

Exercices pour Initiation à l'informatique et à l'algorithmique

Exo 1

Énoncé : Écrivez un programme qui lit l'âge en années et affiche l'âge en jours.

```
1 import java.util.Scanner;
2
3 public class AgeEnJours {
4     public static void main(String[] args) {
5         // Création d'un scanner pour lire l'entrée
           utilisateur
6         Scanner scanner = new Scanner(System.in);
7
8         // Demander à l'utilisateur d'entrer son âge
9         System.out.println("Entrez votre age en annees : ");
10        int ageEnAnnees = scanner.nextInt();
11
12        // Calcul de l'âge en jours (approximation : 1 an =
           365 jours)
13        int ageEnJours = ageEnAnnees * 365;
14
15        // Affichage du résultat
16        System.out.println("Votre age en jours est d'environ
           : " + ageEnJours + " jours.");
17
18        // Fermeture du scanner
19        scanner.close();
20    }
21 }
```

Exo 2

Énoncé : Soit $\triangle KPH$ un triangle rectangle en P tel que : $KP = 7,2$ cm et $HP = 15,4$ cm. Écrivez un programme pour calculer la longueur HK .

```
1 public class TrianglePythagore {
2     public static void main(String[] args) {
3         double KP = 7.2; // Longueur de KP en cm
4         double HP = 15.4; // Longueur de HP en cm
5
6         // Calcul de HK à l'aide du théorème de Pythagore :
7         HK = KP + HP
8         double HK = Math.sqrt(Math.pow(KP, 2) + Math.pow(HP,
9             2));
10
11         // Affichage du résultat
12         System.out.printf("La longueur de l'hypotenuse HK
13             est : %.2f", HK);
14     }
15 }
```

Exo 3

Énoncé : Écrire un programme Java pour calculer l'aire de trois triangles et afficher la somme totale.

```
1 public class SommeAires {
2     public static void main(String[] args) {
3         // Données des triangles
4         double b1 = 5.0, h1 = 7.0; // Base et hauteur du
           triangle 1
5         double b2 = 8.5, h2 = 6.0; // Base et hauteur du
           triangle 2
6         double b3 = 10.0, h3 = 4.5; // Base et hauteur du
           triangle 3
7
8         // Calcul des aires
9         double aire1 = (b1 * h1) / 2;
10        double aire2 = (b2 * h2) / 2;
11        double aire3 = (b3 * h3) / 2;
12
13        // Calcul de la somme des aires
14        double sommeAires = aire1 + aire2 + aire3;
15
16        // Affichage des résultats
17        System.out.println("Aire du triangle 1 : " + aire1 +
           " cm ");
18        System.out.println("Aire du triangle 2 : " + aire2 +
           " cm ");
19        System.out.println("Aire du triangle 3 : " + aire3 +
           " cm ");
20        System.out.println("Somme des aires : " + sommeAires
           + " cm ");
21    }
22 }
```

Exo 4

Énoncé : Écrivez un programme en Java qui demande à l'utilisateur d'entrer un nombre de base et un nombre d'exposant. Le programme doit ensuite calculer la puissance en utilisant une boucle et afficher le résultat.

```
1 import java.util.Scanner;
2
3 public class PuissanceAvecBoucle {
4     public static void main(String[] args) {
5         // Création d'un scanner pour lire l'entrée
5         // utilisateur
6         Scanner scanner = new Scanner(System.in);
7
8         // Demander à l'utilisateur d'entrer la base et l'
8         // exposant
9         System.out.println("Entrez un nombre de base : ");
10        int base = scanner.nextInt();
11        System.out.println("Entrez un nombre d'exposant : ")
12        ;
13        int exposant = scanner.nextInt();
14
15        // Initialiser le résultat à 1
16        int resultat = 1;
17
18        // Calculer la puissance avec une boucle
19        for (int i = 0; i < exposant; i++) {
20            resultat *= base;
21        }
22
23        // Affichage du résultat
24        System.out.println("Le resultat est : " + resultat);
25        scanner.close();
26    }
27 }
```

