

ΕΡΓΑΣΙΑ ΟΝΤΟΚΕΝΤΡΙΚΟΥ ΠΡΟΓΡ/ΣΜΟΥ 2019-2020

(Παράδοση 7/6/2020)

Στα πλαίσια της εργασίας σας ζητείται να σχεδιάσετε και να υλοποιήσετε ένα απλοποιημένο **ηλεκτρονικό κατάστημα** (eshop). Το ηλεκτρονικό κατάστημα εμπορεύεται προϊόντα γραφείου και υποστηρίζει διαχείριση χρηστών, διαχείριση προϊόντων, καταχώρηση και τιμολόγηση παραγγελιών. Η υλοποίηση μπορεί να γίνει είτε σε Java είτε σε C++ και θα μπορούσε να περιλαμβάνει ενδεικτικά τις κλάσεις που περιγράφονται παρακάτω. Δεν είναι υποχρεωτικό να ακολουθήσετε απόλυτα την προτεινόμενη δομή στην υλοποίησή σας, αλλά θα πρέπει να παρέχετε τις λειτουργίες που περιγράφονται, καθώς και την ιεραρχία των κλάσεων. Θα πρέπει να διατηρήσετε την ονοματολογία που δίνεται στην εκφώνηση. Επίσης μπορείτε να υλοποιήσετε επιπλέον πεδία, μεθόδους ή/και κλάσεις που κρίνετε σκόπιμο για την επίτευξη της ζητούμενης λειτουργικότητας.

Κλάση User: Αναπαριστά έναν χρήστη και χαρακτηρίζεται από το όνομά του και το email του. Έχει ως υποκλάσεις τις Buyer και Owner. Δεν μπορεί να έχει στιγμιότυπα.

Κλάση Owner: Υποκλάση της User. Αναπαριστά ένα χρήστη που είναι ο ιδιοκτήτης του eshop. Διαθέτει ένα επιπλέον πεδίο boolean isAdmin true.

Κλάση Buyer: Υποκλάση της User. Διαθέτει πεδίο int bonus το οποίο αποθηκεύει τους πόντους που κερδίζει ο πελάτης από τις αγορές του και πεδίο buyerCategory: {BRONZE, SILVER, GOLD} που αντιπροσωπεύει ορισμένα προνόμια. Επίσης έχει ως πεδίο ένα αντικείμενο της κλάσης ShoppingCart που αναπαριστά το καλάθι αγορών του. Περιλαμβάνει τις παρακάτω μεθόδους με τα κατάλληλα ορίσματα:

- awardBonus(): Αρχικά οι πόντοι είναι μηδέν. Κάθε φορά που πληρώνεται μια παραγγελία αλλάζει το σύνολο των πόντων bonus (προστίθενται οι επιπλέον πόντοι), και τροποποιείται το buyerCategory, αν χρειάζεται.
- setbuyerCategory(): Η κατηγορία πελάτη εξαρτάται από το σύνολο των αγορών που έχει κάνει. Αρχικά είναι bronze. Μετά τους 100 πόντους γίνεται silver και μετά τους 200 γίνεται gold. Ο πελάτης που είναι στην κατηγορία gold δεν πληρώνει μεταφορικά. Ο πελάτης στην κατηγορία silver έχει 50% έκπτωση στα μεταφορικά.
- placeOrder(): Μέθοδος η οποία τοποθετεί ένα συγκεκριμένο προϊόν με συγκεκριμένη ποσότητα στο καλάθι αγορών του χρήστη. Η placeOrder δημιουργεί αντικείμενα τύπου ItemOrdered και τα προσθέτει στο καλάθι, καλώντας τις αντίστοιχες μεθόδους της ShoppingCart.
- Προαιρετικά, μπορούν να φτιαχτούν επιπλέον μέθοδοι που περικλείουν (wrap) και άλλες μεθόδους της ShoppingCart.

