

Services sur réseaux - MMI – S4 – Développement jeu

TP1 : introduction à la programmation objet avec java

Exercice 1 :

1. Ecrire un programme (classe *Numerique.cs*) contenant un ensemble de méthodes statiques :
 - A. Une méthode qui calcule la somme des premiers entiers pairs inférieurs à n (n étant passé en paramètre à la méthode, ex. $n = 10$, $2+4+6+8 = 20$).
 - B. Une méthode qui affiche à l'écran les coefficients de la décomposition d'un entier en puissances de 10 (ex. $542 = 2+4*10+5*100$).
 - C. Une méthode qui indique si un entier est divisible par un autre.
 - D. La méthode *Main* permettant de lancer le programme pour un test particulier. Un ensemble d'arguments doivent ainsi être passés au programme. Utilisez la structure de contrôle *switch* qui en fonction de la valeur du premier argument (chaîne de caractères ayant une des valeurs : SOMME, PUISSANCE ou DIV) détermine la méthode à invoquer et décode les arguments nécessaires à son exécution avant de l'invoquer.
2. Modifier le programme précédent pour qu'il prenne en charge le cas des entiers négatifs en déclenchant une exception *NegativeNumber* et le cas du diviseur nul en déclenchant une exception *NullNumber*. Les deux exceptions doivent être déclarées. Elles doivent également être interceptées et traitées dans le programme principal (*Main*).

Exercice 2 :

Cet exercice a pour but d'illustrer l'utilisation des classes abstraites.

Le salaire d'un ensemble d'employés doit être calculé. Une superclasse abstraite *Employe* sera créée en vue de traiter de manière polymorphe les différents employés de l'entreprise.

La classe *Employe* décrit un salarié avec son nom et permet d'afficher le nom et le salaire de chaque employé (méthode *afficher()*). La méthode abstraite *calculerSalaire()* permet de calculer le salaire d'un employé, lequel dépend du type du salarié en question.

Les employés sont répartis en 3 catégories et donc en 3 classes distinctes :

1. *EmployeFixe* : recevant un salaire fixe connu au moment de sa création (salaire mensuel = salaire) ;
2. *EmployeCommission* : recevant un traitement de base (connu au moment de la création de l'employé) plus un pourcentage lié à ses ventes (salaire mensuel = salaire + ventes * commission). La méthode *setNbVentes(int nbVentes)* permet de spécifier le nombre de ventes ;
3. *EmployeHoraire* : recevant une paie selon le nombre d'heures prestées (salaire mensuel = taux horaire * heures prestées). La méthode *setNbHeures(int nbHeures)* permet de spécifier le nombre d'heures effectuées. Le taux horaire peut être fixé à 15.

Créez une classe publique *GestionSalaires.cs* contenant la méthode *Main()* et qui permettra de tester le programme en créant trois employés : un par catégorie et en spécifiant toutes les données nécessaires. Par la suite, la méthode *afficher* sera appelée pour afficher les informations relatives à chaque employé.