

	<h1>Jeu de rôle</h1>	
---	----------------------	---

Un des tout premiers jeux de rôle sur ordinateur est connu sous le nom d'[Ultima I](#) : The First Age of Darkness, développé par [Richard Garriott](#). Le projet consiste à reproduire un jeu similaire.

Défi N°1 : Compléter les méthodes de la classe Personnage (facile)

```
from random import randint, choice

class Personnage:
    def __init__(self, nom, vie, xp, niveau):
        self.nom=nom
        self.vie=vie
        self.maxVie=vie
        self.xp=xp
        self.niveau=niveau

    def ajouterVie(self, vie):
        #ajoute de la vie dans self.vie sans dépasser self.maxVie

    def retirerVie(self, vie):
        #retire de la vie dans self.vie sans être inférieur à 0

    def monterExperience(self):
        #ajoute 2 points d'expérience
        #Augmente d'un niveau tous les 10 xp
        #Exemple 30xp = niveau 3

    def estVivant(self):
        #retourne vrai si le personnage est vivant

    def estMort(self):
        #retourne vrai si le personnage est mort
```

Valider la classe Personnage avant de passer à la suite.

Défi N°2 : Compléter et tester la classe Guerrier (moyen)

Un guerrier est un personnage avec en plus un attribut de force.

```
class Guerrier(Personnage):
    def __init__(self,nom,force,vie,xp,niveau):
        super().__init__(nom,vie,xp,niveau)
        self.force=force

    def augmenterForce(self):
        #ajoute 1 en force

    def combat(self,adversaire):
        attaque=randint(1, 4)
        degats=attaque*self.niveau*self.force-adversaire.niveau
        print("degat sur le mechant",degats)
        #inflige des dégats au mechant si celui-ci est vivant
        #incrémente le nombre de points d'expérience correspondant aux dégâts infligés
        #Monte si nécessaire en niveau en fonction du nombre de points xp
        #retire de la vie au méchant
        #si le méchant est mort augmenter la force de 1 du guerrier
```

Valider la classe Guerrier avant de passer à la suite.

Défi N°3 : Compléter et tester la classe Magicien (moyen)

Un magicien est un personnage avec en plus un attribut de mana. La nana permet d'attaquer un adversaire. Celle-ci diminue au cours de l'attaque.

```
class Magicien(Personnage):
    def __init__(self,nom,mana,vie,xp,niveau):
        super().__init__(nom,vie,xp,niveau)
        self.maxMana=mana
        self.mana=mana

    def augmenterMana(self):
        #augmente de 10 self.maxMana

    def ajouterMana(self):
        #ajoute 1 en self.mana sans dépasser self.maxMana

    def retirerMana(self,mana):
        #retire mana à self.mana sans descendre en dessous de 0
        #retourne vrai si le magicien à lancé un sort
        #retourne faux si le magicien nen peut plus lancer de sort

    def combat(self,adversaire):
        attaque=randint(1, 4)
        degats=attaque*self.niveau*2-adversaire.niveau
        print("degat du magicien sur le méchant",degats)
        #inflige des dégats au mechant si celui-ci est vivant
        # et que le magicien dispose de nana
        #incrémente le nombre de points d'expérience correspondant aux dégâts infligés
        #Monte si nécessaire en niveau en fonction du nombre de points xp
        #retire de la vie au méchant et diminue de 1 self.mana (consommation de magie)
        #si le méchant est mort augmenter self.maxMana de 10 du magicien.
```

Valider la classe Magicien avant de passer à la suite.

Défi N°4 : Simuler un combat entre 2 personnages (moyen)

Exemple de rendu dans la console :

<pre>combattant=Guerrier("Linflas",2,20,0,1) mechant=Guerrier("Chaos",2,30,0,2) magot=Magicien("Chani",30,30,0,2) def duel(combattant,mechant): #Acompléter (utiliser une boucle tant que) duel(magot,mechant) print(combattant) print(magot) print(mechant)</pre>	<pre>degat heros 6 Chani a touché Chaos / 6 dégats degat mechant -2 degat heros 12 Chani a touché Chaos / 12 dégats degat mechant -3 (négatif le méchant s'est loupé) degat heros 12 Chani a touché Chaos / 12 dégats Chani a tué Chaos Nom :Linflas, Vie :20, Force 2, xp0 Niveau 1 Nom :Chani, Vie :27, Mana 26, xp48 Niveau 4 Nom :Chaos, Vie :0, Force 2, xp5 Niveau 0</pre>
---	--

Défi N°5 : Animation Graphique (très difficile)



A l'aide du programme Fourni « role.py », élaborer votre propre monde en utilisant les classes écrites précédemment.

Aide : la classe Personnage(pygame.sprite.Sprite), doit hériter de la classe [sprite](#) de pygame pour faciliter les collisions. Il existe pleins d'[exemples](#) sur le net.

Défi N°6 : Modifications à volonté (très difficile)

Etablir de nouvelles règles de combat. Changer de niveau ou charger un nouveau monde graphique. Allez chez le marchand pour acheter du matériel, etc...