

Πρότζεκτ Εργαστηρίου Βάσεων Δεδομένων

2024-2025

Περιεχόμενα

Περιεχόμενα	1
1. Εισαγωγή	2
2. Περιγραφή της Βάσης Δεδομένων	3
3. Περιεχόμενο Εργασίας	4
3.1. Μέρος Α: Σχεδιασμός ΒΔ και SQL	5
3.1.2. Νέες απαιτήσεις	5
3.1.3. Δημιουργία Stored Procedure	7
3.1.4. Δημιουργία Trigger	8
3.1.5. Δομή εργασίας – Ζητούμενα Α' Μέρος (Σχεδιασμός ΒΔ και SQL)	9
3.2. Μέρος Β: GUI	9
3.2.1. Δημιουργία Γραφικών Διεπαφών Χρήστη (GUI)	9
3.2.2. Έλεγχος Εγκυρότητας Δεδομένων	10
3.2.3. Προαιρετικό Ζητούμενο με Μπόνους Βαθμό	10
3.2.4. Δομή εργασίας – Ζητούμενα Β' μέρος (GUI)	10
4. Οδηγίες υποβολής	10
5. Εργασία σε ομάδες - Αξιολόγηση	11
6. Εξέταση	13

1. Εισαγωγή

Το πρότζεκτ Εργαστηρίου Βάσεων Δεδομένων έχει δύο στόχους :

- να εξοικειωθείτε με τη σχεδίαση, την υλοποίηση, την εισαγωγή και την επεξεργασία δεδομένων σε μια σχεσιακή Βάση Δεδομένων (ΒΔ), και
- να χρησιμοποιήσετε την τεχνολογία JDBC για να διασυνδέσετε τη ΒΔ με διεπαφές (Graphical User Interfaces) που θα υλοποιήσετε σε Java, ώστε να αποκτήσετε εμπειρία σχεδιασμού μιας πιο ολοκληρωμένης εφαρμογής.

Το πρότζεκτ αποτελείται συνολικά από δύο μέρη: **Μέρος Α - Σχεδιασμός ΒΔ και SQL** και **Μέρος Β - GUIs**.

Τα παραδοτέα σας θα είναι μια αναφορά με το σύνολο της τεκμηρίωσης, το σύνολο του κώδικα SQL και τον κώδικα για το GUI. Η τεκμηρίωση αφορά και στο **Μέρος Α - Σχεδιασμός ΒΔ και SQL** και στο **Μέρος Β - GUIs** που περιγράφονται αναλυτικά παρακάτω. Επιπλέον, για το **Μέρος Β** απαιτείται να εξηγήσετε τη λειτουργία του με τη χρήση screenshots. Αναλυτικότερα τα παραδοτέα και οι προθεσμίες παρουσιάζονται παρακάτω.

2. Περιγραφή της Βάσης Δεδομένων

Η ΒΔ αφορά την **διαχείριση των εργασιών ενός ομίλου εταιρειών που κατέχει ένα πλήθος δισκογραφικών εταιριών**. Το σύστημα χρησιμοποιείται από τον όμιλο για την διαχείριση μουσικών καλλιτεχνών, άλμπουμ και παραγωγών που έχουν συμβόλαιο με κάποια από τις δισκογραφικές εταιρίες του ομίλου. Οι παρακάτω λειτουργικές απαιτήσεις καλύπτονται από το σχεσιακό σχήμα που σας έχει δοθεί από την Προπαρασκευαστική Φάση.

