

.....
.....
Nom du candidat :
Date du test :

Indications

Durée du test	30 min
Réponses	Questionnaire à Choix Multiple

Notation (cadre réservé)

Note	... / 33
Commentaire	

Test JAVA

1. Créer un élément à partir de plusieurs éléments est :

- a. L'encapsulation
- b. L'héritage
- c. L'abstraction
- d. Le polymorphisme
- e. La surcharge (Overloading)
- f. Le remplacement (Overriding)

2. **La capacité d'un langage de programmation à traiter des objets également qui ont la même superclasse est :**

- a. L'encapsulation
- b. L'héritage
- c. L'abstraction
- d. Le polymorphisme
- e. La surcharge (Overloading)
- f. Le remplacement (Overriding)

3. **Avoir une sous-classe offrant une spécialisation d'une méthode de la superclasse avec la même signature est :**

- a. L'encapsulation
- b. L'héritage

- c. L'abstraction
- d. Le polymorphisme
- e. La surcharge (Overloading)
- f. Le remplacement (Overriding)

4. Le processus de regrouper les caractéristiques communes d'objets est :

- a. L'encapsulation
- b. L'héritage
- c. L'abstraction
- d. Le polymorphisme
- e. La surcharge (Overloading)
- f. Le remplacement (Overriding)

5. La capacité d'avoir au sein d'une même classe des méthodes de même nom est :

- a. L'encapsulation
- b. L'héritage
- c. L'abstraction
- d. Le polymorphisme
- e. La surcharge (Overloading)
- f. Le remplacement (Overriding)

6. Supposons que les classes X et Y sont des sous-classes de Z et que Z implémente l'interface. Quelles sont les affirmations vraies ?

- a. Une référence à un objet de classe X peut être projetée vers une référence à un objet de classe Y.
- b. Une référence à un objet de classe Y peut être projetée vers une référence à un objet de classe X.
- c. Une référence à un objet de classe X peut être projetée vers une référence de classe Z.
- d. Une référence à un objet de classe Y peut être projetée vers une référence de classe Z.
- e. Une référence à un objet de classe Z peut être projetée vers une référence à un objet d'interface W.
- f. Une référence à un objet de classe Y peut être projetée vers une référence à un objet d'interface W.

7. Peut-on instancier une classe contenant une méthode abstraite ?

- a. Oui
- b. Non

8. Parmi ces réponses, lesquelles sont vraies en regard des constructeurs ?

- a. Une classe hérite du constructeur de son parent.

- b. Le compilateur fournit un constructeur par défaut si aucun constructeur n'est créé pour la classe.
- c. Tous les constructeurs doivent renvoyer le type void
- d. Un constructeur peut lancer une exception.

9. Qu'est-ce que la sérialisation en Java ?

- a. C'est un processus permettant l'ordonnancement des objets en mémoire.
- b. C'est un mécanisme permettant de rendre les objets persistants.
- c. C'est mécanisme permettant de rendre les objets transactionnels.
- d. C'est le moyen de tracer l'évolution des objets en mémoire.

10. Quelle est la marche à suivre pour sérialiser un Objet ?

- a. Implémenter l'interface Serializable, implémenter les méthodes writeObject et appeler ces méthodes sur l'objet.
- b. L'objet doit implémenter l'interface Serializable. Créer un flux ObjectOutputStream et appeler la méthode writeObject avec l'objet à sérialiser en paramètre.
- c. Appeler Object.writeObject avec l'objet en paramètre.
- d. Appeler System.writeObject avec l'objet en paramètre.

11. Une interface en Java 7 (ou antérieure) peut contenir :

- a. Des déclarations de méthodes et de constantes
- b. Des déclarations de méthodes
- c. Des déclarations de méthodes et de méthodes statiques
- d. Des déclarations de méthodes et de constructeurs
- e. JEE

12. Quelle interface offre la capacité d'extraire les objets par leur clé ?

- a. Set
- b. Enumeration
- c. Collection
- d. Map
- e. List

13. La clause finally d'une instruction try-catch...

- a. S'exécute toujours.
- b. S'exécute si aucune exception n'est lancée dans le bloc try.
- c. S'exécute toujours sauf si son thread se termine.
- d. Est obligatoire.

14. On invoque une méthode statique :

- a. Avec l'identifiant de la classe
- b. Avec une nouvelle instance de la classe
- c. A partir de la classe Object
- d. Aucune de ces solutions

15. Quels sont les packages permettant d'effectuer des traitements XML ?

- a. SAX
- b. SWING
- c. DOM
- d. JNI

16. Quelle est l'annotation qui rend une méthode de webservice soap exposable publiquement?

17. Quel outil est utilisé pour générer des artefacts portables avec les Web Services ?

- a. Wsgen
- b. Soap
- c. Javac
- d. Jws
- e. Rmic

18. Concernant les webServices et l'API JAX-WS, veuillez sélectionner les affirmations correctes :

- a. un client JAX-WS ne peut être une application JAVA SE
- b. les handlers sont des intercepteurs permettant de réaliser des traitements lors de la réception de messages (déclenchés avant l'appel à une opération) uniquement
- c. la structure d'un message SOAP peut ne pas contenir d'header
- d. l'élément portType constituant un WSDL, définit le service web, en particulier les opérations qu'il réalise et le type de messages échangés
- e. apache AXIS n'offre pas d'outils pour créer automatiquement les wsdl correspondant à des classes JAVA, ou inversement
- f. un appel de webservice utilisant AXIS peut se faire via une requête HTTP-GET