Κλάση Item: Αφορά ένα προϊόν και περιλαμβάνει πεδία name (string), price (double), description (string), stock (int), id (int) που αντιπροσωπεύουν το όνομα, την τιμή σε Ευρώ, την περιγραφή, τη διαθέσιμη ποσότητα και τον κωδικό του προϊόντος αντίστοιχα. Η κλάση Item δεν μπορεί να έχει στιγμιότυπα. Διαθέτει μέθοδο String getBasicInfo(), που επιστρέφει τις προηγούμενες πληροφορίες, *δήλωση* μεθόδου

String getDetails() και μέθοδο String toString() (@override) που θα καλεί τις άλλες δύο, ώστε να τυπώνει τις συνολικά διαθέσιμες πληροφορίες ενός προϊόντος κατάλληλα μορφοποιημένες.

Κλάση Pen: Κληρονομεί από την κλάση Item και έχει επιπλέον color (string), tipSize (μέγεθος μύτης σε mm, double). Υλοποιεί την getDetails().

Κλάση Pencil: Κληρονομεί από την κλάση Item και έχει επιπλέον tipSize (μέγεθος μύτης σε mm, double) και type: {H, B, HB}. Υλοποιεί την getDetails().

Κλάση Notebook: Κληρονομεί από την κλάση Item και έχει επιπλέον sections (αριθμός θεμάτων, int). Υλοποιεί την getDetails().

Κλάση Paper: Κληρονομεί από την κλάση Item και έχει επιπλέον weight (σε γραμμάρια, int) και pages (int)

Κλάση EShop: Αντιστοιχεί στο ηλεκτρονικό κατάστημα και περιλαμβάνει πεδίο name, πεδίο owner τύπου Owner, μια λίστα με τα διαθέσιμα προϊόντα itemsList και μια λίστα με τους πελάτες του καταστήματος buyersList. Επίσης περιλαμβάνει μεθόδους για τη διαχείριση και εμφάνιση των προϊόντων και χρηστών:

- addItem(): προσθέτει ένα προϊόν στο κατάστημα. Γίνεται χειρισμός εξαίρεσης αν υπάρχει ήδη.
- getItemById(): ανακτά ένα Item από την itemsList με βάση τον κωδικό του
- removeItem(): αφαιρεί ένα προϊόν
- addBuyer(): προσθέτει έναν υποψήφιο πελάτη στο κατάστημα. Γίνεται χειρισμός εξαίρεσης, αν υπάρχει ήδη.
- removeBuyer(): αφαιρεί έναν πελάτη από το κατάστημα
- updateItemStock(): τροποποιεί τη διαθέσιμη ποσότητα ενός προϊόντος
- showCategories(): Εμφανίζει τις υπάρχουσες κλάσεις των προϊόντων (pen, ...)
- showProductsInCategory(): Εμφανίζει λίστα με τα προϊόντα μιας συγκεκριμένης κατηγορίας
- showProduct(): Εμφανίζει τις συνολικές πληροφορίες για ένα προϊόν
- checkStatus(): Εμφανίζει τους πελάτες, τους πόντους και την κατηγορία τους.

Κλάση ItemOrdered: Αποτελείται από δύο πεδία, ένα Item item και ένα int quantity και συμβολίζει την παραγγελία για ένα συγκεκριμένο Item σε συγκεκριμένη ποσότητα. Προαιρετικά, μπορείτε να εξετάσετε τη χρήση Map στη ShoppingCart και να παραλείψετε την υλοποίηση αυτής της κλάσης.

Κλάση ShoppingCart: Αναπαριστά το καλάθι αγορών του χρήστη. Έχει μια λίστα με αντικείμενα ItemOrdered orderList και μεθόδους:

- addItemOrdered(): Δημιουργεί και προσθέτει ένα αντικείμενο ItemOrdered στη λίστα. Προηγείται έλεγχος αν υπάρχει διαθέσιμο stock. Αν υπάρχει, η αιτούμενη ποσότητα αφαιρείται από το stock του προϊόντος (το προϊόν δεσμεύεται). Αν δεν υπάρχει, εγείρεται εξαίρεση και δεν δημιουργείται αντικείμενο. Πριν από τη δημιουργία αντικειμένου, ελέγχεται αν υπάρχει άλλο αντικείμενο με το ίδιο Item στη λίστα. Αν υπάρχει, τότε απλώς

ανανεώνεται το quantity του υπάρχοντος. Καλείται από την placeOrder() της Buyer.