Exo 5

Énoncé : Écrivez un programme en Java qui demande à l'utilisateur de saisir une chaîne de caractères. Le programme doit compter et afficher le nombre de fois où **xx** apparaît dans la chaîne donnée. Le comptage doit inclure les cas où les **xx** se chevauchent.

```
1 import java.util.Scanner;
2
3 public class CompteXX {
4     public static void main(String[] args) {
5         // Création d'un scanner pour lire l'entrée
5         // utilisateur
6         Scanner scanner = new Scanner(System.in);
7
8         // Demander à l'utilisateur d'entrer une chaîne
9         System.out.println("Entrez une chaîne : ");
10        String chaine = scanner.nextLine();
11
12        // Initialiser le compteur pour le nombre de "xx"
13        int compteur = 0;
14
15        // Parcourir la chaîne pour compter les "xx" avec
15        // chevauchement
16        for (int i = 0; i < chaine.length() - 1; i++) {
17            if (chaine.substring(i, i + 2).equals("xx")) {
18                compteur++;
19            }
20        }
21
22        // Affichage du résultat
23        System.out.println("Le nombre de 'xx' est : " +
23        // compteur);
24        scanner.close();
25    }
26 }
```

Exo 6

Énoncé : Créez un programme qui prend deux entiers en entrée, compare leurs valeurs et indique si le premier est égal, inférieur ou supérieur au second.

```
1 import java.util.Scanner;
2
3 public class CompareIntegers {
4     public static void main(String[] args) {
5         // Création d'un scanner pour lire l'entrée
           utilisateur
6         Scanner scanner = new Scanner(System.in);
7
8         // Demander les deux entiers à l'utilisateur
9         System.out.println("Entrez le premier entier : ");
10        int premier = scanner.nextInt();
11        System.out.println("Entrez le deuxieme entier : ");
12        int deuxieme = scanner.nextInt();
13
14        // Comparaison des deux entiers
15        if (premier == deuxieme) {
16            System.out.println("Les entiers sont egaux.");
17        } else if (premier < deuxieme) {
18            System.out.println("Le premier entier est
                inferieur.");
19        } else {
20            System.out.println("Le premier entier est
                superieur.");
21        }
22
23        // Fermeture du scanner
24        scanner.close();
25    }
26 }
```

Exo 7

Énoncé : Écrivez un programme en Java qui demande à l'utilisateur de saisir deux entiers, a et b. Le programme doit vérifier si l'un des deux nombres est égal à 10 ou si leur somme est égale à 10.

```
1 import java.util.Scanner;
2
3 public class TestValeurs {
4     public static void main(String[] args) {
5         // Création d'un scanner pour lire l'entrée
5         // utilisateur
6         Scanner scanner = new Scanner(System.in);
7
8         // Demander les deux entiers à l'utilisateur
9         System.out.println("Entrez deux entiers : ");
10        int a = scanner.nextInt();
11        int b = scanner.nextInt();
12
13        // Vérification des conditions
14        boolean resultat = (a == 10 || b == 10) || (a + b ==
15                               10);
16
17        // Affichage du résultat
18        System.out.println("Resultat : " + resultat);
19        scanner.close();
20    }
21 }
```

Exo 8

Énoncé : Écrivez un programme qui, à partir d'une chaîne de caractères donnée, affiche une version où tous les x ont été supprimés, sauf ceux situés au tout début ou à la fin de la chaîne.

```
1 import java.util.Scanner;
2
3 public class SupprimeX {
4     public static void main(String[] args) {
5         // Création d'un scanner pour lire l'entrée
5         // utilisateur
6         Scanner scanner = new Scanner(System.in);
7
8         // Demander une chaîne à l'utilisateur
9         System.out.println("Entrez une chaîne : ");
10        String chaine = scanner.nextLine();
11
12        // Vérifier si la chaîne a au moins deux caractères
13        if (chaine.length() <= 1) {
14            System.out.println("Resultat : " + chaine);
15            return;
16        }
17
18        // Garder le premier et le dernier caractère
19        char premier = chaine.charAt(0);
20        char dernier = chaine.charAt(chaine.length() - 1);
21
22        // Extraire la partie intermédiaire et supprimer les
22        // 'x'
23        String milieu = chaine.substring(1, chaine.length()
23        - 1).replace("x", "");
24
25        // Construire la nouvelle chaîne
26        String resultat = premier + milieu + dernier;
27
28        // Afficher le résultat
29        System.out.println("Resultat : " + resultat);
30    }
31 }
```


