Programmation objet

Classe Object

Classe Object

- Classe racine (root) du framework
- Toutes les classes créés implémentent Object par défaut

Méthodes de la classe Object

Modifier and Type	Method and Description
protected Object	clone() Creates and returns a copy of this object.
boolean	equals(Object obj) Indicates whether some other object is "equal to" this one.
protected void	finalize() Called by the garbage collector on an object when garbage collection determines that there are no more references to the object.
Class<?>	getClass() Returns the runtime class of this Object.
int	hashCode() Returns a hash code value for the object.
String	toString() Returns a string representation of the object.

Méthodes de la classe Object

Modifier and Type	Method and Description
void	notify() Wakes up a single thread that is waiting on this object's monitor.
void	notifyAll() Wakes up all threads that are waiting on this object's monitor.
void	wait() Causes the current thread to wait until another thread invokes the notify() method or the notifyAll() method for this object.
void	wait(long timeout) Causes the current thread to wait until either another thread invokes the notify() method or the notifyAll() method for this object, or a specified amount of time has elapsed.
void	wait(long timeout, int nanos) Causes the current thread to wait until another thread invokes the notify() method or the notifyAll() method for this object, or some other thread interrupts the current thread, or a certain amount of real time has elapsed.

ToString

Description de l'objet

Méthode toString

- Retourne une description de l'objet sous forme de chaine de caractère

- Appel implicite de la méthode :

HerosTerre ht = new HerosTerre("Terrarium", 100, 50, 65, 80);
Sout(**ht**) → appel implicite de ht.toString();

Attention! Si la méthode toString n'existe pas, l'appel de sout(ht) affiche la référence de l'objet

Exercice

- Dans le programme de test,
- Créer un héros de Mer, créer un joueur
- Afficher les informations du héros et du joueur en utilisant uniquement le nom d'objet.

- Apporter les corrections nécessaires pour que l'affichage soit pertinent

Finalize

Destruction de l'objet

Méthode finalize

- Appelée automatiquement pas le garbage collector à la destruction de l'objet

- Contient le comportement attendu de l'objet avant sa suppression
  - Libération de la mémoire
  - Fermeture d'une ressource
  - Affichage d'un message
  - …

Equals

Test l'égalité des objets

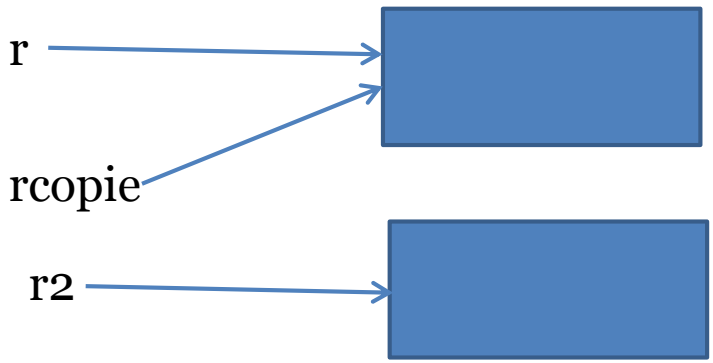
Méthode equals

- Retourne true si les objets comparés sont égaux, false sinon
- Le programmeur définit les éléments à comparer pour attester de l'égalité de deux objets.

Egalité des objets : Rappel

- L'opérateur == entre des objets teste l'égalité des références d'objet

- Rectangle r = new Rectangle(8,6);
- Rectangle rcopie = r;
- Rectangle r2 = new Rectangle(8,6);



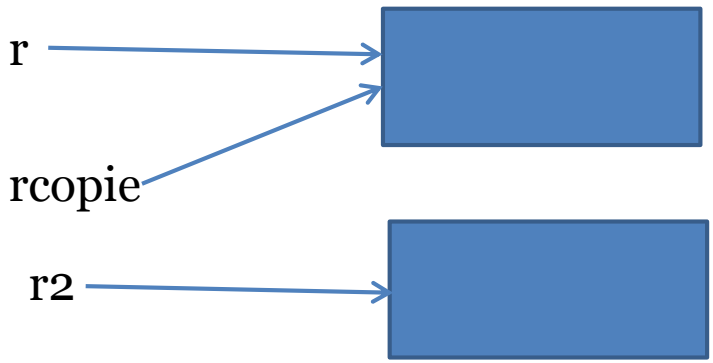
r == rcopie → true

r == r2 →false

Egalité des objets, avec equals

- L'opérateur == entre des objets teste l'égalité des références d'objet

- Rectangle r = new Rectangle(8,6);
- Rectangle rcopie = r;
- Rectangle r2 = new Rectangle(8,6);



r.equals(rcopie) → true

r.equals(r2) → true

Méthode equals

Principe

- Retourne un booléen
- Prend en paramètre un objet de la classe
- Compare l'objet ayant appelé la méthode (this) avec l'objet en paramètre
  - Comparaison de tous les attributs
  - Comparaison des objets reliés (equals)