- removeItemOrdered(): Αφαιρεί ένα αντικείμενο ItemOrdered από μια θέση της λίστας. Ενημερώνει το stock.
- changeItemOrderedQuantity(): Για ένα αντικείμενο σε μια θέση της λίστας τροποποιεί το quantity και ενημερώνει το stock (updateItemStock()). Προηγείται έλεγχος αν υπάρχει διαθέσιμο stock.
- showCart(): Εμφανίζει τα περιεχόμενα του καλαθιού αγορών, δηλαδή τα προϊόντα, τις ποσότητες και τις τιμές τους, το σύνολο και τα μεταφορικά.
- clearCart(): Καλεί τη removeItemOrdered() και αφαιρεί όλα τα αντικείμενα από τη λίστα.
- checkout(): Καλεί την showCart(). Ζητά επιβεβαίωση για πληρωμή και εφόσον είναι θετική, καλεί την awardBonus() του Buyer και διαγράφει τα περιεχόμενα της λίστας (δεν αλλάζει όμως το stock). Οι πόντοι υπολογίζονται σε ποσοστό 10% επί της αξίας της παραγγελίας (χωρίς τα μεταφορικά) και είναι ακέραιος αριθμός.
- calculateNet(): Επιστρέφει την αξία της παραγγελίας.
- calculateCourierCost(): Υπολογίζει το κόστος μεταφορικών το οποίο είναι το 2% της αξίας της παραγγελίας, αλλά δεν μπορεί να είναι μικρότερο από 3 ευρώ. Ελέγχει σε ποια κατηγορία είναι ο πελάτης (Buyer). Αν είναι στην κατηγορία gold, δεν πληρώνει μεταφορικά, ενώ αν είναι στην κατηγορία silver έχει 50% έκπτωση στα μεταφορικά, ακόμα κι αν το ποσό που προκύπτει είναι μικρότερο από 3 ευρώ.

Κλάση Menu: αναπαριστά το μενού της εφαρμογής και περιλαμβάνει μεθόδους για την εκτύπωση των επιλογών, τον χειρισμό της εισόδου του χρήστη και την πλοήγηση στο μενού.

Κλάσεις Εξαιρέσεων: Θα πρέπει να υλοποιηθούν κατάλληλες κλάσεις εξαιρέσεων και να γίνεται χειρισμός τους σε περίπτωση μη επιτρεπτών ενεργειών, αντί να γίνονται εξαντλητικοί έλεγχοι if...else για την ορθή λειτουργία του προγράμματος. Για παράδειγμα, αν δοθεί ποσότητα που υπερβαίνει το stock ή αρνητική εγείρει μια εξαίρεση, αν δοθεί κωδικός προϊόντος που δεν υπάρχει εγείρει εξαίρεση, η προσθήκη νέου προϊόντος με τον ίδιο κωδικό, η προσθήκη ενός χρήστη που ήδη υπάρχει, η εισαγωγή μη έγκυρης επιλογής στο μενού κοκ.

Λειτουργικότητα Εφαρμογής

Διαμορφώστε κατάλληλο μενού σε command line όπου αρχικά θα ζητείται το email. Το σύστημα ταυτοποιεί τον χρήστη. Αν το mail που χρησιμοποιήθηκε δεν αντιστοιχεί στον Owner ή σε κάποιον χρήστη που βρίσκεται στη buyersList, ερωτάται να επιθυμεί να εγγραφεί και προστίθεται στην buyersList ως νέος Buyer. Στη συνέχεια του εμφανίζεται το εξής μενού επιλογών, ανάλογα αν είναι Buyer ή Owner:

