

ΘΕΜΑ 1^ο

Ποιοι από τους παρακάτω ισχυρισμούς είναι σωστοί και ποιοι λάθος (χωρίς εξήγηση).

- (1) Η προσπέλαση μιας μεθόδου γίνεται μόνο με την αποστολή μηνύματος.
- (2) Ένα catch μπλοκ δεν ενεργοποιείται αν δεν εγερθεί εξαίρεση.
- (3) Ο δημιουργός καλείται πάντοτε άμεσα (όχι έμμεσα) στη δημιουργία στιγμιότυπου.
- (4) Μια μέθοδος είναι υποχρεωτικό να έχει παραμέτρους (ορίσματα).
- (5) Ένα buffered stream είναι ένα είδος φίλτρου.
- (6) Κάθε πρόγραμμα Java πρέπει να διαθέτει τουλάχιστον μια κλάση με το όνομα του αρχείου που το περιέχει.
- (7) Ένας δημιουργός μιας κλάσης συμπεριφέρεται όπως μια μέθοδος της κλάσης.
- (8) Όλοι οι μέθοδοι μιας κλάσης πρέπει να δηλώνονται ως private.
- (9) Η αντιγραφή των δεδομένων ενός αντικειμένου σ' ένα άλλο δεν μπορεί να γίνει με τον τελεστή ανάθεσης/καταχώρησης.
- (10) Ένα φίλτρο δεν μπορεί να χρησιμοποιηθεί αυτόνομα, αλλά πάντα σε αναφορά με κάποιο stream αρχείου.
- (11) Ο προσδιοριστής final μιας κλάσης δηλώνει ότι αυτή δεν έχει στιγμιότυπα.
- (12) Ένα try μπλοκ μπορεί να έχει περισσότερα του ενός catch μπλοκς.
- (13) Οι διεπαφές δεν σχηματίζουν ξεχωριστή ιεραρχία από αυτή των κλάσεων.
- (14) Ένα byte stream δεδομένων εξόδου έχει μεγαλύτερη μονάδα αποθήκευσης από ένα byte stream αρχείου εξόδου.
- (15) Η άμεση πρόκληση εξαίρεσης δημιουργείται όταν χρησιμοποιείται η εντολή throw στην επικεφαλίδα μιας μεθόδου.

ΘΕΜΑ 2^ο

Δίνεται η εξής περιγραφή: «Υπάρχουν διάφορα είδη συσκευών, όπως οι ηλεκτρικές, οι μηχανικές και οι ηλεκτρονικές συσκευές. Ηλεκτρικές συσκευές είναι τα είναι τα οι ηλεκ. σκούπες και τα ηλεκ. σίδερα. Μηχανικές συσκευές είναι τα ποδήλατα και οι κλειδαριές, ενώ ηλεκτρονικές οι υπολογιστές. Υπάρχουν δύο ηλεκ. σκούπες, η ΧΟΥΒΕΡ1 και ΖΙΜΕΝΣ2, ένα ποδήλατο, το ΙΝΤΕΑΛ3, ένας υπολογιστής, ο ΠΛΑΙΣΙΟ4, και δύο ηλεκ. Σίδερα, το ΦΙΛΙΠΣ5 και το ΧΟΥΒΕΡ6».

- α) Συμπληρώστε ένα πίνακα, όπως ο πίνακας Α παρακάτω, όπου στην πρώτη στήλη να καταγράψετε όλες τις κλάσεις που αναφέρονται στην παραπάνω περιγραφή και στη δεύτερη στήλη τα στιγμιότυπά τους, σύμφωνα με το παράδειγμα. Αν κάποια κλάση δεν έχει στιγμιότυπα, η θέση στη διπλανή στήλη θα παραμείνει κενή.
- β) Συμπληρώστε ένα πίνακα, όπως ο πίνακας Β παρακάτω, όπου στην πρώτη στήλη να καταγράψετε όλες τις κλάσεις που καταγράψατε στον πίνακα Α και στη δεύτερη στήλη τις άμεσες υπερκλάσεις τους, σύμφωνα με το παράδειγμα. [4]

ΚΛΑΣΗ	ΣΤΙΓΜΙΟΤΥΠΑ
Φοιτητής	Γιώργος, Πέτρος

(Α)

ΚΛΑΣΗ	ΥΠΕΡΚΛΑΣΗ
Φοιτητής	Άνθρωπος

(Β)

- γ) Σχεδιάστε μια ιεραρχία που να περιλαμβάνει όλες τις κλάσεις και τα στιγμιότυπα.
- δ) Αν η ιεραρχία αυτή είναι η τελική, δηλ. δεν υπάρχουν άλλες κλάσεις ή στιγμιότυπα, γράψτε γενικό κώδικα Java που να υλοποιεί την ιεραρχία που σχεδιάσατε. Γενικός κώδικας σημαίνει: (1) μόνο τις (επι)κεφαλίδες με κενό σώμα, για τον ορισμό των κλάσεων, και (2) κλήση των αυτόματων δημιουργών (δηλ. αυτών χωρίς ορίσματα) για τη δημιουργία στιγμιότυπων, όπου απαιτείται.

