ΕΚΠΑΙΔΕΥΤΙΚΟ ΚΕΙΜΕΝΟ
ΑΚΑΔΗΜΙΩΝ ΕΜΠΟΡΙΚΟΥ ΝΑΥΤΙΚΟΥ

ΙΔΡΥΜΑ ΕΥΓΕΝΙΔΟΥ 1954

ΧΡΥΣΟΥΝ ΜΕΤΑΛΛΙΟΝ
ΑΚΑΔΗΜΙΑΣ ΑΘΗΝΩΝ

ΠΛΗΡΟΦΟΡΙΚΗ
ΗΛΕΚΤΡΟΝΙΚΟΙ ΥΠΟΛΟΓΙΣΤΕΣ

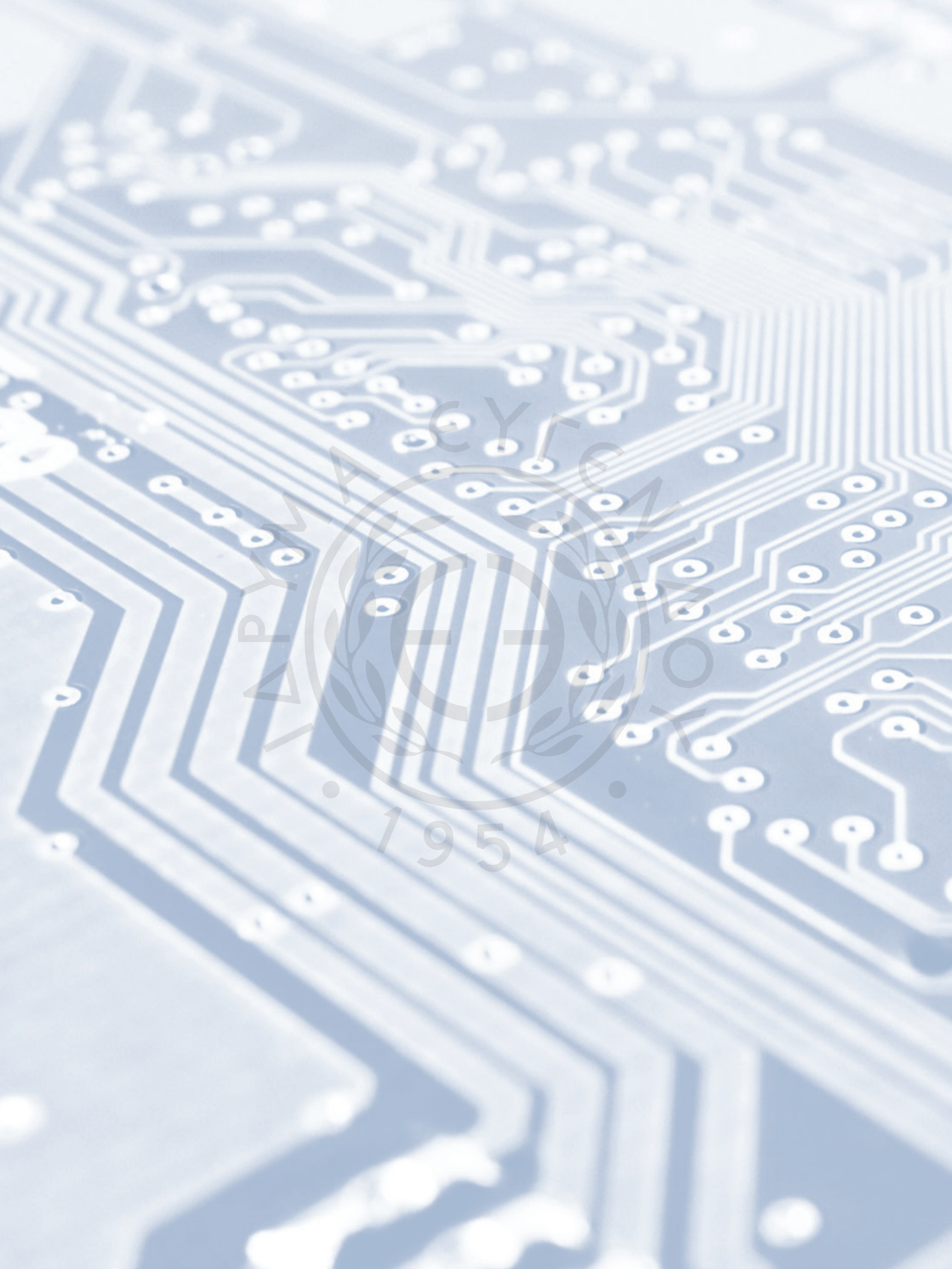
Μ. ΒΑΣΙΛΕΙΑΔΟΥ - Χ. ΥΑΚΙΝΘΟΥ - ΣΤ. ΜΠΑΡΜΠΟΥΝΑΚΗ

β΄ έκδοση

ΠΑΡΑΡΤΗΜΑ

ΠΛΟΙΑΡΧΩΝ + ΜΗΧΑΝΙΚΩΝ





ΠΑΡΑΡΤΗΜΑ

ΕΝΟΤΗΤΑ Ι Λογισμικό

ΕΝΟΤΗΤΑ ΙΙ Η γλώσσα SQL

ΙΔΡΥΜΑ ΕΥΓΕΝΙΔΟΥ
1954

1. Λογισμικό

Με τον όρο ***λογισμικό*** *(software)* ορίζεται το σύνολο των προγραμμάτων και των διαδικασιών που εκτελούνται για την παραγωγική λειτουργία του υπολογιστικού συστήματος. Το λογισμικό είναι κωδικοποιημένο μέσω ενός δυαδικού ψηφιακού συστήματος και έτσι «επικοινωνεί» με το υλικό μέρος. Τα προγράμματα δίνουν τις κατάλληλες εντολές και, με τη συνεργασία του υλικού μέρους του υπολογιστή, εκτελούνται οι διάφορες διαδικασίες και παράγονται τα αντίστοιχα αποτελέσματα. Είναι αυτονόητο ότι για τη σωστή λειτουργία του υπολογιστή, το λογισμικό του πρέπει να είναι πλήρως συμβατό με το υλικό του, έτσι ώστε να γίνει σωστά η ανάγνωση και αποκωδικοποίηση των εντολών.

Το λογισμικό διακρίνεται σε λογισμικό συστήματος και λογισμικό εφαρμογών:

1) ***Λογισμικό συστήματος*** *(System software)*: Βελτιστοποιεί τη λειτουργία του υλικού του υπολογιστή και του υπολογιστικού συστήματος. Εδώ ανήκουν:

α) Λειτουργικά συστήματα.

β) Οδηγοί συσκευών.

γ) Διαγνωστικά εργαλεία.

δ) Υπηρεσίες (services).

ε) Προγράμματα Διαχείρισης γραφικού περιβάλλοντος.

στ) Βοηθητικά προγράμματα.

Με τη χρήση του λογισμικού συστήματος, ο χρήστης δεν διαχειρίζεται ως έναν μεγάλο βαθμό, τα πολύπλοκα στοιχεία του υπολογιστή, όπως η κύρια μνήμη ή οι περιφερειακές συσκευές του. Το σύστημα επιτρέπει στον υπολογιστή να διαχειρίζεται αυτόνομα τους πόρους του.

2) **Λογισμικό εφαρμογών** *(Application Software)*: Εδώ ανήκει μία μεγάλη ποικιλία διαφορετικών προγραμμάτων και αρχείων που εκτελούν συγκεκριμένες εργασίες σύμφωνα με τις απαιτήσεις και τις ανάγκες του χρήστη. Παραδείγματα λογισμικού εφαρμογών είναι:

α) Οι εφαρμογές γραφείου (όπως επεξεργασία κειμένου, υπολογιστικών φύλλων και παρουσιάσεων).

β) Οι εφαρμογές δημιουργίας και επεξεργασίας βάσεων δεδομένων.

γ) Οι εφαρμογές επεξεργασίας φωτογραφίας, εικόνας, ήχου, βίντεο.

δ) Οι εφαρμογές προστασίας στο διαδίκτυο (antivirus, internet security).

ε) Τα εκπαιδευτικά προγράμματα.

στ) Τα παιχνίδια.

Για τις παραπάνω κατηγορίες, διάφορες εταιρίες έχουν ενδιαφερθεί και έχουν δημιουργήσει ποικιλία πακέτων λογισμικού που κυκλοφορούν στην αγορά. Το λογισμικό εφαρμογών που θα χρησιμοποιηθεί, εξαρτάται από τις ανάγκες του εκάστοτε χρήστη. Σε πολλές περιπτώσεις, συνήθως μεγάλες εταιρίες, παραγγέλνουν από κατασκευαστές λογισμικό εφαρμογών προσαρμοσμένο στις δικές τους, συγκεκριμένες ανάγκες. Αυτό το *«κατά παραγγελία»* λογισμικό, προφανώς δεν είναι διαθέσιμο στην ευρύτερη αγορά, αλλά διατίθεται αποκλειστικά στον πελάτη που το παρήγγειλε.

1.1 Βασικές έννοιες λειτουργικών συστημάτων και αρχείων

Το ***Λειτουργικό Σύστημα*** (Λ/Σ) (Operating System – OS) είναι ένα πρόγραμμα που λειτουργεί ως ενδιάμεσος κρίκος μεταξύ των χρηστών και του υλικού ενός υπολογιστικού συστήματος. Ελέγχει και συντονίζει τη χρήση του υλικού μεταξύ των διάφορων προγραμμάτων εφαρμογών των χρηστών, χωρίς να απαιτούνται εξειδικευμένες γνώσεις από τον χρήστη.

Σε γενικές γραμμές ένα Λ/Σ έχει ως στόχους:

1) Την εκτέλεση των προγραμμάτων των χρηστών.

2) Την προστασία των προγραμμάτων και των δεδομένων των διαφόρων χρηστών του υπολογιστικού συστήματος.

3) Τη χρήση του υλικού και των περιφερειακών του υπολογιστικού συστήματος με αποτελεσματικό, αποδοτικό τρόπο.

Όπως φαίνεται και στο σχήμα 1.1, το Λ/Σ κατέχει κυρίαρχη θέση σε ένα υπολογιστικό σύστημα και η ύπαρξή του είναι απαραίτητη για τη σωστή λειτουργία του υπολογιστή. Μαθαίνει από τη συμπεριφορά του χρήστη βελτιστοποιώντας την απόδοση των εφαρμογών που χρησιμοποιούνται πιο συχνά, χωρίς αυτό να γίνεται αντιληπτό από τον χρήστη. Ανακατανέμει τους πόρους του συστήματος με τέτοιον τρόπο, ώστε αυτοί να αξιοποιούνται στον μέγιστο βαθμό. Επιταχύνει τη λειτουργία των σκληρών δίσκων ***αναδιοργανώνοντας τη δομή των αρχείων*** (defragment). Για παράδειγμα, εντοπίζει τα σωστά αρχεία, ελέγχει τη διαθέσιμη χωρητικότητα στα μέσα αποθήκευσης και ειδοποιεί τον χρήστη για σφάλματα που προκύπτουν. Η επικοινωνία του χρήστη με τον υπολογιστή

επιτυγχάνεται μέσω του κατάλληλου περιβάλλοντος του Λ/Σ. Ο χρήστης υπαγορεύει με εντολές και εφαρμογές που χρησιμοποιεί στο περιβάλλον επικοινωνίας και το Λ/Σ με τη σειρά του ερμηνεύει αυτές τις ενέργειες και υπαγορεύει τις αντίστοιχες εντολές στο υλικό του υπολογιστή.

Το Λ/Σ φορτώνεται από το BIOS στη μνήμη RAM κατά την εκκίνηση του υπολογιστή. Φορτώνει το γραφικό περιβάλλον αλληλεπίδρασης χρήστη-υπολογιστή και παραμένει εκεί καθ' όλη τη διάρκεια λειτουργίας του μηχανήματος. Επιπλέον, αναλαμβάνει τη φόρτωση και την εκτέλεση των εκάστοτε εντολών και εφαρμογών που επιθυμεί ο χρήστης.



