



Εργαστήριο 8

Εξαιρέσεις - Πακέτα

(Exceptions - Packages)

8.1. Χειρισμός Εξαιρέσεων

Να δημιουργήσετε μία τάξη 'Υπαλλίλος' η οποία θα περιγράφει έναν υπάλληλο με τα εξής χαρακτηριστικά (attributes): όνομα (name), ημερομηνία γέννησης (birthDate), ώρες εργασίας ανά μήνα (workingHours), ώρες υπερωριών ανά μήνα (overtime), ημερομίσθιο (dayWage), πλήθος τέκνων (childNo), συντελεστής επιδόματος (bonusFactor).

Εφοδιάστε την τάξη με τις απαραίτητες μεθόδους πρόσβασης (access methods) και μεταβολής (mutator methods).

Επίσης θα παρέχονται οι παρακάτω μέθοδοι:

- Υπολογισμός αποδοχών (ypologismosApodohon)
 $((workingHours + overtime) * dayWage * (1 + bonusFactor))$
- Υπολογισμός συντελεστή επιδόματος (ypologismosSintelestiEpidomatos)
- Εμφάνιση ωρών εργασίας (oresErgasias)
- Εμφάνιση στοιχείων υπαλλήλου (emfanisiYpallilou)

Για να υλοποιήσετε τις μεθόδους και την λειτουργικότητα της τάξης του υπαλλήλου ακολουθήστε τις παρακάτω οδηγίες :

- Ο δομητής περιμένει έναν θετικό αριθμό για τα τέκνα του υπαλλήλου. Εάν ο αριθμός είναι αρνητικός τότε να προκαλείται η εξαίρεση που αφορά τα λάθος ορίσματα κατά την κλήση μίας μεθόδου (**IllegalArgumentException**).
- Κατά τον υπολογισμό των αποδοχών να υπολογίζεται ο συντελεστής για το επίδομα ως εξής: $100 - (50 / \text{αριθμό τέκνων})$. Να γίνεται χειρισμός της εξαίρεσης που μπορεί να προκληθεί εάν ο αριθμός των τέκνων είναι 0 (**ArithmeticException**).
- Η ημερομηνία γέννησης θα εισάγεται ως ένα 8-ψήφιο String, (τα πρώτα δύο ψηφία είναι η ημέρα γέννησης, τα επόμενα δύο είναι ο μήνας γέννησης, και τα 4 τελευταία το έτος γέννησης). Κατά την εισαγωγή της ημερομηνίας να ελέγχεται εάν η ημέρα είναι μεταξύ 1-

31 και ο μήνας είναι μεταξύ 1-12, διαφορετικά να προκαλείται μία εξαίρεση (**InvalidBirthDateException**) που να εμφανίζει το κατάλληλο μήνυμα. Για να χειριστείτε την ημερομηνία χρησιμοποιήστε την μέθοδο

```
public String substring(int beginIndex[,int endIndex])
```

- Οι ώρες εργασίας και οι ώρες υπερωριών να καταχωρούνται σε έναν πίνακα 12 θέσεων (μία θέση για κάθε μήνα του έτους). Να υλοποιήσετε μία μέθοδο που θα εμφανίζει το σύνολο των ωρών εργασίας και να προβλέπεται ο χειρισμός εξαίρεσης που προκαλείται όταν γίνει πρόσβαση εκτός των ορίων του πίνακα (**ArrayIndexOutOfBoundsException**).
- Να υπολογίζετε το σύνολο των υπερωριών και εάν είναι μεγαλύτερο από 120 να καλείται μία μέθοδος που να 'ρίχνει' μία ελεγχόμενη εξαίρεση με μήνυμα "Too many hours ... " ή εάν είναι μικρότερες από 0 να εμφανίζει το εξ ορισμού μήνυμα της ίδιας εξαίρεσης. (**InvalidHoursException**).
- Μετά τον υπολογισμό των υπερωριών, είτε προκλήθηκε εξαίρεση είτε όχι να εμφανίζεται το μήνυμα "End of calculation".

8.2. Πακέτα

1. Χρήση τάξεων με ίδιο όνομα

Έστω ότι έχετε δύο κλάσεις Network, η μία περιγράφει ένα δίκτυο H/Y και η δεύτερη ένα δίκτυο συγκοινωνιών. Να δημιουργηθούν οι τάξεις με όνομα Network, οι οποίες να διαθέτουν ένα μέλος Description και εκτός από τον εξ ορισμού δομητή να έχουν μία μέθοδο toString. Τοποθετήστε την πρώτη σε ένα πακέτο με όνομα Computer και την δεύτερη σε ένα πακέτο με όνομα Traffic. Τα μηνύματα που θα εμφανίζονται είναι τα ακόλουθα :

- Τάξη για δίκτυα H/Y πακέτο
 1. Default constructor : This is a Data Network
 2. toString : This network carries bits and bytes
- Τάξη για Δίκτυα Συγκοινωνιών
 1. Default constructor : This is a Traffic Network
 2. toString : This network is used by vehicles

Σε μία τρίτη τάξη με όνομα TestNetwork.java (στον τρέχοντα κατάλογο, του οποίου υποκατάλογοι είναι οι "Computer" και "Traffic") δημιουργήστε αντικείμενα των δύο τάξεων Network και χρησιμοποιήστε την μέθοδο toString για να δείξετε την πρόσβαση μέσω πακέτων στις μεθόδους.

Προσοχή: Η μεταγλώττιση των κλάσεων "Network" να γίνει από γραμμή εντολών

```
javac Computer/Network.java  
javac Traffic/Network.java
```

2. Δημιουργία προσωπικών πακέτων

Για την δημιουργία δικών μας πακέτων απαιτούνται τρεις ενέργειες. Έστω ότι θέλουμε να δημιουργήσουμε ένα πακέτο με όνομα 'tei' και σε αυτό θέλουμε να τοποθετήσουμε 3 τάξεις – Informatics, Nursing και Marketing.

1. Δημιουργήστε έναν υποκατάλογο με όνομα 'tei'.
2. Στην αρχή της κάθε τάξης Informatics, Nursing και Marketing, τοποθετήστε την εξής δήλωση:
`package tei;`
3. Τοποθετήστε τα τρία αρχεία, Informatics.java, Nursing.java και Marketing.java, στον υποκατάλογο tei.

Οι τάξεις αυτές θα έχουν μία μέθοδο toString η οποία θα εμφανίζει το αντίστοιχο μήνυμα :

- Informatics : “Τμήμα Πληροφορικής ”
- Nursing : “Τμήμα Νοσηλευτικής”
- Marketing : “Τμήμα Εμπορίας και Διαφήμισης ”

Δημιουργήστε μία τάξη που να χρησιμοποιεί τις 3 παραπάνω τάξεις.

3 Εσωτερικά πακέτα

Τροποποιήστε την προηγούμενη άσκηση ώστε να υπάρχουν μέσα στο πακέτο 'tei, τρία υπο-πακέτα STEF, SEYP, SDO και οι τρεις τάξεις Informatics, Nursing και Marketing να βρίσκονται στο αντίστοιχο υπο-πακέτο. Συμπεριλάβετε τον κατάλληλο κώδικα για την δημιουργία των πακέτων. Για παράδειγμα για να εισαχθεί η τάξη Informatics:

```
import tei.STEF.Informatics;
```

Δημιουργήστε μία τάξη που να χρησιμοποιεί την παραπάνω δομή.