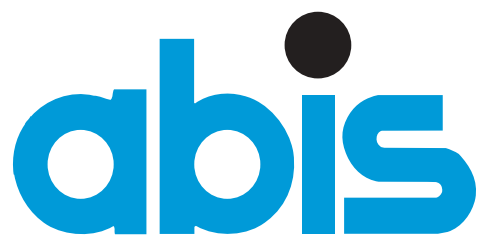


Zelftest Programmeren in Java

Document: n0883test.fm

22/01/2013

ABIS Training & Consulting
P.O. Box 220
B-3000 Leuven
Belgium



TRAINING & CONSULTING

INLEIDING BIJ DE ZELFTEST PROGRAMMEREN IN JAVA

Deze test bestaat uit meerkeuze-vragen. Bij de sommige vragen kunnen meerdere antwoorden correct zijn. Duid uw antwoord(en) aan en vergelijk met de gegeven oplossingen.

In de test worden 20 vragen gesteld. Let op: als er voor een meerkeuzevraag meerdere juiste antwoordmogelijkheden zijn, is dit expliciet aangegeven. Een gestelde vraag is pas correct beantwoord wanneer alle correcte antwoorden gegeven zijn.

Er zitten niet echt strikvragen bij (toch niet bewust), maar lees vraag en antwoord toch aandachtig. Reken ongeveer 30 minuten voor het invullen.

De juiste antwoorden en richtlijnen voor de evaluatie vindt u achteraan dit document.

VRAGEN ZELFTEST PROGRAMMEREN IN JAVA

1. Welk van de volgende statements in verband met arrays is syntactisch verkeerd?

- (a) `Person[] p = new Person[5];`
- (b) `Person p[5];`
- (c) `Person[] p [];`
- (d) `Person p[][] = new Person[2][];`

2. Gegeven het volgende stukje code:

```
public class Test {
    public static void main(String args[]) {
        int i = 0, j = 5 ;
        for( ; (i < 3) && (j++ < 10) ; i++ ) {
            System.out.print(" " + i + " " + j );
        }
        System.out.print(" " + i + " " + j );
    }
}
```

wat is het resultaat?

- (a) 0 6 1 7 2 8 3 8
- (b) 0 6 1 7 2 8 3 9
- (c) 0 5 1 5 2 5 3 5
- (d) de compilatie loopt verkeerd

3. Welk van de volgende declaraties is juist? (2 antwoorden)

- [a] `boolean b = TRUE;`
- [b] `byte b = 255;`
- [c] `String s = "null";`
- [d] `int i = new Integer("56");`

4. Stel een klasse heeft als visibiliteit `public`. In deze klasse definiëren we een methode die `protected` is. Welk van de volgende uitspraken is dan juist?

- (a) Deze methode is enkel en alleen bereikbaar binnen de klasse zelf en binnen alle sub-classes.
- (b) Je mag in een klasse geen methodes met een lagere zichtbaarheid declareren dan de

zichtbaarheid van de klasse zelf.

- (c) Vanuit protected methodes heeft men geen toegang tot public methodes.
- (d) Deze methode is bereikbaar binnen de klasse zelf en vanuit alle klassen die in hetzelfde package gedefinieerd staan als deze klasse.

5. Gegeven het volgende stukje code

```
public class Company{
    public abstract double calculateSalaries();
}
```

welk van de volgende uitspraken is dan juist?

- (a) De keyworden `public` en `abstract` mogen niet samen gebruikt worden.
- (b) De methode `calculateSalaries()` in de klasse `Company` moet een body hebben.
- (c) Er moet een return statement in de methode `calculateSalaries()` worden geplaatst.
- (d) De klasse `Company` moet `abstract` gedefinieerd worden.

6. Gegeven het volgende stukje code

```
public interface Guard{
    void doYourJob();
}

abstract public class Dog implements Guard{}
```

welk van de volgende uitspraken is dan juist?

- (a) Deze code zal niet compileren, omdat de methode `doYourJob()` in de interface `Guard` `abstract` moet gedefinieerd worden.
- (b) Deze code zal niet compileren, omdat de klasse `Dog` de methode `doYourJob()` uit de interface `Guard` moet implementeren.
- (c) Deze code zal niet compileren, omdat we in de declaratie van de klasse `Dog` het keyword `extends` moeten gebruiken i.p.v. `implements`.
- (d) Deze code zal zonder problemen compileren.

