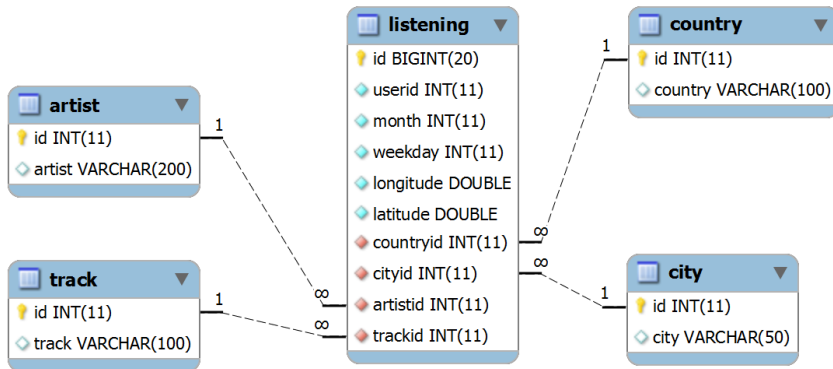


Prova d'esame del 23/06/2015 – Turno A

Si consideri il data-set “**MusicMicro**” messo a disposizione dalla Johannes Kepler University di Linz (Austria). Tale data-set¹ contiene le tracce musicali ascoltate da un insieme di utenti in un periodo lungo quasi un anno, ottenuto analizzando i messaggi condivisi su Twitter. La struttura del database è la seguente:



La tabella ‘listening’ registra ogni ascolto di una canzone (‘trackid’) di un certo cantante (‘artisti’) da parte di un utente (‘userid’).

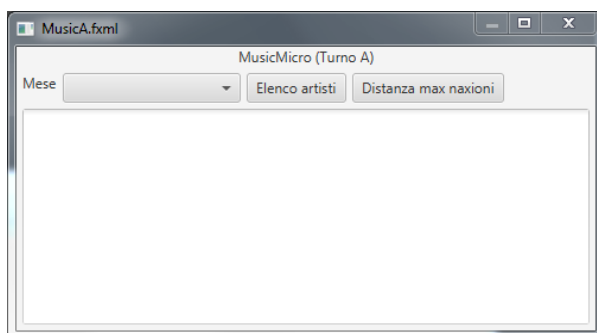
Non è disponibile l’istante esatto dell’ascolto, ma solo il mese (‘month’, 1-12) ed il giorno della settimana (‘weekday’, 0-6).

Le informazioni geografiche (‘longitude’, ‘latitude’, ‘countryid’, ‘cityid’) sono relative al luogo in cui l’utente si trovava al momento dell’ascolto.

Si intende costruire un’applicazione JavaFX che permetta di interrogare tale base dati, e calcolare informazioni a proposito dei trend “social” di ascolto della musica.

L’applicazione dovrà svolgere le seguenti funzioni:

1. Permettere all’utente di selezionare, da una tendina, uno specifico mese, e determinare quali sono stati i 20 artisti più ascoltati in tale mese. Si visualizzi un elenco che comprenda il nome dell’artista ed il numero di ascolti, in ordine di numero di ascolti decrescente. Nota: la tendina deve indicare i nomi dei mesi, non il valore numerico.
2. Dati i 20 artisti sopra selezionati, considerare l’elenco delle nazioni (country) in cui sono stati ascoltati in tal mese. Si costruisca un grafo i cui vertici siano rappresentino tali nazioni, e gli archi abbiano come peso il numero artisti distinti le cui canzoni sono state ascoltate in entrambe le nazioni, nel mese specificato. L’arco non esiste se il numero di utenti è 0. Determinare la massima distanza fra due nodi adiacenti del grafo.



