

## 16. Φύλλα Εργασίας

### ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ 1

#### Δραστηριότητα 1

1. Γράψτε στο περιβάλλον του Greenfoot μέσα στην μέθοδο MyWorld() τον κώδικα που ακολουθεί, χωρίς τα σχόλια. Διαβάστε τα σχόλια για την κατανόηση του κώδικα.

```
String s1 = "cat"; //Δεσμεύει μνήμη για το αντικείμενο s1 και της αποδίδει το περιεχόμενο cat

String s2 = " loves"; //Δεσμεύει μνήμη για το αντικείμενο s2 και της αποδίδει το περιεχόμενο loves

String s3 = " balls!"; //Δεσμεύει μνήμη για το αντικείμενο s3 και της αποδίδει το περιεχόμενο balls

String s4 = s1 + s2 + s3; //Δεσμεύει μνήμη για το αντικείμενο s4 και της αποδίδει το περιεχόμενο
των παραπάνω Strings

String s5 = "cat"; //Δεσμεύει μνήμη για το αντικείμενο s5 και της αποδίδει το περιεχόμενο cat

System.out.println(s4); //Εμφανίζει το περιεχόμενο του String s4

System.out.println("Και η κρυμμένη λέξη είναι: " + s3.charAt(1) + s2.charAt(2) + s1.charAt(1) +
s1.charAt(2)); // Εμφανίζει συγκεκριμένα γράμματα από τα παραπάνω strings, γνωρίζοντας ότι η
αρίθμηση των χαρακτήρων σε ένα αντικείμενο String ξεκινάει από το 0 και όχι από το 1

System.out.println(s1.length()); //Εμφανίζει το μήκος του αντικειμένου s1

System.out.println(s2.length()); //Εμφανίζει το μήκος του αντικειμένου s2

System.out.println(s3.length()); //Εμφανίζει το μήκος του αντικειμένου s3

System.out.println(s4.length()); //Εμφανίζει το μήκος του αντικειμένου s4, δηλαδή την πρόσθεση
των παραπάνω

System.out.println(s1.equals(s5)); //Συγκρίνει τα δύο περιεχόμενα των αντικειμένων s1 και s5

System.out.println(s1 == s5); //Συγκρίνει τα δύο αντικείμενα s1 και s5, δηλαδή την διεύθυνσή
τους στην μνήμη. Δύο αντικείμενα δεν έχουν ποτέ ίδια διεύθυνση
στη μνήμη.
```

#### Δραστηριότητα 2

- Δημιουργήστε ένα κόσμο CarWorld που επεκτείνει τον World.
- Δημιουργήστε για αυτήν την κλάση ένα κατασκευαστή αν δεν υπάρχει ήδη.
- Δημιουργήστε την κλάση Car.
- Δηλώστε τέσσερις νέες μεταβλητές για την κλάση στον κατασκευαστή του κόσμου και προσθέστε τα αντικείμενα στον κόσμο στις θέσεις που επιθυμείτε.
- Δημιουργήστε μία δημόσια κλάση χωρίς τύπο επιστροφής και ορίσματα, που καλώντας την μέσα από το γραφικό περιβάλλον θα κουνάει τα αμάξια 5 θέσεις δεξιά.

#### Δραστηριότητα 3

1. Ποιες από τις παρακάτω μεθόδους ανήκει στην τάξη String;

- Length()
- compareTo()
- equals()
- substring()
- Καμία από αυτές
- Όλες

2. Δεδομένου του κώδικα

```
String s1 = "yes";
```

```
String s2 = "yes";
```

```
String s3 = new String(s1);
```

Ποια/ες από τις παρακάτω προτάσεις είναι αληθείς;

- s1 == s2
- s1 = s2
- s3 == s1
- s1.equals(s2)
- s3.equals(s1)

3. Υποθέστε ότι τα s1 και s2 είναι δύο συμβολοσειρές. Ποια/ες από τις παρακάτω προτάσεις είναι έγκυρη;

- String s3 = s1 +s2;
- String s3 = s1 - s2;
- s1 <= s2
- s1.compareTo(s2);
- int m = s1.length();

## ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ 2

### Δραστηριότητα 1

Έχοντας υπόψιν ότι για την κλήση μεθόδου από ένα στιγμιότυπο αντικειμένου χρησιμοποιείται ο γενικός τύπος **Όνομα\_αντικειμένου.Όνομα\_μεθόδου(παράμετροι)**, π.χ. `car.move(5)` το αντικείμενο `car` καλεί την μέθοδο `move` με όρισμα `5`, δημιουργείτε μέσα στον κώδικα της Δραστηριότητας 2 του προηγούμενου Φύλλου Εργασίας, μέσα στην `act`, μία ξεχωριστή κλήση μεθόδου αυτού του τύπου για κάθε αμάξι.

### Δραστηριότητα 2

Χρησιμοποιείτε τον κώδικα από την προηγούμενη δραστηριότητα και δημιουργείτε μία καινούργια μέθοδο με όνομα `carMove` μέσα στην κλάση `Car` όπου θα περιέχει τις κλήσεις των μεθόδων από τα αντικείμενα που χρησιμοποιήσατε στην προηγούμενη. Η `carMove` θα είναι δημόσια χωρίς τύπο επιστροφής και χωρίς παραμέτρους.

Πραγματοποιείτε κλήση της μεθόδου που δημιουργήσατε από την `act()`.

### Δραστηριότητα 3

Συνεχίζουμε στον κώδικα που χρησιμοποιήσαμε στις προηγούμενες ασκήσεις.

- Φτιάχνουμε μέσα στην κλάση `Car` μια μέθοδο `carMoveValue()` η οποία θα είναι ίδια με την `carMove()`. Διαφέρουν ωστόσο στο γεγονός ότι τώρα η μέθοδος παίρνει όρισμα μια ακέραιη μεταβλητή.
- Διαλέξτε το όνομα της μεταβλητής, προσέξτε ωστόσο μέσα στο όρισμα να γράψετε και τον τύπο της, καθώς εκεί την ορίζετε για πρώτη φορά.
- Αντικαταστήστε τα ορίσματα-νούμερα που έχετε δώσει μέσα στις εντολές της νέας μεθόδου, και τοποθετήστε αντ' αυτού ως όρισμα για αυτές το όρισμα της μεθόδου σας.

## ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ 3

### Δραστηριότητα 1

- Δημιουργήστε έναν κόσμο και τοποθετήστε με όποιον τρόπο επιθυμείτε μία μέλισσα και λίγα λουλούδια.
- Υλοποιήστε έναν μετρητή όπως διδαχθήκατε στην αρχή της Διδακτικής ώρας.
- Χρησιμοποιήστε της μεθόδους από το γραφικό περιβάλλον για να κινηθεί η μέλισσα και να δείτε τον κώδικά σας να λειτουργεί.

### Δραστηριότητα 2

- Δημιουργήστε μέσα σε ένα κόσμο του Greenfoot μια κλάση Animal χωρίς κάποια εικόνα.
- Μέσα στην μέθοδο act() της κλάσης γράψτε κώδικα τέτοιο ώστε να κάνει το αντικείμενο κάποιες απλές και σύντομες κινήσεις.
- Δημιουργήστε μία ακόμη κλάση με όνομα ενός ζώου και όποια εικόνα θέλετε που να είναι υποκλάση της Animal(π.χ. Cat extends Animal).
- Προσθέστε το τελευταίο αντικείμενο στον κόσμο και πατήστε run. Τι παρατηρείτε; Η καινούργια μέθοδος κληρονομεί την Animal. Αυτό ονομάζεται κληρονομικότητα.
- Γράψτε μέσα στην act της νέας κλάσης κάποιες καινούργιες κινήσεις και ξαναδοκιμάστε το πρόγραμμα. Τώρα η νέα κλάση κουνιέται σύμφωνα με τον τελευταίο κώδικα που γράψατε. Αυτό γίνεται γιατί παρόλο που ισχύει η κληρονομικότητα, μπορείτε να επιλέξετε ποιες μεθόδους θα ξανά ορίσετε και ποιες θα κληρονομήσετε ως έχουν.