|  |
| --- |
| TP de Java |
| Année 2018 |
| Apruzzese Raphaël, Melinda ROSSI |

# TP1: Classes et packages

## Package lsg

Quelle est la visibilité optimale pour isAlive dans cette configuration ? Pourquoi ?

* public accessible et fonctionne
* default uniquement pour les interfaces
* protected accessible et fonctionnelle uniquement dans les classes héritées
* private pas accessible

Dans cette configuration la visibilité optimale est protected.

## Packages characters

Quelle est la visibilité optimale pour isAlive dans cette configuration ? Pourquoi ?

* protected ne fonctionne plus car les classes ne sont pas dans les mêmes packages

Donc la visibilité optimale est public.

# TP2: Héritage, Random et E/S

## Classe character

Cherchez ces méthodes dans la documentation officielle Java. Comment cela fonctionne-t-il ?

* getclass fait appelle à la classe courante et getsimplename qui est une méthode de classe qui renvoie le nom de la classe.

Il faudra certainement passer des membres de private à protected. Pourquoi ?

* le passage de private à protected est nécessaire pour faire hériter les variables aux classes filles.

## Classe LearningSoulsGame

Que remarquez-vous au niveau de la durabilité de l’épée ? Pourquoi ?

* Si le héros et le monstre attaquent tous les deux avec la même épée, alors la durabilité est la même pour les deux personnages.

Exemple : durabilité initiale 100

Héros attaque : durabilité 99

Monstre attaque : durabilité 98

Ce phénomène se produit car l’épée est instanciée une seule fois.

# TP3: Spécialisation et classe abstraite

## Classe Monster

Créez les accesseurs en réfléchissant bien à leur visibilité (qui pourra éventuellement être modifiée plus tard…)

* Les accesseurs et les mutateurs sont publics pour pouvoir accéder à la valeur de la variable même en dehors de la classe mère ou des classes filles.

## Absorption de dégâts

Du fait de l’introduction de la méthode computeProtection au niveau de Character, les classes Hero et Monster présentent une erreur. Laquelle ? Pourquoi ?

* Les classes présentent une erreur car la méthode abstraite de la classe abstraite doit être initialisée dans toutes les sous classes.

# TP4: Collections

## Les « Menus Best OF »

Qu’est-il précisé à propos de l’itération ?

* Il n’est pas garanti que l’ordre reste le même dans un Hashset.

Qu’est-il précisé au niveau de l’itération ?

* Il est précisé que grâce à une liste liée, l’ordre d’itération est le même que l’ordre d’insertion.

# TP6: Exceptions

## Test exceptions

Dans le jeu, lancez une attaque du héros : action (1) Que se passe-t-il ?

* Comme l’arme a été set à null, on ne peut pointer vers ses propriétés : cela génère une exception.