Buyer: Εμφανίζει χαιρετισμό, τα στοιχεία του χρήστη, τους πόντους και την κατηγορία του. Στη συνέχεια:

1. Browse Store (Πλοήγηση στο κατάστημα): Χρησιμοποιούνται οι τρεις μέθοδοι show() της EShop. Εμφανίζεται το όνομα του καταστήματος και αριθμημένη λίστα με τις κατηγορίες των προϊόντων. Δίπλα σε κάθε κατηγορία αναγράφεται σε παρένθεση το πλήθος των Items της κατηγορίας.
 - a. Επιλέγοντας μια κατηγορία, εμφανίζεται αριθμημένη λίστα με προϊόντα της κατηγορίας αυτής.
 - i. Επιλέγοντας ένα προϊόν εμφανίζονται οι συνολικές πληροφορίες του και επιπλέον ερωτάται ο χρήστης αν θέλει να το προσθέσει στο καλάθι (γ/η). Αν ναι, συμπληρώνεται η αιτούμενη ποσότητα και καλείται η placeOrder().
2. View Cart (Προβολή καλαθιού αγορών): Καλούνται οι αντίστοιχες μέθοδοι της ShoppingCart. Αρχικά εμφανίζεται αριθμημένη λίστα με τα περιεχόμενα του καλαθιού αγορών (showCart()), διαφορετικά μήνυμα ότι το καλάθι είναι άδειο.
 - a. Επιλέγοντας μια γραμμή παραγγελίας (ItemOrdered) ο χρήστης έχει τη δυνατότητα:
 - i. Να διαγράψει την παραγγελία
 - ii. Να τροποποιήσει την παραγγελία αλλάζοντας την ποσότητά της.
 - b. Επιλογή αδειάσματος όλου του καλαθιού
 - c. Checkout (η επιλογή υπάρχει ξανά και στο κύριο μενού του χρήστη)
3. Checkout (ολοκλήρωση αγοράς)
4. Back. Επιστρέφει κάθε φορά ένα επίπεδο πιο πάνω στο μενού.
5. Αποσύνδεση χρήστη και ερώτημα για σύνδεση άλλου χρήστη.
6. Έξοδος από πρόγραμμα.

Owner: Εμφανίζει χαιρετισμό, τα στοιχεία του χρήστη και ένδειξη ότι είναι Owner. Στη συνέχεια:

1. Browse Store: Όπως πριν, αλλά στο (i) ερωτάται ο χρήστης αν θέλει να τροποποιήσει το επιλεγμένο προϊόν, όχι να το αγοράσει. Αν ναι, συμπληρώνεται η νέα ποσότητα και καλείται η updateItemStock().
2. Check Status: Καλείται η checkStatus() και εμφανίζεται αριθμημένη λίστα με τους πελάτες, τους πόντους και την κατηγορία τους.
 - a. Επιλέγοντας έναν πελάτη, εμφανίζονται τα περιεχόμενα του καλαθιού αγορών του (showCart()).
 - i. Δίνεται η δυνατότητα διαγραφής του επιλεγμένου πελάτη. Σε περίπτωση που διαγραφεί, το stock ενημερώνεται κατάλληλα.
3. Back. Επιστρέφει κάθε φορά ένα επίπεδο πιο πάνω στο μενού.
4. Αποσύνδεση χρήστη και ερώτημα για σύνδεση άλλου χρήστη.
5. Έξοδος από πρόγραμμα.

Κλάση Main

Στη main αρχικά θα δημιουργήσετε ένα αντικείμενο EShop και θα εισάγετε 3 είδη από την κάθε κλάση προϊόντων, έτσι ώστε να είναι αντιπροσωπευτικά, έναν Owner, δύο Buyers και μια παραγγελία για τον καθένα, με αρκετά είδη στο ShoppingCart.