- Για όλους τους μουσικούς καλλιτέχνες (artist) του συστήματος διατηρούμε έναν αύξοντα αριθμό (μοναδικό για κάθε καλλιτέχνη) και τον τύπο του (person ή band). Οι καλλιτέχνες του συστήματος διακρίνονται σε δύο τύπους:
 - ο Πρόσωπο (person), δηλαδή φυσικά πρόσωπα που δραστηριοποιούνται καλλιτεχνικά ατομικά ή σε συγκρότημα ή και τα δύο.
 - ο Συγκρότημα (band), το οποίο αποτελείται από καλλιτέχνες τύπου Πρόσωπο.
- Για κάθε Πρόσωπο (person), το σύστημα αποθηκεύει έναν μοναδικό αριθμό του, το όνομα, το επώνυμό, την ημερομηνία γέννησής, τη χώρα καταγωγής, τυχόν καλλιτεχνικό ψευδώνυμο και αν δραστηριοποιείται καλλιτεχνικά ατομικά ή όχι. Ένας μουσικός καλλιτέχνης-Πρόσωπο, μπορεί να δραστηριοποιείται καλλιτεχνικά ατομικά, μπορεί να συμμετέχει σε ένα ή περισσότερα συγκροτήματα και μπορεί να κάνει και τα δύο. Σε αυτήν την περίπτωση καταγράφουμε το συγκρότημα ή τα συγκροτήματα στα οποία συμμετέχει, την ημερομηνία που μπήκε σε κάθε συγκρότημα και τυχόν ημερομηνία αποχώρησης από αυτά.
- Για κάθε Συγκρότημα (band) καταγράφεται ένας μοναδικός αριθμός, το όνομα, η ημερομηνία που δημιουργήθηκε το συγκρότημα και τυχόν ημερομηνία διάλυσης. Κάθε συγκρότημα μπορεί να αποτελείται από πολλούς καλλιτέχνες-Πρόσωπα.
- Για κάθε Μουσικό Είδος (genre), το σύστημα αποθηκεύει έναν μοναδικό αύξοντα αριθμό του και το όνομά του.
- Ο όμιλος μπορεί να έχει πολλές δισκογραφικές εταιρίες, για αυτό το λόγο για κάθε δισκογραφική εταιρεία (recordCompany), το σύστημα διατηρεί ένα μοναδικό αύξοντα αριθμό, το όνομα της δισκογραφικής εταιρείας, τη διεύθυνσή της, ένα τηλέφωνο, ένα email, την ημερομηνία δημιουργίας της εταιρείας και τυχόν ημερομηνία τερματισμού λειτουργίας. Κάθε μουσικός καλλιτέχνης (Πρόσωπο ή Συγκρότημα) πρέπει να συνεργάζεται με μία μόνο εταιρεία κάθε φορά, αλλά μπορεί και να ξανασυνεργαστεί με εταιρίες που έχει συνεργαστεί στο παρελθόν. Για αυτή τη σχέση διατηρούμε την ημερομηνία έναρξης συνεργασίας με την εταιρεία και την πιθανή ημερομηνία αποχώρησης. Μια δισκογραφική εταιρεία μπορεί και να μην έχει καμία συνεργασία με καλλιτέχνες.
- Οι μουσικοί καλλιτέχνες δημιουργούν άλμπουμ (album). Κάθε άλμπουμ που εισάγεται στη ΒΔ παίρνει ένα μοναδικό αύξοντα ρυθμό, τον τίτλο του, τον καλλιτέχνη που το δημιούργησε, το μουσικό είδος στο οποίο ανήκει το άλμπουμ και την δισκογραφική εταιρεία που παρήγαγε το άλμπουμ. Ένα άλμπουμ πρέπει να ανήκει μόνο σε ένα μουσικό είδος. Κάθε άλμπουμ μπορεί να παράγεται από έναν μόνο καλλιτέχνη και από μόνο μια δισκογραφική εταιρεία. Προφανώς, κάθε καλλιτέχνης και κάθε δισκογραφική εταιρεία μπορούν να παράγουν πολλά άλμπουμ.

- Κάθε άλμπουμ αποτελείται από μουσικά κομμάτια (track). Έτσι, για κάθε κομμάτι το σύστημα διατηρεί ένα μοναδικό αύξοντα αριθμό, τον τίτλο του κομματιού, το άλμπουμ στο οποίο ανήκει, τη χρονική διάρκεια του κομματιού, τη θέση του κομματιού στο άλμπουμ και τυχόν στίχους για το κομμάτι.
- Κάθε δισκογραφική εταιρεία διαθέτει παραγωγούς (producer) που συμμετέχουν στη δημιουργία των άλμπουμ. Για κάθε παραγωγό, το σύστημα διατηρεί ένα μοναδικό αύξοντα αριθμό, το όνομά του, το επώνυμό του, το πλήθος των άλμπουμ που έχει κάνει την παραγωγή, και τη δισκογραφική εταιρεία στην οποία δουλεύει. Ένας παραγωγός πρέπει να δουλεύει μόνο σε μια δισκογραφική εταιρεία κάθε φορά, ενώ μια εταιρία μπορεί και να μην διαθέτει παραγωγούς. Κάθε παραγωγός μπορεί να συνεργαστεί πάλι με εταιρίες που έχει συνεργαστεί στο παρελθόν. Κάθε συνεργασία με εταιρεία που έχει ο παραγωγός συνοδεύεται με την ημερομηνία έναρξης συνεργασίας και τυχόν ημερομηνία λήξης της συνεργασίας. Ένας παραγωγός μπορεί να κάνει την παραγωγή σε ένα ή περισσότερα άλμπουμ, ενώ κάθε άλμπουμ έχει μόνο έναν παραγωγό.
- Τέλος κάθε άλμπουμ έχει και μια κυκλοφορία (albumRelease). Για την κυκλοφορία του άλμπουμ καταγράφουμε έναν μοναδικό αύξοντα αριθμό, την ημερομηνία κυκλοφορίας. Τον τύπο κυκλοφορίας, ο οποίος μπορεί να είναι LP,CD ή MP3. Το status κυκλοφορίας το οποίο μπορεί να είναι official, promotion, bootleg, withdrawn ή cancelled. Τη συσκευασία η οποία μπορεί να είναι book, cardboard, sleeve, digipack, jewel case και not available. Κάθε άλμπουμ μπορεί να έχει μια ή περισσότερες κυκλοφορίες.

