

Σχεδιασμός Λογισμικού


Τεχνολογία Λογισμικού

Βασικές Έννοιες
Αντικειμενοστρεφούς Τεχνολογίας

- Ένα αντικείμενο μοντελοποιείται με **ιδιότητες** (attributes) και **μεθόδους / λειτουργίες** (methods / operations), που **ανήκουν στο αντικείμενο**.
- Για παράδειγμα, ένα αντικείμενο που παριστάνει έναν πελάτη μπορεί να μοντελοποιηθεί ως εξής:

Όνομα αντικειμένου 

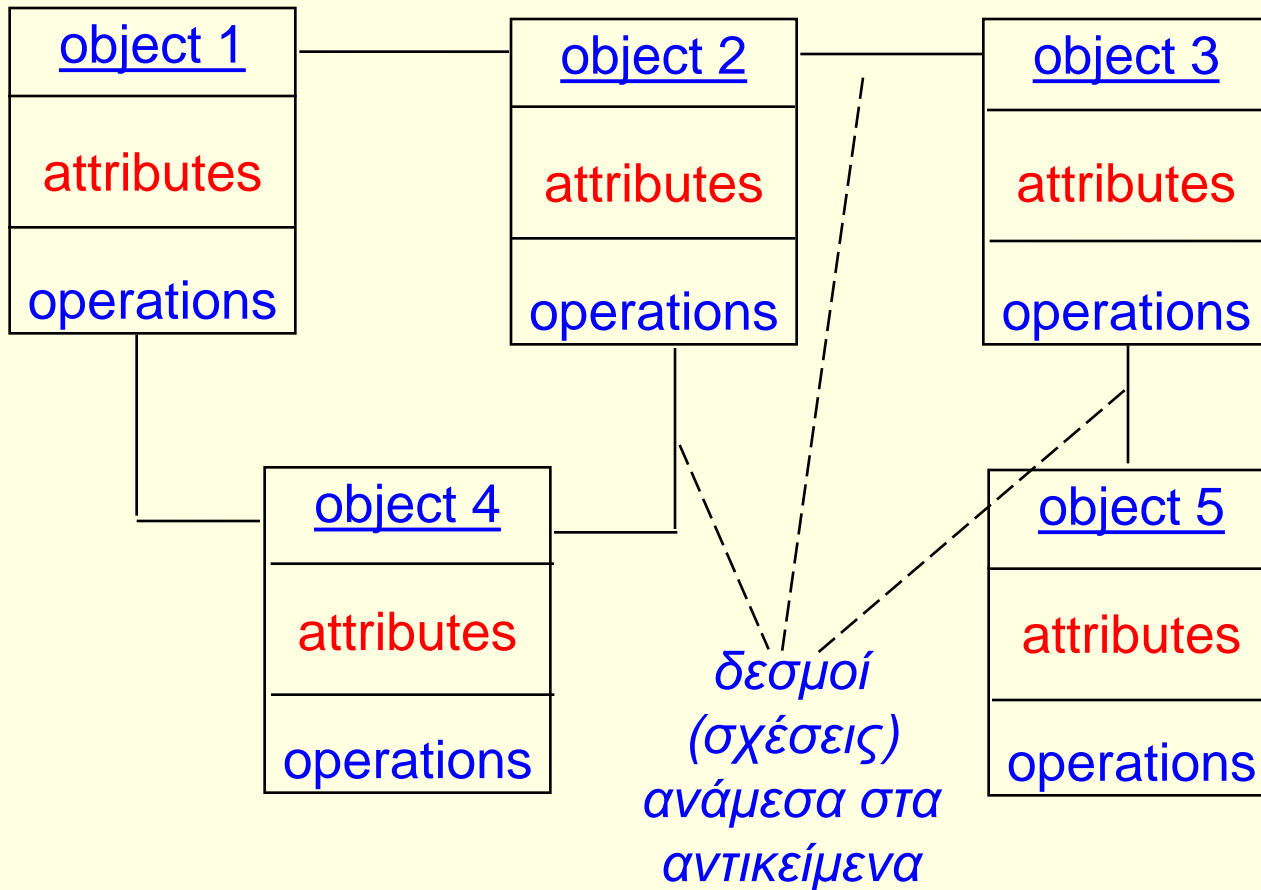
Ιδιότητες 

Μέθοδοι 

<u>customer1</u>
ID number = 4357 name = "Tony Blair" address = "10 Downing St" credit limit = 3000.00
change address increase credit limit

Αποσύνθεση σε Αντικείμενα

📄 Οδηγεί σε σχέδια λογισμικού, όπως το ακόλουθο:



Παραδείγματα Αντικειμένων

📄 Ημερομηνίες (“*date objects*”):

➤ *Ιδιότητες*: day, month, year

➤ *Μέθοδοι*:

- add_day()
- add_days(no_of_days)
- get_month()
- set_month(new_month)

📄 Άλλα παραδείγματα:

- Άνθρωποι / άτομα (“person objects”).
- Δελτία / καρτέλες (“card objects”).

