+24/1/21+

QCM

TEST

Introduction à la programmation
orientée objet
9/11/2017
In

Nom et prénom :
SALORD FLORIA
Groupe:

Cochez les cases en mettant une X.

Le symbole \bigoplus indique que la question peut avoir zéro, une ou plusieurs bonnes réponses. Pour ces questions, cocher une bonne réponse apporte des points positifs; cocher une mauvaise réponse peut apporter des points négatifs. Dans tout le code, les package et les import sont censés être correctement déclarés. Toute classe est supposée être dans le bon package, dans le bon fichier, avec les bons import.

```
Quel est l'accès le plus restrictif par lequel on pourrait remplacer ____ ?
Question 1
 package toto;
 public class Toto {
     ___ String doSomething() {
        // does something
                                                           public public
      package-private (default)
                                                           | private
Question 2
                  Quel est l'accès le plus restrictif par lequel on pourrait remplacer ____?
 package toto;
                                                          package toto;
public class Toto {
                                                          class Foober {
        String doSomething() {
// does something
                                                             private Toto toto = new Toto();
                                                             private String doSomething() {
   toto.doSomething();
}
      private
                                                           x package-private (default)
      public
Question 3
                  Quel est l'accès le plus restrictif par lequel on pourrait remplacer ____?
package toto:
                                                         package foobar;
public class Toto {
                                                         class Foobar {
       _ String doSomething() {
                                                             private Toto toto = new Toto();
       // does something
```

0.33/0.33

0.33/0.33

0.33/0.33

X public

private

private String doSomething() {
 toto.doSomething();

package-private (default)



C	uestion 4 (1)	Lesquelles des	expressions	déclarent.	construisent	et in	nitialisent	un	tableau	?
ν4	nesmon a (1)	Desquence des	CVINCESSIONS	CICCIEII CIIO,	COMBUILLESCIE	CO 11	TI UICLIE CIED	u	Despica	٠

int[] myList = {"1", "2", "3"};

____ int[] myList = (5, 8, 2);

int myList = {4,9,7,0};

0/2

int myList[] = {4, 3, 7};



Question 5 Soit le code à Polytech'Groland pour stocker des notes, afficher les notes, et une classe Main de mise en exécution :

```
package admin;
                                                             package admin;
class Marks {
                                                             class Consulter {
    // this is voodoo, but it correctly intializes marks
                                                                 private final Marks marks;
   private final Map<String, int[]> marks
           = new HashMap<String, int[]>(){{
                                                                 Consulter (Marks marks) {
       put("Barney", new int[][12, 8]);
                                                                    this.marks = marks:
       put("Fred", new int[]{7, 9});
put("Wilme", new int[]{15, 13});
                                                                 void displayMarks(String student) {
                                                                    System.out.print(student + ": ");
                                                                    for (int m : marks.getMarks(student)) {
   int[] getMarks(String student) {
                                                                        System.out.print(m + " ");
       return marks.get(student);
                                                                    System.out.println();
                                                                }
   Set<String> getStudents() {
                                                             }
       return marks.keySet();
                                                             package admin;
                                                             public class Main {
package admin.sploit;
                                                                public static void main(String... args) {
                                                                    Marks marks = new Marks();
public class Sploit {
   // code to be supplied for the following method
                                                                    Consulter consulter = new Consulter(marks);
                                                                     // administration consults student marks
   public void haxMyMarks
                                                                    marks.getStudents().forEach(s
                                                                            -> consulter.displayMarks(s));
                                                                     // Wilma introduces emploit
                                                                    new Sploit().haxMyMarks(marks.getMarks("Wilma"));
                                                                     // administration consults student marks again
                                                                    marks.getStudents().forEach(s
                                                                            -> consulter.displayMarks(s));
                                                                }
                                                             }
```

En fonctionnement normal, tout cela donne le résultat à gauche :

 Barney: 12 8
 Barney: 12 8

 Wilma: 15 13
 Wilma: 20 20

 Fred: 7 9
 Fred: 7 9

Hélas, une élève rusée a trouvé le moyen d'introduire du code dans la classe Sploit pour exploiter une faille dans le système, afin d'améliorer ses notes. Cela donne le résultat à droite. Démontrez comment elle aurait pu arriver à ce résultat en complétant la classe Sploit, sans toucher aux autres classes.

```
public class Splait {

public raid has My Marbs (int I) new Marks ) {

for (int i = 0; i < new Marks . length; i++) {

new Marks [i] = 20;

}
}
```

0/1

Question 6 Soit le code de la question précédente. Quelle parade dans la classe Marks, et seulement dans la classe Marks, aurait pu éviter ce désagrément pour Polytech'Groland?

