

Ενότητα 4. – Σύγχρονα Προγραμματιστικά Περιβάλλοντα (Αντικειμενοστραφής προγραμματισμός)

1. Ποια τα πλεονεκτήματα του Αντικειμενοστραφούς προγραμματισμού;

Στον Διαδικασιακό προγραμματισμό το πρόγραμμα αντιμετωπίζεται ως ένα σύνολο εντολών που εκτελούνται με καθορισμένη σειρά.

Το πρόγραμμα μας σπάει σε πολλαπλές **διαδικασίες**.

- Κάθε διαδικασία λύνει ένα υπο-πρόβλημα και αποτελεί μια λογική μονάδα (**module**).
- Μια διαδικασία μπορούμε να την επαναχρησιμοποιήσουμε σε διαφορετικά δεδομένα.

Το πρόγραμμα μας είναι

τμηματοποιημένο (modular).

Ο διαδικαστικός προγραμματισμός δουλεύει καλά για μικρά προγράμματα. Σε μεγάλα συστήματα είναι δύσκολο να **σχεδιάσουμε**, να **υλοποιήσουμε** και να **συντηρήσουμε** τον κώδικα γιατί δεν είναι εύκολο να προσαρμοστούμε σε αλλαγές, και δεν μπορούμε να προβλέψουμε τις ανάγκες που θα έχουμε.

Η εξέλιξη του προστακτικού προγραμματισμού (imperative programming)



Ο Αντικειμενοστραφής προγραμματισμός έχει τα εξής πλεονεκτήματα

- μοντελοποιεί τον πραγματικό κόσμο, ο κώδικας είναι πιο κατανοητός.
- ο κώδικας είναι πιο εύκολο να λόγω της **ενθυλάκωσης**
- τα δομικά κομμάτια που δημιουργεί (κλάσεις - αντικείμενα) είναι πιο εύκολο να επαναχρησιμοποιηθούν και να συνδυαστούν λόγω της **κληρονομικότητας** και του **πολυμορφισμού**.

2. Τι ονομάζεται Αντικειμενοστραφής προγραμματισμός και τι τα Αντικείμενα;

Αντικειμενοστραφής προγραμματισμός (object-oriented programming) ή αντικειμενοστραφής σχεδίαση είναι μια **μεθοδολογία ανάπτυξης εφαρμογών** η οποία **στηρίζεται** σε αυτόνομες προγραμματιστικές **οντότητες** με δική τους ταυτότητα και συμπεριφορά.

Οι οντότητες αυτές καλούνται **αντικείμενα** (objects), αντιστοιχούν σε φυσικές οντότητες ή έννοιες του φυσικού μας κόσμου, και δομούνται με βάση **δεδομένα** (ιδιότητες) που προσδιορίζουν την υπόστασή τους και **ενέργειες** (κανόνες συμπεριφοράς) που εφαρμόζονται πάνω στα δεδομένα.

Σε μια εφαρμογή, ένα αντικείμενο είναι ο ομαδοποιημένος **συνδυασμός δεδομένων και κώδικα**, τα οποία έχουμε τη δυνατότητα να **χρησιμοποιούμε ενιαία**.

3. Τι ονομάζουμε ιδιότητες και τι μεθόδους στα αντικείμενα;

Τα **δεδομένα** αποτελούν τα χαρακτηριστικά ενός αντικείμενου και αναφέρονται ως **ιδιότητες (properties)** ενώ οι ενέργειες καθορίζουν τη συμπεριφορά του. Οι **ενέργειες** στον αντικειμενοστραφή προγραμματισμό αναφέρονται και ως **μέθοδοι (methods)**.

4. Ποια είναι η βασική αντίληψη στην αντικειμενοστραφή προσέγγιση για την επίλυση προβλημάτων;

Η **επίλυση** ενός προβλήματος επιτυγχάνεται με τη **σύνθεση ικανοτήτων** (ο τρόπος υλοποίησης των οποίων μας είναι άγνωστος) που **διαθέτουν διαφορετικές ανεξάρτητες οντότητες**, οι οποίες **αλληλεπιδρούν** για τον σκοπό αυτό. Οι θεμελιώδεις **αρχές του αντικειμενοστραφούς**

προγραμματισμού **πηγάζουν** από τον καθημερινό μας **φυσικό κόσμο**, καθώς είναι πολύ **κοντά στον τρόπο που σκεφτόμαστε** για να επιλύσουμε προβλήματα της καθημερινότητάς μας.