3. Περιεχόμενο Εργασίας

3.1. Μέρος Α: Σχεδιασμός ΒΔ και SQL

3.1.1. Προπαρασκευαστική φάση: Δημιουργία της ΒΔ

Σας έχει ήδη δοθεί το σχεσιακό μοντέλο για να δημιουργηθεί η ΒΔ και να εισαχθούν εγγραφές σε κάθε πίνακα. Το παρακάτω πλήθος **εγγραφών ανά πίνακα είναι το ελάχιστο πλήθος εγγραφών ανά μέλος.**

Πίνακας	Ελάχιστο πλήθος εγγραφών ανά άτομο
Artist	10
Person	7
Band	3
BandMember	4
Genre	5
RecordCompany	5
ArtistCompany	11
Producer	6
ProducerCompany	8
Album	10
Track	25
AlbumRelease	12

ΠΙΝΑΚΑΣ 1. ΕΛΑΧΙΣΤΟ ΠΛΗΘΟΣ ΕΓΓΡΑΦΩΝ ΑΝΑ ΠΙΝΑΚΑ ΚΑΙ ΑΝΑ ΜΕΛΟΣ

Σημειώνουμε ότι η συγκεκριμένη βάση δεδομένων χρησιμοποιήθηκε σε όλες τις ασκήσεις του Εργαστηρίου. Με τον τρόπο αυτό και από την Προπαρασκευαστική Φάση έχει ήδη επιτευχθεί εξοικείωση με το σχήμα και τα δεδομένα της βάσης και κατ' επέκταση η καλύτερη κατανόηση του προβλήματος που θα διαχειριστείτε.

3.1.2. Νέες απαιτήσεις

Σε αυτήν την ενότητα δίνονται επιπλέον απαιτήσεις που αφορούν την δομή και το χειρισμό της ΒΔ. Με βάση την Περιγραφή (Κεφάλαιο 2) και λαμβάνοντας υπόψη τις νέες λειτουργικές απαιτήσεις που αναφέρονται παρακάτω, θα πρέπει να μελετήσετε και να τροποποιήσετε το σχεσιακό σχήμα της ΒΔ ώστε να υποστηρίζονται και οι νέες απαιτήσεις. Η διαδικασία αυτή περιλαμβάνει **(α) την επέκταση του σχεσιακού σχήματος με επιπλέον πεδία και νέους πίνακες, (β) την εισαγωγή κατάλληλου πλήθους εγγραφών και (γ) την τεκμηρίωση για τις σχεδιαστικές σας αποφάσεις.**

Η δισκογραφική εταιρεία επιθυμεί να επεκτείνει το υπάρχον σύστημα διαχείρισης με λειτουργικότητα παρακολούθησης και διαχείρισης συναυλιών (concert) των καλλιτεχνών της. Το νέο υποσύστημα θα παρακολουθεί τους χώρους διεξαγωγής συναυλιών (venues), τις προγραμματισμένες συναυλίες και τις συμμετοχές των καλλιτεχνών σε αυτές.

3.1.2.1. Απαίτηση: να δημιουργηθεί μηχανισμός για τον προγραμματισμό νέας συναυλίας (consert) κάποιου καλλιτέχνη.

- Οι συναυλίες θα πρέπει να έχουν μια από τρεις καταστάσεις: προγραμματισμένη, ολοκληρωμένη και ακυρωμένη.
- Νέες συναυλίες για έναν καλλιτέχνη θα μπορούν να προγραμματίζονται έως και πέντε μέρες πριν την ημερομηνία έναρξης της συναυλίας.
- Δεν θα μπορεί να προγραμματιστεί νέα συναυλία ή να επαναπρογραμματιστεί ακυρωμένη συναυλία για καλλιτέχνη ο οποίος έχει ήδη τρεις προγραμματισμένες συναυλίες.
- Μια συναυλία θα μπορεί να ακυρωθεί μέχρι και τρεις μέρες πριν την ημερομηνία διεξαγωγής της.

3.1.2.2. Απαίτηση: να δημιουργηθεί μηχανισμός για την εύρεση χώρου διεξαγωγής καλλιτέχνη.

- Συναυλίες σε κατάσταση ακυρωμένη δεν συμμετέχουν στην επεξεργασία.
- Για κάθε συναυλία θα βρίσκεται χώρος διεξαγωγής με 10% χωρητικότητα μεγαλύτερη από την απαιτούμενη.
- Όταν μια συναυλία ολοκληρώνεται ο χώρος διεξαγωγής θα ελευθερώνεται. Το ίδιο και όταν ακυρώνεται.
- Αν βρεθούν παραπάνω από ένας κατάλληλοι χώροι διεξαγωγής για την συναυλία θα επιλέγεται αυτός με τη μεγαλύτερη βαθμολογία, η οποία θα υπολογίζεται σύμφωνα με τα κριτήρια, όπως φαίνονται στον παρακάτω πίνακα.