Exo 9

Énoncé : Écrivez un programme qui compte le nombre de fois où la chaîne code apparaît dans une chaîne donnée, en acceptant n'importe quelle lettre à la place du d (par exemple, cope et coze sont aussi comptés).

```
1 import java.util.Scanner;
2
3 public class CompteCode {
4     public static void main(String[] args) {
5         // Création d'un scanner pour lire l'entrée
5         // utilisateur
6         Scanner scanner = new Scanner(System.in);
7
8         // Demander une chaîne à l'utilisateur
9         System.out.println("Entrez une chaîne : ");
10        String chaine = scanner.nextLine();
11
12        // Initialiser le compteur
13        int compteur = 0;
14
15        // Parcourir la chaîne
16        for (int i = 0; i <= chaine.length() - 4; i++) {
17            // Vérifier si la sous-chaîne correspond à "coXe
18            // " (où X est une lettre)
19            if (chaine.charAt(i) == 'c' && chaine.charAt(i +
20                1) == 'o' &&
21                chaine.charAt(i + 3) == 'e') {
22                compteur++;
23            }
24        }
25
26        // Affichage du résultat
27        System.out.println("Le nombre de 'coXe' est : " +
28            compteur);
29
30        // Fermeture du scanner
31        scanner.close();
32    }
33 }
```

Exo 10

Énoncé : Créez un programme en Java qui lit un nombre entier saisi par l'utilisateur et vérifie s'il est présent dans un tableau de nombres entiers prédéfini. Le programme doit ensuite afficher un message indiquant si le nombre saisi est présent ou non dans le tableau.

```
1 import java.util.Scanner;
2
3 public class ContientNombre {
4     public static void main(String[] args) {
5         // Création d'un scanner pour lire l'entrée
5         // utilisateur
6         Scanner scanner = new Scanner(System.in);
7
8         // Définir le tableau
9         int[] tableau = {3, 7, 15, 20, 42, 8, 13, 27};
10
11        // Demander le nombre à rechercher
12        System.out.println("Entrez un nombre a rechercher :
12        ");
13        int nombreRecherche = scanner.nextInt();
14
15        // Initialiser une variable pour suivre si le nombre
15        // est trouvé
16        boolean trouve = false;
17
18        // Parcourir le tableau pour rechercher le nombre
19        for (int nombre : tableau) {
20            if (nombre == nombreRecherche) {
21                trouve = true;
22            }
23        }
24
25        // Afficher le résultat
26        if (trouve) {
27            System.out.println("Le nombre est present dans
27            le tableau.");
28        } else {
29            System.out.println("Le nombre n'est pas present
29            dans le tableau.");
30        }
31
32        // Fermeture du scanner
33        scanner.close();
34    }
35 }
```

Exo 11

Énoncé : Créez un programme qui commence par rechercher le plus petit et le plus grand nombre dans un tableau donné. Ensuite, il calcule la somme de ces deux nombres. À partir de cette somme, le programme génère un nouveau tableau contenant tous les nombres entiers consécutifs, de 1 jusqu'à la somme obtenue.

```
1 import java.util.Scanner;
2
3 public class TableauCalcul {
4     public static void main(String[] args) {
5         // Création d'un scanner pour lire les entrées
5         // utilisateur
6         Scanner scanner = new Scanner(System.in);
7
8         // Demander la taille du tableau
9         System.out.println("Entrez la taille du tableau : ")
9         ;
10        int taille = scanner.nextInt();
11
12        // Déclarer le tableau
13        int[] tableau = new int[taille];
14
15        // Remplir le tableau avec des valeurs
16        System.out.println("Entrez les éléments du tableau : ")
16        ;
17        for (int i = 0; i < taille; i++) {
18            tableau[i] = scanner.nextInt();
19        }
20
21        // Rechercher le plus petit et le plus grand nombre
21        // dans le tableau
22        int min = tableau[0];
23        int max = tableau[0];
24
25        for (int i = 1; i < tableau.length; i++) {
26            if (tableau[i] < min) {
27                min = tableau[i];
28            }
29            if (tableau[i] > max) {
30                max = tableau[i];
31            }
32        }
33
34        // Calculer la somme des deux nombres
35        int somme = min + max;
36
37        // Créer un tableau allant de 1 à la somme
38        int[] tableauRange = new int[somme];
```

```

39
40     for (int i = 0; i < somme; i++) {
41         tableauRange[i] = i + 1; // Remplir le tableau
42         avec les nombres de 1 à somme
43     }
44
45     // Afficher le tableau résultant
46     System.out.println("Le tableau généré allant de 1 à
47         " + somme + " est : ");
48     for (int i : tableauRange) {
49         System.out.print(i + " ");
50     }
51
52     // Fermeture du scanner
53     scanner.close();

