

Sujet Final A2017 LO43: Durée 2h Documents et Calculatrice interdits

Partie 1: Java

I. Questions de cours:

1. Quels sont les différentes façon de faire des threads? Ecrire le diagramme d'état d'un thread.
2. Y a-t-il un mode de liaison (entre méthode et objet) par défaut en Java? Si oui lequel?

II. Exercice: Dynamique ou pas?

Voici un programme Java composé de 4 classes.

Classe Personnage

```
public class Personnage {
    private String name;

    public Personnage ()
    {
        name = "Personnage Toto";
    }

    public Personnage (String s)
    {
        name = "Personnage:"+s;
    }

    void Combat (Personnage p)
    {
        System.out.println("Personnage Attaque Personnage : "+ p.getName());
    }

    void Combat (Humain h)
    {
        System.out.println("Personnage Attaque Humain : "+ h.getName());
    }

    void Combat (Dragon h)
    {
        System.out.println("Personnage Attaque Dragon : "+ h.getName());
    }

    protected String getName()
    {
        return this.name;
    }
}
```

Classe Humain

```
public class Humain extends Personnage{
```

```

    void Combat (Dragon d)
    {
        System.out.println("Humain "+ this.getName()+" Attaque Dragon " +
d.getName());
    }

    void Combat (Humain h)
    {
        System.out.println("Humain "+ this.getName()+" Attaque Humain "+
h.getName());
    }

    void Combat (Personnage p)
    {
        System.out.println("Humain "+ this.getName()+" Attaque Personnage "
+ p.getName());
    }

    public Humain ()
    {
        super("Humain");
    }

    public Humain (String s)
    {
        super("Humain:"+s);
    }
}

```

Classe Dragon

```

public class Dragon extends Personnage {

    void Combat (Dragon d)
    {
        System.out.println("Dragon Attaque Dragon "+ d.getName());
    }

    void Combat (Humain h)
    {
        System.out.println("Dragon Attaque Humain "+ h.getName());
    }

    void Combat (Personnage p)
    {
        System.out.println();
        System.out.println("Dans Dragon "+ this.getName()+" Attaque
Personnage "+ p.getName());
        p.Combat(this);
    }

    public Dragon ()
    {
        super("Dragon");
    }

    public Dragon (String s)
    {

```

```

        super("Dragon:"+s);
    }
}

```

Classe Principale

```

import java.util.ArrayList;

public class Main {

    public static void main(String[] args) {
        ArrayList <Personnage> list = new ArrayList<Personnage> ();
        Dragon D = new Dragon("Drago");

        list.add(D);
        list.add(new Humain("Bob"));

        list.add(new Humain("Stuart"));

        Humain Kevin = new Humain("Kevin");

        for (Personnage p: list)
        {
            Kevin.Combat(p);
            D.Combat(p);
            System.out.println();
            p.Combat(Kevin);
            p.Combat(D);
            System.out.println();
            D.Combat(Kevin);
            System.out.println();
            for (Personnage p2: list)
            {
                p.Combat(p2);
            }
        }
    }
}

```

Qu'affiche le programme suivant?

Partie 2: UML et conception:

Les films issus des franchises importantes (Avenger, Spider man, Star Wars, Star Trek...) ont des scénarios quasiment standardisés afin que les risques soient limités en terme d'investissement. L'objectif de cet exercice est de décrire ce que pourrait être un logiciel de génération automatique de scénarios pour les films de la saga Star Wars.

Dans chacun des films, il faut :

- Des personnages:
 - des héros (qui sont soit courageux soit avec des pouvoirs (parfois les deux)). Ceux qui ont des pouvoirs peuvent avoir des pouvoirs avec plusieurs niveaux de maîtrise (Maître, Chevalier, Apprenti, Sensible à la force)
 - des méchants (avec plusieurs niveaux de méchanceté) qui peuvent être téméraire, peureux, cupides et qui peuvent avoir des pouvoirs également mais du côté sombre (normal ils sont méchants). Il existe également plusieurs niveaux de maîtrise du pouvoir (Guide suprême, Maître, Apprenti)
 - des militaires qui peuvent être soit méchant soit gentils
 - des personnages secondaires mystérieux (qui ne sont ni-gentils ni-méchant mais parfois un peu des deux)
 - des animaux mignons pour vendre des jouets (ewoks, porg...).
 - chaque personnage est lié à une espèce et à une planète d'origine
 - les personnages peuvent être également des robots
- Des planètes qui possèdent:
 - un type (désertique, glacée, forestière, urbaine, marécageuse)
 - une taille
 - des soleils
 - des lieux (bar, temple, port spatial, désert, forêt, base...)
- Des vaisseaux qui possèdent:
 - un type (chasseur, bombardier, frégate, croiseur, corvette, navette, quadripod...)
 - un lieu d'évolution (terrestre, spatial, aérien, multiple)
 - une capacité (i.e. nombre de vaisseaux de taille plus petite ou de personnages qui peuvent transporter)
 - un point faible
- Des combats qui peuvent être:
 - Entre vaisseaux dans l'espace
 - Entre vaisseaux sur une planète
 - Entre personnages
 - Entre vaisseaux et planète
- Des relations entre personnages combinables entre eux:
 - Maître/Elève (neutre)
 - Amitié (positive)
 - Amour (positive)
 - Haine (négative)
 - Famille (neutre)

Ainsi construire un scénario consiste à utiliser ces ingrédients en appliquant les règles suivantes: Il faut au moins 3 planètes par films, il faut 1 ou 2 lieux par planète, il faut au moins 1 combat entre des vaisseaux et 1 combat entre des personnages, une relation négative entre 2 personnages d'une même famille et/ou une relation positive entre deux personnages de camps différents.

1. Ecrire le diagramme de classe de ce logiciel de génération de scénario en n'oubliant pas la classe principale qui possède une méthode appelée generate qui a pour objectif de fournir le scénario.
2. Ecrire un diagramme d'objet d'une des scènes mythiques de star wars. Si jamais vous n'avez vu aucun des films inventer une scène.
3. Ecrire le diagramme de séquence de cette scène.