

EXERCICES EN JAVA:

Objectifs :

Ces exercices visent à approfondir votre compréhension des bases de Java tout en consolidant vos compétences en variables, conditions, boucles et tableaux. À la fin du fichier, vous trouverez un bref résumé sur les tableaux.

Modalités de soumission :

Créez un dossier portant le numéro de votre groupe.

Ajoutez-y deux fichiers PDF : l'un contenant vos réponses aux QCM et l'autre listant les membres du groupe (Nom, Prénom, Matricule).

Dans ce dossier, créez un sous-dossier contenant les réponses à tous les exercices.

Ensuite, créez un compte GitHub et un dépôt correspondant au numéro de votre groupe. Téléversez votre projet dans ce dépôt, puis envoyez-moi le lien à l'adresse hamidou.bah@altgras.com au plus tard à 1h53.

Attention : Les instructions doivent être suivies à la lettre, faute de quoi je ne corrigerai pas votre travail.

PARTIE A : QCM

1. Qu'est-ce que Java ?

- a) Un langage de programmation
- b) Un système d'exploitation
- c) Un navigateur web
- d) Un logiciel de traitement de texte

2. Qui a créé Java ?

- a) Bill Gates
- b) Linus Torvalds
- c) James Gosling
- d) Mark Zuckerberg

3. Java est principalement utilisé pour :

- a) La création de bases de données
- b) Le développement d'applications mobiles
- c) L'édition de vidéos
- d) L'analyse statistique

4. Qu'est-ce que le bytecode Java ?

- a) Le code source d'un programme Java
- b) Le code intermédiaire généré par le compilateur Java
- c) Le résultat final après l'exécution d'un programme Java
- d) Une bibliothèque standard de Java

5. Quel est le slogan de Java ?

- a) "Just Do It"
- b) "Write Once, Run Anywhere"
- c) "Think Different"
- d) "Connecting People"

6. Quelle est la principale caractéristique de la plateforme Java ?

- a) Sa vitesse d'exécution
- b) Sa capacité à fonctionner sur différentes machines virtuelles
- c) Son interface utilisateur graphique

d) Sa compatibilité avec les navigateurs web

7. Quel est le fichier contenant le code source Java ?

- a) .class
- b) .java
- c) .src
- d) .bytecode

8. Quel outil est utilisé pour exécuter des programmes Java ?

- a) Java Compiler
- b) Java Runtime Environment (JRE)
- c) Java Virtual Machine (JVM)
- d) Java Debugger

9. Qu'est-ce que le JDK en Java ?

- a) Java Developer Kit, contenant le JRE, les outils de développement et les bibliothèques
- b) Java Debugging Kernel, un outil de débogage avancé
- c) Java Development Kernel, une version spécifique pour le développement web
- d) Java Design Kit, pour la conception d'interfaces graphiques

10. Quelle est la méthode principale pour démarrer l'exécution d'un programme Java ?

- a) start()
- b) run()
- c) main()
- d) execute()

11. Comment déclare-t-on une variable constante en Java ?

- a) final int x;
- b) const int x;
- c) immutable int x;
- d) static int x;

12. Quelle est la méthode utilisée pour lire les données depuis le clavier en Java ?

- a) System.in.read()
- b) Console.readLine()
- c) Scanner.nextLine()
- d) input()

13. Quelle est la différence entre les opérateurs '==', '.equals()' et '===' en Java ?

- a) Aucune différence, ils sont interchangeables.
- b) '==' compare la référence mémoire, '.equals()' compare le contenu, '===' est une erreur de syntaxe.
- c) '.equals()' compare la référence mémoire, '==' compare le contenu, '===' compare l'identité des objets.
- d) '==' est utilisé pour les nombres, '.equals()' pour les chaînes de caractères, '===' pour les objets.

14. Comment déclare-t-on un tableau en Java ?

- a) int[] arr = new int[];
- b) int arr[];
- c) int[] arr = new int[10];
- d) int arr[10];

15. Quelle est la syntaxe correcte pour une boucle "for" en Java ?

- a) for (int i = 0; i < 10; i++) {}
- b) for (int i = 0; i < 10) {}

- c) `for (int i = 0; i++) {}`
- d) `for (int i < 10; i++) {}`

16. Comment sort-on d'une boucle "for" avant la fin de son exécution en Java ?

- a) En utilisant le mot-clé "exit"
- b) En utilisant le mot-clé "break"
- c) En utilisant le mot-clé "return"
- d) En utilisant le mot-clé "continue"

17. Qu'est-ce qu'une boucle "while" en Java ?

- a) Une boucle qui s'exécute un nombre fixe de fois
- b) Une boucle qui s'exécute tant qu'une condition est vraie
- c) Une boucle qui s'exécute au moins une fois
- d) Une boucle qui s'exécute jusqu'à ce qu'une condition soit fausse

18. Quelle est la différence entre les boucles "while" et "do-while" en Java ?

- a) Il n'y a pas de différence, ce sont des synonymes.
- b) "while" exécute son bloc de code au moins une fois, tandis que "do-while" vérifie la condition avant l'exécution.
- c) "while" vérifie la condition avant l'exécution, tandis que "do-while" exécute son bloc de code au moins une fois.
- d) "while" exécute son bloc de code tant que la condition est vraie, tandis que "do-while" vérifie la condition après l'exécution.