Σχ. 1.1
Τυπική θέση του Λ/Σ σε έναν υπολογιστή.

Οι ***υπηρεσίες*** που πρέπει να προσφέρει ένα Λ/Σ είναι οι ακόλουθες:

1) *Εκτέλεση προγραμμάτων*, η δυνατότητα δηλαδή του συστήματος να φορτώσει ένα πρόγραμμα στη μνήμη και να το εκτελέσει.

2) *Διεπαφή χρήστη*, το περιβάλλον δηλαδή μέσω του οποίου ο χρήστης αλληλοεπιδρά με το υπολογιστικό σύστημα, η οποία αποτελείται από:

α) Διεπαφή Γραμμής Εντολών (Command Line Interface – CLI).

β) Γραφική Διεπαφή Χρήστη (Graphical User Interface – GUI).

3) *Λειτουργίες Εισόδου/Εξόδου*, η παροχή δηλαδή των μέσων για πραγματοποίηση λειτουργιών εισόδου και εξόδου, εμπλέκοντας αρχεία και συσκευές εισόδου/εξόδου του συστήματος.

4) *Χειρισμός συστήματος αρχείων*, δηλαδή η εξυπηρέτηση προγραμμάτων αναφορικά με ανάγνωση/τροποποίηση/διαγραφή αρχείων και καταλόγων, αναζήτηση, διαχείριση δικαιωμάτων πρόσβασης.

5) *Επικοινωνίες*, δηλαδή η ανταλλαγή πληροφορίας μεταξύ διεργασιών που εκτελούνται είτε στο ίδιο είτε σε διαφορετικά υπολογιστικά συστήματα.

6) *Ανίχνευση σφαλμάτων*:

α) Το Λ/Σ πρέπει να είναι συνεχώς ενήμερο σχετικά με σφάλματα του συστήματος που αφορούν ΚΜΕ, μνήμη, συσκευές εισόδου/εξόδου, προγράμματα.

β) Για κάθε σφάλμα, το Λ/Σ πρέπει να προβαίνει στις απαραίτητες ενέργειες για τη διασφάλιση της ομαλής και απρόσκοπτης λειτουργίας του συστήματος.

7) *Κατανομή πόρων*, δηλαδή τη διαχείριση της εκχώρησης των πόρων του συστήματος (ΚΜΕ, μνήμη, πρόσβαση σε αρχεία) σε πολλαπλούς χρήστες ή διεργασίες.

8) *Ασφάλεια και προστασία*, δηλαδή την παροχή μηχανισμών για:

α) Ελεγχόμενη πρόσβαση στους πόρους του συστήματος.

β) Αυθεντικοποίηση και εξουσιοδότηση χρηστών και

γ) διασφάλιση εμπιστευτικότητας, ακεραιότητας, διαθεσιμότητας.

9) *Απολογισμός*: παρακολούθηση ενεργειών που πραγματοποιούνται στο υπολογιστικό σύστημα και αφορούν σε:

α) Χρέωση υπηρεσιών και

β) εξαγωγή στατιστικών στοιχείων.

Ως προς το *πλήθος χρηστών και διεργασιών που υποστηρίζουν*, τα Λ/Σ διακρίνονται σε τρεις κατηγορίες:

1) ***Ενός χρήστη-μίας εφαρμογής*** *(single user-single task)*: Υποστηρίζονται ένας χρήστης και μία διεργασία κάθε φορά. Βρίσκουν εφαρμογή στους υπολογιστές παλάμης (Personal Digital Assistant – PDA) και στα παλιότερα, απλά κινητά τηλέφωνα.

2) ***Ενός χρήστη-πολλών εφαρμογών*** *(single user-multitasking)*: Υποστηρίζεται ένας χρήστης αλλά πολλές διαφορετικές διεργασίες. Βρίσκουν εφαρμογή στα σημερινά έξυπνα τηλέφωνα (smartphones) και στους προσωπικούς υπολογιστές, όπου ο ίδιος χρήστης μπορεί να έχει ενεργοποιημένες πολλές εφαρμογές, να δουλεύει σε μία και ταυτόχρονα να εκτελούνται έλεγχοι ή άλλες ενέργειες στις υπόλοιπες.

3) ***Πολλών χρηστών-πολλών εφαρμογών*** *(multiuser-multitasking)*: Υποστηρίζονται πολλοί χρήστες και πολλές διαφορετικές διεργασίες ταυτόχρονα. Απαιτούν μεγάλη επεξεργαστική ισχύ και μεγάλη χωρητικότητα σε μνήμη. Βρίσκουν εφαρμογή σε εργαστήρια ή ερευνητικά κέντρα, όπου πολλοί χρήστες χρησιμοποιούν το υπολογιστικό σύστημα για διαφορετικές ενέργειες ταυτόχρονα.

1.2 Παραδείγματα Λ/Σ

Από το 1950, οπότε και εμφανίστηκαν τα πρώτα

Λ/Σ, μέχρι σήμερα, η εξέλιξη στην επιστήμη των υπολογιστών ήταν αλματώδης. Αυτό είχε ως συνέπεια, να υπάρχουν πλέον διαθέσιμα πολλά Λ/Σ, με τεράστιες δυνατότητες και πολύ φιλικά προς τον χρήστη. Στο σχήμα 1.2 παρουσιάζονται κάποια από αυτά που είναι ευρέως γνωστά. Θα μπορούσαμε να πούμε ότι τα Λ/Σ χωρίζονται σε δύο μεγάλες *κατηγορίες*:

1) Λ/Σ που ακολουθούν τη φιλοσοφία του Unix, όπως τα Linux, Solaris, Red Hat κ.ά.. Μπορούν να τρέξουν σε υπολογιστές διαφορετικών αρχιτεκτονικών και χρησιμοποιούνται κατά κύριο λόγο στους servers επιχειρήσεων και οργανισμών.

2) Λ/Σ Windows από την εταιρία Microsoft. Από την έκδοση Windows 98 μέχρι τις σημερινές Windows 8 και Windows 10, αποτελεί το πιο διαδεδομένο Λ/Σ για προσωπικό υπολογιστή. Χρησιμοποιούνται επίσης σε servers βάσεων δεδομένων.

Στη συνέχεια παρουσιάζονται τα δημοφιλέστερα Λ/Σ μέχρι σήμερα.



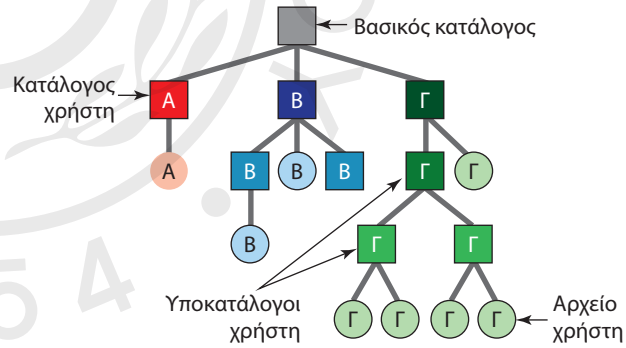
Σχ. 1.2
Λογότυπα γνωστών λειτουργικών συστημάτων.

1) Microsoft Windows – Περιγραφή, διαχείριση, αρχεία, εγγραφή δεδομένων σε διάφορα μέσα, ομάδες εφαρμογών. Το Λειτουργικό Σύστημα Windows 10

Πρόκειται για το πιο δημοφιλές λειτουργικό σύστημα, αφού προτιμάται από τους περισσότερους απλούς χρήστες για τον προσωπικό τους υπολογιστή. Βασισμένο στην ιδέα της απλής και κατανοητής διασύνδεσης μεταξύ χρήστη και υπολογιστή μέσω ενός παραθυρικού περιβάλλοντος, ο καθένας πλέον αντιμετωπίζει τον ηλεκτρονικό υπολογιστή ως ένα εύχρηστο εργαλείο και όχι ως κάτι δυσνόητο και απομακρυσμένο.

Στα Windows δεν χρειάζεται ο χρήστης να έχει γνώσεις προγραμματισμού για να μπορέσει να εκτελέσει αρκετές εργασίες, ούτε είναι απαραίτητη η απομνημόνευση εντολών. Κάθε χρήστης έχει τον ***προσωπικό του λογαριασμό*** (account) και πληκτρολογώντας το όνομά του (login name) και το συνθηματικό του (password), το σύστημα τον αναγνωρίζει και του δίνει πρόσβαση στα αντίστοιχα δεδομένα και εφαρμογές. Μέσω ενός περιβάλλοντος διασύνδεσης που αποτελείται από εικονικά παράθυρα και με τη χρήση πληκτρολογίου και ποντικιού, μπορεί να εκτελεί οποιαδήποτε εντολή επιθυμεί, να παρακολουθεί το ηλεκτρονικό του ταχυδρομείο, να αποθηκεύει και να αντιγράφει δεδομένα από μία εφαρμογή σε μία άλλη ή ακόμα και να εργάζεται σε έναν υπολογιστή από απόσταση. Αυτά οδηγούν αυτόματα σε ευρεία χρήση των Windows από όλους, εξειδικευμένους ή μη με την έννοια του υπολογιστή.

Τα αρχεία είναι ταξινομημένα σε ένα *ιεραρχικό σύστημα καταλόγων*. Ειδικά αρχεία που ονομάζονται ***φάκελοι*** *(folders)* ή ***κατάλογοι*** *(catalogues)* ή ***ευρετήρια*** *(directories)* περιέχουν πληροφορίες για αρχεία (όνομα, μέγεθος, δικαιώματα χρήσης), καθώς και άλλους φακέλους *(υποφάκελοι, subfolders)*. Οι υποφάκελοι επίσης μπορεί να περιέχουν άλλα αρχεία ή άλλους υποφακέλους και μ' αυτόν τον τρόπο δημιουργείται μία πυραμίδα όπως αυτή στο σχήμα 1.3. Υπάρχει ένας βασικός κατάλογος και κάτω από αυτόν κάθε χρήστης έχει τον δικό του αρχικό κατάλογο.



Σχ. 1.3
Ιεραρχικό σύστημα καταλόγων.

Στην περίπτωση που ο χρήστης θέλει να *αποθηκεύσει* ένα αρχείο, χρειάζεται να δηλώσει μόνο το όνομα που επιθυμεί να έχει το αρχείο και την τοποθεσία όπου θέλει να το αποθηκεύσει. Στη συνέχεια, το ΛΣ αναλαμβάνει, μέσω του *συστήματος αρχείων* που διαθέτει (NTFS, FAT32), να αποθηκεύσει το αρχείο καθώς και πληροφορίες από το όνομα και την τοποθεσία, το μέγεθος, την ώρα δημιουργίας, τα δικαιώματα χρηστών κ.ά.. Η πρόσβαση του χρήστη στα αρχεία που είναι αποθηκευμένα στον σκληρό δίσκο ή σε άλλες περι-

φερειακές μονάδες αποθήκευσης, γίνεται μέσω του ***εξερευνητή αρχείων*** *(file explorer)*. Όπως φαίνεται και στο σχήμα 1.4, στην αριστερή στήλη εμφανίζονται οι φάκελοι που χρησιμοποιούνται συνήθως και επάνω μπορούμε να διακρίνουμε τη γραμμή διεύθυνσης, στην οποία φαίνεται η ακριβής διαδρομή αποθήκευσης για το κάθε αρχείο ή φάκελο. Επάνω δεξιά μπορούμε πάντα να αναζητήσουμε κάποιο αρχείο ή φάκελο χρησιμοποιώντας στοιχεία του ονόματός του, ακόμα και αν δεν γνωρίζουμε την πλήρη ονομασία ή την ακριβή τοποθεσία αποθήκευσής του.

