

Prova d'esame del 02/07/2018 – Turno B

Si consideri il database “extflightdelays”, contenente informazioni su aeroporti, compagnie aeree, orari di partenza e di arrivo dei voli effettuati negli Stati Uniti durante il 2015. Il database (disponibile su Kaggle all'indirizzo: <https://www.kaggle.com/usdot/flight-delays/data>) è strutturato secondo il diagramma ER della pagina seguente.

Si intende costruire un'applicazione JavaFX che permetta di interrogare tale base dati, e calcolare informazioni a proposito dei voli tra diversi aeroporti. L'applicazione dovrà svolgere le seguenti funzioni:

PUNTO 1

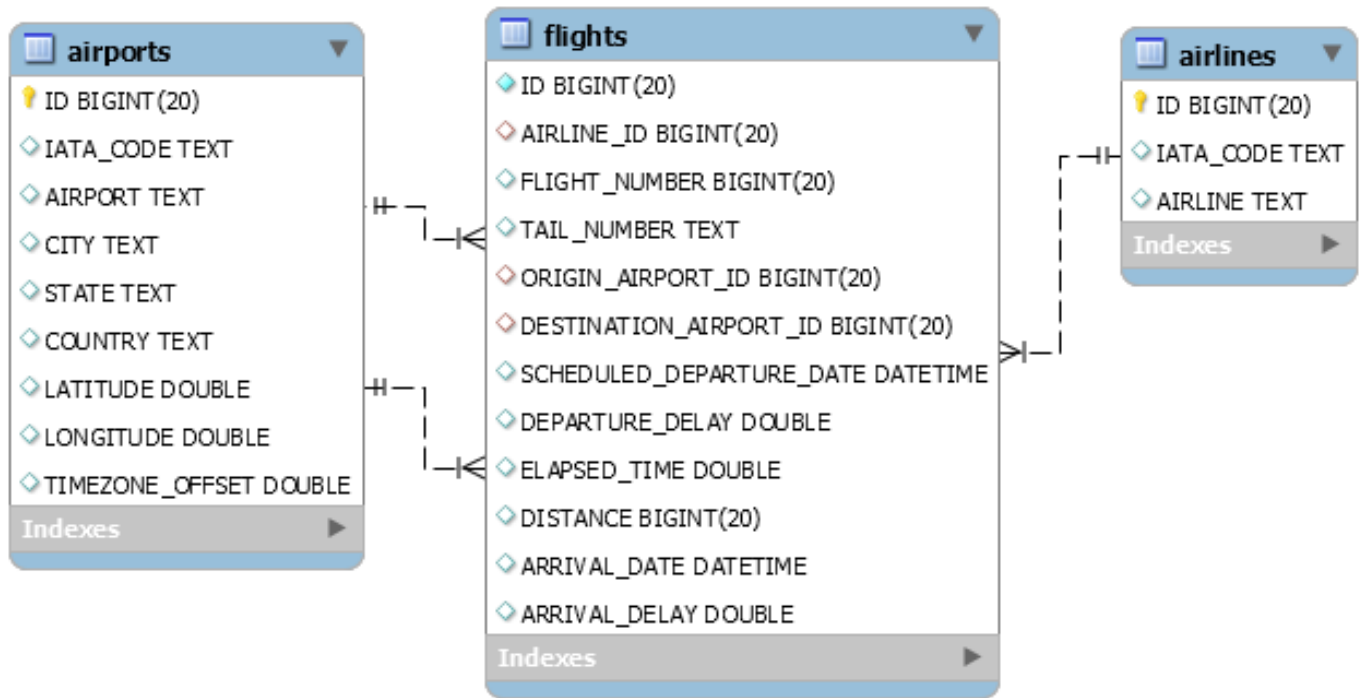
- Permettere all'utente di inserire un numero minimo di voli x , e di selezionare il bottone “Analizza aeroporti”.
- Alla pressione del bottone, creare un grafo che rappresenti gli aeroporti più trafficati. Il grafo deve essere semplice, non orientato e pesato, i vertici devono rappresentare i soli aeroporti la cui somma del numero di voli in partenza e del numero di voli in arrivo sia di almeno x voli, mentre gli archi devono indicare le rotte tra gli aeroporti collegati tra di loro da almeno un volo. Il peso dell'arco rappresenta la durata media di volo tra due aeroporti, calcolata come la media del campo *ELAPSED_TIME* di ciascun volo che li collega (poiché il grafo non è orientato, considerare tutti i voli in entrambe le direzioni: **A->B** e **B->A**).
- Permettere all'utente di selezionare, da un menu a tendina, uno degli aeroporti presenti nel grafo (**a1**).
- Alla pressione del botton “Aeroporti connessi”, stampare l'elenco degli aeroporti adiacenti a quello selezionato, in ordine crescente di durata media di volo.

PUNTO 2

- L'utente inserisce nell'apposita casella di testo un numero totale di ore di volo.
- Alla pressione del bottone “Ottimizza”, sviluppare un algoritmo ricorsivo agente sul grafo creato al punto 1, per cercare l'itinerario di viaggio in cui l'utente possa utilizzare il maggior numero di ore di volo definite in precedenza. Partendo dall'aeroporto **a1**, selezionato nel punto 1c, selezionare un aeroporto **an** di destinazione verso cui vola consumando un numero di ore pari alla durata media di volo da **a1** ad **an**, espressa dal peso dell'arco. In seguito l'utente torna in **a1** consumando lo stesso numero di ore dell'andata, e riparte verso una nuova destinazione. Si supponga che l'utente non visiti mai due volte la stessa località.
- Al termine della ricerca, il programma dovrà stampare l'itinerario, indicando gli aeroporti visitati ed il numero totale di ore di volo consumate.

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.



La tabella *airports* riporta tutti gli aeroporti presenti sul territorio statunitense, la tabella *airlines* tutte le compagnie aeree operanti, mentre la tabella *flights* contiene informazioni sui voli effettuati da una specifica compagnia aerea tra una coppia di aeroporti, riportando le date di partenza, di arrivo ed eventuali ritardi.