Κριτήρια	Βαθμολογία
Για κάθε 1.000 άτομα χωρητικότητα	1
Για κάθε 100 πραγματοποιημένες συναυλίες	3
Για κάθε χρόνο λειτουργίας	2

ΠΙΝΑΚΑΣ 2. ΒΑΘΜΟΛΟΓΙΑ ΑΝΑ ΚΡΙΤΗΡΙΟ

Για παράδειγμα, αν ένας χώρος διεξαγωγής έχει 5.000 άτομα χωρητικότητα, 300 πραγματοποιημένες συναυλίες και λειτουργεί 4 χρόνια, ο βαθμός που υπολογίζεται είναι ($5 \times 1 + 3 \times 3 + 4 \times 2 = 22$).

- Οι συναυλίες οι οποίες έχουν ολοκληρωθεί ή ακυρωθεί χωρίς δυνατότητα επαναπρογραμματισμού τους, διαγράφονται από τον πίνακα που περιέχει τις προγραμματισμένες συναυλίες και μεταφέρονται στο ιστορικό συναυλιών (Απαίτηση 3.1.2.3). Η κατάστασή τους στο ιστορικό θα πρέπει να μετατραπεί από *προγραμματισμένη* ή *ακυρωμένη* σε *ολοκληρωμένη*. Αν η αίτηση ήταν ακυρωμένη θα καταχωρίζεται ως ημερομηνία διεξαγωγής NULL, αλλιώς θα καταχωρίζεται η ημερομηνία διεξαγωγής της.

3.1.2.3. Απαίτηση: να δημιουργηθεί ιστορικό συναυλιών. Θα δημιουργήσετε ένα πίνακα στον οποίον θα καταχωρίζετε τις ολοκληρωμένες συναυλίες. Για κάθε συναυλία του πίνακα θα διατηρείτε: το id της συναυλίας, το όνομα του καλλιτέχνη, το όνομα του χώρου, το πλήθος των εισιτηρίων και την

ημερομηνία διεξαγωγής. Σε αυτόν τον πίνακα θα πρέπει να καταχωρήσετε πάνω από 60.000 εγγραφές

Προκειμένου να διευκολυνθείτε σε αυτήν την διαδικασία, μπορείτε να κάνετε την σχεδιαστική παραδοχή ότι ο πίνακας του ιστορικού δεν χρειάζεται να έχει περιορισμούς αναφορικής ακεραιότητας με άλλους πίνακες της ΒΔ.

Ο όγκος των δεδομένων απαιτείται για να αντιμετωπίσετε το Ερώτημα 3.1.3.4

- 3.1.2.4. Απαίτηση: Για τη βάση δεδομένων είναι υπεύθυνοι οι **Διαχειριστές Βάσης Δεδομένων (DBA)**. Για τους Διαχειριστές Βάσης Δεδομένων (DBA) τηρείται η ημερομηνία που ανέλαβαν το ρόλο (start_date), η οποία δεν μπορεί να είναι null. Επίσης υπάρχει end_date ημερομηνία για όσους DBA έχουν φύγει από τη θέση. Μπορεί να υπάρχουν περισσότεροι από ένας Διαχειριστές Βάσης Δεδομένων την ίδια χρονική περίοδο. Οι ενέργειες των Διαχειριστών Βάσης Δεδομένων καταγράφονται σε έναν **πίνακα log** όπως περιγράφεται στο Ερώτημα 3.1.4.1.

3.1.3. Δημιουργία Stored Procedure

Δημιουργήστε τις παρακάτω stored procedure:

- 3.1.3.1. Stored procedure η οποία παίρνει ως όρισμα το id ενός χώρου διεξαγωγής και θα υπολογίζει και θα επιστρέφει το βαθμό του με βάση τον Πίνακα 2 της Απαίτησης 3.1.2.2.
- 3.1.3.2. Stored procedure στην οποία θα δίνονται ως ορίσματα εισόδου το id ενός καλλιτέχνη, μια ημερομηνία και ένας χαρακτήρας ('i', 'c' ή 'a').
- Αν η είσοδος είναι 'i', τότε θα προγραμματίζει μια συναυλία για τον συγκεκριμένο καλλιτέχνη την συγκεκριμένη ημερομηνία, αφού ελέγξει αν υπάρχει ήδη προγραμματισμένη συναυλία. Διαφορετικά θα εμφανίζει κατάλληλα μηνύματα (ήδη προγραμματισμένης ή ακυρωμένης συναυλίας).
 - Αν η είσοδος είναι 'c', τότε αν υπάρχει προγραμματισμένη συναυλία για τον συγκεκριμένο καλλιτέχνη για την συγκεκριμένη ημερομηνία θα ακυρώνεται και θα εμφανίζει ενημερωτικό μήνυμα. Διαφορετικά θα εμφανίζει κατάλληλα μηνύματα (μη ύπαρξης ή ήδη ακυρωμένης συναυλίας).
 - Αν η είσοδος είναι 'a', τότε αν υπάρχει ακυρωμένη συναυλία για τον συγκεκριμένο καλλιτέχνη την συγκεκριμένη ημερομηνία θα ενεργοποιείται και θα εμφανίζει ενημερωτικό μήνυμα. Διαφορετικά θα εμφανίζει κατάλληλα μηνύματα (μη ύπαρξης ή προγραμματισμένης συναυλίας).
- 3.1.3.3. Stored procedure στην οποία δίνεται ως ορίσματα εισόδου το id μιας συναυλίας και απαιτούμενη χωρητικότητα για τη συναυλία και επιστρέφει σε μια μεταβλητή εξόδου το id χώρου διεξαγωγής και σε μια δεύτερη, την χωρητικότητά του.

- Αν δεν υπάρχει συναυλία με το συγκεκριμένο id ή είναι ακυρωμένη, επιστρέφει id χώρου διεξαγωγής NULL και χωρητικότητα 0.
- Αν έχει κλειστεί χώρος διεξαγωγής για τη συγκεκριμένη συναυλία επιστρέφει το id και την χωρητικότητά του χώρου διεξαγωγής.
- Αν δεν έχει κλειστεί χώρος διεξαγωγής για τη συγκεκριμένη συναυλία, τότε αναζητείται χώρος διεξαγωγής που είναι διαθέσιμος και με χωρητικότητα 10% μεγαλύτερης από την απαιτούμενη (Απαίτηση 3.1.2.2). Αν βρεθούν παραπάνω από ένας κατάλληλοι χώροι διεξαγωγής υπολογίζεται ο βαθμός για κάθε έναν από αυτούς με βάση τον Πίνακα 2 της Απαίτησης 3.1.2.2 και επιστρέφεται το id και την χωρητικότητά του χώρου διεξαγωγής με το μεγαλύτερο βαθμό. Αν χρειαστεί μπορείτε να χρησιμοποιήσετε και την stored procedure του ερωτήματος 3.1.3.1.

3.1.3.4. Σε συνέχεια της Απαίτησης 3.1.2.3 και λόγω του μεγάλου πλήθους εγγραφών, θα πρέπει να δημιουργήσετε επιπλέον ευρετήρια (index) στον πίνακα του ιστορικού των συναυλιών. Για κάθε μια από τις δύο επόμενες stored procedures θα επιλέξετε το κατάλληλο ευρετήριο (index), θα το προσθέσετε στον πίνακα του ιστορικού και θα αιτιολογήσετε την επιλογή σας.

α) Stored procedure στην οποία δίνονται ως παράμετροι εισόδου δύο διαφορετικά πλήθη εισιτηρίων (π.χ. 2.000 και 3.000) και εμφανίζει το ονόματα των καλλιτεχνών των οποίων οι συναυλίες είχαν πλήθος εισιτηρίων ανάμεσα σε αυτό το διάστημα.

β) Stored procedure στην οποία δίνεται ως παράμετρος εισόδου το όνομα ενός χώρου διεξαγωγής συναυλιών και εμφανίζει τις ημερομηνίες που έγιναν συναυλίες στον συγκεκριμένο χώρο.

Σκοπός της χρήσης των ευρετηρίων είναι να αντιμετωπίζονται αποδοτικά τα ζητούμενα (α) και (β). Θα καταγράψετε τους χρόνους για κάθε εκτέλεση με και χωρίς τη χρήση ευρετηρίου (index) στον πίνακα, και θα επισυνάψετε κατάλληλο snapshot. Αν το πλήθος των δεδομένων δεν είναι επαρκές για να εμφανιστεί διαφοροποίηση, θα εισαγάγετε περισσότερα δεδομένα στον πίνακα.

Υπόδειξη: Ανατρέξτε στη δημιουργία ευρετηρίων και πινάκων κατακερματισμού της MySQL
(<https://dev.mysql.com/doc/refman/8.0/en/create-index.html>)

3.1.4. Δημιουργία Trigger

Δημιουργήστε τους παρακάτω trigger:

3.1.4.1. Trigger που θα ενημερώνουν το σχετικό πίνακα καταγραφής ενεργειών (log) για κάθε ενέργεια εισαγωγής, ενημέρωσης ή διαγραφής στους πίνακες person, band, album, concert, venue με την ημερομηνία και ώρα και το username του DBA που την εκτέλεσε.