Όσον αφορά στην ***ονοματολογία***, το μέγιστο επιτρεπτό μήκος ονόματος είναι 255 χαρακτήρες, συμπεριλαμβανομένων διαφόρων σημείων στίξης, όπως η τελεία (.). Να σημειωθεί ότι στα Windows ***δεν επιτρέπονται οι χαρακτήρες*** \ / : * ? " < > στην ονοματολογία αρχείων ή φακέλων, γιατί έχουν ειδική χρήση από το λειτουργικό σύστημα. Επίσης, δεν υπάρχει διάκριση πεζών και κεφαλαίων, το οποίο σημαίνει ότι δεν μπορούν να συνυπάρξουν στην ίδια διαδρομή αποθήκευσης δύο αρχεία με την ίδια ονομασία σε πεζά και κεφαλαία (maria.docx, MARIA.docx).

Οι χαρακτήρες που ακολουθούν μετά την τελευταία τελεία της ονομασίας αποτελούν την ***κατάληξη*** *(suffix)* ή ***επέκταση*** *(extension)* του αρχείου και αποτελούν μία ένδειξη για το είδος του αρχείου ή του προγράμματος με το οποίο δημιουργήθηκε. Στον πίνακα 1.1 παρουσιάζονται κάποιες από τις καταλήξεις που εμφανίζονται πιο συχνά.

Μέσω των Windows έχουμε πρόσβαση σε πληθώρα εφαρμογών ανάλογα με τις ανάγκες του χρήστη, είτε αυτός είναι ένας απλός ιδιώτης είτε είναι ένας επαγγελματίας ή μία εταιρία. Ενδεικτικά, αναφέρονται μερικά ***είδη και παραδείγματα εφαρμογών*** που χρησιμοποιούνται πολύ συχνά.

1) ***Εφαρμογές διαχείρισης και επεξεργασίας εικόνας:*** Paint, Photoshop.

2) ***Εφαρμογές γραφείου:*** Word, Excel, Powerpoint και οι υπόλοιπες εφαρμογές της σουίτας Office, Adobe.

3) ***Εφαρμογές ασφαλείας:*** Antivirus (Kaspersky, Avast, Norton, McAffee, Avira), Firewalls, Intrusion detection.

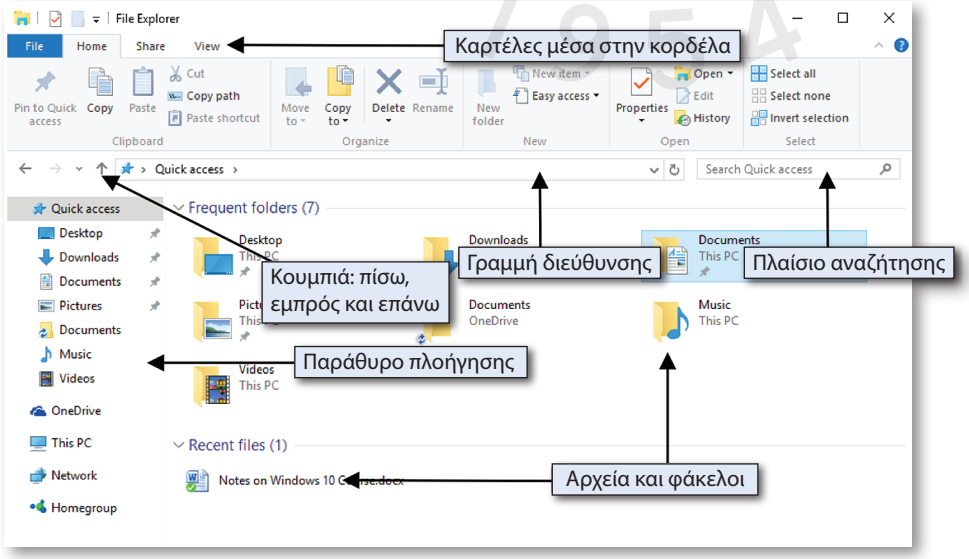
4) ***Διαχείριση συστήματος:*** File Explorer, πίνακας ελέγχου, εφαρμογές backup όπως Acronis.

5) ***Υπολογιστικές εφαρμογές:*** Calculator, Matlab.

6) ***Σχεδιαστικές εφαρμογές:*** Autocad, Archicad, PSpice.

7) ***Απομακρυσμένης διαχείρισης:*** Remote Desktop και Teamviewer, που μπορεί να χρησιμοποιηθούν στην απομακρυσμένη διαχείριση των υπολογιστών του Μηχανοστασίου ή της Γέφυρας.

Τα Windows 10 είναι το νεότερο λειτουργικό σύστημα υπολογιστών της Microsoft (σχ. 1.5). Λόγω του εύχρηστου περιβάλλοντος και της σύγχρονης προσέγγισής τους στον χρήστη, μόλις έναν μήνα μετά την κυκλοφορία τους είχαν ήδη εγκατασταθεί σε 75 εκατομμύρια συσκευές (!). Διαθέτουν ανανεωμένη οθόνη εισόδου, όπου μπορεί κάποιος να χρησιμοποιήσει τον λογαριασμό που διατηρεί στη Microsoft και να επαναφέρει τους κωδικούς του, αν χρειαστεί. Το μενού της Έναρξης είναι πολύ εύχρηστο και πλέον υπάρχει η δυνατότητα να «κατεβάσει» κάποιος μία εφαρμογή ή ένα παιχνίδι απευθείας στον υπολογιστή (μέσω Windows store) και όχι μόνο στις φορητές συσκευές, όπως ίσχυε



Σχ. 1.4
Ο εξερευνητής αρχείων του Λ/Σ Microsoft Windows 10.

Πίνακας 1.1
Καταλήξεις αρχείων και οι σημασίες τους.

.bak	Αρχείο αντιγράφου ασφαλείας
.c	Αρχείο πηγαίου κώδικα σε γλώσσα C
.hlp	Αρχείο βοήθειας
.html	Έγγραφο για το διαδίκτυο σε γλώσσα HTML
.zip	Συμπιεσμένος φάκελος
.doc, .docx	Αρχείο κειμενογράφου Word
.xls, .xlsx	Αρχείο υπολογιστικών φύλλων Excel
.ppt, .pptx	Αρχείο διαχείρισης παρουσιάσεων PowerPoint
.pdf	Αρχείο σε μορφή φορητού εγγράφου (Portable Document Format)
.txt	Γενικό αρχείο κειμένου
.gif, .jpg, .png	Αρχείο εικόνας
.avi, .mp4, .mov, .swf, .DivX, .mpeg	Αρχείο βίντεο
.wav, .mp3, .wma	Αρχείο ήχου

CORTANA

Το μεγάλο πλεονέκτημα των Windows 10 είναι η νέα ψηφιακή βοηθός, Cortana, μέσω της οποίας η αλληλεπίδραση χρήστη και λειτουργικού συστήματος γίνεται πιο ζωντανή και άμεση. Η Cortana «μαθαίνει» και καταχωρεί τις προτιμήσεις του χρήστη με τον οποίο επικοινωνεί, προσφέροντας μια πιο εξατομικευμένη εμπειρία.

μέχρι τώρα. Τα γραφικά είναι επίσης βελτιωμένα και διατίθεται ένας νέος περιηγητής (Microsoft Edge) στην προσπάθεια να κερδίσει τον χρήστη που συνδέεται στο διαδίκτυο μέσω κάποιου άλλου browser.

2) Unix – Linux[1]. Σύντομη αναφορά, διαφορές από τα Microsoft Windows, χαρακτηριστικά.

Το Unix και στη συνέχεια το Linux είναι λειτουργικά συστήματα σχεδιασμένα με διαφορετική φιλοσοφία από αυτή των Windows. Στο Unix (1960 περίπου), ο χρήστης επικοινωνούσε χωρίς γραφικό περιβάλλον, αλλά γράφοντας εντολές σε ένα δύσκολο και απαιτητικό περιβάλλον. Καταλαβαίνουμε ότι ο χρήστης τότε έπρεπε να έχει γνώσεις προγραμματισμού. Το 1984, αναπτύχθηκε από το MIT το X Window System, ένα πρωτόκολλο απεικόνισης για τη δημιουργία «παραθύρων» στην οθόνη του υπολογιστή, στοχεύοντας στους υπολογιστές με λειτουργικά συστήματα τύπου Unix.

Το Linux (1991) είναι η εξελιγμένη εκδοχή του, στην οποία πλέον υπάρχει γραφικό περιβάλλον εργασίας. Κύριο χαρακτηριστικό του Linux είναι ότι είναι ένα λογισμικό *ανοιχτού κώδικα (open source)*, δηλαδή



Σχ. 1.5
Μενού «Έναρξη και Επιφάνεια Εργασίας» στο Λ/Σ Microsoft Windows 10.

1. Δημιουργήθηκε από τον Linus Torvalds. Ο όρος Linux αποτελεί συνδυασμό των λέξεων Linus και Unix.

ο πηγαίος του κώδικας, είναι διαθέσιμος ελεύθερα σε όλους. Βρίσκει ανταπόκριση σε κοινό προγραμματιστών, οι οποίοι τροποποιούν τον πηγαίο κώδικα σύμφωνα με τις ανάγκες τους. Βέβαια, δεδομένου ότι δεν υπάρχει επίσημη τεχνική υποστήριξη από μία μόνο εταιρία, πολλά από τα προβλήματα που προκύπτουν, λύνονται μέσα από βοήθεια που προσφέρει μεμονωμένα ο ένας χρήστης στον άλλο ή εξειδικευμένες εταιρίες υποστήριξης.

Αν συγκρίναμε τα δύο Λ/Σ, Windows και Linux, κάποιες από τις διαφορές που θα επισημαίναμε παρουσιάζονται στον πίνακα 1.2.

3) Apple macOS

Το 1984, η εταιρία Apple κυκλοφόρησε ένα δικό της γραφικό λειτουργικό περιβάλλον, το Classic Mac OS και, ακολουθώντας μία πορεία ετών, η σειρά αντικαταστάθηκε από το *macOS*, το 2001 (σχ. 1.6). Είναι το κυρίαρχο Λ/Σ των υπολογιστών Mac της εταιρίας και, λόγω της χρήσης των αντίστοιχων laptop και προσωπικών υπολογιστών, πρόκειται για το δεύτερο πιο ευρέως διαδεδομένο λειτουργικό σύστημα, μετά τα Microsoft Windows. Είναι κλειστού κώδικα και διατίθεται αποκλειστικά και μόνο για τους υπολογιστές κατασκευής Apple.

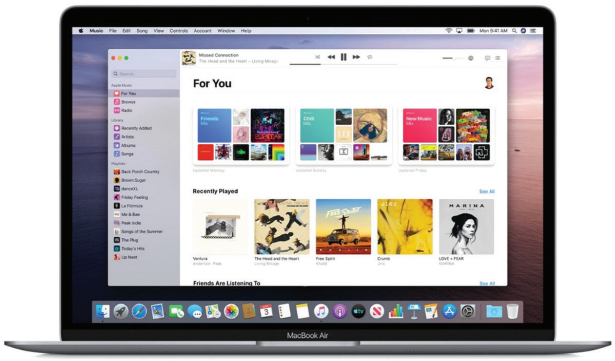
4) Android, iOS[2] και σύντομη αναφορά σε Λ/Σ για tablets

Το *Android* είναι ένα Λ/Σ που χρησιμοποιεί τον πυρήνα του Linux, αναπτύχθηκε από τη Google για συσκευές κινητής τηλεφωνίας και είναι το πιο ευρέως διαδεδομένο λογισμικό στον κόσμο. Επιτρέπει στους κατασκευαστές λογισμικού να συνθέτουν κώδικα με τη χρήση της γλώσσας προγραμματισμού Java, ελέγχοντας τη συσκευή μέσω βιβλιοθηκών λογισμικού που έχει αναπτύξει η Google. Έχει σχεδιαστεί κατά κύριο λόγο για συσκευές με οθόνη αφής (smartphones, tablets), αλλά χρησιμοποιείται και σε κονσόλες παιχνιδιών, ψηφιακές φωτογραφικές μηχανές και άλλες ηλεκτρονικές συσκευές. Η συμβατότητά του με μεγάλο αριθμό κατασκευαστών αποτελεί το χαρακτηριστικό που το φέρνει στην πρώτη θέση στην προτίμηση του κοινού. Υπάρχει διαθέσιμος ένας μεγάλος αριθμός εφαρμογών για android, με τις περισσότερες από αυτές να μπορούν να αποκτηθούν δωρεάν.