Πέραν των παραπάνω παραδειγμάτων θα πρέπει για την επίδειξη της λειτουργικότητας του κώδικα να έχετε προετοιμάσει και δοκιμάσει ικανό πλήθος δεδομένων, ώστε να είναι διαθέσιμα σε κάθε εκτέλεση της εφαρμογής. Θα πρέπει τα δεδομένα που εισάγετε να δίνουν τη δυνατότητα ελέγχου πολλών σεναρίων (παραγγελία σε εκκρεμότητα, διαγραφή παραγγελίας, διαγραφή πελάτη, αλλαγή buyerCategory για τον πελάτη μετά από παραγγελία (αναβάθμιση) κοκ).

❖ **Bonus +20%:** Υλοποιήστε το πρόγραμμα και σε 2^η γλώσσα (Java ή C++).

❖ **ΔΙΕΥΚΡΙΝΗΣΗ:** Όσοι χρωστούν μόνο το project C++, είτε διότι χρωστούν το παλιό μάθημα Οντοκεντρικός Προγραμματισμός II είτε διότι έδωσαν το project Java άλλη χρονιά, θα πρέπει να υλοποιήσουν την εργασία αποκλειστικά σε C++.

ΑΞΙΟΛΟΓΗΣΗ

Η βαθμολόγηση πέρα από την σωστή εκτέλεση του προγράμματος όπως περιγράφεται παραπάνω θα βασιστεί στην σωστή εφαρμογή των εννοιών οντοκεντρικού προγραμματισμού που διδάσκονται στο μάθημα:

1. Σωστή δήλωση των κλάσεων και ιεράρχησή τους όπως ζητείται.
2. Τήρηση της αρχής της απόκρυψης και της ενθυλάκωσης: Αποφυγή χρήσεως public όπου δεν είναι απαραίτητο και χρήση getters και setters για πρόσβαση στα πεδία των κλάσεων.
3. Υπερκάλυψη μεθόδων και πολυμορφική χρήση τους όπου χρειάζεται.
4. Χρήση Static μεταβλητών/μεθόδων για γενική πληροφορία που αφορά κάθε κλάση.
5. Έμφαση στην αντικειμενοστρέφεια και την δυνατότητα επαναχρησιμοποίησης κώδικα.
6. Λίστες / Πίνακες
7. Ποιότητα κώδικα, αναγνωσιμότητα και επαρκής σχολιασμός
8. Η έξοδος του προγράμματος να είναι καλά μορφοποιημένη και εύκολα ερμηνεύσιμη από τον χρήστη

ΠΑΡΑΔΟΣΗ

- Οι εργασίες θα πρέπει να παραδοθούν στο e-Class μέχρι και την Κυριακή 7/6/2020 (23:59).
- Οι εργασίες είναι ομαδικές (τουλάχιστον 2 και το πολύ 3 άτομα).
- Παραδίδετε τα αρχεία (σε. zip ή .rar) 1 φορά ανά ομάδα, φροντίστε όμως στο όνομα του αρχείου να περιλαμβάνονται όλα τα ΑΜ των φοιτητών της ομάδας:
2859_2973_2924_PROJECT_Java.rar // αν η υλοποίηση είναι σε JAVA
2859_2973_2924_PROJECT_CPP.rar // αν η υλοποίηση είναι σε C++
2859_2973_2924_PROJECT.rar // Ενιαίο αρχείο που περιλαμβάνει τα δύο παραπάνω

Σε κάθε εργασία πρέπει να συμπεριλάβετε **αναφορά** όπου να αναφέρετε:

- Τα στοιχεία (ΟΝΟΜΑΤΑ - ΑΜ - email) των μελών της ομάδας
- Διάγραμμα κλάσεων (UML).
- Σύντομη περιγραφή της υλοποίησης σας με τεκμηρίωση των σχεδιαστικών αποφάσεων και της υλοποίησης για κάθε κλάση, ιδίως στην περίπτωση που χρειάστηκε να αποκλίνετε από την εκφώνηση.

ΔΙΑΔΙΚΑΣΙΑ ΒΑΘΜΟΛΟΓΗΣΗΣ

Η διαδικασία βαθμολόγησης θα ανακοινωθεί.

Καλή Επιτυχία