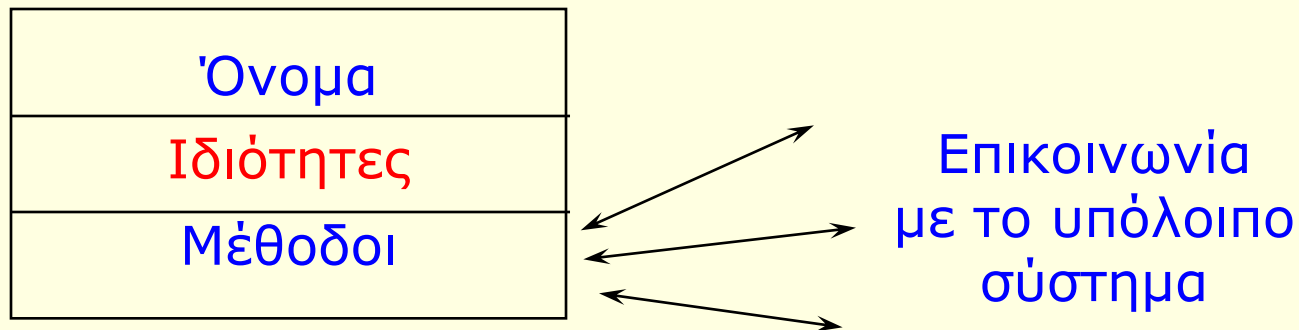
Κλάση ή Τάξη (Class)

- Είναι ένα **περίγραμμα** / μια **φόρμα** (template) για τον προσδιορισμό των ιδιοτήτων και των μεθόδων ενός συνόλου από αντικείμενα.
- Μια κλάση προσδιορίζει:
 - Το όνομα της κλάσης.
 - Το όνομα (και τον τύπο) κάθε ιδιότητας.
 - Το όνομα (και τις παραμέτρους) κάθε μεθόδου.

Customer
ID number : integer name : string address : string credit limit : integer
change address (new_addrs : string) increase credit limit (new_lim : integer)

Βασικά Χαρακτηριστικά Μιας Κλάσης

- ❏ Προσδιορίζει τη **γενική μορφή ενός συνόλου αντικειμένων** – Διαφορετικά στιγμιότυπα (instances) της κλάσης είναι δυνατό να κατασκευασθούν από το γενικό περίγραμμα.
- ❏ Οι μέθοδοι χαρακτηρίζουν την κλάση και αποτελούν τη **διεπαφή** (interface) ανάμεσα στην κλάση και στο υπόλοιπο σύστημα.



Βασικά Χαρακτηριστικά Ενός Αντικειμένου

- ❏ Ένα αντικείμενο είναι ένα συγκεκριμένο **στιγμιότυπο** (instance) μιας κλάσης.
- ❏ Έχει μια συγκεκριμένη **κατάσταση** (state):
 - Αναπαριστάνεται από τις τιμές των ιδιοτήτων του.
- ❏ Κατέχει χώρο στην κύρια μνήμη και επομένως έχει μια μοναδική **ταυτότητα** (identity):
 - 2 αντικείμενα είναι ξεχωριστά έστω και αν έχουν το ίδιο όνομα.
- ❏ Έχει μια **συμπεριφορά** (behaviour) που προσδιορίζεται από τις μεθόδους της κλάσης στην οποία ανήκει.

Άρα:

- ❏ Βασικά χαρακτηριστικά ενός αντικειμένου =
Κατάσταση + Ταυτότητα + Συμπεριφορά

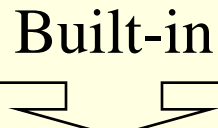
Προσδιορισμός Ιδιοτήτων

Κάθε ιδιότητα έχει:

➤ Ένα **όνομα**.

➤ Έναν **τύπο**.

Built-in



Ο τύπος μπορεί να είναι:

➤ Απλός (π.χ. integer, char, real).

➤ Μια κάπως πολύπλοκη δομή δεδομένων (π.χ. πίνακας – array).

