

Εργαστηριακή άσκηση #1

Θέμα: Εξοικείωση με το περιβάλλον προγραμματισμού BlueJ και τα βασικά χαρακτηριστικά των αντικειμένων.

Η εργαστηριακή αυτή άσκηση αποσκοπεί στην εξοικείωση με το περιβάλλον προγραμματισμού BlueJ και την παρουσίαση μερικών από τα χαρακτηριστικά των αντικειμένων. Έμφαση δίδεται στα παρακάτω:

- Μια εφαρμογή Java είναι μια συλλογή από κλάσεις
 - Αντικείμενα μπορεί να δημιουργηθούν βασίζομενα στις κλάσεις
 - Με βάση μια κλάση μπορεί να δημιουργηθούν πολλά αντικείμενα
 - Τα αντικείμενα έχουν «λειτουργίες» (μεθόδους) που επιδρούν πάνω τους
 - Οι λειτουργίες μπορεί να έχουν «παραμέτρους»
 - Οι παράμετροι έχουν «τύπους»
 - Τα αντικείμενα αποθηκεύουν δεδομένα («πεδία»)
 - Οι λειτουργίες και τα πεδία είναι ίδια για όλα τα αντικείμενα της ίδιας κλάσης
 - Οι τιμές που αποθηκεύονται στα πεδία διαφορετικών αντικειμένων (της ίδιας κλάσης) μπορεί να είναι διαφορετικά.
1. Κατεβάστε (από την σελίδα του μαθήματος, στο σύνδεσμο «κώδικας») και αποθηκεύστε το φάκελο **shapes**. Ο φάκελος **shapes** περιέχει τα αρχεία που θα χρησιμοποιηθούν στην εργαστηριακή άσκηση.
 2. **Εκκίνηση του BlueJ.** Κάντε διπλό-κλικ στο εικονίδιο του BlueJ.
 3. **Άνοιγμα ενός «έργου» (project).** Χρησιμοποιώντας την ακολουθία των μενού **project – open project** προσδιορίστε στο πλαίσιο διαλόγου το φάκελο **shapes** που αντιγράψατε (βήμα 1) ως το έργο που θα ανοιχθεί από το BlueJ. Στο βασικό παράθυρο του BlueJ εμφανίζονται 4 παραλληλόγραμμα με ετικέτες **Square**, **Circle**, **Triangle** και **Canvas**. Κάθε ένα από τα παραλληλόγραμμα αυτά αντιστοιχεί σε μια από τις κλάσεις που χρησιμοποιούνται στο έργο. (Σημείωση: Στην περίπτωση που τα παραλληλόγραμμα των κλάσεων είναι γραμμοσκιασμένα, πατήστε το κουμπί **Compile** του BlueJ.)
 4. **Δημιουργία αντικειμένων.** Κάντε δεξί-κλικ πάνω στο παραλληλόγραμμα της κλάσης **Square**. Εμφανίζεται ένα μενού με τις λειτουργίες τύπου new που μπορούμε να επιλέξουμε για τη δημιουργία αντικειμένων. Επιλέγουμε την λειτουργία **new Square()** η οποία δημιουργεί ένα νέο αντικείμενο της κλάσης **Square**. Εμφανίζεται ένα νέο παράθυρο διαλόγου που προτείνει ένα όνομα για το αντικείμενο που θα δημιουργηθεί. Επιλέξτε **OK**. Στο κάτω μέρος του παραθύρου του BlueJ εμφανίζεται ένα κόκκινο παραλληλόγραμμα που αντιστοιχεί στο αντικείμενο που μόλις δημιουργήθηκε. Ταυτόχρονα εμφανίζεται ένα νέο παράθυρο με την ονομασία **BlueJ Demo Canvas** στο οποίο είναι ζωγραφισμένο ένα τετράγωνο.
 5. **Εκτέλεση μεθόδων.** Τώρα που δημιουργήσαμε ένα αντικείμενο τύπου **Square** μπορούμε να εκτελέσουμε όλες τις δημόσιες (public) μεθόδους του καθώς και δυο επιπλέον λειτουργίες (**Inspect**, **Remove**) που παρέχει το BlueJ. Κάντε δεξί-κλικ πάνω στο αντικείμενο που δημιουργήσαμε (κόκκινο παραλληλόγραμμα στο κάτω μέρος του BlueJ). Θα εμφανιστεί μια λίστα με τις διαθέσιμες μεθόδους.
 - a. Εκτελέστε τις εντολές **moveDown()**, **moveUp()**, **MoveLeft()**, **moveRight()**. Παρατηρείστε ότι μετά την εκτέλεση κάθε εντολής το τετράγωνο στο παράθυρο ζωγραφικής μετακινείται.
 - b. Εκτελέστε την εντολή **changeSize(int)**. Θα εμφανιστεί ένα παράθυρο διαλόγου στο οποίο πρέπει να προσδιορίσουμε την παράμετρο που απαιτείται για την εκτέλεση της μεθόδου (μήκος πλευράς τετραγώνου). Η παράμετρος είναι ένας ακέραιος αριθμός (**int**). Εισάγετε την τιμή 100 και μετά επιλέξτε **OK**. Παρατηρείστε ότι το μέγεθος του τετραγώνου στο παράθυρο ζωγραφικής αλλάζει.
 - c. Εκτελέστε τις μεθόδους **moveVertical(int)** και **moveHorizontal(int)**. Προσδιορίστε κατάλληλες τιμές για τις παραμέτρους.
 - d. Εκτελέστε τη μέθοδο **changeColor(String)**. Η παράμετρος που απαιτείται είναι τύπου συμβολοσειράς (**String**). Εισάγετε την τιμή **"green"** (πληκτρολογήστε και τα διπλά εισαγωγικά). Στο παράθυρο ζωγραφικής το χρώμα του τετραγώνου γίνεται πράσινο.