5. Ποια είναι τα βήματα επίλυσης ενός προβλήματος με αντικειμενοστραφή προσέγγιση;

Για να αναλύσουμε το πρόβλημα το οποίο θέλουμε να επιλύσουμε με αντικειμενοστραφή προσέγγιση πρέπει να αναγνωρίσουμε

- τα **αντικείμενα (οντότητες)** που συμμετέχουν με βάση τον **ρόλο** τους στο συγκεκριμένο σενάριο,
- οι **ιδιότητες** κάθε αντικειμένου, δηλ. τα σχετικά με το συγκεκριμένο πρόβλημα χαρακτηριστικά του, και
- οι υπηρεσίες που προσφέρει ή οι **ενέργειες** που υλοποιεί κάθε αντικείμενο (**μέθοδοι**) προς αξιοποίηση από άλλες, ώστε να αναπτυχθούν οι απαραίτητες συνεργασίες μεταξύ των αντικειμένων για την επίλυση του προβλήματος.



Και εν συνεχεία τον τρόπο με τον οποίο συνδέονται οι οντότητες μεταξύ τους. Η συνεργασία επιτυγχάνεται μέσω κλήσης των μεθόδων τους από τα άλλα αντικείμενα.

6. Ποια η δομή ενός αντικειμενοστραφούς προγράμματος;

Ένα **αντικειμενοστραφές πρόγραμμα** δομείται ως ένα **δίκτυο** συνεργαζόμενων οντοτήτων που είναι τα **αντικείμενα**.

Κάθε **αντικείμενο** έχει ένα συγκεκριμένο ρόλο στην εφαρμογή και **παρέχει μια υπηρεσία** ή εκτελεί μια ενέργεια (**μέθοδο**) που **χρησιμοποιείται από άλλα μέλη** του δικτύου, δηλαδή από άλλα αντικείμενα, για την υλοποίηση της συνεργασίας που θα επιλύσει το πρόβλημα.

7. Τι περιλαμβάνει η διαγραμματική αναπαράσταση των αντικειμένων;

Αφού **εντοπίσουμε τα συστατικά επίλυσης** του προβλήματος, μπορούμε να τα οργανώσουμε σε μια απλή **διαγραμματική**

αναπαράσταση χρησιμοποιώντας **παράλληλόγραμμα** για την αποτύπωση

- των **αντικειμένων**,
- των **ιδιοτήτων** και
- των **μεθόδων** τους

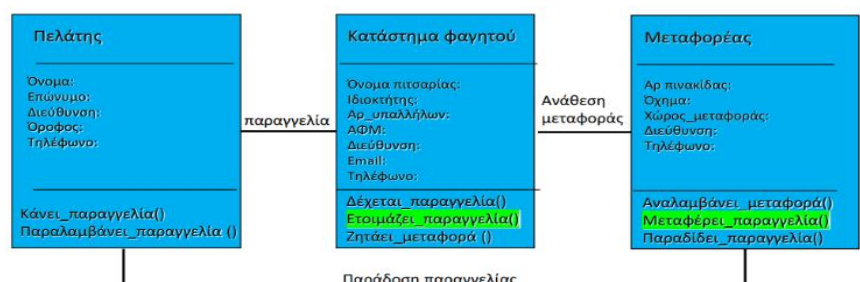
και **γραμμές σύνδεσης** για την περιγραφή του είδους της μεταξύ τους συνεργασίας.



8. Τι περιλαμβάνει η διαγραμματική αναπαράσταση των κλάσεων της εφαρμογής και των μεταξύ τους συνεργασιών;

Για την υλοποίηση της διαγραμματικής αναπαράστασης απαιτείται η **πλήρης προδιαγραφή** καθεμιάς από τις συμμετέχουσες κλάσεις, δηλαδή ο εντοπισμός

- των **ιδιοτήτων**,
- των **μεθόδων** και
- του **είδους της μεταξύ τους συνεργασίας**.



