

# ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΣΤΟ ΔΙΑΔΙΚΤΥΟ & ΣΤΟΝ ΠΑΓΚΟΣΜΙΟ ΙΣΤΟ

Ακαδ. Έτος 2020-21

## Άσκηση 01

Στόχοι άσκησης: Βασικές έννοιες, δημιουργία κλάσεων, κληρονομικότητα, πολυμορφισμός, *constructors*, προσδιοριστές πρόσβασης, *static*, *final*, *this keyword*. Είσοδος από το πληκτρολόγιο, χειρισμός και δημιουργία εξαιρέσεων, χειρισμός εισόδου-εξόδου.

Σε αυτή την άσκηση θα δημιουργήσετε ένα project το οποίο θα περιλαμβάνει τις βασικές κλάσεις που θα χρησιμοποιηθούν σε μία εφαρμογή **διαχείρισης ραντεβού τακτικών ιατρικών εξετάσεων**. Θα δημιουργήσετε τις βασικές κλάσεις που θα χρειαστούν και τη βασική λειτουργικότητά τους.

### Αναλυτικά Βήματα:

#### 1. (προκαταρκτικά βήματα) Έλεγχος εγκατάστασης και καλής λειτουργίας περιβάλλοντος

- 1.1. Έλεγχος εγκατάστασης jdk, έλεγχος εγκατάστασης eclipse (ή άλλου IDE της επιλογής σας), έλεγχος ενσωμάτωσης java documentation, έλεγχος δημιουργίας και εκτέλεσης ενός απλού προγράμματος.

#### 2. Εκκίνηση project

- 2.1. Δημιουργία ενός Java Project με τίτλο JavaBasics.
- 2.2. Δημιουργία ενός πακέτου με τίτλο mainpackage

#### 3. Δημιουργία βασικών κλάσεων

- 3.1. Δημιουργία μίας κλάσης Users, η οποία θα περιλαμβάνει τα βασικά χαρακτηριστικά κάθε κατηγορίας χρήστη της εφαρμογής.
  - Χαρακτηριστικά: όνομα χρήστη (username), κωδικό (password), όνομα (name), επίθετο (surname), και άλλα στοιχεία, και τέλος έναν μετρητή χρηστών (usersCounter) ο οποίος θα ξεκινά από την τιμή 0 και θα αυξάνεται κατά ένα, όταν θα καλείται ο constructor της κλάσης (υπόδειξη: ποιος ορισμός counter ως static μεταβλητής)
  - Μέθοδοι: Σύνδεση / Αποσύνδεση (login(), logout()). Τυπικοί getters και setters για όλες τις μεταβλητές (υπόδειξη: είναι δυνατό να δημιουργηθούν για όλες τις μεταβλητές;)
  - Δημιουργία τυπικού constructor για την κλάση Users.
- 3.2. Δημιουργία υποκλάσης ασθενούς (Patient), η οποία θα χρησιμοποιεί τα χαρακτηριστικά και τις μεθόδους της κλάσης Users αλλά επιπλέον:

