

# Αναφορά για το 1ο σετ εργαστηριακών ασκήσεων JAVA

## Askisi1:

3)Περιπτώσεις:                      Αποτελέσματα:

{"12"}                                      You are  
12  
years old.

{"280"}                                    You are  
280  
years old.

{"16,5"}                                   You are  
16,5  
years old.

{}

Κανένα μήνυμα

4)Περιπτώσεις:                      Αποτελέσματα:

{"12"}                                    You are 12 years old.

Η διαφορά είναι ότι η εντολή System.out.print δεν αλλάζει γραμμή στο τέλος της, σε αντίθεση με την System.out.println.

5)Περιπτώσεις:                      Αποτελέσματα:

{}

You forgot to give your age.

6)Περιπτώσεις:                      Αποτελέσματα:

{"18","Dimitris"}                      Dimitris is 18 years old.

{"18"}                                    You are 18 years old.

{}

You forgot to give your age.

7)Η αλλαγή που έκανα θεωρώντας ότι αν δεν επιθυμεί ο χρήστης να δώσει κάποιο όνομα εισάγει το χαρακτήρα "-",είναι η εξής:

## Αναφορά για το 1ο σετ εργαστηριακών ασκήσεων JAVA

```
if (args.length == 2 && args[0] == "-")
{
    System.out.print("You are ");
    System.out.print(args[1]);
    System.out.println(" years old.");
}
else if (args.length == 1)
{
    System.out.println("You forgot to give your age.");
}
else
{
    System.out.print(args[0]);
    System.out.print(" is ");
    System.out.print(args[1]);
    System.out.println(" years old.");
}
```

Περιπτώσεις:	Αποτελέσματα:
{"Dimitris","18"}	Dimitris is 18 years old.
{"-","18"}	You are 18 years old.
{"-"} ή {"Dimitris"}	You forgot to give your age.

8) Περιπτώσεις:                      Αποτελέσματα:

{"Dimitris","18"}	Dimitris is 18 years old.
{"-","18"}	You are 18 years old.
{"-"} ή {"Dimitris"}	You forgot to give your age.

# Αναφορά για το 1ο σετ εργαστηριακών ασκήσεων JAVA

## Askisi2:

3) Περιπτώσεις:                      Αποτελέσματα:

{}

The size of window w1 is: 5

The size of window w2 is: 4

The size of window w3 is: 6

The size of window w4 is: 5

4) Περιπτώσεις:                      Αποτελέσματα:

{}

The size of window w1 is: 5

The size of window w2 is: 4

The size of window w3 is: 6

The size of window w4 is: 5

Τα αποτελέσματα δεν είναι διαφορετικά, διότι η εντολή `size = x;` είναι ίδια με την εντολή `this.size = x;` σε αυτή την περίπτωση .

5) Περιπτώσεις:                      Αποτελέσματα:

{}

The size of window w1 is: 10

The size of window w2 is: 10

Ανεξαρτήτως του ορίσματος η τιμή που επιστρέφει η μέθοδος `getSize()` είναι η τιμή της μεταβλητής που ορίζεται μέσα σε αυτή δηλαδή 10.

6) Περιπτώσεις:                      Αποτελέσματα:

{}

The size of window w1 is: 5

The size of window w2 is: 4

Στην περίπτωση αυτή η χρήση της εντολής `this` αναγκάζει τη μέθοδο `getSize()` να λάβει ως όρισμα την τιμή της εξωτερικής μεταβλητής `private int size=5`.

# Αναφορά για το 1ο σετ εργαστηριακών ασκήσεων JAVA

## Askisi3:

2) (α) Περιπτώσεις:                      Αποτελέσματα:

```
{}
```

Window size=1

Window size=2

1

2

Το στιγμιότυπο w1 δημιουργείται από τον default constructor οπότε δέχεται ως size την τιμή 1, ενώ το στιγμιότυπο w2 δημιουργείται από τον 2<sup>ο</sup> constructor και δέχεται ως size το όρισμα του δηλαδή την τιμή 2, τα οποία είναι ανεξάρτητα από τα ορίσματα της main.

2) (β) Περιπτώσεις:                      Αποτελέσματα:

```
{}
```

Window size=1

Window message = First MWindow

Window size=3

Window message = Second MWindow

2

3

Το στιγμιότυπο mw1 δημιουργείται από τον default constructor της κλάσης MWindow οπότε ως size έχει την τιμή 2 και ως message την τιμή First MWindow, ωστόσο εμφανίζεται ως window size το 1 λόγω του default constructor της κλάσης Window, αντίθετα με το στιγμιότυπο mw2 το οποίο δημιουργείται από τον δεύτερο constructor της κλάσης MWindow οπότε ως size έχει την τιμή 3 και ως message την τιμή Second MWindow οπότε εμφανίζεται ως window size το 3 λόγω του default constructor της κλάσης Window.

