

ΑΣΚΗΣΕΙΣ Πακέτα (*Packages*)

ΑΣΚΗΣΗ-1^η (Πακέτα - *Packages*)

Να γραφεί το πρόγραμμα Java που δημιουργεί το **πακέτο (φάκελο) Shapes**. Στον φάκελο τοποθετείστε σε ξεχωριστά αρχεία .java (package) τις παρακάτω κλάσεις:

- (1) **interface Emvadon**, που περιέχει την υπογραφή της μεθόδου:
float computeEmvadon(float x, float y); (*υπολογισμός εμβαδού σχήματος*)
- (2) την κλάση **Triangle** που υλοποιεί την διεπαφή Emvadon και την μέθοδο computeEmvadon(float x, float y) (εμβαδόν τριγώνου= $x * y/2$).
- (3) την κλάση **Rectangle** που υλοποιεί την διεπαφή Emvadon και την μέθοδο computeEmvadon(float x, float y) (εμβαδόν ορθογωνίου= $x * y$).

Στον **αρχικό φάκελο (root)** αποθηκεύστε την κλάση **InterfaceEmvadon** που περιέχει την main(). Δημιουργήστε 2 αντικείμενα των τύπων Triangle και Rectangle και με αυτά υπολογίστε και εμφανίστε το αντίστοιχο εμβαδό για το κάθε σχήμα.

ΑΣΚΗΣΗ-2^η (Πακέτα - *Packages*)

Να γραφεί το πρόγραμμα Java που δημιουργεί το **πακέτο (φάκελο) Anadromi**. Στον φάκελο τοποθετείστε σε ξεχωριστά αρχεία .java (package) τις παρακάτω κλάσεις:

- (1) **class Paragontiko**, που περιέχει την μέθοδο **paragontiko(int n)** που υπολογίζει και επιστρέφει το παραγοντικό ενός αριθμού.
- (2) **class Fibonacci**, που περιέχει την μέθοδο **fibonacci(int n)** που υπολογίζει και επιστρέφει τον αριθμό Fibonacci που αντιστοιχεί σε μία θέση της ακολουθίας.
- (3) **class PalidromaStrings**, που περιέχει την μέθοδο **boolean palString(String s)** που επιστρέφει true ή false, ανάλογα αν ένα String είναι παλίνδρομο ή όχι.
- (4) **class ProtoiArithmoi**, που περιέχει την μέθοδο **boolean prwtoi(int n, int d)** που επιστρέφει true ή false αν ένας αριθμός είναι πρώτος ή όχι, για μια δεδομένη ακολουθία αριθμών.
- (5) **class Hanoi**, που περιέχει την μέθοδο **void hanoi(int n, char source, char storage, char destination)** που αναδρομικά υπολογίζει και εμφανίζει τις κινήσεις στους πύργους – Hanoi, για δεδομένο αριθμό δίσκων.
- (6) **class MenouEpilogon**, περιέχει την μέθοδο **menou()** που εμφανίζει το μενού επιλογών των ανωτέρω εργασιών.

Στον **αρχικό φάκελο** (*root*) αποθηκεύστε την κλάση **TestAnadromi** που περιέχει την `main()`. Σε διπλή επανάληψη (`loop – do-while`) εμφανίζεται το μενού επιλογών και επιλέγεται μια εργασία η οποία και εκτελείται. Μετά την εκτέλεση της κάθε εργασίας εμφανίζεται το μενού και ζητείται νέα επιλογή εργασίας, μέχρι να επιλεγεί το τέλος του προγράμματος (6^η επιλογή στο μενού).

ΑΛΥΤΕΣ ΑΣΚΗΣΕΙΣ

ΑΣΚΗΣΗ-1^η (Packages)

Έχοντας σαν πρότυπο την λυμένη 2^η άσκηση, να γραφεί πρόγραμμα Java που υλοποιεί ένα πακέτο - λειτουργιών μετατροπών νομισμάτων- με την βοήθεια ενός μενού επιλογών. Το πρόγραμμα θα αποθηκεύσει στον **φάκελο (πακέτο) Metatropes**, τις κλάσεις μετατροπών που θα έχουν την παρακάτω μορφή:

- (1) **class EuroToDollar**, που περιέχει την μέθοδο ***double eurotodollar(double poso, double isotimia)*** που υπολογίζει και επιστρέφει το αντίστοιχο ποσό σε δολάρια ενός ποσού ευρώ (*poso*), σύμφωνα με την ισοτιμία της ημέρας (*isotimia*).
- (2) **class DollarToEuro**, που περιέχει την μέθοδο ***double dollartoeuro(double poso, double isotimia)*** που υπολογίζει και επιστρέφει το αντίστοιχο ποσό σε ευρώ ενός ποσού δολαρίων (*poso*), σύμφωνα με την ισοτιμία της ημέρας (*isotimia*).
- (3) : : :

Προσθέστε τις κλάσεις και άλλων μετατροπών, π.χ. Γεν Ιαπωνίας, Bitcoin, Κορώνα Σουηδίας, Δολάριο Καναδά, Λίρα Στερλίνα, κλπ. Μην ξεχάσετε την κλάση **MenouEpilogon** που θα υλοποιεί το μενού των τροποποιήσεων.