Υπάρχει, επίσης, το Λ/Σ *iOS*, ένα λογισμικό για κι-

Πίνακας 1.2
Σύγκριση χαρακτηριστικών Windows και Linux.

Ιδιότητα Χαρακτηριστικό	*MS Windows*	*Linux*
Τιμή αγοράς	50 € έως 150+ €.	Πολύ χαμηλή ή και δωρεάν.
Ευκολία χρήσης	Βρίσκονται στην 1η θέση όσον αφορά στην ευκολία χρήσης.	Γίνονται προσπάθειες βελτίωσης.
Αξιοπιστία	Γίνονται σημαντικές προσπάθειες αλλά ακόμα υστερούν.	Ιδιαίτερα αξιόπιστο. Μπορεί να τρέχει το σύστημα για μήνες χωρίς καν επανεκκίνηση.
Λογισμικό (προγράμματα, παιχνίδια κ.λπ.)	Μεγάλος αριθμός εξαιτίας της πλειονότητας των χρηστών.	Μικρός αριθμός λόγω περιορισμένου κοινού.
Κόστος λογισμικού	Παρά τη μεγάλη ποικιλία, πολλές φορές δεν είναι δωρεάν.	Δωρεάν ή με ένα ελάχιστο ποσό.
Ασφάλεια	Ευάλωτο σε ιούς και άλλες απειλές.	Πιο ασφαλές.
Υποστήριξη hardware	Υποστηρίζει μεγάλη ποικιλία hardware.	Περιορισμένη υποστήριξη.
Ανοιχτός κώδικας	Δυνατότητα τροποποίησης μόνο σε ό,τι επιτρέπει ο κατασκευαστής.	Δυνατότητα τροποποίησης όλου του πηγαίου κώδικα.
Υποστήριξη	Service packs, patches, security updates κατευθείαν από τον κατασκευαστή.	Υποστήριξη μέσα από αναζήτηση στο διαδίκτυο και εξειδικευμένες εταιρίες.



Σχ. 1.6
Ενδεικτική οθόνη του Λ/Σ MacOS Catalina.

νητά τηλέφωνα που αναπτύχθηκε και διανέμεται από την Apple Inc. Αρχικά παρουσιάστηκε για το iPhone αλλά πλέον υποστηρίζει και άλλες συσκευές της εταιρίας. Σε αντίθεση με το Android της Google, η Apple δεν δίνει την άδεια εγκατάστασης του iOS σε συσκευές εκτός Apple. Είναι εύκολο στη χρήση και με πολλές διαθέσιμες εφαρμογές.

Για τα tablets υπάρχει επίσης διαθέσιμο το Λ/Σ BlackBerry Tablet OS. Είναι βασισμένο σε Λ/Σ πραγματικού χρόνου, δηλαδή παρέχει υπηρεσίες άμεσων μηνυμάτων (Instant Messaging), μηνυμάτων που παραδίδονται σχεδόν αμέσως, χωρίς μεγάλο χρόνο αναμονής. Επίσης, το BlackBerry OS παρέχει υπηρεσίες εξαιρετικά ασφαλούς ηλεκτρονικού ταχυδρομείου.

Τέλος, αξίζει να σημειωθεί ότι το Λ/Σ Windows 10 Mobile, που είχε αναπτυχθεί από τη Microsoft, με πολλά καλά χαρακτηριστικά, αλλά με το Android απέναντί του, δεν κατάφερε να συγκεντρώσει το ενδιαφέρον των χρηστών και αποσύρθηκε τον Ιανουάριο του 2020.

1.3 Ασφάλεια διαχείρισης αρχείων

Στους ηλεκτρονικούς υπολογιστές αποθηκεύονται δεδομένα σε αρχεία που μπορεί να έχουν σημαντική αξία για τους χρήστες. Εντολές πληρωμής, τιμολόγια, φωτογραφίες ή κάποια αρχεία που οι ιδιοκτήτες τους θα θέλουν να προστατεύσουν. Τα αρχεία στους Η/Υ κινδυνεύουν από:

1) Φυσικές καταστροφές.
2) Βλάβες υλικού.
3) Λανθασμένους ή εσκεμμένους χειρισμούς χρηστών.
4) Κακόβουλο λογισμικό (ιούς).

Ο μόνος τρόπος για να προστατευτούν πλήρως τα σημαντικά αρχεία από τις παραπάνω απειλές είναι να τηρούνται αντίγραφα ασφαλείας. Το Λ/Σ παρέχει διάφορες επιλογές για να προστατευτούν τα αρχεία από κάποιες απειλές, οι βασικότερες από τις οποίες είναι:

1) ***Αντίγραφα ασφαλείας*** *(backup)*. Στα περισσότερα σύγχρονα Λ/Σ υπάρχει βοηθητικό πρόγραμμα για λήψη *Αντιγράφων Ασφαλείας* των αρχείων που θέλει ο χρήστης, σε τακτά χρονικά διαστήματα.

2) ***Κωδικοί πρόσβασης*** *(password)*. Συνοδεύουν πάντα ένα *Όνομα Χρήστη* και δίνονται μαζί για να επιτραπεί η σύνδεση στον ηλεκτρονικό υπολογιστή. Αποτελούνται από γράμματα, αριθμούς και σύμβολα. Υπάρχει διάκριση μεταξύ πεζών και κεφαλαίων.

3) ***Έλεγχος πρόσβασης.*** Ανάλογα με το σύστημα αρχείων, ο ιδιοκτήτης ενός αρχείου μπορεί να δώσει ή να αφαιρέσει δικαιώματα πάνω σε αυτό (για χρήστες και ομάδες χρηστών).

1.4 Δίκτυα και Λ/Σ

Τα περισσότερα σύγχρονα Λ/Σ επιτρέπουν τη σύνδεση ενός υπολογιστή τόσο σε τοπικά δίκτυα όσο και στο διαδίκτυο, ενσωματώνοντας στον κώδικά τους την υλοποίηση των απαιτούμενων αντίστοιχων πρωτοκόλλων. Τα πρωτόκολλα επικοινωνίας ορίζουν ένα σύνολο κανόνων συμφωνημένων και από τα δύο επικοινωνούντα μέρη, στους οποίους στηρίζεται η επικοινωνία των συσκευών (συνήθως, αλλά όχι πάντα, υπολογιστών) σε ένα δίκτυο. Οι κανόνες αυτοί καθορίζουν τη μορφή, τον χρόνο και τη σειρά μετάδοσης των πληροφοριών στο δίκτυο. Εκτελούν, επίσης, έλεγχο και διόρθωση σφαλμάτων στη διάρκεια μετάδοσης των πληροφοριών.

Τα πρωτόκολλα αυτά σήμερα είναι ενσωματωμένα στα Λ/Σ και αποτελούν το τμήμα του Λ/Σ που ονομάζεται ***Λειτουργικό Σύστημα Δικτύου*** (ΛΣΔ). Το ΛΣΔ διευθύνει κάποιες βασικές λειτουργίες όπως την ασφάλεια του δικτύου, την απομακρυσμένη πρόσβαση σε βάσεις δεδομένων, τον διαμοιρασμό αρχείων, εφαρμογών και συσκευών όπως εκτυπωτές, τη δημιουργία ομάδων χρηστών και άλλων λειτουργιών του δικτύου.

Αυτές οι λειτουργίες, μαζί με πολλά περισσότερα χαρακτηριστικά πλέον, περιλαμβάνονται σε όλα τα κοινά Λ/Σ, όπως είναι τα Windows και το Linux. Τα σύγχρονα Λ/Σ, προκειμένου να υποστηρίξουν τις καθημερινές λειτουργίες των χρηστών, όπως πλοήγηση στο διαδίκτυο, μεταφορά αρχείων, απομακρυσμένη σύνδεση και ηλεκτρονικό ταχυδρομείο, υποστηρίζουν τα περισσότερα γνωστά και διαδεδομένα πρωτόκολλα, όπως είναι το TCP/IP για τη διασύνδεση των συσκευών σε χαμηλό επίπεδο, το HTTP για την πλοήγηση, το SMTP για το ηλεκτρονικό ταχυδρομείο, το FTP για τη μεταφορά αρχείων κ.λπ..

Ενότητα ΙΙ Η Γλώσσα SQL (Structured Query Language)

Η SQL είναι μία γλώσσα σχεδιασμένη για επικοινωνία με βάσεις δεδομένων και διαχείρισή τους. Πρόκειται για την πιο συνηθισμένη γλώσσα που χρησιμοποιείται από συστήματα διαχείρισης σχεσιακών βάσεων δεδομένων, όπως Oracle, Access, Sybase κ.λπ. Με τη χρήση της μπορούμε να δημιουργήσουμε πίνακες σε μία ΒΔ, να θέσουμε ερωτήματα στη ΒΔ, να εισάγουμε ή να ενημερώσουμε αρχεία της ΒΔ, καθώς και να προβούμε σε οποιαδήποτε άλλη ενέργεια χειρισμού των δεδομένων στη βάση.

Εντολές διαχείρισης δεδομένων

Βασική προϋπόθεση για οποιαδήποτε ενέργεια είναι η αποθήκευση των δεδομένων σε πίνακες. Σε μία ενημερωμένη ΒΔ λοιπόν, μπορούμε να χρησιμοποιήσουμε κατάλληλες εντολές για εξαγωγή πληροφοριών, προσθήκη ή διαγραφή δεδομένων από τους πίνακες ή αλλαγή των ήδη αποθηκευμένων δεδομένων στη ΒΔ.

Στη συνέχεια, θα δούμε κάποια παραδείγματα σύνταξης της εντολής SELECT, με την οποία κάνουμε ερωτήσεις στη ΒΔ και παίρνουμε τις αντίστοιχες απαντήσεις.

Η εντολή SELECT

Η εντολή SELECT χρησιμοποιείται για ανάκτηση από μία ΒΔ δεδομένων που ικανοποιούν τα κριτήρια που εμείς θέτουμε. Η βασική μορφή μίας εντολής **SELECT** είναι:

SELECT *column1, column2…*
FROM *table(s)*
[WHERE *condition*]
[GROUP BY *group _type*]
[HAVING *where_definition*]
[ORDER BY *order_type*];

Ας θεωρήσουμε, για παράδειγμα, ότι έχουμε τον παρακάτω πίνακα εργαζομένων με όνομα ΥΠΑΛΛΗΛΟΣ1 (Πίνακας 1). Τα πεδία του πίνακα είναι ΑΦΜ, Ονοματεπώνυμο, Διεύθυνση, Πόλη.