- e. Επιλέξατε τη λειτουργία **Inspect** από το μενού του αντικειμένου που έχουμε δημιουργήσει. Σε ένα νέο παράθυρο εμφανίζονται οι τιμές των μεταβλητών (πεδία) που χρησιμοποιεί η κλάση **Square** για να παραστήσει ένα τετράγωνο. Μην κλείσετε το παράθυρο αυτό. Μετακινείτε το τετράγωνο που δημιουργήσαμε εκτελώντας τις μεθόδους του (π.χ. `moveDown()`) και παρατηρείστε πως αλλάζουν οι τιμές των μεταβλητών `xPosition` και `yPosition`.
- f. Κλείστε το παράθυρο που δίνει τις τιμές των πεδίων του αντικειμένου.
6. **Επεξεργασία κώδικα.** Κάντε δεξί-κλικ πάνω στην κλάση **Square** (όχι στο αντικείμενο) και επιλέξτε τη λειτουργία **Open Editor**. Ο επεξεργαστής κειμένου (editor) του BlueJ ανοίγει σε ένα ξεχωριστό παράθυρο και περιέχει τον πηγαίο κώδικα της κλάσης **Square**. Διπλό-κλικ πάνω στην κλάση έχει το ίδιο αποτέλεσμα.
 - a. Εντοπίστε στην αρχή του κώδικα τις δηλώσεις των πεδίων της κλάσης **Square**.
 - b. Εντοπίστε τον κώδικα για τη δημιουργία ενός νέου αντικείμενου. Παρατηρείστε ότι κάθε νέο τετράγωνο που δημιουργείται έχει μήκος πλευράς 30 pixels και η πάνω-αριστερή του γωνία βρίσκεται στο σημείο (60, 50).
 - c. Μετατρέψτε τον κώδικα έτσι ώστε στο πεδίο size να δίδεται η τιμή 70 (`size=70;`).
 - d. Από το μενού του editor επιλέξτε **Class...Save**. Παρατηρείστε ότι το αντικείμενο που είχαμε δημιουργήσει δεν υπάρχει πλέον στο παράθυρο του BlueJ και ότι η κλάση **Square** εμφανίζεται γραμμοσκιασμένη. Η γραμμοσκίαση υποδηλώνει ότι η κλάση δεν έχει μεταφραστεί. Επιλέξτε τη λειτουργία **Compile** για την κλάση **Square**. Όταν η μετάφραση ολοκληρωθεί, δημιουργείται ένα νέο αντικείμενο και ελέγξτε τις τιμές των πεδίων του (επιλογή **Inspect** από το μενού του αντικειμένου).
 - e. Από το παράθυρο του editor που περιέχει τον κώδικα της κλάσης **Square**, να διαγράψετε το “;” από το τέλος της εντολής `xPosition=60;`. Σώστε τον κώδικα (**Class...Save**) και μεταφράστε τον (**Compile**). Η απαλοιφή του “;” εισάγει ένα συντακτικό λάθος στον κώδικα. Ο μεταφραστής αντιλαμβάνεται το λάθος, και η μετάφραση δεν ολοκληρώνεται. Στο παράθυρο του editor η γραμμή του κώδικα στην οποία εντοπίστηκε το λάθος εμφανίζεται σκιασμένη. Στο κάτω μέρος του παραθύρου του editor εμφανίζεται σχόλιο που περιγράφει το λάθος που εντόπισε ο compiler. Επανεισάγετε το “;”, σώστε τον κώδικα, και μεταφράστε τον ξανά. Παρατηρείστε ότι η μετάφραση ολοκληρώνεται με επιτυχία.
