Exercice - 1 Jeu de Craps

Le jeu de craps est un jeu de dés qui se déroule comme suit :

- Un joueur lance deux dés. Soit S la somme des points marqués par les dés. Si S est égale à 7 ou 11, le joueur gagne la partie. Si S est égale à 2, 3 ou 12, le joueur perd la partie. Si S est égale à 4, 5, 6, 8, 9 ou 10, le joueur relance les dés jusqu'à obtenir 7 ou de nouveau S : si le joueur fait 7 il perd la partie, s'il fait S il gagne.

1. Ecrire la classe **De** définie par
   - **MAXFACE** qui représente le nombre maximum de faces
   - **valeurFace** qui représente la valeur courante de la face

   Cette classe comporte les méthodes suivantes :
   (a) un constructeur sans paramétre qui affecte la valeur de la face à 1.
   (b) les getters et les setters.
   (c) la fonction lancer qui retourne la valeur de face de manière aléatoire.
   (d) la fonction afficher.
2. Ecrire la classe **Craps** définie par deux dés de type **De** et qui comporte les méthodes suivantes :
   (a) un constructeur sans paramétre qui génère la valeur des deux dés de manière aléatoire
   (b) les getters et les setters
   (c) la méthode lancer qui lance les deux dés.
   (d) une fonction jeuCraps qui renvoie le score GAGNE ou PERDU.
   (e) la méthode afficher pour afficher l'état des deux dés.
3. Ecrire la classe **TestCraps** qui demande à l'utilisateur de jouer un nombre de parties et de calculer le pourcentage de parties gagnées et le pourcentage de parties perdues.

Exercice - 2 Classe Date

Une date sera simplement représentée par :

- **jour** : un entier compris entre 1 et 31
- **mois** : un entier compris entre 1 et 12.
- **annee** : un entier

1. Ecrire la classe **Date** dans un package **personne** qui comporte les méthodes suivantes :
   (a) les constructeurs et les accesseurs,
   (b) la fonction toString pour convertir une date en un objet String au format "jour/mois/annee".
   (c) la fonction boolean equals(Object objet). Cette classe doit tester si objet est bien une instance de la classe Date. Et dans ce cas, elle renvoie vrai si les jour, mois et année sont égaux.
   (d) la fonction int difference(Date d) qui calcule le nombre de jours écoulés entre la date implicite et la date d fournie en paramètre. Vous aurez besoin de la fonction **nbJours** qui calcule le nombre de jours écoulés depuis une date de référence (laquelle ? mystère...) à partir du jour, du mois et de l'année d'une date (D'après "http ://files.codes-sources.com/fichier.aspx ?id=18523f=Date.c)".

```
static int nbJours(int jour, int mois, int annee){
return (int)((1461 * (annee + 4800 + (mois - 14) / 12)) / 4 +
(367 * (mois - 2 - 12 * ((mois - 14) / 12))) / 12 -
(3 * ((annee + 4900 + (mois - 14) / 12) / 100)) / 4 +
jour - 32075);
}
```

2. Ecrire la classe **TestDate** dans un package **test** qui contient une fonction main destinée pour tester la classe Date. Le programme de test devra notamment permettre de saisir deux dates au clavier, et d'afficher la différence entre les deux.

Exercice - 3 Classe Personne

---

1. Ecrire la classe **Personne** dans le package **personne**. Une personne est décrite par :

   - nom de type String,
   - prenom de type String
   - date de naissance de type Date

   La classe doit contenir

   (a) les accesseurs et les constructeurs,

   (b) la fonction *toString*

   (c) la fonction *equals* (qui renvoie vrai si tous les champs des deux personnes sont égaux).

2. Ecrire la classe **TestPersonne** dans le package **test** pour tester la classe Personne Vous testerez en particulier la situation suivante :

   - Créez un objet Date (par exemple d, et peu importe sa valeur)
   - Créez deux personnes ayant toutes les deux d comme date de naissance
   - Modifiez d (par exemple l'anné) puis affichez les deux personnes

   Vous devez également pouvoir saisir une personne au clavier.

Exercice - 4 Classes Etudiant, Employe, CollectionPersonne

---

1. Ecrire une classe **Etudiant** dans le package **personne** sachant qu'un un étudiant est une personne qui est en plus représentée par :

   - email : une adresse mail (String)

   Elle comporte les méthodes suivantes :

   (a) Les constructeurs et les accesseurs

   (b) Les fonctions *toString* et *equals* qui doivent être implémentées (en tenant compte naturellement de l'héritage)

2. Ecrire une classe **Employe** dans le package **personne** sachant qu'un employé est une personne qui est en plus représentée par :

   - anciennete : une ancienneté (int)

   Elle comporte les méthodes suivantes :

   (a) Les constructeurs et les accesseurs

(b) Les fonctions toString et equals doivent être implémentées (en tenant compte naturellement de l'héritage)

3. Ecrire une classe **CollectionPersonne** qui permet de gérer une collection de personnes. Cette classe est éfinie par : T : un tableau de Personne de type **ArrayList**. Cette classe comporte

   (a) un constructeur

   (b) une fonction ajouter qui ajoute une personne à la collection

   (c) une fonction afficher qui affiche la collection

4. Dans le package **test**, créez une nouvelle classe **TestCollection** pour tester Etudiant et Employe. En particulier, afin de vérifier le comportement de l'héritage en JAVA, vous contruirez un tableau de personnes dans lequel vous rangerez dans un ordre arbitraire, des personnes, des agents, et des étudiants. Ensuite, vous afficherez ce tableau (grâce à toString()). Vous comparerez également des personnes entre elles (grâce à equals()).