Πίνακας 1: ΥΠΑΛΛΗΛΟΣ1

ΑΦΜ	ΟΝΟΜΑΤΕΠΩΝΥΜΟ	ΔΙΕΥΘΥΝΣΗ	ΠΟΛΗ
12345678	ΒΑΣΙΛΕΙΑΔΟΥ ΜΑΡΙΑ	ΑΓΙΑΣ ΤΡΙΑΔΟΣ 9	ΘΕΣΣΑΛΟΝΙΚΗ
23456789	ΥΑΚΙΝΘΟΣ ΧΑΡΑΛΑΜΠΟΣ	ΤΣΙΜΙΣΚΗ 113	ΘΕΣΣΑΛΟΝΙΚΗ
34567890	ΒΑΣΙΛΕΙΑΔΟΥ ΕΥΑΓΓΕΛΙΑ	ΑΡΙΣΤΟΤΕΛΟΥΣ 12	ΑΘΗΝΑ
45678901	ΒΑΓΙΑΝΟΣ ΔΗΜΗΤΡΙΟΣ	ΧΑΛΚΟΚΟΝΔΥΛΗ 56	ΑΘΗΝΑ
56789012	ΜΠΕΣΗΣ ΓΙΩΡΓΟΣ	ΘΕΣΣΑΛΟΝΙΚΗΣ 44-46	ΠΕΙΡΑΙΑΣ

Ερώτημα 1: Να εμφανιστεί λίστα με όλα τα στοιχεία όλων των υπαλλήλων.

Λύση 1:

SELECT * FROM ΥΠΑΛΛΗΛΟΣ1;

Το σύμβολο (*) στη γλώσσα των υπολογιστών συνήθως σημαίνει «όλα» ή «οτιδήποτε». Οπότε, εδώ ουσιαστικά ζητάμε όλα τα πεδία, δηλαδή όλες τις στήλες, από τον πίνακα ΥΠΑΛΛΗΛΟΣ1.

Αν θέλουμε να εισάγουμε κάποια επιπλέον κριτήρια έτσι ώστε να ανακτήσουμε ένα υποσύνολο των

εγγραφών (γραμμών) του πίνακα, τότε χρησιμοποιούμε τη ρήτρα **WHERE**. Τα αποτελέσματα που επιστρέφονται είναι αυτά που ικανοποιούν τη συνθήκη της WHERE.

Οι πιο συνηθισμένοι τελεστές σύγκρισης που χρησιμοποιούνται στη WHERE φαίνονται στον πίνακα 2:

Πίνακας 2: Τελεστές σύγκρισης

Τελεστής	Ενέργεια - έλεγχος
=	Ισότητα
>	Μεγαλύτερο από
<	Μικρότερο από
>=	Μεγαλύτερο ή ίσο
<=	Μικρότερο ή ίσο
!=	Όχι ίσο, διάφορο
IS NULL	Ελέγχει αν το πεδίο είναι κενό
BETWEEN	Η τιμή ανάμεσα σε δύο όρια
IN	Η τιμή ανήκει σε ένα συγκεκριμένο σύνολο
LIKE	Ελέγχει αν η τιμή ταιριάζει σε μια ακολουθία

Ερώτημα 2: Να εμφανιστεί λίστα με τα ονοματεπώνυμα και τις διευθύνσεις των υπαλλήλων που έχουν έδρα τη Θεσσαλονίκη.

Λύση 2:

```
SELECT ΟΝΟΜΑΤΕΠΩΝΥΜΟ, ΔΙΕΥΘΥΝΣΗ FROM ΥΠΑΛΛΗΛΟΣ1 WHERE ΠΟΛΗ = 'ΘΕΣΣΑΛΟΝΙΚΗ';
```

Ερώτημα 3: Να εμφανιστεί λίστα με τα ονοματεπώνυμα των υπαλλήλων που έχουν έδρα την Αθήνα ή τον Πειραιά.

Λύση 3:

Υπάρχουν δύο τρόποι για να πάρουμε την ίδια απάντηση στο ερώτημα:

```
SELECT ΟΝΟΜΑΤΕΠΩΝΥΜΟ FROM ΥΠΑΛΛΗΛΟΣ1 WHERE ΠΟΛΗ IN ('ΑΘΗΝΑ', 'ΠΕΙΡΑΙΑΣ');
```

ή

```
SELECT ΟΝΟΜΑΤΕΠΩΝΥΜΟ FROM ΥΠΑΛΛΗΛΟΣ1 WHERE ΠΟΛΗ = 'ΑΘΗΝΑ' OR ΠΟΛΗ = 'ΠΕΙΡΑΙΑΣ';
```

Με τον πρώτο τρόπο, γίνονται αποδεκτές οι πόλεις που ανήκουν (IN) στο σύνολο ΑΘΗΝΑ-ΠΕΙΡΑΙΑΣ, ενώ με τον δεύτερο τρόπο εξετάζουμε χωριστά την τιμή ΠΟΛΗ, μία φορά για την τιμή ΑΘΗΝΑ και μία φορά για την τιμή ΠΕΙΡΑΙΑΣ.

Προσοχή! Ο τελεστής OR μας δίνει αποτέλεσμα TRUE (ότι ισχύει δηλαδή), όταν τουλάχιστον μία από τις συνθήκες που συνδέει, ισχύει.

Ερώτημα 4: Να εμφανιστεί λίστα με τα ονοματεπώνυμα των υπαλλήλων των οποίων το επώνυμο αρχίζει από 'Β'.

Λύση 4:

```
SELECT ΟΝΟΜΑΤΕΠΩΝΥΜΟ FROM ΥΠΑΛΛΗΛΟΣ1 WHERE ΟΝΟΜΑΤΕΠΩΝΥΜΟ LIKE 'Β%';
```

Ας υποθέσουμε ότι στον παραπάνω πίνακα ΥΠΑΛΛΗΛΟΣ1, υπάρχει ακόμα ένα πεδίο με ονομασία ΜΙΣΘΟΣ, και οι αντίστοιχες τιμές για κάθε υπάλληλο.

Ερώτημα 5: Να εμφανιστεί λίστα με τους υπαλλήλους οι οποίοι έχουν μισθό 50.000 και άνω.

Λύση 5:

```
SELECT ΟΝΟΜΑΤΕΠΩΝΥΜΟ FROM ΥΠΑΛΛΗΛΟΣ1 WHERE ΜΙΣΘΟΣ >= 50.000;
```

Ερώτημα 6: Να εμφανιστεί λίστα με τους υπαλλήλους οι οποίοι έχουν μισθό μεταξύ 40.000 και 60.000.

Λύση 6:

```
SELECT ΟΝΟΜΑΤΕΠΩΝΥΜΟ FROM ΥΠΑΛΛΗΛΟΣ1 WHERE ΜΙΣΘΟΣ BETWEEN 40000 AND 60000;
```

Λογικοί τελεστές AND, OR, NOT

Όταν απαιτείται να απαντήσουμε σε πιο πολύπλοκα ερωτήματα, χρησιμοποιώντας συνδυασμό κριτηρίων, εφαρμόζουμε τους λογικούς τελεστές AND, OR και NOT. Ο λογικός τελεστής **AND** επιστρέφει ως αποτέλεσμα TRUE (δηλαδή ότι ισχύει), όταν όλα τα γεγονότα τα οποία συνδέει, ισχύουν. Αντίστοιχα, ο λογικός τελεστής **OR** επιστρέφει ως αποτέλεσμα TRUE, όταν τουλάχιστον ένα από τα γεγονότα τα οποία συνδέει, ισχύει. Τέλος, ο λογικός τελεστής **NOT** επιστρέφει το λογικό αντίθετο του γεγονότος στο οποίο εφαρμόζεται.

Σε μία έκφραση WHERE, μπορούν να χρησιμοποιηθούν παραπάνω από μία φορά AND και OR. Όπως και στις μαθηματικές, έτσι και στις λογικές πράξεις υπάρχει προτεραιότητα κατά την εκτέλεσή τους. Οι πράξεις με τη μεγαλύτερη προτεραιότητα εκτελούνται πρώτες, ανεξάρτητα από τη θέση τους στην έκφραση.

Η προτεραιότητα εκτέλεσης των παραπάνω λογικών πράξεων είναι NOT-AND-OR. Αν θέλουμε να καθορίσουμε άλλη σειρά εκτέλεσης, όπως και στα μαθηματικά, θα πρέπει να χρησιμοποιήσουμε παρενθέσεις.

Ερώτημα 7: Να εμφανιστεί λίστα με τους υπαλλήλους οι οποίοι μένουν στην Αθήνα και έχουν μισθό τουλάχιστον 50.000.

Λύση 7:

```
SELECT * FROM ΥΠΑΛΛΗΛΟΣ1 WHERE ΠΟΛΗ = 'ΘΕΣΣΑΛΟΝΙΚΗ' AND ΜΙΣΘΟΣ >= 50000;
```

Ερώτημα 8: Να εμφανιστεί λίστα με τους υπαλλήλους των οποίων το ονοματεπώνυμο αρχίζει από Β, μένουν στην Αθήνα και έχουν μισθό τουλάχιστον 50.000.

Λύση 8:

```
SELECT * FROM ΥΠΑΛΛΗΛΟΣ1 WHERE ΟΝΟΜΑΤΕΠΩΝΥΜΟ LIKE 'Β%' AND (ΠΟΛΗ = 'ΑΘΗΝΑ' AND ΜΙΣΘΟΣ >= 50000);
```

Ερώτημα 9: Να εμφανιστεί λίστα με τους υπαλλήλους που δεν μένουν στον Πειραιά ούτε στην Αθήνα.

Λύση 9:

```
SELECT * FROM ΥΠΑΛΛΗΛΟΣ1 WHERE NOT ΠΟΛΗ = 'ΠΕΙΡΑΙΑΣ' AND NOT ΠΟΛΗ = 'ΑΘΗΝΑ';
```

Χειρισμός διπλοεγγραφών στα αποτελέσματα

Υπάρχουν περιπτώσεις όπου μία τιμή είναι η ίδια για πολλές εγγραφές του πίνακα. Ανάλογα με τα ερωτήματα που θα απευθύνουμε στη ΒΔ, κάποιες φορές θα θέλαμε η πληροφορία αυτή που επαναλαμβάνεται, να μην εμφανίζεται στην απάντηση.

Ας πάρουμε, για παράδειγμα, τον πίνακα ΥΠΑΛΛΗΛΟΣ2 (Πίνακας 3), που δείχνει πολλαπλές καταχωρήσεις της ίδια τιμής στις θέσεις των υπαλλήλων.

Πίνακας 3: ΥΠΑΛΛΗΛΟΣ2

ΚΩΔ_ΥΠΑΛΛΗΛΟΥ	ΜΙΣΘΟΣ	ΠΡΙΜ	ΘΕΣΗ
123	78000	12000	ΔΙΕΥΘΥΝΤΗΣ
234	89000	13000	ΔΙΕΥΘΥΝΤΗΣ
345	90000	14000	ΔΙΕΥΘΥΝΤΗΣ
456	56000	10500	ΤΜΗΜΑΤΑΡΧΗΣ
567	34000	8000	ΠΡΟΣΩΠΙΚΟ
678	45000	9000	ΠΡΟΣΩΠΙΚΟ
789	23000	5600	ΕΚΠΑΙΔΕΥΤΗΣ
890	25000	5600	ΕΚΠΑΙΔΕΥΤΗΣ

Παρατηρούμε ότι κάποιες θέσεις, όπως 'ΔΙΕΥΘΥΝΤΗΣ', υπάρχουν σε παραπάνω από μία εγγραφές. Αν το ερώτημα αφορά στη λίστα με τις θέσεις των υπαλλήλων, θα θέλαμε η θέση 'ΔΙΕΥΘΥΝΤΗΣ' να εμφανιστεί μόνο μία φορά και όχι περισσότερες. Για τον λόγο αυτό, χρησιμοποιούμε τη λέξη-κλειδί **DISTINCT**.

Παράδειγμα

```
SELECT DISTINCT ΘΕΣΗ FROM ΥΠΑΛΛΗΛΟΣ2;
```

Απόδοση ονομασίας σε αποτέλεσματα

Σε περίπτωση που θέλουμε να αποδώσουμε έναν τίτλο (ψευδώνυμο) στα αποτελέσματα, χρησιμοποιούμε τον όρο **AS**. Ο όρος AS ακολουθείται από διπλά εισαγωγικά ("") μέσα στα οποία γράφουμε τον τίτλο που θέλουμε να έχει η στήλη των αποτελεσμάτων. Εναλλακτικά, μπορούμε να χρησιμοποιήσουμε τετράγωνες αγκύλες ([]).

