

Travail pratique # 3

Deuxième partie

Ce travail peut être fait en équipe d'au plus 2 personnes.

But : Le but du présent travail est de vous faire faire des classes, vous faire utiliser et manipuler des tableaux ainsi que des fichiers de texte.

Énoncé :

Un de vos amis qui vient de s'ouvrir une petite animalerie de quartier, nommée "Minizou" vous a demandé de l'aider à gérer ses animaux. Le logiciel que vous lui proposez, après mûre réflexion, sera élaboré en Java et fonctionnera à base de menus. L'interaction avec l'utilisateur se fera à la console pour le moment puisque vous n'avez pas beaucoup de temps pour concevoir une première version. Après discussions avec votre ami, vous vous entendez sur le fait que les informations suivantes devront être maintenues pour chacun des animaux :

- La sorte d'animal (0 = chat, 1 = chien, 2 = oiseau, 3 = serpent, 4 = iguane, 5 = hamster, 6 = lapin, 7 = souris, 8 = furet, 9 = autre)
- Des informations sur l'animal (une chaîne de caractères)
- Le nom de l'animal (une chaîne de caractères)
- La date de naissance de l'animal (Date)
- La date de vente de l'animal (Date)
- Si l'animal est vendu ou non (booléen)
- Le prix de l'animal (réel)
- Un numéro unique d'identification (entier)

Pour cette deuxième partie, vous devrez concevoir un programme (des méthodes `public static` avec une méthode `main`) qui utilisera les classes `Animal` (que vous avez conçue pour la première partie) et `Date`. Ce programme devra fonctionner à base de menu. Voici les différents menus et sous-menus :

Menu	Sous-menu	Paramètre(s)	Description	Validations
N		<p><i>nom</i></p> <p><i>sorte</i></p> <p><i>informations</i></p> <p><i>année</i></p> <p><i>prix</i></p>	<p>Ajouter un nouvel animal à la liste d'animaux. Les informations suivantes seront demandées :</p> <ul style="list-style-type: none"> - le nom - la sorte d'animal - les informations sur l'animal - sa date de naissance (saisir l'année puis le mois et finalement, le jour) <p>- le prix de l'animal</p> <p>Lorsque les informations saisies sont valides, le nouvel animal est ajouté. S'il ne reste pas de place (à vérifier avant de saisir les données), un message doit être affiché et vous devez faire ce qui est mentionné plus bas selon que vous travaillez en équipe ou non.</p> <p>L'animal n'a pas de date de vente car il est considéré non vendu.</p>	<p>Vérifier d'abord s'il reste de la place dans la structure qui gère la liste d'animaux.</p> <ul style="list-style-type: none"> - n'importe quelle chaîne de caractères - un entier entre 0 et 9 incl. seulement - n'importe quelle chaîne de caractères - un entier ≥ 0 pour l'année, un nombre entre 1 et 12 inclusivement pour le mois et un nombre entre 1 et le nombre de jours du mois pour le jour - doit être un réel > 0 <p>Dans tous les cas de données non valides, un message d'erreur doit être affiché.</p>
I		<i>numéro</i>	Afficher les informations de l'animal possédant le numéro spécifié.	Le numéro doit exister. Sinon, un message d'erreur doit être affiché.
J		<i>nom</i>	Afficher les informations sur les animaux	

