

Esame del 14/09/2022 – Turno B

Si consideri il database **iTunes.sql**, presente nella cartella “database” del progetto base e tratto dalla piattaforma iTunes di Apple. Esso contiene informazioni su artisti (Artist), album musicali (Album) e canzoni (Track), ed è stato estratto a partire dai dati di un utente reale. Il diagramma ER del database è illustrato nella pagina seguente.

Si intende costruire un’applicazione JavaFX che permetta di svolgere le seguenti funzioni:

PUNTO 1

- Permettere all’utente di inserire, nell’apposito campo di testo, una durata **d espressa in secondi**.
- Alla pressione del bottone “Crea Grafo”, si crei un grafo **semplice, non orientato e non pesato**, i cui vertici sono tutti gli album musicali (tabella *Album*) la cui durata media dei brani ad esso appartenenti sia superiore a d .
- Due album $a1$ e $a2$ sono collegati tra loro se almeno una canzone di $a1$ e una canzone di $a2$ sono state inserite da un utente all’interno di una stessa playlist (tabella *PlaylistTrack*).
- Permettere all’utente di selezionare, dall’apposita tendina, un album $a1$ tra quelli presenti nel grafo. Alla pressione del bottone “Analisi Componente”, si stampino:
 - la dimensione della componente connessa a cui appartiene $a1$;
 - il numero complessivo di brani di tutti gli album appartenenti alla componente connessa di $a1$.

possiamo elencare tutte le playlist correlate ad un album, poi controlliamo se la lista ha playlist in comune

PUNTO 2

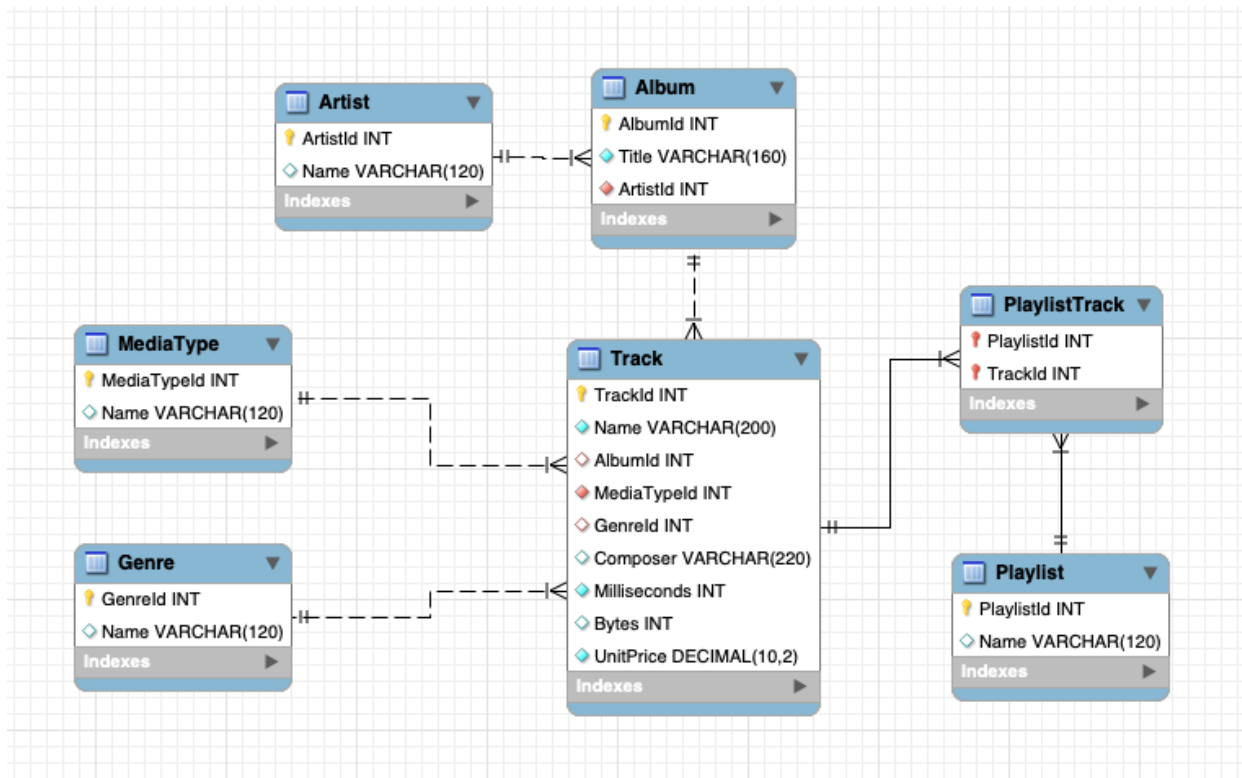
Permettere all’utente di inserire una durata complessiva $dTOT$, espressa in minuti. Alla pressione del bottone “Set di Album”, utilizzare un algoritmo ricorsivo per estrarre un set di album dal grafo che abbia le seguenti caratteristiche:

- include $a1$;
- include solo album appartenenti alla stessa componente connessa di $a1$;
- include il maggior numero possibile di album;
- garantisca che la somma della durata media (definita come media della durata dei brani, come nel punto 1.b) degli album in esso contenuti non sia superiore a $dTOT$.

Nella realizzazione del codice, si lavori a partire dalle classi (Bean e DAO, FXML) e dal database contenuti nel progetto di base. È ovviamente permesso aggiungere o modificare classi e metodi.

Tutti i possibili errori di immissione, validazione dati, accesso al database, ed algoritmici devono essere gestiti, non sono ammesse eccezioni generate dal programma. Nelle pagine seguenti, sono disponibili due esempi di risultati per controllare la propria soluzione.

Le tabelle **Artist**, **Album** e **Track** contengono rispettivamente informazioni su artisti, album musicali e canzoni contenute. Ogni canzone è collegata a un genere (tabella **Genre**) ed è memorizzata con uno specifico formato (tabella **MediaType**). La tabella **Playlist** contiene informazioni sulle playlist salvate dall'utente. La tabella **PlaylistTrack** modella la relazione molti a molti tra playlist e canzoni.



ESEMPI DI RISULTATI PER CONTROLLARE LA PROPRIA SOLUZIONE:

2022-09-14
Esame 14/09/2022 iTunes - Turno B

Durata (d)

Album (a1)

Soglia (dTOT)

Grafo creato!
Vertici : 10
Archi : 45

2022-09-14
Esame 14/09/2022 iTunes - Turno B

Durata (d)

Album (a1)

Soglia (dTOT)

Componente connessa - Heroes, Season 1
Dimensione componente = 10
Album componente = 185.0

2022-09-14
Esame 14/09/2022 iTunes - Turno B

Durata (d)

Album (a1)

Soglia (dTOT)

Grafo creato!
Vertici : 123
Archi : 6171

2022-09-14
Esame 14/09/2022 iTunes - Turno B

Durata (d)

Album (a1)

Soglia (dTOT)

Componente connessa - Allegri: Miserere
Dimensione componente = 111
Album componente = 788.0