

Laboratoire 10 - Les classes et les objets

1. Il s'agit de définir un animal dont les caractéristiques (attributs) seront :

- son nom (une chaîne de caractères),
- la sorte (un entier : 1 pour chat, 2 pour chien, 0 pour inconnu ou autre),
- la race (une chaîne de caractères),
- le propriétaire (une chaîne de caractères).

Vous devez définir la classe `Animal` où chacun des attributs ne sera pas "private".

Ensuite, créez une instance de la classe `Animal` dans BlueJ (utilisez le menu contextuel en cliquant avec le bouton de droite sur la classe `Animal` : `new Animal()` qui est le constructeur par défaut) et allez consulter le contenu de l'objet `Animal` que vous avez créé (menu contextuel : *inspect*).

2. Définissez une classe `TesterAnimal` dans laquelle il y aura la méthode `main` permettant de tester votre classe `Animal` et apprendre à la manipuler. Votre méthode `main` devra déclarer une variable de type `Animal` (un objet de la classe `Animal`) puis lui affecter les valeurs suivantes :

- nom : "choupette"
- sorte : 1 (chat)
- race : "domestique"
- propriétaire : "Louise"

(Vous pouvez le faire dans votre méthode `main` de la classe `TesterAnimal` puisque les attributs (variables d'instance) ne sont pas privés.)

3. Définissez la méthode d'instance (dans la classe `Animal`) `afficher ()` qui affiche à la console les attributs de l'objet `Animal` puis testez-la en l'appelant de votre méthode `main` après avoir fait les affectations de la question 2. Vous pouvez aussi tester cette méthode en créant une instance de la classe `Animal` (vous pouvez utiliser l'instance que vous avez créée en 1.) et en utilisant le menu contextuel en cliquant avec le bouton de droite sur l'instance créée.

4. Définissez le constructeur sans paramètres pour le type `Animal`. Les valeurs à utiliser pour l'initialisation seront des chaînes vides et la sorte 0. Testez ce constructeur avec votre méthode `main` et ensuite avec le menu contextuel de la classe.

5. Définissez un deuxième constructeur pour le type `Animal`. Le constructeur devra prendre les 4 informations en paramètres. Testez ce constructeur avec votre méthode `main`.

6. Définissez les 4 méthodes d'instance permettant d'obtenir les attributs de l'objet de la classe `Animal`.

7. Dans la classe `TesterAnimal`, définissez une méthode de classe (statique) qui prend en paramètre un objet de la classe `animal` et qui affiche les attributs de l'animal passé en paramètre en utilisant les méthodes d'instance que vous avez définies en 6.

8. Définissez une méthode d'instance dans la classe `Animal` qui permet de modifier le nom de l'animal.

Supplément

1. Définissez une classe `Fraction`. Les attributs sont le numérateur (entier) et le dénominateur (entier positif). Ces variables d'instances seront privées. Définissez deux constructeurs : un sans paramètre fabriquant la fraction 0/1; l'autre prenant en paramètre le numérateur ainsi que le dénominateur. Pour ce deuxième constructeur, si le numérateur ainsi que le dénominateur sont négatifs, il doit mettre ces deux attributs positifs; si le dénominateur est négatif et le numérateur est positif, il doit mettre le signe au numérateur. Définissez une méthode d'instance `simplifier()` qui simplifie la fraction (vous devrez calculer le pgcd (plus grand commun diviseur) entre deux nombres entiers). Définissez les quatre méthodes mettant en oeuvre les quatre opérateurs binaires arithmétiques de base : +, -, * et /.
2. Définissez une classe `CarteAjouer` ayant les attributs suivants :
 - valeur : l'une des 13 valeurs parmi as, 2, 3, 4, ..., 9, 10, valet, dame, roi
 - couleur : l'une des 4 valeurs suivantes : coeur, carreau, trèfle et pique.

Ensuite, définissez le constructeur prenant en paramètre la valeur et la couleur, les deux méthodes "observateur", les deux méthodes "modificateur".