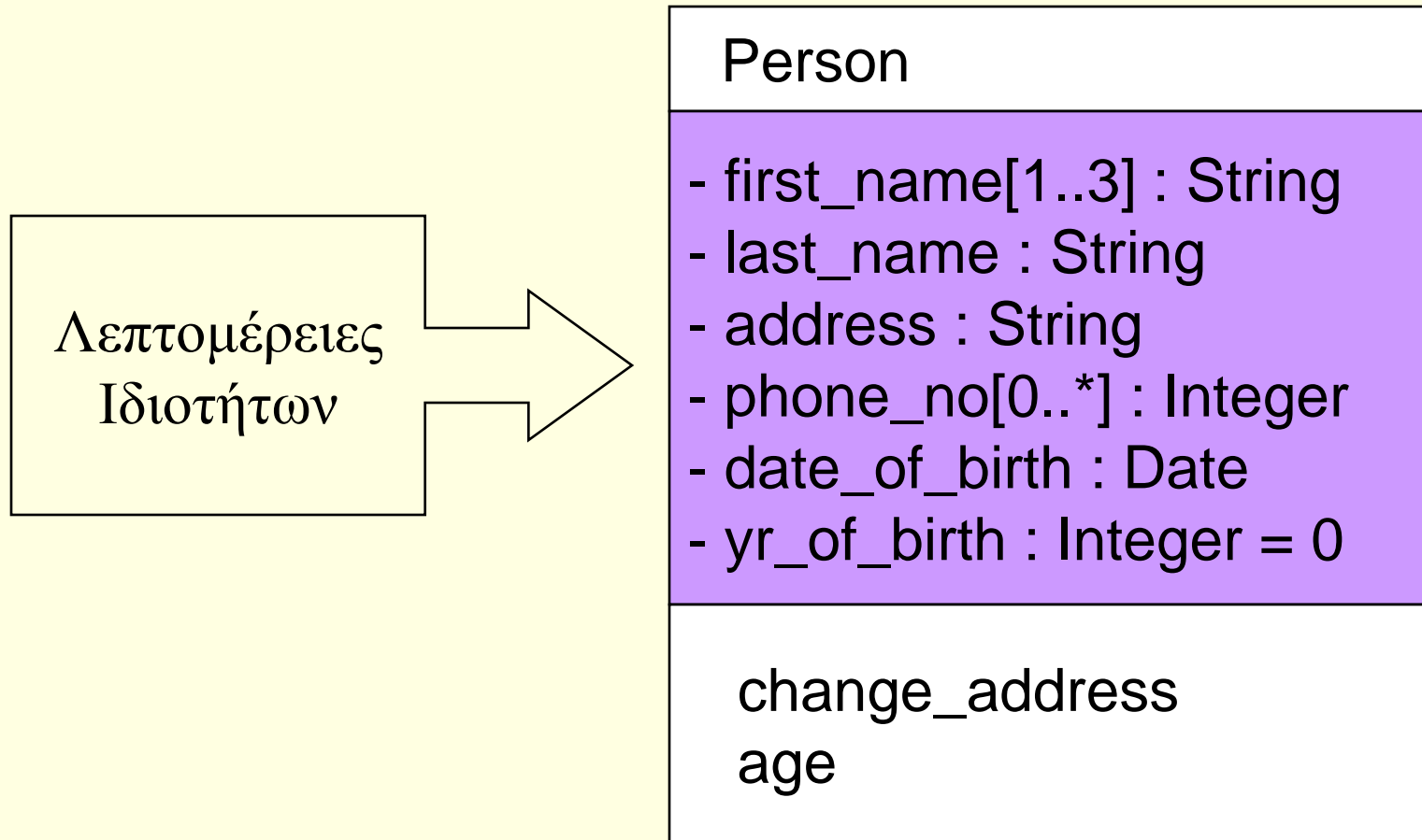
➤ Μια άλλη κλάση.

Στα αρχικά στάδια της μοντελοποίησης ...

➤ Αρκεί ο προσδιορισμός μόνο των βασικών ιδιοτήτων με το όνομά τους.

➤ Περισσότερες λεπτομέρειες μπορούν να προστεθούν σταδιακά κατά τη βελτίωση / **εκλέπτυνση** της σχεδίασης.

