

Objets (sans fonctions) (TD 5)

Exercices table: 3, 4, 5.

Rappel: Respectez les consignes de la page *Fonctionnement* et celles données dans les feuilles précédentes.

Dans ce TD, on n'écrira aucune fonction (excepté les constructeurs), et on ne considérera pas de tableaux d'objets. En effet, ni les unes (sur les objets) ni les autres n'ont été vus en cours.

1. Placez-vous dans le répertoire adéquat demandé dans la page *Environnement*.
2. **Familiarisation** Tapez et exécutez le programme suivant de la feuille d'exemples du cours, en respectant les consignes de la page *Fonctionnement*: le dernier.
3. Le but de cet exercice est d'obtenir le programme de la dernière question. Cette construction est décomposée en questions progressives. Après chaque question, compilez, exécutez et testez votre programme avant de commencer la question suivante.

On considère des comptes bancaires simples, avec un numéro, un solde et un salaire (des entiers). Le "salaire" est la rentrée mensuelle du détenteur du compte. (Pour simplifier on ne représente pas de client dans cet exercice.) Chaque compte a aussi une autorisation de découvert, dont le montant est un pourcentage du salaire. Le taux (pourcentage) est le même pour tous les comptes.

Dans cet exercice et dans toute la suite, chaque fois qu'un compte sera saisi, ce sera dans l'ordre: numéro, solde, salaire. De même pour les affichages.

Dans cet exercice et dans toute la suite, "afficher toutes les informations de l'application" signifie afficher le taux, puis tous les comptes existants, y compris bien sûr le montant de leur autorisation de découvert, et le cas échéant les autres entités présentes.

- (a) Ecrivez un programme complet, qui crée un compte et l'affiche. Les informations du compte apparaîtront directement dans le code, sans saisie au clavier.
 - (b) On s'intéresse maintenant au taux permettant de calculer l'autorisation de découvert: c'est 10 %. Complétez votre programme.
 - (c) On considère maintenant que le taux est saisi au clavier.
 - (d) On considère maintenant deux nouveaux comptes, dont les informations sont saisies au clavier. Affichez toutes les informations de l'application. Affichez le nom du client qui a le salaire le plus élevé.
 - (e) Modifiez maintenant, par une ou des instructions dans votre programme, l'autre compte saisi pour qu'il ait le même salaire que le premier. Affichez toutes les informations de l'application.
4. Cet exercice n'a aucun lien avec le précédent. Le placer dans un nouveau fichier (et de même dans toute la suite). On considère des clients, avec un nom, une ville, et un numéro de compte (un entier). (Pour simplifier, deux clients ne peuvent avoir le même compte.) Saisir deux clients. Afficher toutes les informations de l'application.
 5. Regrouper les deux exercices précédents dans une seule application (vous pourrez vous inspirer du dernier programme de la feuille d'exemples du cours). Chaque client sera saisi avec son compte (dans l'ordre à chaque fois: nom, ville, numéro, solde, salaire). (Vous retirerez de l'exercice 3 le

compte non saisi, ainsi que les comparaisons et modifications de comptes;) (Pour simplifier, on ne fera aucune vérification, par exemple que les numéros sont différents, ainsi que dans toute la suite.)

6. Dans cet exercice, on n'utilise aucun objet, *ni fonction*. (On revient donc au matériel Java du TD 3.) On reprend l'exercice 3 ci-dessus, mais on va gérer un ensemble de comptes comme suit. On va représenter l'ensemble des comptes par trois tableaux: *numéro* pour les numéros de compte, *solde* pour les soldes, et *salaire* pour les salaires. Le *i*-ème compte sera représenté par les trois valeurs: *numero[i]*, *solde[i]*, *salaires[i]*. (C'est le même principe que celui utilisé dans l'exercice 13 du TD 3, puis dans le "récapitulatif global" du TD 4b.)

Saisir le taux et un nombre quelconque de comptes au clavier. Afficher toutes les informations de l'application. Afficher le montant de l'autorisation de découvert maximale.

7. Reprenez l'exercice précédent, mais bien sûr avec des fonctions.

8. *Déroulement mémoire. Travail personnel.*

On considère l'exécution du programme suivant (il compile sans erreur).

- (a) Donner la configuration de la mémoire immédiatement après l'exécution de la ligne contenant le commentaire: `// ici`. Vous utiliserez un croquis détaillé et l'algorithme vu en cours et TD. En particulier, on allouera obligatoirement les cases mémoire dans l'ordre croissant à partir de 101. Comme en cours et en TD, si une case contient successivement plusieurs valeurs, on les écrira de gauche à droite dans la case en les barrant d'un seul trait au fur et à mesure. De même, on ne réutilisera pas les zones de fonctions.
- (b) Donner le dernier état de la mémoire avant l'arrêt du programme.
- (c) Dire ce qu'affiche ce programme.

```
class Bidule {
    static int c = 1;
    int a1;
    int a2;
}

class Memoire {
    public static void main (String[] args ) {
        Bidule b = new Bidule();
        b.a2 = b.a1;
        b.a1 = Bidule.c;
        Bidule.c++;
        int n = 0;
        n++;
        b.a1 = Bidule.c;
        Bidule d = b;
        d.a1++;
        Bidule.c++;
        int m = n;
        System.out.println(m+" "+n+" "+b.a1+" "+b.a2+" "+Bidule.c); // ici
        n++;
        b = null;
        System.out.println(m+" "+n+" "+b.a1+" "+b.a2+" "+Bidule.c);
    }
}
```

Si vous avez fini tous les exercices ci-dessus avant la fin du TD

9. Etendre l'exercice 4 (clients seuls) à un ensemble de clients, sans objets, comme c'est fait pour les comptes dans l'exercice 6, et avec des fonctions, comme dans l'exercice 7.
10. Regrouper l'exercice précédent avec l'exercice 7 dans une seule application. Les comptes correspondront aux clients pour simplifier. Chaque client sera affiché avec son compte. Afficher toutes les informations de l'application, ainsi que le montant de l'autorisation de découvert maximale, mais *en passant par les clients, sans passer par le tableau des comptes*.
11. On considère des billets avec pour chacun le nom du client, la destination au départ de Paris, le kilométrage, et le prix, calculé à partir du prix du kilomètre.

Saisir le prix du kilomètre au clavier. Saisir deux billets au clavier. Affichez toutes les informations de l'application. Affichez le nom de la personne qui va le plus loin. Modifiez ensuite, par une ou des instructions dans votre programme, l'autre billet pour qu'il corresponde à la destination (et au kilométrage) du premier. Réaffichez bien sûr les deux billets pour vérifier.