

Algorithmique Programmation (2^{nde} partie)

Exercices dirigés

Programmation objet

23 Janvier 2008

1 Enregistrements et objets

Un compte bancaire est représenté par son numéro, par le nom de son titulaire et par son solde courant. On peut faire des opérations de débit, de crédit sur un compte, ainsi que des virements d'un compte à un autre. On peut également afficher les propriétés d'un compte (numéro, titulaire et solde).

Ecrire un programme qui crée un compte pour Marie avec 100 euros et un autre pour Pierre avec 200 euros, ensuite débite les deux comptes de 10 euros et fait un virement de 50 euros du compte de Pierre vers celui de Marie. A la fin, le programme affiche les deux comptes.

1. Ecrire d'abord le programme en définissant les comptes comme des enregistrements.
2. Modifier le programme pour utiliser la notion d'objet.
3. Modifier le programme du point précédent pour générer automatiquement les numéros de compte : le premier compte créé aura le numéro 1, le second 2, etc.

2 Conception de classe

Un salarié est représenté par son nom, son numéro de badge et son salaire. On peut lui modifier le numéro de badge ou son salaire. Le salaire est modifié une fois par an, quand on applique une augmentation d'un pourcentage donné, le même pour tous les salariés. Ce pourcentage peut être lui aussi modifié chaque année.

Ecrire la classe `Salarie` respectant les spécifications ci-dessus.

3 Méthodes statiques et non-statiques

Parmi les méthodes de la classe suivante, lesquelles peuvent être statiques et lesquelles ne peuvent en aucun cas être statiques ?

```
class Exo2{  
    int x, y;  
    String nom;  
    void afficher(){  
        Terminal.ecrireString(nom + " " + x + " " + y);  
    }  
}
```

```

}
void ajouter(Exo2 obj){
    x = x + obj.x;
    y = y + obj.y;
    nom = nom + obj.nom;
}
Exo2 nouveau(int n){
    Exo2 res = new Exo2();
    res.x = n;
    res.y = n*2;
    res.nom = "Auto_" + n;
    return res;
}
boolean plusGrand(Exo2 obj){
    if (obj.x == x){
        return y > obj.y;
    } else {
        return x > obj.x;
    }
}
boolean compare(Exo2 obj1, Exo2 obj2){
    if (obj1.x == obj2.x){
        return obj1.y > obj2.y;
    } else {
        return obj1.x > obj2.x;
    }
}
}
}

```