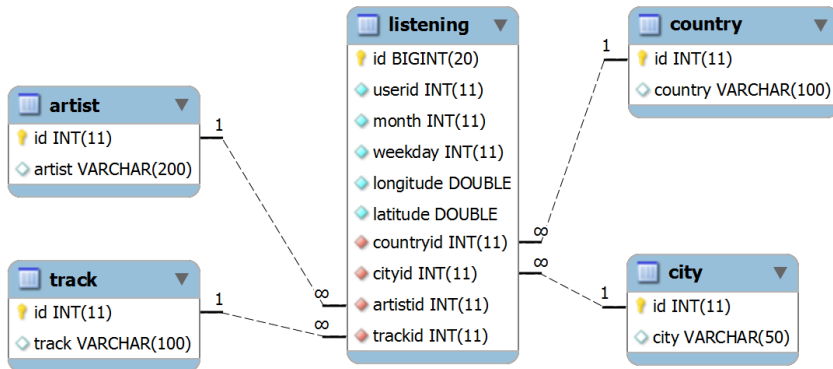
Nella realizzazione del codice, si lavori a partire dalle classi (Bean e DAO, FXML) ed il database contenuti nel progetto scaricabile. È ovviamente permesso aggiungere o modificare classi e metodi.

Tutti i possibili errori di immissione o validazione dati devono essere gestiti, non sono ammesse eccezioni generate dal programma.

¹ <http://www.cp.jku.at/datasets/musicmicro/index.html>

Prova d'esame del 23/06/2015 – Turno B

Si consideri il data-set “**MusicMicro**” messo a disposizione dalla Johannes Kepler University di Linz (Austria). Tale data-set² contiene le tracce musicali ascoltate da un insieme di utenti in un periodo lungo quasi un anno, ottenuto analizzando i messaggi condivisi su Twitter. La struttura del database è la seguente:



La tabella ‘listening’ registra ogni ascolto di una canzone (‘trackid’) di un certo cantante (‘artist’) da parte di un utente (‘userid’).

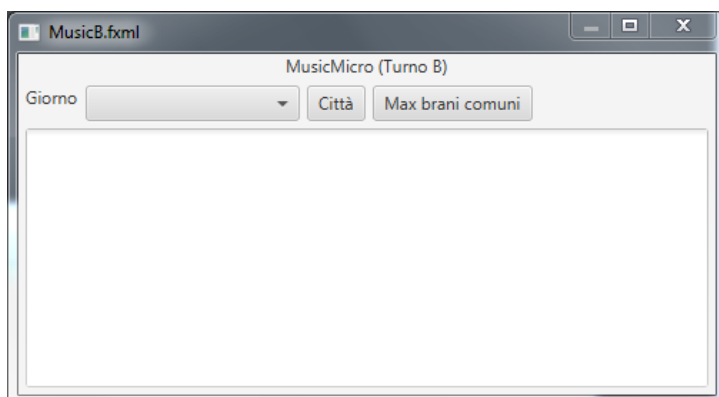
Non è disponibile l’istante esatto dell’ascolto, ma solo il mese (‘month’, 1-12) ed il giorno della settimana (‘weekday’, 0-6).

Le informazioni geografiche (‘longitude’, ‘latitude’, ‘countryid’, ‘cityid’) sono relative al luogo in cui l’utente si trovava al momento dell’ascolto.

Si intende costruire un’applicazione JavaFX che permetta di interrogare tale base dati, e calcolare informazioni a proposito dei trend “social” di ascolto della musica.

L’applicazione dovrà svolgere le seguenti funzioni:

1. Permettere all’utente di selezionare, da una tendina, uno specifico giorno della settimana, e determinare quali sono state le 20 città nelle quali si sono ascoltati più brani. Si visualizzi un elenco che comprenda il nome del brano ed il numero di ascolti, in ordine di numero di ascolti decrescente. Nota: la tendina deve indicare i nomi dei giorni, non il valore numerico.
2. Si costruisca un grafo i cui vertici siano composti dalle 20 città sopra selezionate e che contenga, per ciascuna coppia di città, un arco pesato il cui peso è pari al numero di brani ascoltati comuni tra le due città, nel giorno specificato. Nel caso in cui il numero di brani comuni sia pari a zero, l’arco non deve essere creato. Identificare poi il sottoinsieme di 3 città nelle quali la somma dei pesi degli archi incidenti sia massima.