9. Τι ονομάζεται ενθυλάκωση;

Σε μια αντικειμενοστραφή εφαρμογή κάθε **αντικείμενο** αποτελεί **ξεχωριστή οντότητα** και περιέχει ενσωματωμένες τις ιδιότητες (δεδομένα) και τους κανόνες συμπεριφοράς του (μεθόδους). Η **δυνατότητα ενός αντικειμένου να συνδυάζει εσωτερικά τα δεδομένα και τις μεθόδους χειρισμού του** καλείται **ενθυλάκωση** (encapsulation).

10. Τι ονομάζεται κλάση;

Ο **γενικός τύπος** ενός αντικειμένου καλείται **κλάση** (class) και **καθορίζει τις αρχικές ιδιότητες και τη συμπεριφορά κάθε αντικειμένου** που προέρχεται από αυτή.

Μια κλάση αποτελεί ένα **αφαιρετικό (abstract) στοιχείο** (τύπο) και μπορεί να **παράγει ένα απεριόριστο πλήθος δομικά ίδιων αντικειμένων**.

11. Με ποια μεθοδολογία τοποθετούνται στις κλάσεις οι μέθοδοι που υλοποιούν τις σχέσεις μεταξύ τους;

Όταν μια λειτουργία απαιτεί **ενέργειες και από τις δύο κλάσεις** που συμμετέχουν σε μια σχέση, τότε πρέπει να **υπάρχουν αντίστοιχες μέθοδοι και στις δύο κλάσεις**. Όταν δεν απαιτείται ενέργεια και από τις δύο κλάσεις που συμμετέχουν σε μια σχέση, τότε πρέπει να αποφασίσουμε σε ποια κλάση θα οριστεί η μέθοδος.

Σε αυτές τις περιπτώσεις η μέθοδος **ορίζεται στην κλάση που ολοκληρώνει την ενέργεια**. Αυτό συμβαίνει επειδή η ολοκλήρωση της ενέργειας θα έχει ως αποτέλεσμα να μεταβληθούν οι τιμές κάποιων ιδιοτήτων, συνεπώς **η μέθοδος και οι ιδιότητες πρέπει να βρίσκονται στην ίδια κλάση**. Με αυτόν τον τρόπο, κάθε κλάση επιτρέπει την **πρόσβαση στις ιδιότητές της μέσω των μεθόδων που διαθέτει**.

12. Ποιες ιδιότητες πρέπει να υπάρχουν σε κάθε κλάση;

Σε κάθε περίπτωση, όμως, **χρειάζονται ιδιότητες** που αφενός να **ορίζουν μοναδικά τα αντικείμενα** κάθε κλάσης και αφετέρου να **χρησιμοποιούνται από τις μεθόδους για την υλοποίηση των σχέσεων μεταξύ των κλάσεων**.

13. Τι ονομάζεται κληρονομικότητα και ποια τα χαρακτηριστικά από τις κλάσεις πρόγονοι-απόγονοι;

Η **δυνατότητα δημιουργίας ιεραρχιών αντικειμένων** καλείται **κληρονομικότητα** (inheritance). Με βάση την κληρονομικότητα, **μία κλάση μπορεί να περιγραφεί γενικά και στη συνέχεια μέσω αυτής της κλάσης να οριστούν υποκλάσεις** αντικειμένων.

Η κλάση **απόγονος (υποκλάση)** **κληρονομεί** και μπορεί να χρησιμοποιήσει όλα τα δεδομένα (**ιδιότητες**) και τις μεθόδους **που περιέχει η κλάση πρόγονος (υπερκλάση)**.

14. Πώς αποτυπώνεται η σχέση κληρονομικότητας στη διαγραμματική αναπαράσταση;

Η διαγραμματική αναπαράσταση της σχέσης κληρονομικότητας που μόλις περιγράψαμε γίνεται με τη βοήθεια του ειδικού συμβόλου γενίκευσης.

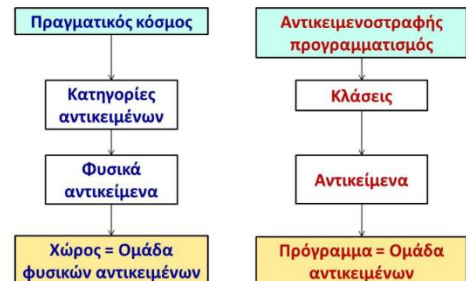
15. Πότε μπορεί να θεωρηθεί ως έγκυρη μια υποκλάση;

Μια κλάση A μπορεί να είναι έγκυρη υποκλάση της B αν έχει νόημα να πούμε «**ένα A είναι ένα (is_a) B**» (προσοχή το αντικείμενο της κλάσης δεν αποτελεί υποκλάση!).

