|  |  |
| --- | --- |
|  |  |
|  | C:\Users\hamadi\AppData\Local\Microsoft\Windows\INetCache\Content.Word\logo_inti_vector-04.jpg  Institut des Nouvelles Technologies  de l’Information |

QCM

JAVA JBDC Swing

Session 2016

## Guide d'utilisation

* Notez qu'en fonction de la question, il est possible de cocher une ou plusieurs réponses. Ce sera à vous de déterminer le nombre de bonnes réponses possibles.

## Quelques conseils de mise en œuvre

* Ne restez pas bloqué sur une question pour laquelle vous n'avez pas de réponse. Passez à la question suivante. Vous reviendrez par la suite sur les questions sans réponse.
* Identifiez les questions auxquelles vous avez répondu sans certitude. Que ces réponses soient justes ou non, cela vous permettra de mémoriser les questions sur lesquelles vous devrez revenir dans la correction du TEST.
* Ne cherchez pas à aller le plus vite possible. Le temps imparti doit être utilisé au maximum pour vous permettre d'effectuer une synthèse des points abordés dans le module.

**QCM**

**1. Par qui java a-t-il été développé ?**

* Google
* Oracle
* Microsoft
* Sun Microsystems

**2. Que peut-on mettre dans une interface ?**

* Des définitions de variables et des méthodes abstraites
* Au moins une méthode abstraite
* Des méthodes abstraites et des constantes
* Des méthodes abstraites

**3. Qu’affichera le code suivant ?**

public class UneClasse {

private String attribut = “un mot”;

public UneClasse(){

this.attribut = “un autre mot”;

}

}

public class AutreClasse extends UneClasse {

public String getAttribut(){

return this.attribut;

}

}

// ….

public static void main(String[] args) {

UneClasse instance = new AutreClasse();

System.out.println( instance.getAttribut() );

* Un mot
* Un autre mot
* Une erreur car AutreClasse n’a pas accès à attribut
* Une erreur car AutreClasse n’est pas du type UneClasse

}

**4. Quelle phrase est vraie ?**

* Une classe peut étendre plusieurs classes mais ne peut implémenter qu’une interface
* Une classe ne peut étendre qu’une seule classe mais implémenter plusieurs interfaces
* Une classe peut étendre plusieurs classes et implémenter plusieurs interfaces
* Une classe ne peut étendre qu’une seule classe et n’implémenter qu’une seule interface

**5. Qu’est-ce que l’encapsulation ?**

* Le fait de mettre une visibilité publique seulement aux variables auxquelles on veut permettre l’accès
* Le fait d’avoir une classe ou une liste de classes en attribut d’une autre classe
* Le fait de ne donner accès aux informations d’un objet qu’en passant par des méthodes dédiées
* Le fait d’appeler des attributs d’une classe mère depuis l’une de ses classes filles

**6. Quelle phrase est vraie ?**

* La classe abstraite peut implémenter des méthodes contrairement à l’interface
* La classe abstraite ne peut pas définir de constante contrairement à l’interface
* La classe abstraite peut être instanciée contrairement à l’interface
* Il n’y a pas de différence entre ces deux classes

**7. En SQL, quelle requête permet d’effectuer une jointure entre premiere\_table et seconde\_table en affichant toutes les entrées de table2 ?**

* SELECT \* FROM premiere\_table pt INNER JOIN seconde\_table st ON pt.id\_prem=st.id\_prem;
* SELECT \* FROM premiere\_table pt LEFT OUTER JOIN seconde\_table st ON pt.id\_prem=st.id\_prem;
* SELECT \* FROM premiere\_table pt RIGHT JOIN seconde\_table st ON pt.id\_prem=st.id\_prem;
* SELECT \* FROM premiere\_table pt UNION seconde\_table st ON pt.id\_prem=st.id\_prem;

**8. Qu’est-ce qu’un DAO ?**

* Une classe qui centralisera les requêtes vers la base de données pour une entité
* Une classe de JDBC permettant la communication avec la base de données
* Une classe qui ne peut être instanciée qu’une seule fois
* Une classe qui fera le lien entre les différentes entités

**9. Si ClasseB hérite de ClasseA qui implémente InterfaceA, lequel de ces codes est possible ?**

* ClasseB classe = new ClasseA();
* ClasseA classe = new InterfaceA();
* ClasseB classe = new InterfaceA();
* InterfaceA classe = new ClasseB();

**10. Quel code n’est pas correct pour instancier une classe ?**

* ClasseA classe = new ClasseA();
* ClasseA classe = Class.forName(“fr.machin.ClasseA”).getInstance();
* ClasseA classe = instanciate(ClasseA);
* ClasseA classe = ClasseA.class.newInstance();

**11. Quelle est l’erreur de cette boucle *for* ?**

for( x = 10; x == 0; x­­ ){

x­­;

}

* On ne peut pas modifier x dans le corps de la boucle
* Il faut donner un type à x
* Une boucle *for* doit aller en s’incrémentant uniquement
* x ne sera jamais égal à 0

**12. Quelle ligne est correcte pour récupérer une connexion via JDBC ?**

* DriverManager.getConnection(“jdbc:mysql:127.0.0.1/nom\_bdd”, “username”, “password”);
* DriverManager.getConnection(“jdbc:mysql://127.0.0.1”, “nom\_bdd”, “username”, “password”);
* DriverManager.getConnection(“jdbc:mysql:127.0.0.1”, “nom\_bdd”, “username”, “password”);
* DriverManager.getConnection(“jdbc:mysql://127.0.0.1/nom\_bdd”, “username”, “password”);

**13. Quelle est la bonne syntaxe pour effectuer un *foreach* sur une List<Personne> laListe en *java* ?**

* foreach(Personne personne : laListe){ }
* for(Personne personne in laListe){ }
* for(Personne personne : laListe){ }
* foreach(Personne personne in laListe){ }

**14. Quelle est l’erreur dans le code JDBC suivant ?**

//ouverture de la connexion dans la variable cnx

PrepareStatement stmt = cnx.prepare(“INSERT INTO table (champ1,champ2) VALUES

(?,?)”);

stmt.setString(1, “texte”);

stmt.setInt(2, 10);

stmt.executeUpdate();

ResultSet rs = stmt.getGeneratedKeys();

rs.next();

return rs.getInt(1);

* La bonne méthode est stmt.executeQuery()
* Il manque la boucle sur le ResultSet
* Il manque un argument à la méthode cnx.prepare()
* Il n’y a pas d’erreur

**15. Pour *int* variable = 9 Que vaut**

(10 >= variable && variable != 9) || (variable > 9 || variable < 9)

* True
* False
* 9
* Erreur, il manque le *if*

**16. Une table de jointure est…**

* Un type de requête SQL
* Une manière de représenter une relation n...n en base de données
* Une manière de représenter une relation 1...n en base de données
* Une table sans clé primaire

**17. Cochez l’une des classes permettant de faire une fenêtre avec *Swing* ?**

* JPanel
* JAmbon
* JFrame
* BorderLayout

**18. Comment faire référence à une classe parente dans une classe enfant ?**

* parent
* herited
* this
* super

**19. Qu’est-ce que la refactorisation ?**

* Le fait de rendre le code automatique
* La revue du code pour simplifier sa maintenabilité et éviter les répétitions
* La revue du code pour voir si toutes les exigences attendues sont présentes
* L’ajout de fonctionnalités supplémentaires à un code

**20. Si une méthode qu’on utilise lève une exception, on pourra se passer du try..catch en choisissant de faire que notre méthode *throw* l’exception en question.**

* Vrai
* Faux

FIN