Correction TP1

On suppose que toutes vos classes sont dans le même package : ici `com.mco`. A la fin du TP, vous aurez deux classes (donc deux fichiers) :

- `Main.java` qui contient le programme principal,
- `Dice.java` pour le dé

Pour exécuter ce programme chez vous :

- créez un projet intellij comme pour le TP1 et placez le package de base à `com.mco`
- créez une classe `Dice` dans le même package (voir cette partie du cours)
- copiez/collez les 2 fichiers.
- exécutez votre programme.

Les classes

On montrera la classe Dice et son utilisation avec la méthode main de la classe Main.

Dice.java

```
package com.mco;

import java.util.Random;

public class Dice {
    private int position; // Méthode générée automatiquement
    private Random randomGenerator;

    public Dice(int positionInitiale) {
        position = positionInitiale;
        randomGenerator = new Random();
    }

    public Dice() {
        this(1);
    }

    public int getPosition() { // Méthode générée automatiquement
        return position;
    }

    public void setPosition(int position) { // Méthode générée
automatiquement
        this.position = position;
    }
```

```
    public void roll() {
        position = randomGenerator.nextInt(6) + 1;
    }

    @Override
    public String toString() { // Méthode générée automatiquement
        return "Dice{" +
                "position=" + position +
                '}';
    }

    @Override
    public boolean equals(Object o) { // Méthode générée automatiquement
        if (this == o) return true;
        if (o == null || getClass() != o.getClass()) return false;

        Dice dice = (Dice) o;

        return getPosition() == dice.getPosition();

    }

    @Override
    public int hashCode() { // Méthode générée automatiquement
        return getPosition();
    }
}
```

Spécificités :

- deux constructeurs, celui sans paramètre appelant celui avec un paramètre,
- le générateur de nombre aléatoire est placé en attribut privé de la classe.
- les méthodes `getPosition`, `setPosition`, `toString`, `equals` et `hashcode` ont été générées automatiquement.

Main.java

```
package com.mco;

public class Main {

    static void testDice() {
        System.out.println("---- Basic Dice tests ----");

        Dice firstDice = new Dice();
        System.out.println(firstDice); // position=1
        firstDice.setPosition(5);
        System.out.println(firstDice); // position=5

        Dice secondDice = new Dice(firstDice.getPosition());
```

```
        System.out.println(firstDice == secondDice); // false
        System.out.println(firstDice.equals(secondDice)); // true
        firstDice.setPosition(2);
        System.out.println(firstDice.equals(secondDice)); // false
    }

    static void testRandom() {
        System.out.println("---- Random tests ----");


        final int NUMBER_TRIES = 100;

        int numberOf1 = 0;
        Dice dice = new Dice();

        for (int i=0; i < NUMBER_TRIES; i+=1 ) {
            dice.roll();

            if (dice.getPosition() == 1) {
                numberOf1 += 1;
            }
        }

        System.out.println("Number of 1: " + numberOf1 + "/" +
NUMBER_TRIES);  // 100 / 6 ~= 17
        System.out.println("Probability: " + (float)numberOf1 /
NUMBER_TRIES); // 1 / 6 ~= 1.167
    }

   public static void main(String[] args) {
        testDice();
        testRandom();
    }
}
```

Spécificités :

- des commentaires pour vérifier que les réponses sont correctes.
- utilisation de méthodes statiques dans la classe Main pour rendre la lecture plus aisée,
- mot clé `final` pour les constantes
- on caste un entier en réel (`float`) pour que la division soit réelle et non entière.

From:
https://wiki.centrale-med.fr/informatique/ - **WiKi informatique**

Permanent link:
https://wiki.centrale-med.fr/informatique/restricted:mco-2:correction:tp1

Last update: **2016/02/15 11:20**