



Accueil ► Cours ► Apprenez l'objet avec Java ► Activité : Créez un jeu de combat épique

Apprenez l'objet avec Java



🕒 20 heures 📊 Moyenne

Licence



1. Énoncé

- 👤 Vous trouverez dans cet énoncé toutes les informations pour réussir votre activité. Prenez le temps de lire jusqu'au bout avant de débiter. Une fois terminé, envoyez votre fichier (au format ZIP uniquement) au bas de cette page.

Bienvenue dans MagiWorld !

Dans cet exercice, vous allez devoir créer un **jeu de combat épique** entre Guerrier 🧔, Rôdeur 🦉 et Mage 🧙 !

Ce jeu est prévu pour **2 joueurs** et fonctionne dans **le terminal**. En voici un aperçu :

```

Création du personnage du Joueur 1
Veuillez choisir la classe de votre personnage (1 : Guerrier, 2 : Rôdeur, 3 : Mage)
1
Niveau du personnage ?
10
Force du personnage ?
10
Agilité du personnage ?
0
Intelligence du personnage ?
0
Woarg je suis le Guerrier Joueur 1 niveau 10 je possède 50 de vitalité, 10 de force, 0 d'agilité et 0 d'intelligence !
Création du personnage du Joueur 2
Veuillez choisir la classe de votre personnage (1 : Guerrier, 2 : Rôdeur, 3 : Mage)
3
Niveau du personnage ?
10
Force du personnage ?
0
Agilité du personnage ?
10
Intelligence du personnage ?
10
Abracadabra je suis le Mage Joueur 2 niveau 10 je possède 50 de vitalité, 0 de force, 0 d'agilité et 10 d'intelligence !
Joueur 1 (50 Vitalité) veuillez choisir votre action (1 : Attaque Basique, 2 : Attaque Spéciale)
1
Joueur 1 utilise Coup d'Épée et inflige 10 dommages.
Joueur 2 perd 10 points de vie
Joueur 2 (40 Vitalité) veuillez choisir votre action (1 : Attaque Basique, 2 : Attaque Spéciale)
1
Joueur 2 utilise Boule de Feu et inflige 10 dommages.
Joueur 1 perd 10 points de vie
Joueur 1 (40 Vitalité) veuillez choisir votre action (1 : Attaque Basique, 2 : Attaque Spéciale)
2
Joueur 1 utilise Coup de Rage et inflige 20 dommages.
Joueur 2 perd 20 points de vie
Joueur 1 perd 5 points de vie
Joueur 2 (20 Vitalité) veuillez choisir votre action (1 : Attaque Basique, 2 : Attaque Spéciale)
2
Joueur 2 utilise Soin et gagne 20 en vitalité.
Joueur 1 (35 Vitalité) veuillez choisir votre action (1 : Attaque Basique, 2 : Attaque Spéciale)
2
Joueur 1 utilise Coup de Rage et inflige 20 dommages.
Joueur 2 perd 20 points de vie
Joueur 1 perd 5 points de vie
Joueur 2 (20 Vitalité) veuillez choisir votre action (1 : Attaque Basique, 2 : Attaque Spéciale)
1
Joueur 2 utilise Boule de Feu et inflige 10 dommages.
Joueur 1 perd 10 points de vie
Joueur 1 (20 Vitalité) veuillez choisir votre action (1 : Attaque Basique, 2 : Attaque Spéciale)
2
Joueur 1 utilise Coup de Rage et inflige 20 dommages.
Joueur 2 perd 20 points de vie
Joueur 2 est mort
Joueur 1 perd 5 points de vie
Joueur 2 a perdu !

```

Comment se déroule le jeu :

- Le jeu commence par proposer au **joueur 1** de choisir entre **Guerrier**, **Rôdeur** ou **Mage**.
- Il doit ensuite choisir le **niveau**, la **force**, **l'agilité** et enfin **l'intelligence** de son personnage.
- C'est ensuite au tour du **second joueur** de choisir la **classe** et les **caractéristiques** de son personnage.
- La partie démarre ensuite, le **joueur 1 commence**.
- Chaque joueur joue **tour à tour** et choisit entre son **Attaque Basique** ou son **Attaque spéciale**.
- Quand un joueur **n'a plus de point de vie**, la **partie est terminée** et l'autre joueur l'emporte.