Παράδειγμα

```
SELECT DISTINCT ΘΕΣΗ AS "Κατηγορίες θέσεων" FROM ΥΠΑΛΛΗΛΟΣ2;
```

Επιστρέφει το ίδιο αποτέλεσμα με πριν, μόνο που τώρα η στήλη με τις θέσεις έχει ονομασία «Κατηγορίες θέσεων».

Προσθήκη πεδίων με υπολογιζόμενες τιμές

Με τη χρήση του όρου **AS**, μπορούμε να δημιουργήσουμε νέα πεδία (στήλες) στον πίνακα των αποτελεσμάτων, τα περιεχόμενα των οποίων προκύπτουν μέσω πράξεων στα αρχικά δεδομένα. Με τον όρο **AS**, καθορίζουμε και την ονομασία της νέας στήλης.

Ερώτημα 10: Κατά πόσο θα αυξηθεί ο μισθός των διευθυντών αν η αύξηση είναι 10%;

Λύση 10:

```
SELECT ΚΩΔ_ΥΠΑΛΛΗΛΟΥ, ΜΙΣΘΟΣ, ΘΕΣΗ, ΜΙΣΘΟΣ*10% AS "Αύξηση"
FROM ΥΠΑΛΛΗΛΟΣ2 WHERE ΘΕΣΗ='ΔΙΕΥΘΥΝΤΗΣ';
```

Εκτός από αριθμητικές πράξεις και συναρτήσεις, υποστηρίζονται και πράξεις ή συναρτήσεις με συμβολοσειρές (ακολουθίες χαρακτήρων).

Η πράξη || ή & οδηγεί σε συνένωση συμβολοσειρών (string concatenation).

Στο ακόλουθο παράδειγμα, όλα τα στοιχεία του πελάτη «ενώνονται» και εμφανίζονται τελικά ως μία ενιαία ταχυδρομική διεύθυνση.

Ονομασία πίνακα: ΠΕΛΑΤΕΣ			
ΠΕΛΑΤΗΣ	ΔΙΕΥΘΥΝΣΗ	ΠΟΛΗ	ΤΚ
ΑΡΓΥΡΙΟΥ ΑΝΔΡΟΚΛΗΣ	ΜΑΡΤΙΟΥ 23	ΘΕΣΣΑΛΟΝΙΚΗ	54641

```
SELECT ΠΕΛΑΤΗΣ, ΔΙΕΥΘΥΝΣΗ || '-' || ΠΟΛΗ || ', Τ.Κ. :' || ΤΚ AS "Ταχυδρομική Διεύθυνση"
FROM ΠΕΛΑΤΕΣ;
```

Με αποτέλεσμα:

ΠΕΛΑΤΗΣ	**Ταχυδρομική Διεύθυνση**
ΑΡΓΥΡΙΟΥ ΑΝΔΡΟΚΛΗΣ	ΜΑΡΤΙΟΥ 23-ΘΕΣΣΑΛΟΝΙΚΗ, Τ.Κ. :54641

Πράξεις μπορούμε να εκτελέσουμε οπουδήποτε, όχι μόνο στις στήλες που προκύπτουν από υπολογισμούς. Μπορούν να χρησιμοποιηθούν, για παράδειγμα, σε λογικές συνθήκες που περιέχονται στις διάφορες εντολές.

Ας πάρουμε για παράδειγμα, τον παρακάτω πίνακα που ονομάζεται ΦΙΛΜ (Πίνακας 4). Στον πίνακα αυτόν είναι καταχωρημένες οι τιμές και η διαθέσιμη ποσότητα ανά φιλμ σε ένα κατάστημα ενοικιάσεων ταινιών.

Πίνακας 4: ΦΙΛΜ

Ονομασία Πίνακα: ΦΙΛΜ						
Κωδ. ταινίας	όνομα	είδος	ποσότητα	τιμή	έτος κυκλοφορίας	γλώσσα
001	Ο Ιρλανδός	Δράμα	5	4	2019	1
002	Τζόκερ	Έγκλημα	6	3	2019	1
003	Top Gun	Δράση	4	2	1986	1
004	Pretty woman	Αισθηματική	2	1	1990	1
005	Ο Νονός	Αστυνομική	5	2	1972	1
006	Τρελές Σφαίρες	Κωμωδία	2	1	1988	1

Ερώτημα 11: Να εμφανιστούν τα φιλμ για τα οποία η αποθηκευτική αξία είναι μεγαλύτερη ή ίση των 10 μονάδων.

Λύση 11:

```
SELECT όνομα, είδος, τιμή*ποσότητα AS "Αξία αποθήκης" FROM ΦΙΛΜ WHERE τιμή*ποσότητα >=10;
```

Στην παραπάνω έκφραση, εντοπίζεται πρώτα ο πίνακας ΦΙΛΜ. Στη συνέχεια, η SQL ανατρέχει στην έκφραση WHERE και εφαρμόζει τη λογική συνθήκη, υπολογίζοντας τα αντίστοιχα γινόμενα. Τέλος, εκτελεί τις εκφράσεις στη λίστα SELECT, επιλέγοντας μόνο τις τιμές που ικανοποιούν τα κριτήρια της WHERE.

Παρατηρούμε ότι, σ' αυτήν την περίπτωση, δεν μπορούμε να κάνουμε χρήση ψευδωνύμων στηλών στην πρόταση WHERE, γιατί αυτή εκτελείται πριν από τη SELECT, όπου ορίζονται τα ψευδώνυμα.

Ταξινόμηση αποτελεσμάτων

Τα δεδομένα αποθηκεύονται στους πίνακες κατά γραμμές, οι οποίες καταγράφονται με τη σειρά που δίνονται τα δεδομένα στη ΒΔ και χωρίς κάποια ιδιαίτερη ταξινόμηση. Με τον ίδιο τρόπο που είναι αποθηκευμένα, με τον ίδιο τρόπο εξάγονται και στον πίνακα των αποτελεσμάτων που προκύπτουν από τα ερωτήματα. Αν θέλουμε η εμφάνιση των αποτελεσμάτων να γίνει με συγκεκριμένη σειρά κατά γραμμές, τότε χρησιμοποιούμε την **ORDER BY**:

Ερώτημα 12: Να εμφανιστούν τα στοιχεία του πίνακα ΦΙΛΜ κατά φθίνουσα σειρά τιμής του φιλμ:

Λύση 12:

```
SELECT * FROM ΦΙΛΜ ORDER BY τιμή DESC;
```

Όπως φαίνεται στην εντολή SELECT, η ταξινόμηση γίνεται κατά τιμή. Η φθίνουσα ταξινόμηση καθορίζεται από τη λέξη-κλειδί DESC (descending). Σε περίπτωση αύξουσας ταξινόμησης, θα επιλέγαμε τη λέξη-κλειδί ASC (ascending).

Μπορούμε να ορίσουμε πολλές στήλες ταξινόμησης, χωρίζοντάς τες με κόμμα (,) μετά την ORDER BY. Η SQL ταξινομεί πρώτα ως προς την πρώτη στήλη που θα ορίσουμε στην έκφραση ORDER BY, και σε περίπτωση ταύτισης των τιμών της πρώτης στήλης, προχωρά στη δεύτερη στήλη κ.λπ.

Συγκέντρωση και ομαδοποίηση δεδομένων

Κάποιες φορές θα χρειαστεί να απαντήσουμε σε ερωτήματα όπως «Ποια είναι η μέση τιμή μιας παραγγελίας;» ή «Πόσα προϊόντα έχουν απόθεμα μεγαλύτερο του 10;» κ.λπ. Για να μπορέσουμε να πάρουμε απαντήσεις σε τέτοιου είδους ερωτήσεις, η SQL παρέχει κάποιες συναρτήσεις που εφαρμόζονται σε ομάδες δεδομένων. Οι πιο συνηθισμένες είναι αυτές που φαίνονται στον πίνακα 5:

Πίνακας 5: Συναρτήσεις που χρησιμοποιούνται συνήθως

Συνάρτηση	Περιγραφή
AVG(στήλη)	Μέση τιμή δεδομένων στη συγκεκριμένη στήλη.
COUNT(…)	Αν μέσα στην παρένθεση προσδιορίσουμε μία στήλη, επιστρέφει τον αριθμό των κελιών που δεν είναι κενά (non-NULL). Αν προσθέσουμε τη λέξη DISTINCT πριν από το όνομα της στήλης, επιστρέφει τον αριθμό των ξεχωριστών τιμών μόνο για τη συγκεκριμένη στήλη. Αν απλά, γράψουμε * μέσα στην παρένθεση, επιστρέφει τον αριθμό των εγγραφών ανεξάρτητα από το εάν υπάρχουν κενά (NULL) κελιά ή όχι.
MIN(στήλη)	Ελάχιστη τιμή δεδομένων στη συγκεκριμένη στήλη.
MAX(στήλη)	Μέγιστη τιμή δεδομένων στη συγκεκριμένη στήλη.
SUM(στήλη)	Άθροισμα όλων των δεδομένων τιμών στη συγκεκριμένη στήλη.

Ερώτημα 13: Να εμφανιστεί η μέση τιμή όλων των ταινιών και το σύνολο των διαθέσιμων τεμαχίων, με τις αντίστοιχες επικεφαλίδες στις στήλες των αποτελεσμάτων.

Λύση 13:

```
SELECT AVG(τιμή) AS "Μέση Τιμή φιλμ", SUM(ποσότητα) AS "Σύνολο ταινιών" FROM ΦΙΛΜ;
```

Ερώτημα 14: Να εμφανιστούν η ελάχιστη και η μέγιστη τιμή πώλησης όλων των ταινιών.

Λύση 14:

SELECT MIN(τιμή), MAX(ποσότητα) FROM ΦΙΛΜ;

Ερώτημα 15: Να εμφανιστούν τα διαθέσιμα είδη ταινιών με την αντίστοιχη επικεφαλίδα. Κάθε είδος να εμφανίζεται μόνο μία φορά.

Λύση 15:

SELECT COUNT(DISTINCT είδος) AS "Πλήθος ειδών" FROM ΦΙΛΜ;

Έτσι, στην περίπτωση που υπήρχαν τρεις ταινίες που ανήκαν στο είδος «Δράση», το είδος θα εμφανιζόταν μία φορά και όχι τρεις.

Στην περίπτωση που δεν μας ενδιέφερε η εμφάνιση κάθε είδους μόνο μία φορά, θα μπορούσαμε να χρησιμοποιήσουμε και τις εκφράσεις:

SELECT COUNT(*) AS "Πλήθος ειδών" FROM ΦΙΛΜ;

ή

SELECT COUNT(είδος) AS "Πλήθος ειδών" FROM ΦΙΛΜ;

Ερώτημα 16: Πόσες ταινίες δράσης υπάρχουν στο κατάστημα; Τα αποτελέσματα να εμφανίζονται με τις αντίστοιχες επικεφαλίδες.

Λύση 16:

SELECT SUM(ποσότητα) AS "Πλήθος ταινιών δράσης" FROM ΦΙΛΜ WHERE είδος='Δράση';

Αν θέλουμε πιο λεπτομερείς πληροφορίες, μπορούμε να χρησιμοποιήσουμε την ***GROUP BY***. Σ' αυτήν την περίπτωση μπορούμε να πάρουμε πληροφορίες για όλα τα δεδομένα μιας συγκεκριμένης στήλης και όχι για όλα τα δεδομένα όλων των εγγραφών (σειρών). Η στήλη που μας ενδιαφέρει είναι αυτή που προσδιορίζουμε αμέσως μετά την GROUP BY.

