

Travaux pratiques

séance n°5

Méthodes

Exercice 1 : passage de paramètres par valeur

```
public class Exo1{
    public static void main(String[] args){
        int n1 = 1;
        int n2 = 3;
        System.out.println("avant permutation, n1="+n1+" et n2="+n2);
        permuter(n1,n2);
        // question 3 : quelle est la valeur de n1 et n2 ?
        System.out.println("après permutation, n1="+n1+" et n2="+n2);
    }

    public static void permuter( int x,int y ){
        // question 1 : quelle est la valeur de x et y ?
        int temp = x;
        x = y;
        y = temp;
        // question 2 : quelle est la valeur de x et y ?
    }
}
```

Avant d'exécuter ce petit programme, répondre aux 3 questions soumises en commentaire.
Pour répondre correctement, il faut avoir compris le mécanisme de la transmission de paramètres par valeur.

Exercice 2 : passage de paramètres par valeur

```
public class Exo2{
    public static void main(String[] args){
        String s1 = "hello";
        String s2 = "world";
        System.out.println("avant permutation, s1="+s1+" et s2="+s2);
        permuter(s1,s2);
        // question 3 : quelle est la valeur de s1 et s2 ?
        System.out.println("après permutation, s1="+s1+" et s2="+s2);
    }

    public static void permuter( String x,String y ){
        // question 1 : quelle est la valeur de x et y
        System.out.println("in - avant permutation, x="+x+" et y="+y);
        String temp = x;
        x=y;
        y=temp;
        // question 2 : quelle est la valeur de x et y
        System.out.println("in - après permutation, x="+x+" et y="+y);
    }
}
```

Mêmes questions qu'au précédent exercice. En quoi cet exercice est-il différent ?

Exercice 3 : passage de paramètres par valeur

```
public class Exo3{
    public static void main(String[] args){
        String abc = "abcd";
        String wxy = "wxyz";
        char[] y = abc.toCharArray();
        char[] z = wxy.toCharArray();
        System.out.println("avant z="+z);
        System.out.println("avant z[0]="+z[0]);
        permuter(y,z);
        // question 3 : quelles sont les valeurs de y et y[0] ?
        System.out.println("après y="+y);
        System.out.println("après y[0]="+y[0]);
    }

    public static void permuter(char[] a, char[] b){
        // question 1 : quelles sont les valeurs de a et a[0] ?
        char tmp;
        for(int i=0;i<a.length;i++){
            tmp = a[i];
            a[i] = b[i];
            b[i] = tmp;
        }
        // question 2 : quelles sont les valeurs de a et a[0] ?
    }
}
```

Mêmes questions qu'au précédent exercice. Expliquez le résultat.

Exercice 4 : passage de paramètres par valeur

```
public class Exo4{
    public static void main(String[] args){
        Double d1 = new Double(5);
        Double d2 = new Double(12);
        System.out.println("avant permutation, d1="+d1+" et d2="+d2);
        permuter(d1,d2);
        // question 3 : quelle est la valeur de d1 et d2 ?
        System.out.println("après permutation, d1="+d1+" et d2="+d2);
    }

    public static void permuter( Double x, Double y ){
        // question 1 : quelle est la valeur de x et y
        System.out.println("in - avant permutation, x="+x+" et y="+y);
        Double temp = x;
        x=y;
        y=temp;
        // question 2 : quelle est la valeur de x et y
        System.out.println("in - après permutation, x="+x+" et y="+y);
    }
}
```

Exercice 5 : génération aléatoire d'un mot de passe

L'ensemble des caractères traités par les programmes java est codé selon le format Unicode. Ce format permet le codage de 65536 caractères. Leur expression hexadécimale s'étend entre 0 et FFFF. Leur expression décimale entre 0 à 65535.

L'Unicode de 'a' est : `(int) 'a'`. les lettres minuscules suivantes sont les entiers consécutifs.

Pour générer aléatoirement un entier, on utilisera la méthode `Math.random()` qui retourne un réel compris entre 0.0 et 1.0 :

```
0.0<=Math.random()<1.0
```

L'expression suivante permettra de générer un entier correspondant à un caractère quelconque de l'Unicode :

```
(int) (Math.random() * (65535+1))
```

Les "Unicode" des lettres minuscules sont des entiers consécutifs; de même pour les majuscules; de même pour les chiffres.

En Java, les opérations arithmétiques peuvent être appliquées aux valeurs de type `char`.

Si l'on veut générer une lettre minuscule, on évaluera l'expression :

```
'a'+Math.random() * ('z'-'a'+1)
```

Le résultat étant un caractère, il faudra la convertir en `char` :

```
(char) ('a'+Math.random() * ('z'-'a'+1))
```

Nous allons construire pas à pas un programme capable de générer un mot de passe de 8 caractères. composé aléatoirement de lettres minuscules, majuscules et de chiffres

Question 1: écrire une méthode qui retourne un caractère aléatoire.

Question 2: écrire une méthode qui retourne un caractère aléatoire compris entre deux caractères quelconques

Question 3: écrire une méthode qui retourne une lettre minuscule aléatoire

Question 4: écrire une méthode qui retourne une lettre majuscule aléatoire

Question 5: écrire une méthode qui retourne un chiffre aléatoire

Question 6: écrire le programme qui affiche le mot de passe aléatoire de 8 caractères.

Question 7: modifier le programme de manière à générer un mot de passe d'une longueur aléatoire comprise entre 8 et 24 caractères.