3.1.4.2. Trigger ο οποίος θα αποτρέπει τον προγραμματισμό νέας συναυλίας, αν η ημερομηνία που γίνεται ο προγραμματισμός της συναυλίας απέχει λιγότερο

από πέντε μέρες από την ημερομηνία διεξαγωγής της ή αν ο συγκεκριμένος καλλιτέχνης έχει ήδη τρεις προγραμματισμένες συναυλίες.

- 3.1.4.3. Trigger ο οποίος θα αποτρέπει την ακύρωση μιας συναυλίας, αν η ημερομηνία που γίνεται η ακύρωση απέχει λιγότερο από τρεις μέρες από την ημερομηνία διεξαγωγής της. Επίσης, θα αποτρέπει τον επαναπρογραμματισμό ακυρωμένης συναυλίας αν ο συγκεκριμένος καλλιτέχνης έχει ήδη τρεις προγραμματισμένες συναυλίες.

3.1.5. Δομή εργασίας – Ζητούμενα Α' Μέρος (Σχεδιασμός ΒΔ και SQL)

Θα παραδώσετε **μια αναφορά με το σύνολο της τεκμηρίωσης και το σύνολο του κώδικα SQL**. Στην αναφορά σας θα αναγράφονται Ονοματεπώνυμο, ΑΜ και έτη φοίτησης για όλα τα μέλη της ομάδας. Ακολουθώντας, η αναφορά σας θα αναπτύσσεται σε κεφάλαια με βάση τα ζητούμενα. Αναλυτικά τα ζητούμενα για το Α' μέρος είναι τα εξής:

- Στην αναφορά σας στον **Κεφάλαιο 1** θα περιλαμβάνετε το **σχεσιακό διάγραμμα της συνολικής αναθεωρημένης ΒΔ** (μετά τις επεκτάσεις που προαναφέρθηκαν). Επίσης, θα περιλαμβάνετε περιγραφή όλων των παραδοχών της σχεδίασής σας, των νέων πινάκων και των τροποποιήσεων που κάνατε σε υφιστάμενους πίνακες και σχολιασμό των νέων πινάκων που δημιουργήσατε για να μπορέσετε να υλοποιήσετε τα ζητούμενα. Τέλος, θα **περιλαμβάνει τις εντολές sql για τη δημιουργία της ΒΔ** (create, insert, update, alter).
- Στο **Κεφάλαιο 2** της αναφοράς σας θα δώσετε τον κώδικα και παραδείγματα από την εκτέλεση των **stored procedures** της Ενότητας 3.1.3.
- Στο **Κεφάλαιο 3** της αναφοράς σας θα δώσετε τον κώδικα και παραδείγματα (screenshots) από την εκτέλεση των **triggers** της Ενότητας 3.1.4.

Οδηγίες για την υποβολή θα βρείτε στο Κεφάλαιο 4.

3.2. Μέρος Β: GUI

Η ΒΔ θα χρησιμοποιείται από τους Διαχειριστές Βάσης Δεδομένων (DBA). Για να καλυφθούν οι παρακάτω ανάγκες θα χρειαστεί να δημιουργηθούν διεπαφή/φες. Σημειώνεται ότι η υλοποίηση αυτών των διεπαφών μπορεί να απαιτήσει κάποιες αναθεωρήσεις στη σχεδίαση (π.χ., να προστεθούν κάποια πεδία σε πίνακες).

3.2.1. Δημιουργία Γραφικών Διεπαφών Χρήστη (GUI)

Αναπτύξτε σε Java χρησιμοποιώντας IDE της επιλογής σας (NetBeans, κ.τ.λ.) την εξής λειτουργία: Το σύστημα θα πρέπει να παρέχει μια Γραφική Διεπαφή Χρήστη, από την οποία θα μπορεί να επιλέξει οποιονδήποτε πίνακα της ΒΔ. Μετά την επιλογή του πίνακα, θα εμφανίζονται στην διεπαφή τα περιεχόμενά του και ο χρήστης θα μπορεί να εισάγει, να τροποποιεί και να διαγράφει δεδομένα (στον συγκεκριμένο πίνακα).

3.2.2. Έλεγχος Εγκυρότητας Δεδομένων

Σχεδιάστε την εφαρμογή σας, με τέτοιον τρόπο ώστε, κατά την διαδικασία εισαγωγής ή τροποποίησης που αναφέρεται παραπάνω (3.2.1) και όπου είναι αυτό δυνατό, να περιοριστεί η εισαγωγή δεδομένων από το χρήστη, με σκοπό να αποφεύγονται τα λάθη και να επιταχύνονται οι διαδικασίες (π.χ. να γίνεται επιλογή από μενού ή λίστες). Οι επιλογές που θα είναι διαθέσιμες θα προέρχονται από τα δεδομένα που υπάρχουν στη ΒΔ. Για παράδειγμα, κατά την συνεργασία παραγωγών με δισκογραφικές εταιρίες, να εμφανίζεται σε μια λίστα οι καταχωρισμένοι παραγωγοί και σε μια άλλη οι δισκογραφικές εταιρίες.