7. Να εντοπίσετε τα τμήματα του κώδικα που αντιστοιχούν στις μεθόδους `moveVertical(int)` και `moveHorizontal(int)`. Προσπαθήστε να κατανοήσετε το πώς ακριβώς λειτουργούν. Προς το παρόν, αγνοείστε τις λεπτομέρειες των μεθόδων `erase()` και `draw()`. Απλά αναφέρουμε ότι η μέθοδος `erase()` σβήνει το τετράγωνο από το παράθυρο ζωγραφικής ενώ η μέθοδος `draw()` το ζωγραφίζει.
8. Να σχεδιάσετε μία νέα μέθοδο `moveTo(int x, int y)` η οποία μετακινεί το τετράγωνο έτσι ώστε η πάνω-αριστερή του γωνία βρίσκεται στο σημείο με συντεταγμένες `(x,y)`.
9. Να εντοπίσετε το τμήμα του κώδικα που αντιστοιχεί στη μέθοδο `moveUp()`. Παρατηρήστε ότι η υλοποίηση της μεθόδου βασίζεται στη μέθοδο `moveVertical()`. Να τροποποιήσετε τον κώδικα `moveUp()` έτσι ώστε να κάνει χρήση της μεθόδου `moveTo(int, int)` που σχεδιάστηκε στο βήμα 8 της εργαστηριακής άσκησης. Μεταφράστε τον κώδικα και δημιουργήστε ένα νέο αντικείμενο. Βεβαιωθείτε ότι η νέα υλοποίηση της `moveUp()` έχει τα ίδια αποτελέσματα με την παλιά. Επαναλάβετε για τις μεθόδους `moveDown()`, `moveLeft()`, και `moveRight()`.
10. Εξετάστε τις κλάσεις **Triangle** και **Circle**. Τα βήματα 5, 6, 7, 8, και 9 να επαναληφθούν (κατάλληλα τροποποιημένα) για τις κλάσεις αυτές.
11. Ζωγραφίστε ένα σπίτι (ή κάτι που μοιάζει με σπίτι) χρησιμοποιώντας αντικείμενα που θα δημιουργήσετε. Μπορείτε να δημιουργήσετε περισσότερα από ένα αντικείμενα για κάθε κλάση, και κάνοντας χρήση των μεθόδων τους να τους δώσετε τις σωστές διαστάσεις, το επιθυμητό χρώμα και να τα μετακινήσετε στο κατάλληλο σημείο του παραθύρου ζωγραφικής.
12. (Σημείωση: για την περάτωση του βήματος αυτού χρειάζεται η γνώση στοιχείων που δεν έχει καλυφθεί έως τώρα). Να γραφεί μια μέθοδος `drawHouse()` η οποία ζωγραφίζει στο παράθυρο ζωγραφικής το σπίτι που σχεδιάσατε στο βήμα 11 της εργαστηριακής άσκησης. Τα ακριβή στοιχεία που προσδιορίζουν κάθε αντικείμενο του σπιτιού μπορεί να παρατηρηθούν μέσω της εντολής `inspect` για το αντικείμενο αυτό.
13. Να εντοπίσετε τα τμήματα του κώδικα που αντιστοιχούν στις μεθόδους `slowMoveVertical(int)` και `slowMoveHorizontal(int)`. Προσπαθήστε να κατανοήσετε το πώς ακριβώς λειτουργούν.