Στον **αρχικό φάκελο** (*root*) αποθηκεύστε την κλάση **TestMetatropes** που περιέχει την `main()`. Σε διπλή επανάληψη (`loop – do-while`) εμφανίζεται το μενού επιλογών και επιλέγεται μια εργασία μετατροπής η οποία και εκτελείται. Μετά την εκτέλεση της κάθε εργασίας εμφανίζεται το μενού και ζητείται νέα επιλογή εργασίας, μέχρι να επιλεγεί το τέλος του προγράμματος.

ΑΣΚΗΣΗ-2^η (Packages)

Έχοντας σαν πρότυπο την λυμένη 2^η άσκηση, να γραφεί πρόγραμμα Java που υλοποιεί ένα πακέτο - λειτουργιών μετατροπών μονάδων- με την βοήθεια ενός μενού επιλογών. Επιλέξτε μόνοι σας τις μονάδες μετατροπής, π.χ. μίλια σε χιλιόμετρα και το αντίστροφο, λίτρα σε γαλόνια και το αντίστροφο, ίντσες σε εκατοστά και το αντίστροφο, κλπ. Μην ξεχάσετε την κλάση **MenouEpilogon** που θα υλοποιεί το μενού των τροποποιήσεων.

Στον **αρχικό φάκελο** (*root*) αποθηκεύστε την κλάση **TestMetatropesMonadon** που περιέχει την `main()`. Σε διπλή επανάληψη (`loop – do-while`) εμφανίζεται το μενού επιλογών και επιλέγεται μια εργασία μετατροπής η οποία και εκτελείται. Μετά την εκτέλεση της κάθε εργασίας εμφανίζεται το μενού και ζητείται νέα επιλογή εργασίας, μέχρι να επιλεγεί το τέλος του προγράμματος.

ΑΣΚΗΣΗ-3^η (Η άλυτη άσκηση του 4^{ου} Εργαστηρίου – Να λυθεί με την χρήση package)

Έχοντας σαν πρότυπο την λυμένη 2η άσκηση, να γίνει το πρόγραμμα αναζήτησης και εμφάνισης των στοιχείων ενός υπάλληλου σε ένα πίνακα **N** - αντικειμένων τύπου Employee. Ορίστε την κλάση **Employee** με private πεδία: (1) Όνομα, String, (2) Επώνυμο, String, (3) Κωδικός Υπαλ, int και (4) Μισθός, double.

Το πρόγραμμα θα εκτελεί (1) σειριακή αναζήτηση ως προς το Επώνυμο, (2) ταξινόμηση ως προς τον Κωδικό Υπαλ. και (3) δυαδική αναζήτηση ως προς τον Κωδικό Υπαλ. (σε ταξινομημένο ως προς τον Κωδικό-πίνακα). Οι εργασίες θα υλοποιούνται από ένα **μενού επιλογών**, ως προς την αναζήτηση και ένα **2^ο μενού επιλογών θα επιτρέπει την επιλογή της μεθόδου ταξινόμησης (όλες οι διδαχθείσες)**.

Οι εργασίες θα καλούνται κατάλληλα, δηλ. δεν θα μπορείς να κάνεις δυαδική αναζήτηση, αν δεν έχει **εκτελεστεί πρώτα η ταξινόμηση** ως προς τον κωδικό (όποια ταξινόμηση θέλετε εσείς). Θα εμφανίζονται τα κατάλληλα μηνύματα που θα καθοδηγούν τις εργασίες.

Για την σειριακή αναζήτηση: Το πρόγραμμα θα ζητά το Επώνυμο του υπάλληλου που αναζητάμε. Αν το βρει θα εμφανίσει όλα τα στοιχεία του, διαφορετικά το μήνυμα "Den γparxei o Ypallilos".

Για την δυαδική αναζήτηση: Το πρόγραμμα θα ζητά τον κωδικό του και θα υλοποιεί τα ίδια με την σειριακή αναζήτηση (εμφάνιση στοιχείων).