3.2.3. Προαιρετικό Ζητούμενο με Μπόνους Βαθμό

Να προσθέσετε λειτουργικότητα (π.χ., κατά την επιλογή χώρων για τη διεξαγωγή μιας συναυλίας να εμφανίζονται όλοι οι διαθέσιμοι χώροι διεξαγωγής συναυλιών με την βαθμολογία τους) ή επιπλέον διεπαφές (π.χ., νέα διεπαφή που να εμφανίζει όλα τα στοιχεία ενός χώρου διεξαγωγής συναυλιών και τις συναυλίες που έχουν προγραμματιστεί σε αυτόν).

3.2.4. Δομή εργασίας – Ζητούμενα Β' μέρος (GUI)

- Στο **Κεφάλαιο 4** της αναφοράς σας, θα περιγράψετε την τεκμηρίωση και τον κώδικα σε Java που αναπτύξατε στο Μέρος Β. Επίσης, θα περιγράψετε ένα σενάριο χρήσης, και θα παραθέσετε screenshots που παρουσιάζουν τη λειτουργικότητα της εφαρμογής σας.
- Τέλος, αν κάνετε το 3.2.3, στο κεφάλαιο 5 να περιγράψετε και να τεκμηριώσετε την λειτουργικότητα που προσθέσατε. Επίσης να παραθέσετε screenshot που παρουσιάζουν τη επιπλέον λειτουργικότητα.

4. Οδηγίες υποβολής

Η υποβολή του πρότζεκτ έχει προθεσμία μέχρι και τις **20/01/2025 και 11.00 προ μεσημβρίας** και θα γίνει υποχρεωτικά σε **ΔΥΟ** εργασίες στο **eclass** με τίτλους.

- Υποβολή Τεκμηρίωσης MONO DOC ή PDF
- Υποβολή ΣΥΝΟΛΟΥ αρχείων ZIP

ΠΡΟΣΟΧΗ! Ανεξάρτητα από το αν υλοποιήσατε μόνο το Μέρος Α ή και τα δύο (Α και Β) θα υποβάλετε **ΥΠΟΧΡΕΩΤΙΚΑ** και στις **ΔΥΟ** εργασίες στο **eclass**.

Παραδοτέα ανά εργασία στο eclass:

I) Στην εργασία Υποβολή Τεκμηρίωσης MONO DOC ή PDF

Υποβολή ενός .doc. ή .pdf αρχείου το οποίο θα έχει **το σύνολο της τεκμηρίωσης και το σύνολο του κώδικα SQL** και θα περάσει από έλεγχο ομοιότητας. Η τεκμηρίωση αφορά και στο Α και στο Β μέρος (αν το έχετε υλοποιήσει). Η δομή της αναφοράς σε κεφάλαια περιγράφεται στις Ενότητες 3.1.5 και 3.2.4. Οι εντολές insert για τις 60.000 εγγραφές δεν χρειάζεται να μπουν στην αναφορά.

II) Στην εργασία Υποβολή ΣΥΝΟΛΟΥ αρχείων ZIP

Υποβολή ενός .zip αρχείου που περιλαμβάνει τα προηγούμενα (.doc. ή .pdf αρχείο το οποίο θα έχει **το σύνολο της τεκμηρίωσης και το σύνολο του κώδικα SQL**), **το σύνολο του κώδικα για τα Μέρη Α και Β** (αν το έχετε υλοποιήσει) και **όλα τα αρχεία που απαιτούνται για την επίδειξη του κώδικα.**

Την υποβολή την κάνει **το ίδιο μέλος της ομάδας ΚΑΙ** στις δύο εργασίες στο **eclass.**

Το όνομα του κάθε αρχείου υποβολής (.doc-.pdf και .zip) θα αποτελείται

- από τα ονόματα και ΑΜ όλων των μελών της ομάδας με την ένδειξη _Α και
- με την ένδειξη _ΑΒ, αν έχετε υλοποιήσει ΚΑΙ το GUI.