Ας πάρουμε για παράδειγμα τον παρακάτω πίνακα που ονομάζεται ΕΡΓΑΖΟΜΕΝΟΙ (Πίνακας 6). Εδώ εμφανίζονται οι υπάλληλοι με τις αποδοχές τους και το τμήμα στο οποίο εργάζονται.

ΕΡΓΑΖΟΜΕΝΟΙ (ΚΩΔ_ΥΠΑΛΛΗΛΟΥ, ΕΠΩΝΥΜΟ, ΑΠΟΔΟΧΕΣ, ΚΩΔ_ΤΜΗΜΑΤΟΣ)

Πίνακας 6: ΕΡΓΑΖΟΜΕΝΟΙ

ΚΩΔ_ΥΠΑΛΛΗΛΟΥ	ΕΠΩΝΥΜΟ	ΑΠΟΔΟΧΕΣ	ΚΩΔ_ΤΜΗΜΑΤΟΣ
23	ΥΑΚΙΝΘΟΣ	35000	1
34	ΒΑΣΙΛΕΙΑΔΟΥ	36000	1
45	ΠΕΔΙΑΔΙΤΗΣ	23000	2
56	ΑΡΓΥΡΙΟΥ	24000	1
67	ΧΙΛΙΤΙΔΗΣ	31000	2

Ερώτημα 17: Να εμφανίζεται ο μέσος όρος αποδοχών των υπαλλήλων ανά τμήμα, με την αντίστοιχη επικεφαλίδα.

Λύση 17:

SELECT ΚΩΔ_ΤΜΗΜΑΤΟΣ, AVG(ΑΠΟΔΟΧΕΣ) AS "Μ.Ο. Αποδοχών" FROM ΕΡΓΑΖΟΜΕΝΟΙ

GROUP BY ΚΩΔ_ΤΜΗΜΑΤΟΣ ORDER BY ΚΩΔ_ΤΜΗΜΑΤΟΣ;

Ερώτημα 18: Να εμφανίζεται το σύνολο των μισθών των υπαλλήλων ανά τμήμα, με την αντίστοιχη επικεφαλίδα.

Λύση 18:

SELECT ΚΩΔ_ΤΜΗΜΑΤΟΣ, SUM(ΑΠΟΔΟΧΕΣ) AS "Σύνολο Αποδοχών" FROM ΕΡΓΑΖΟΜΕΝΟΙ

GROUP BY ΚΩΔ_ΤΜΗΜΑΤΟΣ ORDER BY ΚΩΔ_ΤΜΗΜΑΤΟΣ;

Κριτήρια πριν την ομαδοποίηση

Επειδή η WHERE δεν μπορεί να περιέχει συγκεντρωτικές συναρτήσεις ως κριτήριο, η λύση σε τέτοιου είδους προβληματισμούς δίνεται χρησιμοποιώντας τη ρήτρα **HAVING**. Η ρήτρα αυτή μας επιτρέπει να ορίσουμε κριτήρια επιλογής γραμμών μέσα σε ομαδοποιημένα δεδομένα. Αν χρησιμοποιήσουμε τη HAVING, αυτή θα πρέπει να έπεται της GROUP BY.

Ερώτημα 19: Από τον πίνακα ΦΙΛΜ να εμφανιστεί η συνολική ποσότητα ταινιών εκτός από αυτές που ανήκουν στο είδος 'Δράση' και αυτές που δεν ανήκουν σε κάποιο είδος. Τα αποτελέσματα να εμφανιστούν με την αντίστοιχη επικεφαλίδα.

Λύση 19:

SELECT είδος, SUM(ποσότητα) AS "Σύνολο ορισμένων ταινιών" FROM ΦΙΛΜ

GROUP BY είδος HAVING είδος<>›Δράμα› AND είδος Is Not Null;

Παράδειγμα 5

Από τον πίνακα ΠΡΟΣΩΠΙΚΟ (Πίνακας 7), θέλουμε τα τμήματα που απασχολούν περισσότερους από 2 υπαλλήλους

Πίνακας 7: ΠΡΟΣΩΠΙΚΟ

ΚΩΔ_ΥΠΑΛΛΗΛΟΥ	ΕΠΩΝΥΜΟ	ΑΠΟΔΟΧΕΣ	ΚΩΔ_ΤΜΗΜΑΤΟΣ
23	ΥΑΚΙΝΘΟΣ	35000	1
34	ΒΑΣΙΛΕΙΑΔΟΥ	36000	1
45	ΠΕΔΙΑΔΙΤΗΣ	23000	2
56	ΑΡΓΥΡΙΟΥ	24000	1
67	ΧΙΛΙΤΙΔΗΣ	31000	2
67	ΒΟΥΒΑΛΙΔΗΣ	35000	3

SELECT ΚΩΔ_ΤΜΗΜΑΤΟΣ FROM ΠΡΟΣΩΠΙΚΟ GROUP BY ΚΩΔ_ΤΜΗΜΑΤΟΣ

HAVING COUNT(ΚΩΔ_ΥΠΑΛΛΗΛΟΥ)>2;

Συνδέσεις πινάκων

Σε κάποιες περιπτώσεις, για να απαντήσουμε σε ένα ερώτημα, θα χρειαστεί να συνδυάσουμε δεδομένα από περισσότερους του ενός πίνακες. Έστω, για παράδειγμα, ότι ζητάμε λίστα με τα ονόματα, τηλέφω-

να και τα τμήματα στα οποία είναι εγγεγραμμένοι οι σπουδαστές μιας σχολής. Τα στοιχεία των τμημάτων βρίσκονται στον πίνακα ΤΜΗΜΑ (Πίνακας 8), ενώ τα προσωπικά στοιχεία των σπουδαστών βρίσκονται στον πίνακα ΣΠΟΥΔΑΣΤΗΣ (Πίνακας 9), όπου:

ΤΜΗΜΑ(ΚΩΔ_ΤΜΗΜΑ, ΤΜΗΜΑ)

ΣΠΟΥΔΑΣΤΗΣ(ΑΜΣ, ΟΝΟΜΑΤΕΠΩΝΥΜΟ, ΦΥΛΟ, ΤΗΛΕΦΩΝΟ, ΚΩΔ_ΤΜΗΜΑ)

Πίνακας 8: ΤΜΗΜΑ

ΚΩΔ_ΤΜΗΜΑ	ΤΜΗΜΑ
01	ΣΩΣΤΙΚΩΝ
02	ΠΛΟΙΑΡΧΩΝ
03	ΜΗΧΑΝΙΚΩΝ
04	ΗΛΕΚΤΡΟΛΟΓΩΝ
05	ΚΕΣΕΝ

Πίνακας 9: ΣΠΟΥΔΑΣΤΗΣ

ΑΜΣ	ΟΝΟΜΑΤΕΠΩΝΥΜΟ	ΦΥΛΟ	ΤΗΛΕΦΩΝΟ	ΚΩΔ_ΤΜΗΜΑ
2345	ΣΙΑΦΛΙΑΚΗΣ ΣΩΤΗΡΙΟΣ	Α	6912345678	01
3456	ΑΡΓΥΡΙΟΥ ΑΝΔΡΟΚΛΗΣ	Α	6923456789	04
4567	ΤΣΟΥΛΗΣ ΝΙΚΟΛΑΟΣ	Α	2392034567	02
5678	ΠΑΠΑΔΟΠΟΥΛΟΣ ΔΗΜΗΤΡΙΟΣ	Α	2392045678	01
6789	ΑΛΕΞΙΟΥ ΛΙΝΑ	Γ	6934567890	04
7890	ΥΑΚΙΝΘΟΣ ΧΑΡΑΛΑΜΠΟΣ	Α	6945678901	03
8901	ΒΑΣΙΛΕΙΑΔΟΥ ΜΑΡΙΑ	Γ	2392056789	03
9012	ΑΘΑΝΑΣΙΑΔΗΣ ΙΩΑΝΝΗΣ	Α	6956789012	03
0123	ΡΩΣΣΙΑΔΟΥ ΚΩΝΣΤΑΝΤΙΑ	Γ	6967890123	02
1234	ΧΑΡΑΛΑΜΠΙΔΗΣ ΙΑΚΩΒΟΣ	Α	2392067890	05

Για να πάρουμε απαντήσεις, θα χρειαστεί να συνδυάσουμε δεδομένα και από τους δύο πίνακες. Επομένως, θα χρειαστεί να συνδυάσουμε τους πίνακες χρησιμοποιώντας τις μεταξύ τους σχέσεις.

Ερώτημα 20: Να εμφανιστεί λίστα με την ονομασία του τμήματος, το ονοματεπώνυμο και το τηλέφωνο του κάθε σπουδαστή.

Λύση 20:

```
SELECT ΤΜΗΜΑ, ΟΝΟΜΑΤΕΠΩΝΥΜΟ, ΤΗΛΕΦΩΝΟ
FROM ΣΠΟΥΔΑΣΤΗΣ, ΤΜΗΜΑ WHERE ΤΜΗΜΑ.ΚΩΔ_ΤΜΗΜΑ=ΣΠΟΥΔΑΣΤΗΣ.ΚΩΔ_ΤΜΗΜΑ;
```

Καταλαβαίνουμε ότι πρόκειται για σύνδεση πινάκων, όταν βλέπουμε πως μετά τη λέξη FROM ακολουθούν παραπάνω από ένας πίνακες, χωρισμένοι με κόμμα (,). Η πρόταση που ακολουθεί τη λέξη

FROM, μας δίνει όλους τους δυνατούς συνδυασμούς των εγγραφών των πινάκων ΣΠΟΥΔΑΣΤΗΣ και ΤΜΗΜΑ. Εμείς δεν ενδιαφερόμαστε για όλους τους συνδυασμούς παρά μόνο για εκείνους στους οποίους ο ΚΩΔ_ΤΜΗΜΑ του σπουδαστή συμπίπτει με τον ΚΩΔ_ΤΜΗΜΑ της σχολής.

Αυτό δηλώνεται με τη συνθήκη ΤΜΗΜΑ.ΚΩΔ_ΤΜΗΜΑ=ΣΠΟΥΔΑΣΤΗΣ.ΚΩΔ_ΤΜΗΜΑ, η οποία χρησιμοποιώντας την τελεία (.) διαχωρίζει τα πεδία που έχουν ίδια ονομασία αλλά ανήκουν σε διαφορετικούς πίνακες.

Επιτρέπεται, ακόμη, η χρήση ομαδοποίησης και σύνοψης δεδομένων, όπως στο επόμενο παράδειγμα.

Ερώτημα 21: Να εμφανιστεί λίστα με το πλήθος των σπουδαστών σε κάθε τμήμα. Στα αποτελέσματα να υπάρχει και η αντίστοιχη επικεφαλίδα.

Λύση 21:

```
SELECT ΤΜΗΜΑ, COUNT(ΑΜΣ) AS "Σύνολο σπουδαστών"
FROM ΣΠΟΥΔΑΣΤΗΣ, ΤΜΗΜΑ
WHERE ΤΜΗΜΑ.ΚΩΔ_ΤΜΗΜΑ=ΣΠΟΥΔΑΣΤΗΣ.ΚΩΔ_ΤΜΗΜΑ
GROUP BY ΤΜΗΜΑ;
```

Συνδυασμένα ερωτήματα

Θα θεωρήσουμε, για παράδειγμα, τους πίνακες 10, 11 και 12. Στον πίνακα 10 εμφανίζονται οι υπάλληλοι μιας εταιρίας συνολικά και στους πίνακες 11 και 12 εμφανίζεται ποιοι από αυτούς συμμετείχαν σε δύο σεμινάρια επιμόρφωσης.