```

Exo 12

Énoncé : Créez un programme qui demande à l'utilisateur de saisir trois phrases via la console. Écrivez ces phrases, ligne par ligne, dans un fichier nommé extttphrases.txt à l'aide de extttBufferedWriter. Finalement, lisez le contenu du fichier ligne par ligne à l'aide de extttBufferedReader et affichez chaque ligne dans la console.

```
1 import java.io.*;
2 import java.util.Scanner;
3
4 public class FileReadWrite {
5     public static void main(String[] args) throws
6         IOException {
7         // Nom du fichier
8         String fileName = "phrases.txt";
9
10        // 1.  criture  dans le fichier
11        BufferedWriter writer = new BufferedWriter(new
12            FileWriter(fileName));
13        Scanner scanner = new Scanner(System.in);
14        System.out.println("Veuillez entrer 3 phrases :");
15        for (int i = 1; i <= 3; i++) {
16            System.out.print("Entrez la phrase " + i + " : "
17                );
18            String phrase = scanner.nextLine();
19            writer.write(phrase); //  crire  la phrase dans
20                le fichier
21            writer.newLine();      // Aller à la ligne
22                suivante
23        }
24        writer.close(); // Fermer le BufferedWriter
25
26        // 2. Lecture du fichier
27        System.out.println("\nContenu du fichier :");
28        BufferedReader reader = new BufferedReader(new
29            FileReader(fileName));
30        String line;
31        while ((line = reader.readLine()) != null) {
32            System.out.println(line); // Afficher chaque
33                ligne
34        }
35        reader.close(); // Fermer le BufferedReader
36        scanner.close(); // Fermer le Scanner
37    }
38 }
```

Exo 13

Énoncé : Créez une fonction qui prend deux nombres comme arguments (extttnum, extttlength) et renvoie un tableau de multiples de extttnum jusqu'à ce que la longueur du tableau atteigne extttlength.

```
1 import java.util.Arrays;
2
3 public class MultiplesArray {
4     public static int[] generateMultiples(int num, int
5         length) {
6         // Créer un tableau pour contenir les multiples
7         int[] multiples = new int[length];
8
9         // Remplir le tableau avec les multiples
10        for (int i = 0; i < length; i++) {
11            multiples[i] = num * (i + 1);
12        }
13
14        return multiples;
15    }
16
17    public static void main(String[] args) {
18        int[] result = generateMultiples(5, 10);
19        System.out.println("Multiples de 5 : " + Arrays.
20            toString(result));
21    }
22 }
```

Exo 14

Énoncé : Créez un programme qui traite un tableau de chaînes représentant des accords musicaux. Pour chaque accord dans le tableau, ajoutez le chiffre exttt7 à la fin, sauf si l'accord se termine déjà par exttt7. Si le tableau est vide, le programme doit renvoyer un tableau vide.

```
1 import java.util.Arrays;
2
3 public class JazzifyChords {
4     public static String[] jazzify(String[] chords) {
5         if (chords.length == 0) {
6             return new String[0]; // Retourner un tableau
              vide si aucun accord
7         }
8
9         // Créer un tableau pour stocker les accords modifié
              s
10        String[] jazzifiedChords = new String[chords.length
              ];
11
12        for (int i = 0; i < chords.length; i++) {
13            if (chords[i].endsWith("7")) {
14                jazzifiedChords[i] = chords[i]; // Garde l'
              accord tel quel
15            } else {
16                jazzifiedChords[i] = chords[i] + "7"; //
              Ajoute "7"
17            }
18        }
19
20        return jazzifiedChords;
21    }
22
23    public static void main(String[] args) {
24        String[] accords = {"G", "F", "C", "F7"};
25        System.out.println("Accords jazzifiés : " + Arrays.
              toString(jazzify(accords)));
26    }
27 }
```

Exo 15

Énoncé : Étant donné une chaîne, créez une fonction pour inverser la casse. Toutes les lettres minuscules doivent être majuscules et vice versa.
Exemple : `reverseCase("MANY THANKS")` renvoie "many thanks".

```
1 public class ReverseCase {
2     public static String reverseCase(String str) {
3         StringBuilder result = new StringBuilder();
4
5         for (int i = 0; i < str.length(); i++) {
6             char c = str.charAt(i);
7
8             if (Character.isUpperCase(c)) {
9                 result.append(Character.toLowerCase(c)); //
10                    Convertir en minuscule
11             } else if (Character.isLowerCase(c)) {
12                 result.append(Character.toUpperCase(c)); //
13                    Convertir en majuscule
14             } else {
15                 result.append(c); // Garder les caractères
16                    non alphabétiques
17             }
18         }
19
20         return result.toString();
21     }
22
23     public static void main(String[] args) {
24         System.out.println(reverseCase("MANY THANKS")); //
25            Affiche "many thanks"
26         System.out.println(reverseCase("sPoNtAnEoUs")); //
27            Affiche "SpOnTaNeOuS"
28     }
29 }
```