

## I. Questions de cours

1. Quelle est la différence entre une classe abstraite et une interface ? Quelle est la syntaxe Java pour définir une interface ?
2. Quelle est la différence entre une collection de type **ArrayList** et une collection de type **HashSet**?
3. Que signifie le mot clef "static" associé à un attribut ?
4. Quel est le droit d'accès le plus restrictif pour donner l'accès aux membres d'une autre classe de même package ?
5. Expliquer le principe de la séquence : try-catch-finally

## II. Questions à choix multiples

1. Soit les classe Master et Slave Suivantes:

```
import java.io.*;
class Master {
    String doFileStuff() throws FileNotFoundException {
        return "a";
    }
}
class Slave extends Master {
    public static void main(String[] args) {
        String s = null;
        try {
            s = new Slave().doFileStuff();
        } catch ( Exception x) {
            s = "b";
        }
        System.out.println(s);
    }
}
// Insérer une instruction ici
}
```

Choisir le(s) instruction(s) à insérer dans *//Insérer une instruction ici* ,pour que la classe Slave compile et affiche la valeur de la variable (b)?

- A. String doFileStuff() { return "b"; }
- B. String doFileStuff() throws IOException { return "b"; }
- C. String doFileStuff(int x) throws IOException { return "b"; }
- D. String doFileStuff() throws FileNotFoundException { return "b"; }

2. Soit la classe Exemple Suivante et sachant que Short est une classe fille de Number.

```
1. class Exemple {  
2. public static void main(String[] args) {  
3. Short s = 15;  
4. Boolean b;  
5. // insert code here  
6. }  
7. }
```

Choisir le(s) instruction(s) à insérer dans la ligne 5 pour que la classe Exemple compile?

- A. b = (Number instanceof s);
- B. b = (s instanceof Short);
- C. b = s instanceof (Short);
- D. b = (s instanceof Number);
- E. b = s instanceof (Object);
- F. b = (s instanceof String);

3. Soit les classes Plant, Flower et Tulip Suivantes:

```
1. class Plant {  
2. String getName() { return "plant"; }  
3. Plant getType() { return this; }  
4. }  
5. class Flower extends Plant {  
6. // insert code here  
7. }  
8. class Tulip extends Flower { }
```

Choisir le(s) instruction(s) à insérer dans la ligne 6 pour que la classe Flower compile?

- A. Flower getType() { return this; }
- B. String getType() { return "this"; }
- C. Plant getType() { return this; }
- D. Tulip getType() { return new Tulip(); }

4. Soit les classes X, Y et Chrome Suivantes:

```
1. class X { void do1() { } }  
2. class Y extends X { void do2() { } }  
3.  
4. class Chrome {  
5. public static void main(String [] args) {  
6. X x1 = new X();  
7. X x2 = new Y();  
8. Y y1 = new Y();  
9. // insert code here
```



**Examen Java de base - D2J**  
**20/01/2019**  
**Durée 1H30min**  
**Documents non autorisés**



**UNIVERSITÉ HASSAN 1<sup>ER</sup>**  
**FACULTÉ DES SCIENCES**  
**ET TECHNIQUES DE SETTAT**

10. } }

Choisir le(s) instruction(s) à insérer dans la ligne 9 pour que la classe Flower compile?

- A. x2.do2();
- B. (Y)x2.do2();
- C. ((Y)x2).do2();
- D. Les instructions A,B,C ne compilent pas.

5. Soit les classes Top et Bottom2 Suivantes:

```
class Top {  
    public Top(String s) { System.out.print("B"); }  
}  
public class Bottom2 extends Top {  
    public Bottom2(String s) { System.out.print("D"); }  
        public static void main(String [] args) {  
            new Bottom2("C");  
            System.out.println(" ");  
        }  
}
```

Choisir le résultat issu de la classe Bottom2 ?

- A. BD
- B. DB
- C. BDC
- D. DBC
- E. Erreurs de compilation.

**Fin de L'examen.Bonne Chance**