ΘΕΜΑ 3^ο

α) Γράψτε μια μέθοδο Java (`changeString`), που παίρνει σαν όρισμα ένα αλφαριθμητικό `s` και επιστρέφει το αλφαριθμητικό με τις εξής αλλαγές: κάθε 'k' έχει αντικατασταθεί με 'x', αν τα 'k' είναι περισσότερα από τα 'x', και κάθε 'x' έχει αντικατασταθεί με 'k', αν τα 'x' είναι περισσότερα από τα 'k'. Αν δεν υπάρχουν ούτε 'k' ούτε 'x', τότε επιστρέφει το μήνυμα `true`.

β) Γράψτε ένα πρόγραμμα Java που να διαβάζει 50 λέξεις από ένα αρχείο (`words.dat`), να αλλάζει τα 'k' των λέξεων σε 'x', αν τα 'k' είναι περισσότερα από τα 'x', και αντίστροφα, και να τις τυπώνει στην οθόνη, εκτός από αυτές που δεν έχουν ούτε 'k' ούτε 'x'. Κάθε λέξη είναι σε διαφορετική γραμμή στο αρχείο. Να χρησιμοποιηθεί η μέθοδος στο (α).

[Στο (α) για τη διαχείριση του αλφαριθμητικού να χρησιμοποιηθούν μόνο από τις εξής μεθόδους: `charAt (int index)`, `equals (String s)`, `indexOf (char ch)`, `length()`, `replace (char oldChar, char newChar)`, `substring (int beginIndex)`]

ΘΕΜΑ 4^ο

Δίνεται το παρακάτω πρόγραμμα Java.

(α) Εξηγήστε συνοπτικά ποιό είναι το αποτέλεσμα (εσωτερικό και εξωτερικό) της κάθε γραμμής κώδικα 1-17 της μεθόδου `main` της κλάσης 'RunParent'.

(β) Αν στην κλάση `Child` υπήρχε η δήλωση «`double value = 3.0;`», τι θα άλλαζε στα αποτελέσματα; Εξηγήστε.

(γ) Αν η «`String message;`» γίνει «`private String message;`», θα υπάρξει κάποιο πρόβλημα κατά την εκτέλεση της `main`; Αν ναι, πού και γιατί;

(δ) Αν η έκφραση «`super(value)`» στην `Child` έλειπε, θα άλλαζε κάποιο αποτέλεσμα (εσωτερικό ή εξωτερικό); Ποιο και γιατί;

(ε) Σχεδιάστε την ιεραρχία (κλάσεων και στιγμιοτύπων) που θα δημιουργήσει η Java με την εκτέλεση της `main`.

<pre>class Parent{ double value = 0; public Parent(){ value = 1.0; System.out.println("Parent value " +value);} public Parent(double value){ System.out.println("Parent value " + value);} public void setValue (double value){ this.value -= value;}} class Child extends Parent{ String message; public Child (String message){ this.message = message; System.out.println("Child "+message);} public Child (double value){ super(value); this.value = value++; System.out.println("Child value " +value);} public void setMessage (String message){ this.message = message;}</pre>	<pre>public void setValue1 (double value) {this.value += value;} public void setValue2 (double value) {super.setValue (--value);}} public class RunParent{ public static void main(String args []){ (1) Parent P = new Parent(); (2) Child C1 = new Child(5.5); (3) Child C2 = new Child("Second Child"); (4) System.out.println (C1.value); (5) System.out.println (C2.value); (6) System.out.println (C1.message); (7) System.out.println (C2.message); (8) System.out.println (P.value); (9) C1.setMessage ("First Child"); (10) System.out.println (C1.message); (11) C2.setValue (7.5); (12) System.out.println (C2.value); (13) System.out.println (P.value); (14) C1.setValue1 (3); (15) System.out.println (C1.value); (16) C1.setValue2 (8.5); (17) System.out.println (C1.value);}}</pre>
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------