



Acquérir les concepts fondamentaux et la syntaxe de base en Java

Mini-Projet 1 : Emergence des concepts objet en Java

Modalités :

- Travail par îlot, puis par classe mixée
- Production individuelle
- Durée ~2 heures

Objectifs de l'activité :

- Emergence des concepts fondamentaux de la « Programmation Orientée Objet » déjà appréhendés lors des sessions précédentes (PHP) par des manipulations concrètes (ordinateur éteint)
- Formulation en langage simple des notions de classe ; classe abstraite ; objet ; attribut ; méthode ; héritage ; polymorphisme...

Sujet :

- Faire une liste de caractéristiques vous définissant bien (qualités, défauts, caractéristiques physiques ou intellectuelles), et des capacités et/ ou aptitudes personnelles (mode de transport pour venir au campus ? Je fais du vélo, je code, j'aime l'informatique...) de chaque élève de l'îlot. Utiliser des post-its.
- Synthèse par salle : Au tableau, les élèves viennent coller leurs post-its, et effectuent un regroupement par thème...
- À partir de cette liste, l'animateur fait émerger les différents concepts objet : les élèves effectuent des regroupements selon des critères ou catégories. Les concepts objet sont amenés en faisant l'analogie avec les regroupements effectués par les élèves.

Livrables :

- Document proposant un premier niveau de définition « en langage simple » des différents concepts objets (environ 2 lignes par définition). A déposer sous le nom **livrable1**, dans votre répertoire individuel sous **java→livrables**

Projet 1 : Affichage des personnages

Introduction:

Le but de l'ensemble des activités qui suivent est de créer un jeu, inspiré des jeux de plateaux de l'univers « Donjons et Dragons », en utilisant le langage Java. Nous allons commencer par créer et afficher des personnages.

Modalités :

- Travail en autonomie
- Production individuelle

Objectifs de l'activité :

- Manipuler la syntaxe de base de Java
- Manipuler les concepts objets du langage Java
- Générer la Javadoc
- Validation des compétences n° 2, 3, 4, 5, 6, 8, 9, 10, 12

Consignes:

1. Etude de tutoriels :

- a. Codecademy : Introduction to Java, Conditional and Control, Object-oriented Java ou Sololearn : Basic Concepts, Conditionals and Loops, Arrays
- b. OpenClassrooms : « Les variables et les opérateurs », « Lire les entrées clavier », « Votre première classe », « Les tableaux »

2. Manipulation de la syntaxe de base de Java

Nous allons manipuler deux types de personnages (guerrier ou magicien) et leur équipement par défaut (arme ou sort).

Le guerrier possède une arme et peut être équipé d'un bouclier pour se défendre. Le magicien possède un sort et peut avoir un philtre pour se défendre.

Chaque type de personnage est caractérisé par les attributs suivants :

- Guerrier :
 - o Nom
 - o Image
 - o Niveau de vie (5-10)
 - o Force d'attaque (5-10)
 - o Arme
 - o Bouclier
- Magicien :
 - o Nom
 - o Image
 - o Niveau de vie (3-6)
 - o Force d'attaque (8-15)
 - o Sort
 - o Philtre

Les armes et sorts sont caractérisés par un nom, et un niveau d'attaque. Les boucliers et philtres sont simplement des Strings.

3. Spécifications du programme

- Le programme va tout d'abord permettre la création d'un personnage, avec la saisie via lecture des entrées clavier (classe Scanner), des informations du personnage (Guerrier ou Magicien).
Pour le personnage en cours de création, un choix permettra de :
 - o Afficher les infos du personnage (affichage via des println)
 - o Modifier les infos du personnage (toutes les infos doivent être modifiables)

- Faire la saisie de plusieurs personnages (boucle) jusqu'à ce que l'utilisateur choisisse de quitter le programme.

Pour cela, vous utiliserez une classe « Guerrier » qui aura 3 constructeurs (sans paramètres, avec nom, et avec nom/image/niveau de vie/force d'attaque) et une classe « Magicien » qui aura de même 3 constructeurs.

Vous structurerez votre code en utilisant des fonctions.

Vous allez créer de même une classe « Arme » (indépendante de la classe "Guerrier") et ses méthodes associées, ainsi qu'une classe « Sort » indépendante de la classe « Magicien » et ses méthodes associées.

Attention !! Toutes vos classes devront être structurées correctement (Getters/Setters, constructeurs, méthode toString(), ...).

4. Etude de tutoriels :

- Codecademy : Object-oriented Java ou Sololearn : Classes and Objects
- OpenClassrooms : « Votre première classe »

Ne pas oublier

- Testez votre programme de manière exhaustive
- Documentez vos classes et générez la Javadoc

Livrables :

- Codes commentés des classes « Main », « Guerrier », « Magicien », « Arme » et « Sort ». A déposer sous git.
- Javadoc de votre projet à déposer dans **java → livrables**
- Glossaire des différentes syntaxes utilisées. A déposer sous le nom **glossaire**, dans votre répertoire individuel sous **java → livrables**

Ressources :

- Codecademy :
 - Introduction to Java, Conditionals and Control flow, Object-Oriented Java
 - <https://www.codecademy.com/learn/learn-java>
- Sololearn:
 - Basic Concepts, Conditionals and Loops, Arrays, Classes and Objects
 - <https://www.sololearn.com/Course/Java/>
- Openclassrooms :
 - <https://openclassrooms.com/courses/apprenez-a-programmer-en-java/les-variables-et-les-operateurs>
 - <https://openclassrooms.com/courses/apprenez-a-programmer-en-java/lire-les-entrees-clavier>
 - <https://openclassrooms.com/courses/apprenez-a-programmer-en-java/les-tableaux-3>
 - <https://openclassrooms.com/courses/apprenez-a-programmer-en-java/votre-premiere-classe>
- Tutoriel d'Oracle pour Java :
 - <http://docs.oracle.com/javase/tutorial/java/index.html>

Synthèse (activité de groupe)

Modalités :

- Travail en groupe

Objectifs de l'activité :

- Travail de synthèse des concepts clé et de la syntaxe Java

Déroulement :

- Mot clé rédigé par les étudiants sur post-it
- Travail en grand groupe de définition et reformulation du concept, ou de rédaction de la (ou des) ligne(s) de code Java