

ΟΝΤΟΚΕΝΤΡΙΚΟΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ

ΕΡΓΑΣΤΗΡΙΑΚΕΣ ΑΣΚΗΣΕΙΣ JAVA

ΟΝ/ΜΟ: ΑΠΟΣΤΟΛΟΣ ΝΤΕΛΗΔΗΜΟΣ

A.M: 1047199 ΕΤΟΣ: 4^ο

ΑΣΚΗΣΗ 1)

Έδωσα το όρισμα {"12"}) και μου έβγαλε Error: illegal character: "\u201c". Το ίδιο και με διαφορετικά ορίσματα. Όταν δεν έδωσα όρισμα, δεν μου εμφάνισε τίποτα. Μετά την αλλαγή στα `system.out.print` εμφανίστηκε κανονικά το `You are 12 years old`. Μετά την προσθήκη του `else`, το `terminal` μου εμφάνισε `You forgot to give your age`. όταν δεν δίνω `argument`. Στην διαμόρφωση της `main()` εάν βάλω 2 `arguments` μου βγάζει `You are 12 years old. giannis is 12 years old.` , με 1 `argument` εμφανίζει μόνο την πρώτη πρόταση και χωρίς `arguments` `You forgot to give your age.` . Για να δίνουμε πρώτα το όνομα και μετά την ηλικία ,αντέστρεψα τα `arguments` 1 και 0 στην κλάση `Age`. Έκανα τις αλλαγές και το πρόγραμμα τρέχει όπως πρέπει.

ΑΣΚΗΣΗ 2)

Τα αποτελέσματα είναι

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 είναι ίδια ,δεν έπαιξε ρόλο η χρήση του `this`. Δεν επηρεάζεται γιατί δεν έχουμε αναθέσει κάποια τιμή στο `x`.

Τα αποτελέσματα με τον νέο κώδικα ήταν :

The size of window w1 is: 10

The size of window w2 is: 10

Παρατηρώ ότι εφόσον έχω αναθέσει τιμή στο `size` στην κλάση `getSize()` το αποτέλεσμα των `w1` και `w2` `getSize()` παραμένει το ίδιο.

Με την προσθήκη του `this` παίρνουμε τα αποτελέσματα:

The size of window w1 is: 5

The size of window w2 is: 4

Παρατηρώ ότι το `this` μπλοκάρει το `int size = 10;` στην κλάση `getSize()` και παίρνει την τιμή `int size=5;` . Ορθά αποτελέσματα καθώς το `this` αναφέρεται στο πεδίο `size`.

ΑΣΚΗΣΗ 3)

2)α) Το αποτέλεσμα όταν έτρεξα το RunWindow ήταν:

Window size=1

Window size=2

1

2

Με τις `Window w1=new Window();` και `Window w2=new Window(2);` στην class window εκτελεί τα αντίστοιχα constructors και printarei τα window size. Αντίστοιχα τα `println` printaroun 1 και 2.

2)β) Το αποτέλεσμα είναι :

Window size=1

Window size=2

1

2

Window size=1

Window message = First MWindow

Window size=3

Window message = Second MWindow

2

3

Το `MWindow mw1=new MWindow("First MWindow");` πάει στο constructor `public MWindow(String message)` και το άλλο στο 2^ο constructor της `MWindow`. Επειδή έχουμε το `extends` εκτελούνται και τα αντίστοιχα constructors από την κλάση `Window`.

2)γ) Επιπρόσθετα με τα παραπάνω αποτελέσματα, τώρα εμφανίζονται και τα :

4

6

8

Τώρα χρησιμοποιείται η μέθοδος `setSize1(4)` η οποία δίνει τιμή στο `size=4` και έπειτα χρησιμοποιείται η `mw1.setSize2(2);` η οποία χάρης το `super` στέλνεται στην `setSize` της κλάσης `Window` και αυξάνει το `size` κατά 2, δηλαδή 6. Έπειτα αυξάνεται πάλι κατά 2 και γίνεται 8. Λογικά θα έπρεπε να βγαίνει 4 4 4 στο τέλος όπως στην 3) αλλά δεν μπορώ να το κατανοήσω.

3)Τα αποτελέσματα είναι :

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

Φαίνεται ότι το `super(size)` συνδέεται με την `protected int size = 7`; Και δίνει την τιμή 7 στο `mw2.size`.

Το `mw1.setSize1(4)`; δίνει την τιμή 4 στο `size` και τα `mw1.setSize2(2)`; Και `mw1.setSize(2)`; Δίνουν την τιμή 2+2 λόγω `super(size)` και την `prinSize`.

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

To w1.printSize(); Printarei 1 λόγω του ότι ήταν 1 πιο πριν.

To mw1.printSize(); Είναι 4 λόγω της προηγούμενης τιμής του.

To mw1.printSize1(); Είναι 5 λόγω του super.size που δίνει τιμή στο size αυτήν που έχει στο class πατέρα.

To mw1.printSize2(); Είναι όπως το προηγούμενο μόνο που καλεί την μέθοδο printSize της κλάσης Window.

5)

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

MSize=4

Το `mw1.printSize1(); (((Window)this).size)` μας δίνει 5 ,δηλαδή το size που είναι στην κλάση `Window`, ενώ το `mw1.printSize2(); {((Window)this).printSize();}` παίρνει το size που είναι στην κλάση `MWindow` και το μεταφέρει στην μέθοδο `printSize` της κλάσης `Window`.

6)

Τα αποτελέσματα που βγαίνουν είναι τα ίδια :

`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

`MSize=4`