

ΕΡΓΑΣΤΗΡΙΑΚΕΣ ΑΣΚΗΣΕΙΣ JAVA

ΑΣΚΗΣΗ 1

3. Αν δεν δοθεί όρισμα δεν εμφανίζεται τίποτα στην οθόνη, αλλιώς εμφανίζεται το πρώτο όρισμα στη θέση του args[0].

```
{"12"} => You are  
        12  
        years old.
```

```
{"45"} => You are  
        45  
        years old.
```

4. "You are 12 years old." - Εκτυπώνεται μια αντί για 3 γραμμες.

5. {} => You forgot to give your age

```
6. {"18", "egw"} => egw is 18 years old.  
   {"12"} => You are 12 years old.
```

7.

```
String name = null;
```

```
String age = null;
```

```
if(args.length >= 1){  
    name = args[0];  
}
```

```
if(args.length >=2){  
    age = args[1];  
}
```

```
if (name == null && age != null)  
    {  
        System.out.print("You are ");  
        System.out.print(age);  
        System.out.println(" years old.");  
    } else if (name == null && age == null)  
    {  
        System.out.println("You forgot to give your age.");  
    }  
else  
    {  
        System.out.print(name);
```

```
System.out.print(" is ");  
System.out.print(age);  
System.out.println(" years old.");  
}
```

8. Τα ίδια αποτελέσματα.

ΑΣΚΗΣΗ 2

4. Οχι δεν έπαιξε ρόλο γιατί η αναφορά στην instance μεταβλητή size δεν αλλάζει.
5. Το size είναι πάντα 10 γιατί στην getSize επιστρέφεται συνέχεια η τιμή 10 επειδή χρησιμοποιείται συνέχεια η δεύτερη μεταβλητή size η οποία δηλώθηκε εντός του scope της μεθόδου.
6. Επιστρέφεται το αναμενόμενο size επειδή αλλάζει το this αλλάζει το scope από την method στο instance.

ΑΣΚΗΣΗ 3

2. a)

Window size=1

Window size=2

1

2

Στην πρώτη περίπτωση χρησιμοποιήθηκε ο δημιουργός χωρίς παραμέτρους και το size έγινε αυτόματα 1.

Στην δεύτερη, το size έγινε 2 επειδή χρησιμοποιήθηκε ο δημιουργός με παράμετρο το size (το οποίο δώθηκε ως 2).

b)

Window size=1

Window size=2

1

2

Window message = First MWindow

Window size=3

Window message = Second MWindow

2

3

Πέρα από τις πρώτες 4 γραμμές που εξηγήθηκαν στο πρώτο ερώτημα:

Στο πρώτο MWindow κλήθηκε ο δημιουργός με μονή παράμετρο string για αυτό και εκτυπώθηκε το "Window message = First MWindow" και έγινε το size 2

Στο δευτερο MWindow κληθηκε ο δημιουργος με 2 παραμετρους (Int και string), γιαυτο εκτυπωθηκε το "Window size=3" (λογω του super), καθως και το Window message = Second MWindow. Επειδη περαστηκε ως παραμετρος size to 3 , για αυτο εκτυπωθηκε 3 κατωτιν.

c)

Window size=1

Window size=2

1

2

Window size=1

Window message = First MWindow

Window size=3

Window message = Second MWindow

2

3

4

6

8

Περα απο τις πρωτες 10 γραμμες που εξηγηθηκαν στα ερωτηματα a και b:

mw1.setSize1(4) => εκτυπωνει 4 γιατι θετει το instance variable size (το οποιο ειναι προσβασιμο απο την υπερκλαση Window αφου ειναι protected) σε 4.

mw1.setSize2(2) => προσθετει 2 στο size. Το super.setSize(z) απλα καλει την υπερκλαση. Ομως η μεταβλητη size μενει απειραχτη και λειτουργει οπως περιμενουμε επειδη δεν γινεται override στην υποκλαση MWindow

Η mw1.setSize(2), λειτουργει οπως η mw1.setSize2() απλα καλει αμεσως την μεθοδο της υπερκλασης χωρις να περναι μεσα απο την υποκλαση και μετα να καλει super

3)

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

Οι μονες γραμμες που αλλαζουν ειναι οι 2 τελευταιες. Αυτο γινεται επειδη στην συγκεκριμενη περιπτωση γινεται override η μεταβλητη που ειχε δηλωθει στην υπερκλαση, απο την υποκλαση. Αυτο εχει ως αποτελεσμα οποιαδηποτε κληση μεθοδου υπερκλασης (ειτε αμεσα ειτε μεσα μεθοδο υποκλασης που καλει super) να μην μπορει να αλλαξει την τιμη της instance μεταβλητης size της κλασης MWindow

4.

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
Size=5

Για τις 4 τελευταιες γραμμες:
Ολες εκτυπωνουν αυτο που θα επρεπε.

Συγκεκριμενα οι περιπτωσεις

mw1.printSize1();
mw1.printSize2();
Καλουν την υπερκλαση, Window, η οποια επιστρεφει την τιμη 5 για το size (undefined), αφου δεν ειναι προσβασιμη η overridden μεταβλητη απο την υπερκλαση.

5.

Αλλάζει μόνο η τελευταία γραμμή, και γίνεται: MSize=4 . Στην πρώτη περίπτωση συνεχίζουμε να βλέπουμε ότι size = 5 επειδή προσπαθούμε να δούμε την τιμή της size χειρίζοντας την MWindow ως Window, ενώ έχουμε κάνει override το size στην MWindow.

Στην δεύτερη περίπτωση χειριζόμαστε το MWindow ως Window αλλά καλούμε την printSize, η οποία έχει γίνει override στην MWindow αρα δεν χρησιμοποιείται η printSize της υπερκλάσης, αλλά της υποκλάσης.