Πίνακας 10: ΥΠΑΛΛΗΛΟΣ

ΚΩΔ_ΥΠΑΛ	ΕΠΩΝΥΜΟ	ΟΝΟΜΑ	ΤΗΛΕΦΩΝΟ	ΔΙΕΥΘΥΝΣΗ
1234	ΣΙΑΦΛΙΑΚΗΣ	ΣΩΤΗΡΙΟΣ	6912345678	ΘΕΣΣΑΛΟΝΙΚΗΣ 34, ΕΠΑΝΩΜΗ
2345	ΑΡΓΥΡΙΟΥ	ΑΝΔΡΟΚΛΗΣ	6923456789	ΜΑΡΤΙΟΥ 23, ΘΕΣΣΑΛΟΝΙΚΗ
3456	ΤΣΟΥΛΗΣ	ΝΙΚΟΛΑΟΣ	2392034567	Μ. ΜΠΟΤΣΑΡΗ 10, ΘΕΣΣΑΛΟΝΙΚΗ
4567	ΠΑΠΑΔΟΠΟΥΛΟΣ	ΔΗΜΗΤΡΙΟΣ	2392045678	ΑΡΙΣΤΟΤΕΛΟΥΣ 12, ΝΕΑ ΜΗΧΑΝΙΩΝΑ
5678	ΑΛΕΞΙΑΔΟΥ	ΑΝΤΩΝΙΑ	6934567890	ΑΓ. ΠΑΡΑΣΚΕΥΗΣ 11, ΤΑΓΑΡΑΔΕΣ
6789	ΥΑΚΙΝΘΟΣ	ΓΕΩΡΓΙΟΣ	6945678901	ΒΑΣ. ΓΕΩΡΓΙΟΥ 57, ΘΕΣΣΑΛΟΝΙΚΗ
7890	ΒΑΣΙΛΕΙΑΔΟΥ	ΕΥΑΓΓΕΛΙΑ	2392056789	ΛΑΜΠΡΑΚΗ 6, ΘΕΣΣΑΛΟΝΙΚΗ
8901	ΓΕΩΡΓΙΟΥ	ΙΩΑΝΝΗΣ	6956789012	ΜΗΤΡΟΠΟΛΕΩΣ 23, ΦΙΛΥΡΟ
9012	ΡΩΣΣΙΑΔΟΥ	ΚΩΝΣΤΑΝΤΙΑ	6967890123	ΜΠΟΥΜΠΟΥΛΙΝΑΣ 23, ΚΑΡΔΙΑ
1111	ΧΑΡΑΛΑΜΠΙΔΗΣ	ΙΑΚΩΒΟΣ	2392067890	ΠΕΡΙΚΛΕΩΣ 17, ΠΕΡΑΙΑ
2222	ΑΘΑΝΑΣΙΑΔΟΥ	ΑΛΕΞΙΑ	23107493544	ΚΟΛΟΚΟΤΡΩΝΗ 16, ΘΕΣΣΑΛΟΝΙΚΗ

Πίνακας 11: Ερώτημα – ΣΕΜΙΝΑΡΙΟ1

ΚΩΔ_ΥΠΑΛ	ΕΠΩΝΥΜΟ	ΟΝΟΜΑ	ΤΗΛΕΦΩΝΟ	ΔΙΕΥΘΥΝΣΗ
1234	ΣΙΑΦΛΙΑΚΗΣ	ΣΩΤΗΡΙΟΣ	6912345678	ΘΕΣΣΑΛΟΝΙΚΗΣ 34, ΕΠΑΝΩΜΗ
3456	ΤΣΟΥΛΗΣ	ΝΙΚΟΛΑΟΣ	2392034567	Μ. ΜΠΟΤΣΑΡΗ 10, ΘΕΣΣΑΛΟΝΙΚΗ
5678	ΑΛΕΞΙΑΔΟΥ	ΑΝΤΩΝΙΑ	6934567890	ΑΓ. ΠΑΡΑΣΚΕΥΗΣ 11, ΤΑΓΑΡΑΔΕΣ
2345	ΑΡΓΥΡΙΟΥ	ΑΝΔΡΟΚΛΗΣ	6923456789	ΜΑΡΤΙΟΥ 23, ΘΕΣΣΑΛΟΝΙΚΗ
7890	ΒΑΣΙΛΕΙΑΔΟΥ	ΕΥΑΓΓΕΛΙΑ	2392056789	ΛΑΜΠΡΑΚΗ 6, ΘΕΣΣΑΛΟΝΙΚΗ

Πίνακας 12: Ερώτημα – ΣΕΜΙΝΑΡΙΟ2

ΚΩΔ_ΥΠΑΛ	ΕΠΩΝΥΜΟ	ΟΝΟΜΑ	ΤΗΛΕΦΩΝΟ	ΔΙΕΥΘΥΝΣΗ
1234	ΣΙΑΦΛΙΑΚΗΣ	ΣΩΤΗΡΙΟΣ	6912345678	ΘΕΣΣΑΛΟΝΙΚΗΣ 34, ΕΠΑΝΩΜΗ
8901	ΓΕΩΡΓΙΟΥ	ΙΩΑΝΝΗΣ	6956789012	ΜΗΤΡΟΠΟΛΕΩΣ 23, ΦΙΛΥΡΟ
6789	ΥΑΚΙΝΘΟΣ	ΓΕΩΡΓΙΟΣ	6945678901	ΒΑΣ. ΓΕΩΡΓΙΟΥ 57, ΘΕΣΣΑΛΟΝΙΚΗ
5678	ΑΛΕΞΙΑΔΟΥ	ΑΝΤΩΝΙΑ	6934567890	ΑΓ. ΠΑΡΑΣΚΕΥΗΣ 11, ΤΑΓΑΡΑΔΕΣ
9012	ΡΩΣΣΙΑΔΟΥ	ΚΩΝΣΤΑΝΤΙΑ	6967890123	ΜΠΟΥΜΠΟΥΛΙΝΑΣ 23, ΚΑΡΔΙΑ
7890	ΒΑΣΙΛΕΙΑΔΟΥ	ΕΥΑΓΓΕΛΙΑ	2392056789	ΛΑΜΠΡΑΚΗ 6, ΘΕΣΣΑΛΟΝΙΚΗ
1111	ΧΑΡΑΛΑΜΠΙΔΗΣ	ΙΑΚΩΒΟΣ	2392067890	ΠΕΡΙΚΛΕΩΣ 17, ΠΕΡΑΙΑ

1) Όταν θέλουμε να συνδυάσουμε τα αποτελέσματα από 2 ή περισσότερες εντολές SELECT, τότε χρησιμοποιούμε τον τελεστή **UNION**. Ως αποτέλεσμα εμφανίζεται η ένωση των συνόλων των αποτελεσμάτων κάθε ξεχωριστής εντολής. Για να μπορέσουμε να χρησιμοποιήσουμε τον τελεστή UNION, κάθε SELECT που συμμετέχει πρέπει να αφορά στον ίδιο αριθμό στηλών (πεδίων), τα οποία βρίσκονται στις SELECT με την ίδια σειρά και περιέχουν παρόμοιους τύπους δεδομένων.

Η γενική σύνταξη της εντολής με χρήση του τελεστή UNION είναι:

```
SELECT column1, column2… FROM table1
UNION
SELECT column1, column2… FROM table2;
```

Ερώτημα 22: Να εμφανιστεί λίστα με τα ονοματεπώνυμα των υπαλλήλων που παρακολούθησαν τουλάχιστον ένα σεμινάριο, χωρίς να εμφανίζονται δύο φορές τα ονόματα αυτών που συμμετείχαν και στα δύο σεμινάρια.

Λύση 22:

Η εντολή είναι:

```
SELECT ΕΠΩΝΥΜΟ, ΟΝΟΜΑ FROM ΣΕΜΙΝΑΡΙΟ1
UNION
SELECT ΕΠΩΝΥΜΟ, ΟΝΟΜΑ FROM ΣΕΜΙΝΑΡΙΟ2;
```

Έτσι όπως είναι γραμμένη η εντολή, μας δίνει ως αποτέλεσμα όλους τους υπαλλήλους που παρακολούθησαν τουλάχιστον ένα σεμινάριο. Οι υπάλληλοι ΣΙΑΦΛΙΑΚΗΣ, ΑΛΕΞΙΑΔΟΥ, ΒΑΣΙΛΕΙΑΔΟΥ, που παρακολούθησαν και τα δύο σεμινάρια, εμφανίζονται μία φορά στον πίνακα των αποτελεσμάτων. Αν θέλαμε να επιτρέπονταν διπλότυπες τιμές στον πίνακα των αποτελεσμάτων, αντί για UNION θα έπρεπε να χρησιμοποιήσουμε **UNION ALL**.

2) Αν θέλουμε να πάρουμε την ***τομή*** των συνόλων των αποτελεσμάτων κάθε ξεχωριστής εντολής, να πάρουμε δηλαδή ως απάντηση τα κοινά τους στοιχεία, τότε μπορούμε να δουλέψουμε διαφορετικά, μέσα από την WHERE.

Ερώτημα 23: Να εμφανιστεί λίστα με τα ονοματεπώνυμα των υπαλλήλων που παρακολούθησαν και τα δύο σεμινάρια.

Λύση 23:

Εδώ θέλουμε ο πίνακας των αποτελεσμάτων να περιέχει μόνο τις κοινές εγγραφές των πινάκων Σεμινάριο1 και Σεμινάριο2. Η απάντηση είναι η ακόλουθη:

```
SELECT ΕΠΩΝΥΜΟ, ΟΝΟΜΑ FROM ΣΕΜΙΝΑΡΙΟ1
WHERE
ΚΩΔ_ΥΠΑΛ IN (SELECT ΚΩΔ_ΥΠΑΛ FROM ΣΕΜΙΝΑΡΙΟ2);
```

Ο τελεστής **IN** μας επιτρέπει να προσδιορίσουμε πολλαπλές τιμές σε μία έκφραση WHERE. Στην παραπάνω απάντηση, το IN χρησιμοποιείται για να συγκρίνει αν οι κωδικοί των υπαλλήλων που παρακολούθησαν το σεμινάριο 1 ανήκουν στο σύνολο των κωδικών των υπαλλήλων που παρακολούθησαν το σεμινάριο 2. Δηλαδή, εντοπίζει τους κοινούς κωδικούς υπαλλήλων στα δύο σύνολα των σεμιναρίων.

Η σειρά με την οποία εκτελούνται συνθήκες που περιέχουν IN, είναι από την τελευταία προς την πρώτη. Στη δική μας περίπτωση, εκτελείται πρώτη η εντολή SELECT ΚΩΔ_ΥΠΑΛ FROM ΣΕΜΙΝΑΡΙΟ2 μέσα στην παρένθεση εντοπίζοντας τους υπαλλήλους που παρακολούθησαν το σεμινάριο 2, και στη συνέχεια, το πρόγραμα εξετάζει ποιοι από τους υπαλλήλους που παρακολούθησαν το σεμινάριο 1, ανήκουν και στο σύνολο του σεμιναρίου 2. Έτσι, εκτελείται η πράξη τομής των δύο συνόλων και εντοπίζονται τα κοινά τους στοιχεία.

3) Αν θέλουμε να πάρουμε ως αποτέλεσμα τις εγγραφές που αποτελούν εξαίρεση μιας κατάστασης, μπορούμε να χρησιμοποιήσουμε την λογική έκφραση NOT σε συνδυασμό με τη WHERE και την IN.

Ερώτημα 24: Να εμφανιστεί λίστα με τα ονοματεπώνυμα των υπαλλήλων που παρακολούθησαν το πρώτο σεμινάριο και δεν παρακολούθησαν το δεύτερο.

Λύση 24:

```
SELECT ΕΠΩΝΥΜΟ, ΟΝΟΜΑ FROM ΣΕΜΙΝΑΡΙΟ1
WHERE ΚΩΔ_ΥΠΑΛ NOT IN (SELECT ΚΩΔ_ΥΠΑΛ FROM ΣΕΜΙΝΑΡΙΟ2);
```

Ερώτημα (για εξάσκηση): Να εμφανιστεί λίστα με τα ονοματεπώνυμα των υπαλλήλων που δεν παρακολούθησαν κανένα σεμινάριο.



