

## Σύντομη παρουσίαση της πρώτης εργασίας στον ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΗ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟ

Αρχικά έφτιαξα την κλάση `animal` και την κλάση `Main`. Στην κλάση `animal` δήλωσα 5 private μεταβλητές ως ιδιότητες την κλάσης `animal`. Δημιούργησα έναν constructor περνώντας ως ορίσματα τις ιδιότητες του κάθε ζώου επειδή όλα τα ζώα έχουν τις συγκεκριμένες ιδιότητες. Στην συνέχεια δημιούργησα getters και setters. Τους setters τους χρειάζομαι ώστε στην επεξεργασία να αλλάζω τις τιμές που έχω περάσει στο κάθε ζώο μέσω του constructor. Ενώ τους getters τους χρειάζομαι για να παίρνω τις τιμές των ιδιοτήτων του κάθε αντικείμενου. Στην `Main` δημιούργησα τις στατικές μεθόδους (συναρτήσεις) `intInput()` και `doubleInput()` για τον χειρισμό εξαιρέσεων σε ακέραιες και πραγματικές τιμές ως είσοδο, την `AnimalSearch` με όρισμα την λίστα με τα ζώα ώστε να κάνει αναζήτηση του ζώου που έχει έναν συγκεκριμένο κωδικό (τον οποίο δίνει ο χρήστης) και να το επιστρέφει (το συγκεκριμένο ζώο/αντικείμενο), και τέλος την `answerEquals` η οποία επιστρέφει `true` αν ο χρήστης δώσει την τιμή "ΝΑΙ" ως είσοδο αλλιώς επιστρέφει `false` αν ο χρήστης δώσει την τιμή "ΟΧΙ" ή αν δώσει μη έγκυρη τιμή.

Στη συνέχεια δημιούργησα την `main` στην οποία αρχικά έφτιαξα 10 αντικείμενα (ζώα), τα πέρασα σε μια λίστα και μετά αποθήκευσα σε εάν αρχείο όλη την λίστα... Στη συνέχεια έβαλα όλο αυτό το κώδικα σε σχόλια ώστε το πρώτο πράγμα που να κάνω στην `main` να είναι φόρτωση σε μια άλλη λίστα το αρχείο που είχα ήδη αποθηκεύσει (που έχει τον κατάλογο ζώων). Υστέρα χρησιμοποίησα έναν βρόχο επανάληψης `do-while` (ώστε να εμφανίζεται επαναλαμβανόμενα το μενού επιλογών στο χρήστη μέχρι εκείνος να δώσει την τιμή 7) μέσα στο οποίο χρησιμοποίησα μια `switch` ανάλογα με την επιλογή που δώσει ο χρήστης από το μενού επιλογών. Έτσι αν ο χρήστης επέλεξε την πρώτη επιλογή εμφανίζω τα διαθέσιμα ζώα του ζωολογικού κήπου, αν επιλέξει την δεύτερη του ζητάμε να δώσει τιμές για τις ιδιότητες του νέου ζώου με κατάλληλο έλεγχο για `exceptions` και εγκυρότητας των δεδομένων για το βάρος (εύρος 0,1-100000) και την ηλικία (εύρος 1-400) και δημιουργώ το νέο αντικείμενο (ζώο) προσθέτοντας το τελικά στην λίστα. Αν όμως ο χρήστης επιλέξει την τρίτη επιλογή τότε κάνουμε αναζήτηση ζώου βάση ονόματος το οποίο δίνει ο χρήστης ως `input`. Αν το ζώο με το συγκεκριμένο όνομα δεν βρεθεί ζητάμε από τον χρήστη να δώσει νέο όνομα μέχρι να βρεθεί κάποιο ζώο. Το ίδιο κάνουμε και στην τέταρτη περίπτωση με τη διαφορά ότι αυτή τη φορά καλούμε την συνάρτηση `search` η οποία βρίσκει το ζώο με τον συγκεκριμένο κωδικό, το επιστρέφει στο κυρίως πρόγραμμα και εμφανίζουμε τις ιδιότητες του συγκεκριμένου ζώου. Αν ο χρήστης επιλέξει την πέμπτη επιλογή αρχικά βρίσκουμε το ζώο με τον συγκεκριμένο κωδικό μέσω την συνάρτησης `search` ρωτάμε τον χρήστη ποιες ιδιότητες του συγκεκριμένου ζώου θέλει να επεξεργαστεί μέσω της συνάρτησης `answerEquals` και του δίνουμε την επιλογή για επεξεργασία χρησιμοποιώντας τους setters.

Αν ο χρήστης έχει επιλέξει την έκτη επιλογή βρίσκουμε το ζώο με τον συγκεκριμένο κωδικό μέσω την συνάρτησης `search` και αφού το βρούμε το αφαιρούμε από την λίστα ενώ αν επιλέξει την έβδομη επιλογή τερματίζει το πρόγραμμα εμφανίζοντας

ένα μήνυμα «αντίο». Τέλος αν ο χρήστης δεν εισάγει κάποια από αυτές τις επιλογές του εμφανίζουμε μήνυμα ότι δεν έδωσε έγκυρη επιλογή.

Αφού βγούμε από την switch χρησιμοποιούμε μια if έτσι ώστε αν ο χρήστης έχει επιλέξει επεξεργασία, διαγραφή ή προσθήκη νέου ζώου τότε ενημερώνουμε το αρχείο με την ανανεωμένη λίστα.