19. Quel attribut peut prendre la TransactionAttributeType :

- a. TransactionAttributeType.MANDATORY
- b. TransactionAttributeType.NEW
- c. TransactionAttributeType. SUPPORTS
- d. TransactionAttributeType.SUSPEND

20. Parmis l'API JMS (Java Message Service), quelles sont les méthodes ne pouvant pas être utilisées pour consommation de messages ou qui ne font pas partie de l'API JMS :

- a. receive()
- b. receiveWait()
- c. receiveNoWait()
- d. acknowledge()
- e. onMessage()

21. Parmi les suivantes, quelles affirmations sont correctes :

- a. Si on a plusieurs consommateurs abonnés à une Queue JMS, seul le premier consommateur qui lit le message le traitera.
- b. Le message JMS se perd si la MDB renvoie une RuntimeException au moment de sa consommation
- c. Par défaut, si la queue est disponible alors que le client ne l'est pas, le message reste dans la queue tant qu'il n'est pas consommé

22. Parmi les affirmations suivantes concernant JUnit, la ou lesquelles sont correctes :

- a. Les annotations sont apparues avec Junit 4.x
- b. JUnit sert à la fois de développer des tests unitaires et de développer aussi les tests d'intégration et de validation
- c. JUnit execute une méthode annotée @After après l'exécution de toutes les méthodes de la classe de test
- d. Pour désactiver tous les tests d'une classe, on peut l'annoter avec l'annotation@Ignore (sans annoter les méthodes)

23. Pour écrire un bon test JUnit, sélectionner les bonnes réponses ci-dessous :

- a. Il faut écrire le tests avant de commencer les développements
- b. Essayer, le plus que possible, de regrouper les tests de plusieurs classes dans une seule classe de test
- c. Ne pas mettre les classes dans le même répertoire que les tests
- d. Imaginer des scénarios catastrophiques pour vérifier que l'application résiste.

24. Parmi les suivantes, quelles affirmations JUnit sont correctes :

- a. Si la méthode est de type private, son exécution échoue
- b. Si la méthode de test a un type de retour différent de void, son exécution échoue
- c. Il est possible de tester les méthodes private
- d. Si la méthode est de type protected, il faut absolument que la classe de test soit dans le même package

25. Encadrez les fonctions que permet de réaliser Maven

- a. Gérer le cycle de vie des builds
- b. Modifier à la volée les classes java compilées
- c. Automatiser certaines tâches intervenant dans le cycle de développement
- d. Permet d'intégrer des outils supplémentaires pour la gestion continue des builds
- e. Compiler les classes java en .class
- f. Centralise tous les plugins en un seul lieu de téléchargement

26. Encadrez les scopes de dépendances que Maven met à disposition :

- a. required
- b. compile
- c. test
- d. package
- e. application
- f. mandatory
- g. runtime
- h. request
- i. provided

27. Encadrez la bonne réponse. Jenkins est :

- a. Un serveur d'analyse et de supervision de la qualité des logicielles
- b. Un serveur d'intégration continue
- c. Un outil de gestion des bugs
- d. Un plugin maven utilisé pour l'exécution des tests automatiques d'intégration
- e. Un serveur d'application
- f. Un conteneur web de Servlets

28. Cochez la ou les bonnes réponses. Le Refactoring d'une application java permet de :

- a. Améliorer la qualité du code de l'application et faciliter sa maintenance
- b. Augmenter le couplage entre les différents composants de l'application
- c. Restructurer le code sans modifier le comportement de l'application
- d. Ajouter des nouvelles fonctionnalités à l'application
- e. Supprimer des anciennes fonctionnalités de l'application

29. Parmis le code et les affirmations ci-dessous, lesquelles sont correctes ?

```
@Entity  
public class Window {  
    @Id  
    private int winNo;  
    @ManyToOne  
    private House aHouse;  
}  
@Entity
```

```

public class House {
    @Id
    private int houseNo;
    @OneToMany(mappedBy="aHouse")
    private List<Window> windows; }
```

- a. Nous sommes en présence d'une association unidirectionnelle entre les entités House et Window ;
- b. Nous sommes en présence d'une association bidirectionnelle entre les entités House et Window ;
- c. L'association maître est définie par l'entité House ;
- d. L'association maître est définie par l'entité Window ;

30. Lequel des design patterns est implémenté par le code ci-dessous :

```

public final class MaClasse {
    private static MaClasse instance = new MaClasse ();

    public static MaClasse getInstance() {
        return instance;
    }

    private MaClasse () {
    }
}
```

- a. Factory
- b. Composite
- c. Proxy
- d. Singleton

31. Lequel des design patterns est implémenté par le code ci-dessous :

```

public abstract class Animal {
    public abstract String makeSound(); }

public class Dog extends Animal {
    @Override
    public String makeSound() {
        return "Woof";
    }
}

public class Cat extends Animal {
    @Override
    public String makeSound() {
        return "Meow";
    }
}
```

```
public class AnimalType {  
    public Animal getAnimal(String type) {  
        if ("canine".equals(type)) {  
            return new Dog();  
        } else {  
            return new Cat();  
        }  
    }  
}
```

- a. Builder
- b. Proxy
- c. Factory
- d. Decorator.

32. Soit un objet O qui implémente Runnable, quelle méthode de la classe thread faut-il appeler pour que O s'exécute en thread indépendant ?

- a. Start()
- b. Run()
- c. Init()
- d. Main()

33. Lequel des mots clés suivants n'est pas une commande DML ?

- a. INSERT
- b. CREATE
- c. UPDATE
- d. DELETE