Déclarer les tableaux de notes conne final la la exemple: put ("Barney", new final int[](12,8})

Question 7

Quelles affirmations s'appliquent aux sets (la classe HashSet):

	sont indexés	${\it exclusivement}$	par	des	entiers
	non-négatifs				
_					

le nombre d'éléments est donné par .size()

- peuvent stocker des doublons (deux fois le même élément)
- sont initialsés par, eg, new Person (14)

sont de taille fixe

- peuvent stocker des primitifs, eg, double
- sont déclarés par, eg, Person[] p
- x sont dans le package java.util
- l'ordre de stockage des éléments est bien défini



```
Question 8 Corrigez toutes les erreurs dans le code :
```

```
roid fruit Greater (double [] marks, double mean) {

for (marks [index] > mean) {

System. out. pintln (marks [index]);

}

}
```

```
Question 9  Soit la déclaration :

private final String[] names = {"Fred"};

Quelles expressions sont permises dans une méthode de la même classe ?

names = new String[1];  names = "Barney";

names[0] = "Barney";  names = new String("Barney");

Question 10  Soit la déclaration :

private String[] names = {"Fred"};

Quelles expressions sont permises dans une méthode de la même classe ?

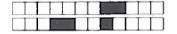
names[0] = "Barney";  names = new String("Barney");

names = new String[1];
```

names = "Barney";

1.33/2

names[0] == "Barney";



	Question 11 Quelles affirmations s'appliquent aux tableaux :						
0.833/1	 sont indexés exclusivement par des entiers non-négatifs sont dans le package java.util peuvent stocker des doublons (deux fois le même élément) l'ordre de stockage des éléments est bien défini 	 					
0.5/0.5	Question 12 On souhaite écrire une application dans un package foobar. Il est obligatoir de compiler les fichiers avec la commande javac -package foobar *.java écrire import foobar.*; en début de tous les fichiers source écrire package foobar; en début de tous les fichiers source mettre tous les fichiers sources dans un même .jar Question 13 Nous souhaitons compiler le code source :						
	<pre>package main; class Main { private Toto toto; public static void main(String args) { // some code } }</pre>	package main; class Toto {}					
0.5/0.5							
0.5/0.5	☐ java bin/Main.class☐ java -cp bin Main.class☐ java -cp bin main.Main	question précédente, ça serait : java main.Main java -cp bin Main java -cp bin Main.java uent aux listes (la classe ArrayList) :					
1/1	 sont indexés exclusivement par des entiers non-négatifs sont déclarés par, eg, Person[] p sont créés par, eg, new Person(14) sont dans le package java.util le nombre d'éléments est donné par .size() 	 □ sont de taille fixe ☒ peuvent stocker des doublons (deux fois le même élément) □ peuvent stocker des primitifs, eg, double ☒ l'ordre de stockage des éléments est bien défini 					

Question 16 Le code

```
class Protagonist {
   private String name = "Fred";
   private void print() {
       System.out.println("My name is " + name);
   public static void main(String... args) {
      print();
7
```

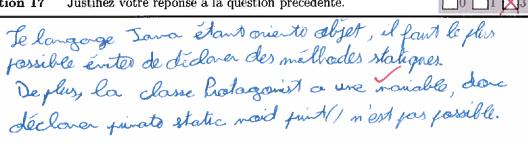
ne compile pas. Le compilateur dit

Protagonist.java:11: error: non-static method print() cannot be referenced from a static context print();

Deux possibilités se présentent, laquelle est le meilleur choix :

Déclarer : private static void print() Modifier main: new Protagonist().print()

Question 17 Justifiez votre réponse à la question précédente.



1/1