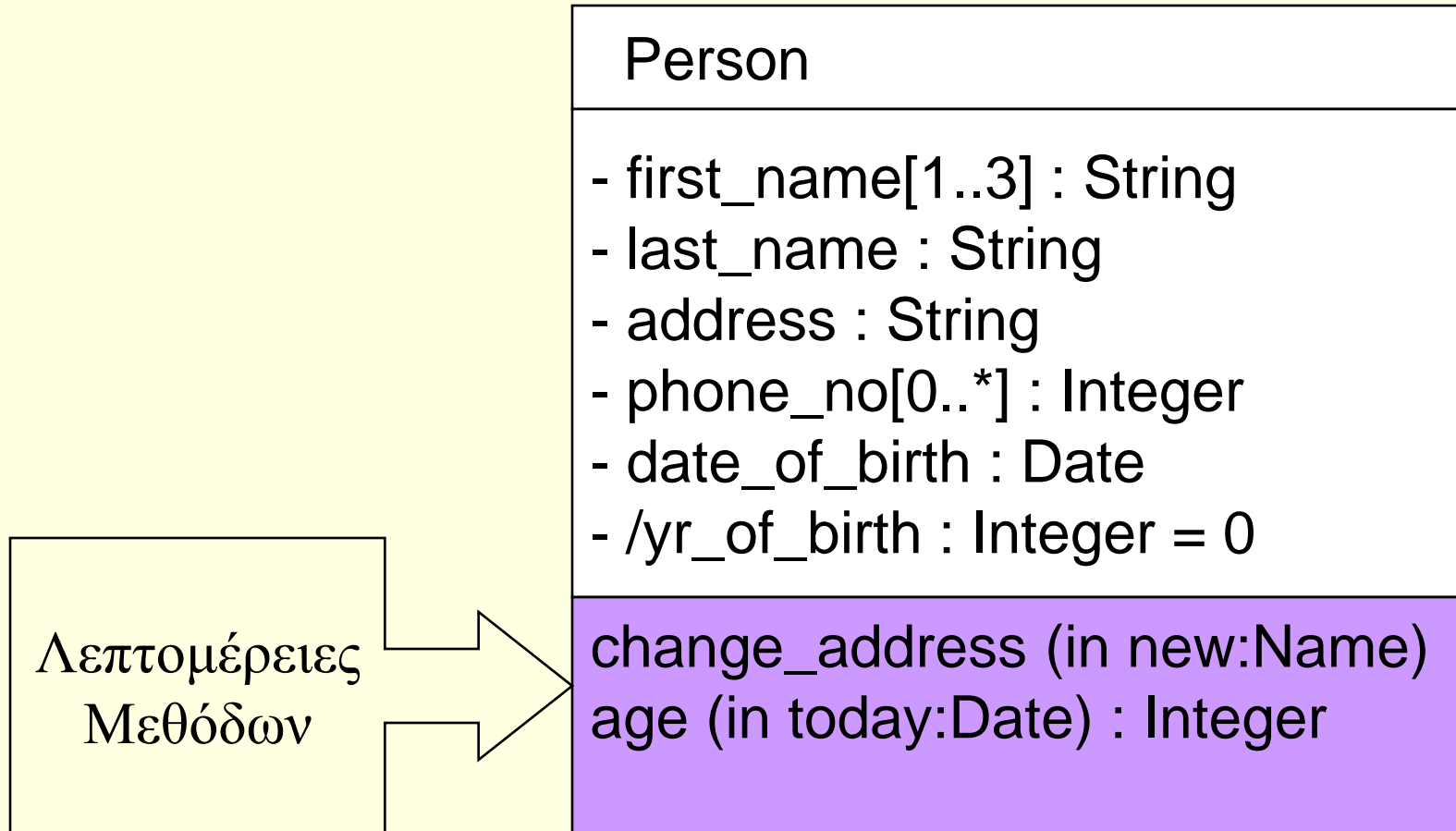
Λεπτομερής Σύνταξη Ιδιοτήτων σε Διαγράμματα Κλάσης



Προσδιορισμός Μεθόδων

- ❏ Μια μέθοδος προσδιορίζεται με τον καθορισμό της **υπογραφής** της (signature) ή της επικεφαλίδας της (header), η οποία αποτελείται από:
 - Το όνομα της μεθόδου.
 - Μια λίστα από παραμέτρους (που ίσως είναι κενή).
 - Έναν τύπο επιστροφής (return type), αν απαιτείται.
- ❏ Η λίστα των παραμέτρων αποτελείται από τα ονόματα των παραμέτρων και τους τύπους τους.
- ❏ Στα αρχικά στάδια της μοντελοποίησης αρκεί ο προσδιορισμός μόνο των βασικών μεθόδων με το όνομά τους – περισσότερες λεπτομέρειες μπορούν να προστεθούν σταδιακά κατά τη βελτίωση / εκλέπτυνση της σχεδίασης.
- ❏ Ο κώδικας των μεθόδων δε χρειάζεται κατά τη μοντελοποίηση.