2) (γ) Περιπτώσεις:                      Αποτελέσματα:

```
{}
```

4

6

8

Η χρήση της μεθόδου setSize1(int y) της κλάσης MWindow αλλάζει τη μεταβλητή size του στιγμιότυπου mw1 σε 4, έπειτα χρησιμοποιείται η μέθοδος της κλάσης MWindow, setSize2(int z) η οποία λόγω της εντολής super επικαλύπτει τη μέθοδο της υποκλάσης και ενεργοποιεί την μέθοδο setSize(int x) της

## Αναφορά για το 1ο σετ εργαστηριακών ασκήσεων JAVA

υπερκλάσης Window, οπότε η μεταβλητή size αυξάνεται κατά 2 από το 4 στο 6 και τελικά καλείται η μέθοδος setSize(int x) της κλάσης Window και αυξάνει τη μεταβλητή size του στιγμιότυπου mw1 σε 8.

3) Περιπτώσεις:	Αποτελέσματα:
{ }	Window size=1
	Window size=2
	1
	2
	Window size=1
	Window message = First MWindow
	Window size=3
	Window message = Second MWindow
	2
	7
	4
	4
	4

Εξαιτίας της δήλωσης protected int size = 7;, εντολή super(size); στην κλάση MWindow(int size, String message) αλλάζει τη μεταβλητή size του στιγμιότυπου mw2 σε 7 αντί για 3, ενώ κατά την κλίση των μεθόδων setSize2(2); και setSize(2); η μεταβλητή size της κλάσης κλάση MWindow(int size, String message) έχει αλλάξει σε 4 λόγω της κλίσης της μεθόδου setSize1(4), εξαιτίας της δήλωσης της μεταβλητής ως protected.

4) Περιπτώσεις:	Αποτελέσματα:
{ }	Size=1
	MSize=4
	5
	Size=5

Το στιγμιότυπο w1 έχει δημιουργηθεί από τον default constructor της κλάσης Window οπότε ως size έχει την τιμή 1. Κατά τη διάρκεια εκτέλεσης του κώδικα η μεταβλητή protected int size = 7 αλλάζει τιμή από 7 σε 4, οπότε και η τιμή της μεταβλητής size του στιγμιότυπου mw1 είναι 4. Η τιμή του στιγμιότυπου mw1 στην κλάση window είχε αρχικά την τιμή του default constructor, δηλαδή 1, κατά τη

## Αναφορά για το 1ο σετ εργαστηριακών ασκήσεων JAVA

διάρκεια εκτέλεσης του κώδικα αυξάνεται κατά τέσσερα λόγω της μεθόδου `mw1.setSize(4)`; της κλάσης `RunWindow`, οπότε τελικά εκτυπώνεται το 5 λόγω της δήλωσης `super` για τη μεταβλητή `size` στην μέθοδο `mw1.printSize1()`. Στην εντολή `mw1.printSize2()`; , η ύπαρξη της δήλωσης `super` έχει ως αποτέλεσμα στην εκτέλεση της μεθόδου `printSize` της κλάσης `Window` με την μεταβλητή `size` ήδη στην τιμή 5.

5) Περιπτώσεις:                      Αποτελέσματα:

```
{}
```

5

MSize=4

Κατά την εκτέλεση της μεθόδου `printSize1()` λόγω της δήλωσης `this`, εκτυπώνεται η τιμή της μεταβλητής `protected int size`, η οποία είχε αρχικά την τιμή 1 για το στιγμιότυπο `mw1`, ενώ κατά την εκτέλεση του κώδικα άλλαξε τιμή από 1 σε 5. Κατά την εκτέλεση της μεθόδου `printSize2()` λόγω της δήλωσης `this`, καλείται η μέθοδος `printSize()` της κλάσης `MWindow` και εκτυπώνεται η τιμή της μεταβλητής `protected int size = 7` η οποία κατά την εκτέλεση του κώδικα έχει νέα τιμή, την τιμή 4.

6) Περιπτώσεις:                      Αποτελέσματα:

```
{}
```

Window size=1

Window size=2

1

2

Window size=1

Window message = First MWindow

Window size=3

Window message = Second MWindow

2

7

4

4

4

Size=1

MSize=4

5

# Αναφορά για το 1ο σετ εργαστηριακών ασκήσεων JAVA

MSize=4

Παράγονται τα ίδια αποτελέσματα.

## Απαντήσεις για το 1ο σετ εργαστηριακών ασκήσεων JAVA

### A Μέρος

1.C,E

2.C

3.D

4.B

5.A

6.B

### B Μέρος

1. Η μέθοδος calculate υπολογίζει το συνολικό εμβαδό των 6 εδρών του αντικειμένου box.

4. Η μέθοδος toString(), που προσπαθούμε να δημιουργήσουμε στην κλάση Box υπάρχει ήδη στην κλάση java.lang.Object, οπότε δεν μπορούμε να χρησιμοποιήσουμε το ήδη δηλωμένο όνομα toString(), για να ονομάσουμε μια μέθοδο με λιγότερα δικαιώματα.

5. Η ειδική μέθοδος toString() που δημιουργήσαμε επικαλύπτει την γενικότερη μέθοδο toString() που υπάρχει στην κλάση Object, η οποία κληρονομείται σε όλες τις υποκλάσεις της, οπότε εκτελείται η μέθοδος που δημιουργήσαμε αντί για την αντίστοιχη γενικότερη.