Exemple, classe Heros

```
public boolean equals(Heros h2)
{
if (this.nom.equals(h2.getNom())
                &&
this.defense==h2.getDefense()
                &&
this.pointsVie==h2.getPointsVie())
            return true;
          return false;
        }
```

Exercice

Classe HerosTerre

- Écrire et tester la méthode equals dans la classe HerosTerre pour vérifier que les attributs aient des valeurs identiques

Classe Joueur

- Ecrire et tester la méthode equals dans la classe Joueur pour vérifier que les attributs et le héros choisi soient identiques

Clone

Intérêt du clonage

- Faire une copie par valeur d'un objet
- Ajout d'une méthode clone dans la classe

Copie d'un objet

Utilisation de l'opérateur =

```
HerosTerre h1 = new
HerosTerre("Terrum",
150,50,14,32);
//copie de l'objet
HerosTerre copie = h1;

h1.setDefense(70);
```

Résultat

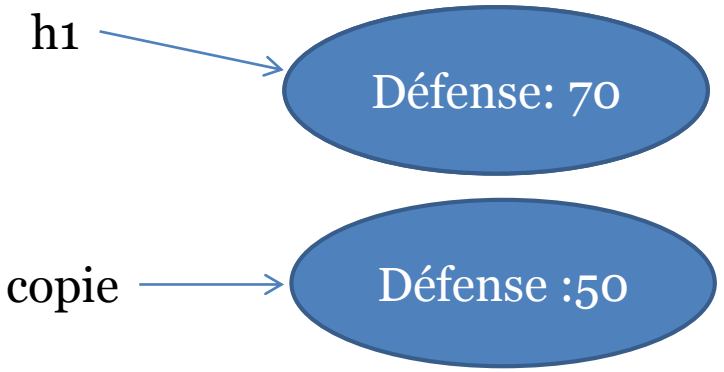
Copie d'un objet, ce que l'on veut

Utilisation de l'opérateur =

```
HerosTerre h1 = new
HerosTerre("Terrum",
150,50,14,32);
//copie de l'objet
HerosTerre copie = copie de h1;

h1.setDefense(70);
```

Résultat



Principes du clonage

- 2 références d'objet différentes
  - x.clone() != x doit renvoyer true
- Classes identiques
  - x.clone().getClass() == x.getClass() renvoie true par convention
- 2 objets identiques
  - x.clone().equals(x) renvoie true par convention

Création du clonage

- La classe doit implémenter l'interface Cloneable
- Par convention, l'objet retourné est obtenu grace à l'appel de super.clone()

Principe du clonage

Attributs de classes immuables

- Copie par valeur faite par l'appel de super.clone()
- Copie de surface (shallow copy)

Attributs de classes non immuables

- Copie par valeur à ajouter dans la méthode clone
- Copie en profondeur (deep copy)

Classes immuables

- Correspond à des objets immuables
- Un objet immuable est un objet dont les membres ne peuvent être modifiés après création

- Exemple de classes immuables : String, Integer…

Clonage dans la classe HerosTerre

- La classe doit implémenter Cloneable
- Création d'une méthode clone qui retourne un HerosTerre

Méthode clone de HerosTerre

- Création d'un objet HerosTerre
- Appel de la méthode super.clone() à l'intérieur d'un bloc try catch
  - cast de l'objet renvoyé pour utiliser le type HerosTerre
  - Capter l'exception CloneNotSupportedException

Test du clonage

- Dans le main
  - Créer un objet HerosTerre htest
  - Cloner l'objet dans un objet hcopie
  - Changer le nombre de points de vie de htest
  - Vérifier que hcopie n'ait pas été modifié

Classe non clonable

- L'exception CloneNotSupportedException est levée lors de l'appel de la méthode clone sur un attribut d'une classe non clonable (la classe n'implémente pas l'interface Cloneable)

Exercice

- Créer la méthode clone dans les classes HerosMer et Joueur

- Tester le clonage d'objets de ces 2 classes