- Χαρακτηριστικά: Επιπρόσθετα θα χρησιμοποιεί το χαρακτηριστικό αριθμός ΑΜΚΑ. Το χαρακτηριστικό αυτό θα δίδεται μία φορά για κάθε αντικείμενο Patient και στη συνέχεια δεν θα μπορεί να αλλάξει.
  - Ο constructor θα πρέπει να τροποποιηθεί κατάλληλα ώστε να περιλαμβάνει τον αριθμό ΑΜΚΑ, τον οποίο θα αρχικοποιεί τη στιγμή της δημιουργίας κάθε αντικείμενου τύπου Patient (Υπόδειξη: κάνετε override τον constructor της υπερκλάσης Users)
  - Μέθοδοι της κλάσης: Εγγραφή χρήστη (registration) και άλλες μεθόδους που απαιτούνται, όπως ενδεικτικά, αναζήτηση διαθέσιμου ραντεβού για συγκεκριμένο ιατρό, αναζήτηση ραντεβού για οποιονδήποτε διαθέσιμο ιατρό μίας ειδικότητας, προβολή προγραμματισμένων ραντεβού, προβολή ιστορικού ραντεβού κτλ. Ποιες επιπλέον μεθόδους ή χαρακτηριστικά θα πρέπει να δημιουργήσετε για την κλάση Patient;
- 3.3. Δημιουργία υποκλάσης ιατρού (Doctor), με επιπλέον χαρακτηριστικό την ειδικότητα (specialty). Ποια επιπλέον χαρακτηριστικά και ποιες νέες μεθόδους χρειάζεται η κλάση Doctor; Να δημιουργηθούν αντίστοιχα. (Ενδεικτικές μέθοδοι: καταχώρηση διαθεσιμότητας ιατρού για ραντεβού ανά μήνα, προβολή προγράμματος ραντεβού κτλ)
- 3.4. Δημιουργία υποκλάσης διαχειριστή (Admin). Ο διαχειριστής θα εισαγάγει στο σύστημα τους Ιατρούς. Ποια επιπλέον χαρακτηριστικά και ποιες νέες μεθόδους χρειάζεται η κλάση Admin; Να δημιουργηθούν αντίστοιχα.
- 3.5. Δημιουργία της κλάσης ραντεβού (Appointment), η οποία θα χρησιμοποιείται για την καταγραφή ενός ραντεβού. Η κλάση αυτή ενδέχεται να χρησιμοποιεί ως χαρακτηριστικά αντικείμενα των κλάσεων Patient και Doctor.
- 3.6. Δημιουργία άλλων πιθανώς κλάσεων που είναι αναγκαίες για την εφαρμογή σύμφωνα με το σχεδιασμό σας.

#### 4. Δοκιμή και τεκμηρίωση των βασικών κλάσεων

- 4.1. Δημιουργήστε μία κλάση με όνομα CreateUsers η οποία θα περιλαμβάνει τη συνάρτηση main(). Μέσω της κλάσης αυτής θα δημιουργήσετε αντικείμενα τύπου Patient, Doctor, Admin κτλ. Δημιουργήστε ένα αντικείμενο από κάθε κλάση και χρησιμοποιείστε τις μεθόδους που έχετε δημιουργήσει. Με τη χρήση ενδεικτικών μηνυμάτων, δείξτε τη δημιουργία και τα χαρακτηριστικά των αντικείμενων.
- 4.2. Προσθέστε κατάλληλη τεκμηρίωση (documentation) για τις κλάσεις Patient, Doctor, Admin και Appointment και για όσες μεθόδους απαιτείται.

#### 5. Έλεγχος εισόδου και χειρισμός εξαιρέσεων

- 5.1. Μέσω της κύριας κλάσης CreateUsers, χρησιμοποιείστε ένα αντικείμενο τύπου Scanner ώστε να δημιουργήσετε αντικείμενα τύπου Patient και Doctor, λαμβάνοντας τα ορίσματα του constructor από το πληκτρολόγιο. Χρησιμοποιείστε exceptions με τη βοήθεια των εντολών try...catch για να ελέγξετε την ορθότητα των παραμέτρων εισόδου.
- 5.2. Προσθέστε κατάλληλη τεκμηρίωση (documentation) όπου απαιτείται.

#### 6. Δημιουργία αντικειμένων με τη χρήση αρχείου εισόδου

- 6.1. Δημιουργήστε ένα txt αρχείο το οποίο περιλαμβάνει τα χαρακτηριστικά ενός αντικειμένου τύπου Patient (με κενά μεταξύ των χαρακτηριστικών). Δημιουργήστε ένα αντικείμενο, διαβάζοντας τα χαρακτηριστικά του από αυτό το αρχείο.
- 6.2. Χρησιμοποιείτε το μηχανισμό εξαιρέσεων για να ελέγξετε για πιθανά σφάλματα ανάγνωσης, εγγραφής, και σωστού τύπου των χαρακτηριστικών.

## 7. Διάγραμμα Κλάσεων

- 7.1. Περιλάβετε στο παραδοτέο σας το διάγραμμα κλάσεων (Class Diagram).