renvoyer aussi les caractéristiques de l'arme en main.

Test :

Créer un guerrier et 3 épées.

Ajouter les 3 armes au personnages et faites lui choisir la seconde.

Afficher toutes les caractéristiques du personnage.

- Le guerrier, quand il est victime d'une attaque, voit son niveau de vie baisser d'un certain nombre de points (passés en argument).
- Le personnage doit pouvoir attaquer un autre personnage. Les dégâts subis par le personnage correspondent au niveau d'attaque de l'arme en main. Toutefois, si le guerrier « Lannister », attaque avec l'épée « Durandal », les dégâts sont doublés. Idem si « Conan » attaque avec « Excalibur ».

Test :

Créer 2 guerriers et une épée qui sera affectée au second.

Faire en sorte que le premier soit victime d'une attaque de la part du second et afficher leurs caractéristiques. Vérifier les dégâts subis dans toutes les configurations possibles.

D) Interface graphique

- Créer une IHM (interface graphique) comme ci-dessous.
- Créer une fonction « histoire() » qui crée une liste, lit le fichier « histoire.txt » et met les lignes du fichier dans cette liste.

```
file = open(nomFichier, "r")
liste = file.readlines()
file.close()
```

- Afficher la première ligne de la liste et la retirer de cette liste (méthode pop()).

Test :

Choisir votre personnage : Conan, Lannister ou Oui-Oui

Pas de personnage

saisir votre choix

Validez

E) Mise en place de l'histoire

- Créer 5 armes dans le programme principal que vous mettrez dans une liste :

```

dague = Epee("Dague", 1)
aiguille = Epee("Aiguille", 3)
canif = Epee("Canif", 2)
excalibur = Epee("Excalibur", 20)
durandal = Epee("Durandal", 15)

```
- Créer 4 personnages que vous mettrez dans une liste. Chaque personnage possède dès le début du jeu une arme :

```

ouioui = Guerrier("Oui-Oui", 20)
ouioui.ajouterArme(dague)
lannister = Guerrier("Lannister", 30)
lannister.ajouterArme(excalibur)
conan = Guerrier("Conan", 25)
conan.ajouterArme(canif)
mechant = Guerrier("méchant", 28)
mechant.ajouterArme(aiguille)

```
- Créer une variable globale `persoChoisi` (qui stockera le personnage choisi dans l'histoire).
- Coder la fonction `valider()` associée au bouton qui gère toute l'histoire. Le plus simple est de créer un compteur qui s'incrémente de 1 à chaque fois qu'on valide une réponse. Des tests permettront de savoir où on en est dans l'histoire.

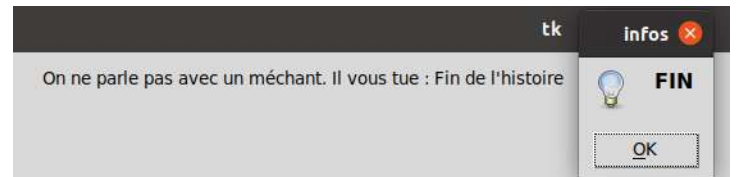
1) Choix du personnage.

On parcourt la liste à la recherche d'un personnage de même nom. La variable `persoChoisi` est mise à jour. Ces caractéristiques sont affichées dans le fenêtre.

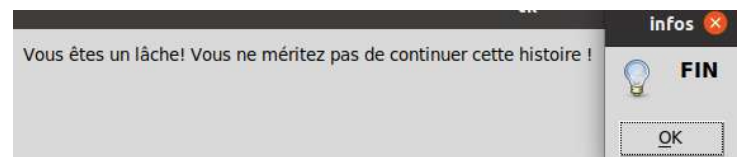
2) Si Excalibur est ramassée, on l'ajoute à la liste d'armes du personnage.

Conan 25 arme en main : Excalibur 20

3) Choix A :



Choix C :



Choix B : Vous choisissez une arme pour combattre : Il faut tester si le méchant meurt ou non. Un des 2 affichages apparaît alors, puis l'histoire se termine ... pour le moment.