Για παράδειγμα:

Ομάδα χωρίς Β' μέρος

υποβολή των αρχείων :

Papadopoulos_10101010_Xristopoulos_10101011_A.pdf και

Papadopoulos_10101010_Xristopoulos_10101011_A.zip στις δυο εργασίες

Ομάδα με Β' μέρος

υποβολή των αρχείων:

Dimitropoulos13131313_Papaioannou12121212_AB.pdf και

Dimitropoulos13131313_Papaioannou12121212_AB.zip στις δυο εργασίες

5. Εργασία σε ομάδες - Αξιολόγηση

Το πρότζεκτ έχει σχεδιαστεί για να εργαστείτε σε ομάδες των 3 ατόμων. Αν θέλετε, μπορείτε να εργαστείτε σε ομάδες 2 ατόμων με διαφοροποίηση στις μονάδες κάθε θέματος σε σχέση με μια ομάδα 3 ατόμων. Η διαφοροποίηση υπάρχει λόγω του όγκου εργασίας για μια ομάδα 2 ατόμων. Επιπλέον, για ομάδα 2 ατόμων, υπάρχει μπόνους αν υλοποιηθεί το ερώτημα 3.2.2, ενώ για ομάδα 3 ατόμων το συγκεκριμένο ερώτημα (3.2.2) είναι υποχρεωτικό και χωρίς μπόνους. Παρακάτω αναλύεται η διαφοροποίηση της βαθμολογίας στις δυο περιπτώσεις ομάδων:

								Σύνολο
Ομάδα	3.1.1	3.1.2	3.1.3	3.1.4	3.2.1	3.2.2	3.2.3 (μπόνους)	
3 ατόμων	1	1	4	1.5	2	0.5	1	7.5+2.5+1=11
2 ατόμων	1	1	4	2	2	1 (μπόνους)	1	8+3+1=12

- Μια ομάδα 3 ατόμων χρειάζεται να απαντήσει σωστά όλα τα θέματα (3.1.1 έως και 3.2.2) για να πετύχει βαθμολογία 10.
- Μια ομάδα 2 ατόμων χρειάζεται να απαντήσει σωστά όλα τα θέματα από 3.1.1 έως και 3.2.1 για να πετύχει βαθμολογία 10, ενώ αν απαντήσει σωστά και το θέμα 3.2.2 πετυχαίνει βαθμολογία 11.
- Η επιπλέον λειτουργικότητα 3.2.3 βαθμολογείται έως 1 μονάδα (μπόνους) και προστίθεται ανεξάρτητα από το μέγεθος της ομάδας.

Το πρότζεκτ είναι υποχρεωτικό και αποτελεί το 50% της τελικής βαθμολογίας σας στο **Εργαστήριο Βάσεων Δεδομένων**. Το άλλο 50% προκύπτει από τις Εργαστηριακές Ασκήσεις 1 έως 5. Δεν είναι απαραίτητο να έχετε σε όλα τα παραπάνω βαθμό μεγαλύτερο από 5. Αρκεί ο συνολικός βαθμός εργαστηρίου να είναι προβιβάσιμος, όπως περιγράφηκε στις αρχικές οδηγίες του Εργαστηρίου. Επομένως, αν η βαθμολογία σας στο πρότζεκτ είναι μεγαλύτερη από 10 βελτιώνει σημαντικά τον τελικό βαθμό του Εργαστηρίου, ειδικά αν κάποιες ασκήσεις σας έχουν χαμηλή βαθμολογία αφού συνυπολογίζεται με αυτές.

6. Εξέταση

Η προφορική εξέταση θα γίνει κατά τη διάρκεια της εξεταστικής σε ημερομηνία που θα ανακοινωθεί. Η προφορική εξέταση είναι ΥΠΟΧΡΕΩΤΙΚΗ για όλα τα μέλη της ομάδας. **Μετά το τέλος της προφορικής εξέτασης θα υπολογιστεί ο τελικός βαθμός πρότζεκτ για κάθε μέλος της ομάδας ατομικά.**

Κατά την προφορική εξέταση, κάθε ομάδα **θα κάνει επίδειξη του πρότζεκτ και της λειτουργίας του σε ΕΝΑ δικό της υπολογιστή** είτε εξ αποστάσεως είτε εκ του σύνεγγυς. **Αν κάποιο μέλος τη ομάδας δεν συμμετάσχει στην εξέταση, βαθμολογείται με μηδέν και δεν περνάει το Εργαστήριο.**

Κατά την εξέταση θα πρέπει η ομάδα να έχει ετοιμαστεί για να κάνει επίδειξη κάθε ερωτήματος και ακολούθως θα γίνει βαθμολόγηση με βάση τον παραπάνω πίνακα βαθμολόγησης. Ο κάθε εξεταζόμενος θα ερωτηθεί πάνω στον κώδικα που επιδεικνύει η ομάδα. Ο βαθμός του κάθε μέλους της ομάδας μπορεί να διαφοροποιείται, ανάλογα με την προφορική του εξέταση.

Η αξιολόγηση προϋποθέτει την παρουσίαση κώδικα SQL (και java για όσους κάνουν το Β' μέρος) που είναι χωρίς λάθη και λειτουργεί.

Θα βγει σχετική ανακοίνωση για τη διαδικασία προφορικής εξέτασης, μετά την υποβολή των πρότζεκτ.