Les caractéristiques :

Chaque personnage possède **5 caractéristiques** :

- **Niveau** : choisi par le joueur (min 1, max 100)
- **Vie** : égale au niveau du joueur * 5
- **Force** : choisie par le joueur (min 0, max 100)
- **Agilité** : choisie par le joueur (min 0, max 100)

- **Intelligence** : choisie par le joueur (min 0, max 100)

Attention le total force + agilité + intelligence doit être égal au niveau du joueur.

Par exemple, un mage niveau 18 ne peut pas avoir 5 de force et 14 d'intelligence : le total fait 19 et non 18.

Les sorts :

Chaque joueur possède **2 sorts** :

- Une **attaque basique** qui cause des dommages à l'adversaire.
- Une **attaque spéciale** aux effets variés.

Voici le détail des sorts :

Guerrier

- Attaque Basique - **Coup d'Épée** : Effectue des dommages égaux à la force du joueur sur l'adversaire.
- Attaque Spéciale - **Coup de Rage** : Effectue des dommages égaux à la force du joueur fois deux sur l'adversaire. Le joueur lançant l'attaque perd de la vitalité : la valeur de sa force divisée par 2.

Rôdeur

- Attaque Basique - **Tir à l'Arc** : Effectue des dommages égaux à l'agilité du joueur sur l'adversaire.
- Attaque Spéciale - **Concentration** : Le joueur gagne son niveau divisé par 2 en agilité.

Mage

- Attaque Basique - **Boule de Feu** : Effectue des dommages égaux à l'intelligence du joueur sur l'adversaire.
- Attaque Spéciale - **Soin** : Le joueur soigne sa vie et regagne sa quantité d'intelligence fois 2 en points de vie. Attention, il ne peut pas avoir plus de vie qu'il n'en avait au départ.

Quelques conseils :

- Pensez à bien créer **des classes** pour représenter les concepts du jeu. Ce sera plus facile pour vous organiser.
- Pensez à mettre en place **des tests** dès qu'il y a de la logique dans votre jeu.
- **Ne dupliquez pas votre code**, utilisez au maximum des **méthodes**.
- Pensez à utiliser le **polymorphisme** pour représenter vos classes de personnages.
- Si vous avez des doutes sur la **compréhension** du sujet, référez-vous au **screenshot**.

Livrable

Le dossier contenant votre projet devra être archivé dans un dossier au format .zip. Attention à bien ajouter votre dossier .git à l'intérieur de cette archive (une partie du barème est consacrée au versionnage).

Compétences évaluées



Cette activité permet de mettre en œuvre les compétences suivantes. Votre activité sera évaluée par vos pairs en fonction des critères associés à chaque compétence.

Réaliser une conception Java facilitant l'organisation du projet

La relation d'héritage est utilisée de manière pertinente.

Rédiger un code propre et facilement utilisable par d'autres développeurs

1. Une classe est définie pour chaque partie logique du projet.
2. Les noms des classes, méthodes, attributs et variables respectent les conventions d'écriture et ont un sens.
3. Chaque action est identifiée par une méthode qui lui est propre.
4. Aucune méthode ne dépasse 20 lignes de code.
5. Le code est facile à comprendre.

Développer un programme Java fonctionnel

1. Le projet fonctionne comme indiqué dans l'énoncé.
2. Les textes définis dans la capture d'écran sont respectés.

Documenter un code Java avec la Javadoc

Les classes et méthodes publiques sont documentées.

Utiliser Git pour la gestion des versions en Java

1. Tous les éléments nécessaires au fonctionnement du projet sont dans git.
2. Les commits sont réguliers.
3. Le nom de chaque commit rend explicite le changement apporté par le commit.

Rédiger des tests unitaires en Java

1. Le nom de chaque test permet de comprendre à quoi sert le test.
2. Chaque méthode contenant de la logique est testée.

Votre activité



[creez-un-jeu-de-combat-epique-JB-2018-08-28T152018.zip](#)



ACCUEIL DE L'ACTIVITÉ

ÉTAPE SUIVANTE



Le professeur

Anthony Cursan

Développeur freelance spécialisé dans les problématiques Big Data.

Découvrez aussi ce cours en...



Vidéo

OpenClassrooms

Le blog OpenClassrooms

Recrutement

Devenez mentor

Nous contacter

Professionnels

Affiliation

For Business


En plus

Qui sommes-nous ?

Fonctionnement de nos cours

Conditions Générales d'Utilisation

Politique de Protection des Données Personnelles

 Français ▼