Λεπτομερής Σύνταξη Μεθόδων



Ενθυλάκωση (1)

- Η κλάση υποκρύπτει λεπτομέρειες υλοποίησης.
- Η ιδιότητα αυτή ονομάζεται **ενθυλάκωση** (encapsulation).
- Είναι χωρίς αμφιβολία το πιο ισχυρό χαρακτηριστικό της αντικειμενοστρεφούς προσέγγισης.
- Ποιές είναι οι λεπτομέρειες οι οποίες υποκρύπτονται;
 - Οι ιδιότητες.
 - Ο κώδικας που υλοποιεί κάθε μέθοδο.

Ενθυλάκωση (2)

Account
<i>balance : integer</i>
deposit(amount : integer) withdraw(amount : integer)

Κόκκινο (italic) = κρυφό
Μπλέ = ορατό

void withdraw (int amount)
{
 if (amount<0)
 cout<<"error";
 else
 if (amount>balance)
 cout<<"error";
 else
 balance -= amount;
}

void deposit (int amount)
{
 if (amount<0)
 cout<<"error";
 else
 balance += amount;
}

Ορατότητα (Visibility)

- ❏ Η ενθυλάκωση μπορεί να εξεταστεί με βάση την **ορατότητα** που επιτρέπει στον εξωτερικό (της κλάσης) κόσμο.
- ❏ Η ορατότητα ελέγχεται χρησιμοποιώντας τις λέξεις κλειδιά:
 - **private** (ιδιωτικό).
 - **public** (δημόσιο).
- ❏ Οτιδήποτε δηλωθεί ως ιδιωτικό είναι ορατό μόνο στο εσωτερικό της κλάσης και όχι έξω από την κλάση.
- ❏ Οι ιδιότητες πρέπει να δηλώνονται πάντα ως **ιδιωτικές**.

Μέθοδοι Αναζήτησης και Ενημέρωσης

- ❏ **Οι ιδιότητες πρέπει να είναι ιδιωτικές.**
- ❏ Επομένως, για να προσδιοριστεί η τιμή τους, αλλά και για να αλλάξει από τον εξωτερικό (της κλάσης) κόσμο, πρέπει να παρέχονται ειδικές για το σκοπό αυτό μέθοδοι.
- ❏ Οι μέθοδοι αυτοί ονομάζονται:
 - Μέθοδοι **αναζήτησης** και **ενημέρωσης** (**query** and **update** methods).
 - ή
 - Μέθοδοι **απόκτησης** και **τοποθέτησης** (**get** and **set** methods).

Πολυμορφισμός

☞ Στα αντικειμενοστρεφή συστήματα επιτρέπεται σε **διαφορετικές μεθόδους να έχουν το ίδιο όνομα**, αν:

- Ανήκουν σε διαφορετικές κλάσεις, πιθανότατα στα πλαίσια μιας ιεραρχίας κληρονομικότητας (inheritance), π.χ. κάθε μία από διαφορετικές κλάσεις μπορεί να έχει μια μέθοδο για εμφάνιση (display) πληροφορίας.
- Ανήκουν στην ίδια κλάση, αλλά η λίστα των παραμέτρων τους είναι διαφορετική, π.χ. διαφορετικές μορφές (για την ίδια κλάση) μιας μεθόδου αρχικοποίησης που ονομάζεται “init”.

☞ Τα πιο πάνω είναι παραδείγματα **πολυμορφισμού** (polymorphism) – Μέθοδοι που έχουν υλοποιηθεί διαφορετικά έχουν το ίδιο όνομα.

Διάγραμμα Κλάσης

- ❏ Το διάγραμμα κλάσης (class diagram) είναι ένα διάγραμμα που χρησιμοποιείται για την απεικόνιση κλάσεων και των μεταξύ τους σχέσεων.
- ❏ Αποτελεί ένα από τα κυριότερα παραδοτέα της αντικειμενοστρεφούς ανάλυσης και σχεδίασης (OOA&D).
- ❏ Κατά τη μετάβαση από τη φάση ανάλυσης στη φάση σχεδίασης γίνεται προσθήκη περισσότερων λεπτομερειών στο διάγραμμα κλάσης.
- ❏ Αρχικά (κατά τη φάση της ανάλυσης) γίνεται **μοντελοποίηση του προβλήματος και όχι της λύσης**.