			ayant <i>nom</i> comme nom	
L	*		Afficher les informations sur tous les animaux (autrement dit, ce menu affiche le contenu de la liste d'animaux).	
	<i>n</i>		Afficher les informations sur les animaux de la sorte <i>n</i>	<i>n</i> doit être un nombre entier entre 0 et 9 inclusivement. Sinon, un message d'erreur doit être affiché.
	v		Afficher les informations sur les animaux vendus	
F		<i>numéro</i>	Afficher la date de naissance de l'animal possédant le numéro spécifié.	Le numéro doit exister. Sinon, un message d'erreur doit être affiché.
C		<i>chaîne</i>	Afficher les informations sur les animaux ayant <i>chaîne</i> dans leurs informations	
T		<i>année</i>	Afficher tous les animaux qui sont nés au courant de l'année spécifiée.	L'année doit être un nombre de 4 chiffres. Sinon, un message d'erreur doit être affiché.
X			<u>Si vous travaillez en équipe seulement</u> Afficher tous les animaux qui ont été vendus avant l'âge d'un an.	
K			<u>Si vous travaillez en équipe seulement</u> Afficher, pour chaque sorte d'animal, le total des prix pour les animaux vendus.	
V		<i>numéro</i> <i>date</i>	Vendre l'animal à la date spécifiée	Le numéro doit exister. La date doit être valide et supérieure ou égale à la date de naissance.
M		<i>numéro</i> <i>nom</i>	Modifier le nom de l'animal ayant le numéro spécifié, par le nouveau nom fourni	Le numéro doit exister.
P		<i>numéro</i> <i>prix</i>	Modifier le prix demandé.	Le numéro doit exister. Le prix doit être > 0. Si l'animal est déjà vendu, le prix ne pourra pas être modifié. On doit le mentionner à l'utilisateur et on retourne au menu principal.
?			Afficher le choix des menus	
Q			Quitter	

N.B. Chaque saisie de paramètre doit être faite par boucle de validation. Par exemple, pour le menu N, tant et aussi longtemps que la sorte n'est pas valide, on redemande celle-ci avant de demander la date, etc. On ne revient donc pas au menu principal. Vous devrez valider les saisies non numériques lors de la saisie de nombres.

Vous devez gérer la liste d'animaux à l'aide d'un tableau d'animaux `Animal[]`.

À la fin, lorsque l'utilisateur entre 'Q' comme menu, signifiant qu'il veut quitter le programme, celui-ci doit écrire dans le fichier "c:\animaux.txt" le contenu du tableau d'animaux, c'est-à-dire, écrire les informations, sous format texte, de tous les animaux dans le fichier, une information par ligne, avec une ligne blanche entre chaque animal.

Si vous travaillez seul(e) : Vous devrez déclarer une constante représentant la longueur du tableau d'animaux puis utiliser cette constante pour déclarer le tableau d'animaux. Lorsque l'utilisateur veut créer un nouvel animal, vous devez vérifier s'il reste de la place dans le tableau. S'il ne reste plus de place, ceci voudra dire qu'il ne sera plus possible d'ajouter de nouveaux animaux.

Si vous travaillez en équipe : Vous devez demander à l'utilisateur le nombre maximal d'animaux qui seront stockés (longueur du tableau d'animaux). Lorsque le maximum sera atteint et que l'utilisateur désire à nouveau ajouter un animal, vous devrez "allouer" plus de place dans votre tableau de d'animaux. Puisqu'il n'est pas possible en Java de modifier la longueur d'un tableau, vous devrez tout d'abord déclarer un nouveau tableau de longueur égale au tableau d'abonnés + 5 qu'on appellera tableau de travail, ensuite copier le contenu du tableau d'animaux au début de ce tableau de travail puis changer la référence du tableau d'animaux par la référence du tableau de travail.

Vous pouvez afficher les informations sur un animal comme vous le voulez à la console mais il faut que ce soit clair. Une partie des points sera allouée sur la clarté et l'allure des résultats à la console ainsi que la facilité d'utilisation de votre programme.

Aide

Voici l'algorithme de haut niveau :

```
Demander à l'utilisateur le nombre maximal d'animaux dans la liste (équipes seulement)
afficher le menu
saisir et valider le choix de menu
tant que le choix n'est pas de quitter faire
    si le choix est 'N' alors
        effectuer le menu N
    sinon si le choix est 'I' alors
        effectuer le menu I
        ... (pour les autres menus)
    sinon si le choix est '?' alors
        afficher le menu
    fin si
    saisir et valider le choix
fin tant que
stocker les animaux dans un fichier texte
```

Voici quelques entêtes de méthodes qui pourraient vous être utiles :

```
public static void effectuerMenuN ( Animal[] listeAnimaux ) ... (ou bien Animal[] au lieu de void si vous travaillez en équipe)
public static void stockerDansFichier ( Animal[] listeAnimaux ) ...
public static int saisirEtValiderAnnee ( ) ...
public static Date saisirEtValiderDate ( ) ...
public static void afficherMenus() ...
public static char saisirEtValiderMenu ( )
public static int chercherNumero ( Animal[] listeAnimaux, int numeroCherche ) ...
etc...
```

Exigences logicielles, conseils et contraintes

N'oubliez pas qu'en BlueJ, vous pouvez tester les méthodes de classes publiques en utilisant le menu contextuel avec le bouton de droite de la souris. Vous pouvez donc les tester avant de faire exécuter votre méthode `main`.