19. Quelle est la différence entre les types de données primitifs et les objets en Java ?

- a) Les types de données primitifs ont des méthodes, tandis que les objets n'en ont pas.
- b) Les types de données primitifs sont des références vers des objets.
- c) Les types de données primitifs sont stockés sur la pile, tandis que les objets sont stockés sur le tas.
- d) Les types de données primitifs sont des types de données de base, tandis que les objets sont des instances de classes.

PARTIE B : Exercices

Variables :

1. Écrivez un programme pour échanger les valeurs de deux variables sans utiliser de variable temporaire.
2. Créez un programme pour calculer et afficher la surface d'un rectangle en utilisant des variables pour la longueur et la largeur.
3. Demandez à l'utilisateur de saisir un nombre et affichez son carré.
4. Implémentez un programme pour convertir des degrés Celsius en Fahrenheit et vice versa.
5. Écrivez un programme pour calculer la moyenne de trois nombres.

Conditions :

1. Créez un programme qui vérifie si un nombre est pair ou impair.

2. Demandez à l'utilisateur de saisir son âge et déterminez s'il est majeur ou mineur.
3. Écrivez un programme pour trouver le plus grand parmi trois nombres saisis par l'utilisateur.
4. Implémentez un calculateur de frais d'expédition en fonction du poids du colis (utilisez des tranches de poids et des tarifs différents).
5. Créez un programme qui identifie si une année saisie par l'utilisateur est bissextile ou non.

Boucles :

1. Écrivez un programme pour afficher les nombres de 1 à 10 en utilisant une boucle for.
2. Implémentez un programme pour calculer la somme des nombres de 1 à 100 à l'aide d'une boucle while.
3. Créez un programme qui affiche la table de multiplication d'un nombre saisi par l'utilisateur.
4. Écrivez un programme pour trouver la factorielle d'un nombre donné en utilisant une boucle do-while.
5. Demandez à l'utilisateur de deviner un nombre et utilisez une boucle pour lui donner des indices jusqu'à ce qu'il trouve le bon nombre.

Tableaux : Résumé

Les tableaux en Java sont des structures de données permettant de stocker une collection ordonnée d'éléments de même type. Voici un résumé des points clés concernant les tableaux en Java :

Déclaration et Initialisation :

Les tableaux en Java sont déclarés en spécifiant le type de données des éléments et peuvent être initialisés de différentes manières.

Exemple de déclaration : `int[] tableauDeNombres;`

Allocation de mémoire :

Les tableaux Java doivent être alloués avec de la mémoire avant de pouvoir être utilisés.

Exemple d'allocation : `tableauDeNombres = new int[5];`

Accès aux Éléments :

Les éléments du tableau sont accessibles par leur index (position).

L'index commence à 0 et se termine à `tailleTableau - 1`.

Exemple d'accès à un élément : `int premierElement = tableauDeNombres[0];`

Longueur du Tableau :

La propriété `length` est utilisée pour obtenir la taille d'un tableau.

Exemple : `int taille = tableauDeNombres.length;`

Parcours d'un Tableau :

Les boucles (par exemple, `for`, `while`) sont utilisées pour parcourir les éléments d'un tableau.

Exemple de parcours avec une boucle `for` :

java

Copy code

```
for (int i = 0; i < tableauDeNombres.length; i++) {  
    // Faire quelque chose avec tableauDeNombres[i]  
}
```

Tableaux Multidimensionnels :

Java prend en charge les tableaux multidimensionnels, tels que les tableaux à deux dimensions (matrices).

Exemple de déclaration : `int[][] matrice = new int[3][3];`

Manipulation des Tableaux :

Les tableaux offrent des méthodes pour copier, trier et rechercher des éléments.

Les classes utilitaires comme `Arrays` proposent des méthodes pour effectuer ces opérations.

Limitations :

Les tableaux en Java ont une taille fixe après l'allocation et ne peuvent pas être redimensionnés directement.

Ils ne peuvent contenir qu'un seul type de données.

Les tableaux en Java sont des structures fondamentales utilisées pour stocker des collections d'éléments du même type, offrant un accès facile aux éléments par leur index et permettant diverses opérations de manipulation

Tableaux : Exercices (Lisez le résumé sur les tableaux en JAVA avant de faire ces exercices)

1. Créez un programme pour additionner deux tableaux et stocker le résultat dans un troisième tableau.
2. Écrivez un programme pour trouver la plus grande valeur dans un tableau d'entiers.
3. Implémentez un programme pour trier un tableau d'entiers par ordre croissant.
4. Écrivez un programme pour fusionner deux tableaux triés en un seul tableau trié.
5. Créez un programme pour compter le nombre d'occurrences d'un élément dans un tableau.