7. Gegeven deze klassen:

```
public class Person{
    public void talk(){ System.out.print("I am a Person "); }
}
public class Student extends Person {
    public void talk(){ System.out.print("I am a Student "); }
}
```

Wat is dan het resultaat van volgend stukje code:

```
public class Test{
    public static void main(String args[]){
        Person p = new Student();
        p.talk();
    }
}
```

```
    }  
}
```

- (a) I am a Person
- (b) I am a Student
- (c) I am a Person I am a Student
- (d) I am a Student I am a Person

8. Gegeven het volgende stukje code:

```
public class Person{  
    private String firstName;  
    public Person(String fn){ firstName = fn; }  
}  
  
public class Student extends Person{  
    private String studentNumber;  
    public Student(String number) { studentNumber = number; }  
}
```

Welk van de volgende uitspraken is dan juist? (2 antwoorden)

- [a] Deze code zal compileren als we in de klasse `Person` een no-argument constructor definiëren.
- [b] Deze code zal compileren als we in de klasse `Student` een no-argument constructor definiëren.
- [c] Deze code zal compileren als we in de constructor van `Student` de volgende lijn code als eerste statement schrijven:
`super();`
- [d] Deze code zal compileren als we in de constructor van `Student` de constructor van `Person` aanroepen.

9. Geef de correcte kenmerken van een enumeration type (2 antwoorden)

- [a] `enum` kan statische velden en methodes definiëren
- [b] `enum` kan een publieke constructor bevatten
- [c] `enum` kan interfaces implementeren
- [d] `enum` is een referentie naar een variabele set van constanten

10. Gegeven volgend stukje code:

```
class Person { public int number; }  
  
public class Test{  
    public void doIt(int i , Person p){  
        i = 5;  
        p.number = 8;  
    }  
  
    public static void main(String args[]){
```

```

        int x = 0;
        Person p = new Person();
        new Test().doIt(x, p);
        System.out.println(x + " " + p.number);
    }
}

```

Wat is dan de uitkomst?

- (a) 0 8
- (b) 5 0
- (c) 0 0
- (d) 5 8

11. Gegeven het volgende stukje code:

```

class SalaryCalculationException extends Exception{}

class Person{
    public void calculateSalary() throws SalaryCalculationException {
        //...
        throw new SalaryCalculationException();
        //...
    }
}

class Company{
    public void paySalaries(){
        new Person().calculateSalary();
    }
}

```

Welk van de volgende uitspraken is juist (2 antwoorden):

- [a] Deze code zal zonder problemen compileren.
- [b] Deze code zal compileren indien we in `paySalaries()` als returnwaarde een boolean terug geven in plaats van `void`.
- [c] Deze code zal compileren indien we in `paySalaries()` een `try-catch-block` toevoegen.
- [d] Deze code zal compileren indien we in de signatuur van de `paySalaries()` methode `throws SalaryCalculationException` toevoegen.

12. Welke uitspraken in verband met static methoden zijn correct? (2 antwoorden)

- [a] static methoden zijn moeilijk onderhoudbaar, omdat hun implementatie niet veranderd mag worden.
- [b] static methoden mogen worden aangeroepen via een object-referentie naar een object van de klasse waarin die methode gedefinieerd staat.
- [c] static methoden zijn altijd public, omdat ze op klasse-niveau gedefinieerd zijn.
- [d] static methoden hebben geen rechtstreekse toegang tot methoden die niet static zijn, en die binnen dezelfde klasse gedefinieerd staan.

13. Gegeven volgend stukje code:

```
class Person{ public void talk(){} }

public class Test{
    public static void main(String args[]){
        Person p = null;
        try{
            p.talk();
        } catch(NullPointerException e){
            System.out.print("There is a NullPointerException. ");
        } catch(Exception e){
            System.out.print("There is an Exception. ");
        }
        System.out.print("Everything went fine. ");
    }
}
```

Welk van volgende uitspraken is dan juist.

- (a) Als je dit programma runt is de uitkomst:
There is a NullPointerException. Everything went fine.
- (b) Als je dit programma runt is de uitkomst:
There is a NullPointerException.
- (c) Als je dit programma runt is de uitkomst:
There is a NullPointerException. There is an Exception.
- (d) Deze code zal niet compileren, want Java kent geen pointers.