0. Les seules classes que vous pouvez utiliser sont : `Tp3, Clavier, Animal, Date, String, java.text.DecimalFormat, Integer, Double, Float, NumberFormatException, IOException`. (Veuillez me consulter si vous voulez utiliser une autre classe)
1. Les différentes saisies et validations (lecture d'une date, lecture d'un menu, lecture d'un sous-menu, lecture d'une sorte, etc.) **doivent** se faire chacune dans une méthode dans votre classe qui contient votre méthode `main`. (Voir exemples d'en-têtes ci-haut.)
2. Le traitement de chaque menu **doit** se faire chacun dans une méthode. (Voir exemples d'en-têtes ci-haut.)
3. Vous **devez** avoir la méthode suivante et l'appeler, soit de la méthode `main`, soit d'une autre méthode :

```
/**
 * Affiche les informations sur l'animal a, à la console
 */
public static void afficherAnimal ( Animal a ) {
    ...
}
```

4. Vous devez suivre les normes de programmation énoncées dans le document disponible sur le site : <http://cyberzoide.developpez.com/java/javastyle/>
5. **Sont interdits :**
 - Les `break` ailleurs que dans un `switch`,
 - Les opérateurs `++` ou `--` ailleurs que dans la troisième partie de l'en-tête d'un `for`, les opérateurs `+=`, ... (i.e. il faut écrire `i = i + 1;` au lieu de `i++` ou `++i`),
 - L'opérateur `?`,
 - Un `return` dans un `for`, un `while`, un `do...while`, un `switch`, un `if`.
 - L'énoncé continue est aussi interdit.
 - Les variables de classe (toute variable utilisée doit être déclarée dans une méthode) dans la classe qui contiendra votre méthode main.
6. Utiliser des variables réelles uniquement lorsque requis. Autrement dit, un compteur ne doit pas être `float` ou `double`.
7. Vous devez utiliser la classe `Clavier` fournie sur le site du cours pour la saisie des données. Vos programmes seront testés avec cette classe et dans l'environnement BlueJ.
8. À moins d'avis contraire, l'affichage des résultats doit se faire à la console.

Ce que vous devez remettre

Partie 2 (80 points) : Le programme `Tp3.java` ainsi que la classe `Animal` sur disquette, CD ou clé USB, à la racine (ne pas placer vos fichiers dans des répertoires) + listages du programme ainsi que de la classe `Animal`.) Mettez le tout dans une enveloppe cachetée.

- **Faites imprimer en mode paysage s'il y a lieu**
- **Si vos lignes sont trop longues et sont coupées sur deux lignes, écrivez des lignes plus courtes**
- **Utilisez la fonte courrier**

Remarques :

Même si aucun point n'est attribué pour la qualité de la présentation du travail, ce critère sera considéré pour la correction et vous pouvez perdre jusqu'à 10 points pour cela.

Aucun travail écrit à la main ne sera accepté.

Les règlements sur le plagiat seront strictement appliqués.

Je vous rappelle que, selon une directive du département, la qualité du français constitue un critère d'évaluation.

Un programme ne compilant pas se verra attribuer la note 0 pour la partie 2.

Aucun programme reçu par courriel ne sera accepté.

Aucune disquette ou CD "zippé(e)" ne sera accepté(e).

Aucune disquette ou CD contenant des virus ne sera accepté(e).

Un travail dont la disquette ou le CD n'est pas accompagné(e) du listage du programme ne sera pas accepté.

Dates et modalités de remise :

Partie 2 : Samedi le 29 avril 2006 avant l'examen final

Aucun travail ne sera accepté après le mercredi 3 mai, 16h

Instructions pour la remise

Si vous remettez votre enveloppe dans la chute à courrier après 17h le vendredi 28 avril, celui-ci sera horodaté le lundi matin seulement (donc, une journée en retard)

Aucun travail glissé sous ma porte de bureau ne sera accepté.