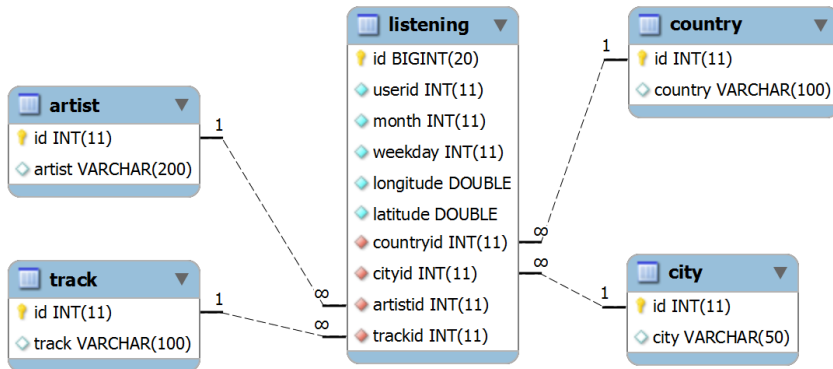
Nella realizzazione del codice, si lavori a partire dalle classi (Bean e DAO, FXML) ed il database contenuti nel progetto scaricabile. È ovviamente permesso aggiungere o modificare classi e metodi.

Tutti i possibili errori di immissione o validazione dati devono essere gestiti, non sono ammesse eccezioni generate dal programma.

² <http://www.cp.jku.at/datasets/musicmicro/index.html>

Prova d'esame del 23/06/2015 – Turno C

Si consideri il data-set “**MusicMicro**” messo a disposizione dalla Johannes Kepler University di Linz (Austria). Tale data-set³ contiene le tracce musicali ascoltate da un insieme di utenti in un periodo lungo quasi un anno, ottenuto analizzando i messaggi condivisi su Twitter. La struttura del database è la seguente:



La tabella ‘listening’ registra ogni ascolto di una canzone (‘trackid’) di un certo cantante (‘artistid’) da parte di un utente (‘userid’).

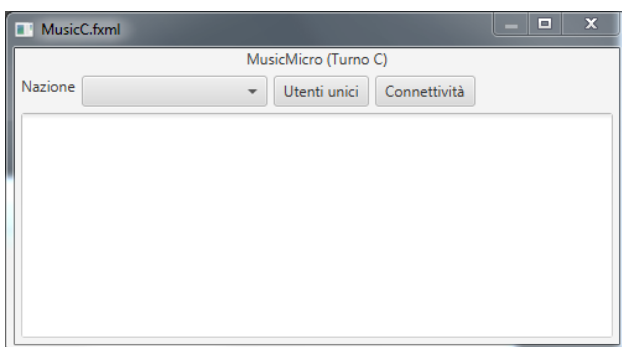
Non è disponibile l’istante esatto dell’ascolto, ma solo il mese (‘month’, 1-12) ed il giorno della settimana (‘weekday’, 0-6).

Le informazioni geografiche (‘longitude’, ‘latitude’, ‘countryid’, ‘cityid’) sono relative al luogo in cui l’utente si trovava al momento dell’ascolto.

Si intende costruire un’applicazione JavaFX che permetta di interrogare tale base dati, e calcolare informazioni a proposito dei trend “social” di ascolto della musica.

L’applicazione dovrà svolgere le seguenti funzioni:

1. Permettere all’utente di selezionare, da una tendina, una specifica nazione (country), e trovare per ciascun artista il numero di utenti unici (‘userid’ distinti) che hanno ascoltato tale artista nella nazione. Si visualizzi un elenco (lungo al massimo 20 righe) che comprenda il nome dell’artista, ed il numero di utenti unici, ordinati per numero decrescente.
2. Si costruisca un grafo i cui vertici siano composti dai 20 artisti sopra selezionati, ed ogni coppia di artisti sia connessa da un arco pesato con il numero di utenti unici che hanno ascoltato entrambi tali brani, nella nazione considerata. Nel caso in cui il numero di utenti comuni sia pari a zero, l’arco non deve essere creato. Determinare se il grafo così creato è connesso, e stampare in ogni caso l’elenco dei vertici di una componente connessa.



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

Tutti i possibili errori di immissione o validazione dati devono essere gestiti, non sono ammesse eccezioni generate dal programma.

³ <http://www.cp.jku.at/datasets/musicmicro/index.html>