14. Welke uitspraken i.v.m. het gebruik van **Generics** zijn correct? (2 antwoorden)

- [a] Generics zijn getypeerde subklassen van de klassen uit het Collections framework
- [b] Generics worden gebruikt om collecties te parametriseren zodat bij compilatie een type checking mogelijk is van de objecten in de collectie.
- [c] Generics kunnen gebruikt worden om type checking te doen van de objecten in de collectie bij uitvoering
- [d] Generics laten toe om op een gemakkelijkere manier over een volledige collectie te itereren via de 'enhanced for' lus.

15. Welke collectie klasse associeert values met keys, en rangschikt de keys volgens hun natuurlijke orde?

- (a) java.util.HashSet
- (b) java.util.LinkedList
- (c) java.util.TreeMap
- (d) java.util.SortedSet

16. Welke uitspraak in verband met GUI componenten is fout?
- (a) Swing bestaat sinds versie 1.2 van de jdk.
 - (b) AWT staat voor Abstract Window Toolkit
 - (c) Je mag AWT componenten op Swing containers plaatsen.
 - (d) De AWT klassen zijn deprecated.
17. Welk van de volgende uitspraken i.v.m. events zijn juist? (2 antwoorden)
- [a] Event objecten worden op een queue geplaatst, waar ze door subscribers (objecten van klassen die de interface `Subscriber` implementeren) worden afgehaald.
 - [b] Een listener van een event moet steeds de methode `public void listen(EventObject obj)` implementeren.
 - [c] Elk event object moet een object zijn van een subklasse van `EventObject`.
 - [d] Elke event listener kan te weten komen wat de bron is van het event, door de methode `getSource()` aan te roepen op het event object.
18. Hoe kan men een object serialiseren?
- (a) Men moet de klasse van het object de interface `Serializable` laten implementeren.
 - (b) Men moet de methode `serializeObject()` (die wordt overgeërfd uit de klasse `Object`) aanroepen op het object.
 - (c) Men moet van de klasse `Serializer` de statische methode `serialize(Object obj)` aanroepen, met als argument het object dat men wil serializeren.
 - (d) Men moet helemaal niets doen, want alle objecten zijn by default serializeerbaar.
19. Welke uitspraken in verband met IO zijn juist (2 antwoorden)?
- [a] `OutputStream` is de abstracte superklasse van alle klassen die een outputstream van bytes representeren.
 - [b] Subklassen van de klasse `Reader` worden gebruikt om character streams te lezen.
 - [c] Om characters naar een outputstream te schrijven, moet je gebruik maken van de klasse `CharacterOutputStream`.
 - [d] Om een object naar een file te schrijven, maak je gebruik van de klasse `ObjectFileWriter`
20. Gegeven het volgende stukje code:

```
public class MyThread extends Thread{
    public String text;
    public void run(){
        System.out.print(text);
    }
}

public class Test{
```



```
public static void main(String args[]){
    MyThread t1 = new MyThread(); t1.text = "one ";
    MyThread t2 = new MyThread(); t2.text = "two ";
    t1.start();
    t2.start();
    System.out.print("three ");
}
}
```

Welke van de volgende uitspraken is dan waar:

- (a) Als men dit programma uitvoert, krijgt men altijd als uitkomst one two three
- (b) Als men dit programma uitvoert, krijgt men altijd als uitkomst three one two
- (c) De uitkomst van dit programma is onbepaald.
- (d) Het compileren van dit programma zal mislukken.

EVALUATIE.

Hier zijn de correcte antwoorden op alle vragen:

1. b
2. a
3. c d
4. d
5. d
6. d
7. b
8. a d
9. a c
10. a
11. c d
12. b d
13. a
14. b d
15. c
16. d
17. c d
18. a
19. a b
20. c

Geef 1 punt per juist antwoord. Voor vragen met meerdere juiste antwoorden moeten alle antwoorden juist zijn voor er 1 punt wordt toegekend. Halve punten worden niet toegekend.

Indien uw score meer is dan 80%, dan hoeft u de cursus [Programmeren in Java](#) niet te volgen.

Indien uw score tussen 50% en 80% ligt, kan het volgen van de cursus [Programmeren in Java](#) uw kennis vervolledigen.

Is uw score minder dan 50%, dan is de cursus [Programmeren in Java](#) een echte aanrader.