16. Τι ονομάζεται πολυμορφισμός;

Πολυμορφισμός (polymorphism) είναι μια **ιδιότητα** του αντικειμενοστραφούς προγραμματισμού με την οποία **μία λειτουργία μπορεί να υλοποιείται με πολλούς διαφορετικούς τρόπους**.

Με άλλα λόγια πολυμορφισμός σημαίνει **πολλές διαφορετικές μορφές** ή **πολλές διαφορετικές συνθήκες**. Ο πολυμορφισμός μας επιτρέπει να **επαναπροσδιορίσουμε** τον τρόπο με τον οποίο

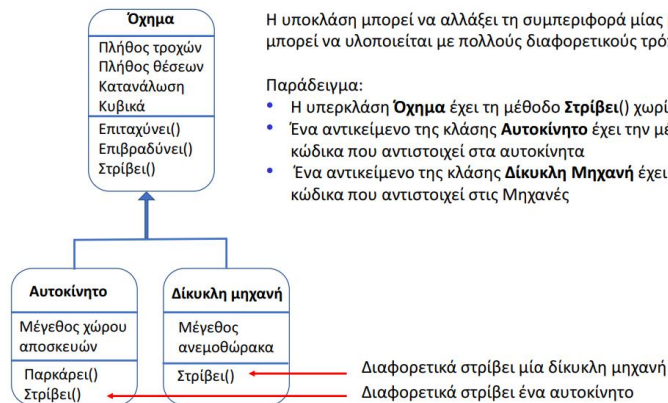


λειτουργούν κάποια πράγματα, είτε **αλλάζοντας τον τρόπο λειτουργίας** τους είτε **αλλάζοντας τα εργαλεία** τα οποία χρησιμοποιούνται για την επίτευξη του στόχου.

17. Τι προσφέρει ο **πολυμορφισμός** στις μεθόδους των αντικειμένων στον αντικειμενοστραφή προγραμματισμό;

Τα αντικείμενα μπορούν δηλαδή να υποστηρίζουν συμπεριφορές (μεθόδους) με κοινό όνομα και τον ίδιο βασικό σκοπό αλλά με διαφορετική λειτουργική υλοποίηση.

Κάθε φορά που καλείται μια πολυμορφική λειτουργία, το πρόγραμμα αποφασίζει ποια από τις διαφορετικές μεθόδους με την ίδια ονομασία θα ενεργοποιηθεί, με βάση την κλάση του αντικειμένου στην οποία απευθύνεται η εφαρμογή της λειτουργίας.



§ 6.5

18. Ποια είναι **πρωτεύοντα** δομικά στοιχεία ενός προγράμματος στην **αντικειμενοστραφή σχεδίαση**;

Η αντικειμενοστραφής σχεδίαση εκλαμβάνει ως πρωτεύοντα δομικά στοιχεία ενός προγράμματος τα **δεδομένα**, από τα οποία δημιουργούνται με κατάλληλη μορφοποίηση τα **αντικείμενα** (objects).

19. Ποια τα **πλεονεκτήματα** της αντικειμενοστραφούς σχεδίασης;

Η αντικειμενοστραφής σχεδίαση αποδείχθηκε ότι επιφέρει καλύτερα αποτελέσματα, αφού τα προγράμματα που δημιουργούνται είναι περισσότερο ευέλικτα και επαναχρησιμοποιήσιμα.

20. Ο **αντικειμενοστραφής προγραμματισμός** χρησιμοποιεί **τεχνικές του κλασσικού διαδικασιακού προγραμματισμού**;

Ο αντικειμενοστραφής προγραμματισμός εκτός από τον τρόπο που χειρίζεται τα δεδομένα, συνεχίζει να χρησιμοποιεί την **ιεραρχική** σχεδίαση, τον **τμηματικό** προγραμματισμό και να ακολουθεί τις αρχές του δομημένου προγραμματισμού.

Δημιουργία διαγραμμάτων αντικειμένων/κλάσεων από σενάρια

Ο Μανόλης πήγε στο κατάστημα ρούχων fashion. Δοκίμασε μία άσπρη μπλούζα και ένα μπλε παντελόνι τα οποία και αγόρασε από το μαγαζί.

