M070 - ESAME DI STATO DI ISTITUTO TECNICO INDUSTRIALE

CORSO DI ORDINAMENTO

Indirizzo: INFORMATICA

Tema di: INFORMATICA GENERALE E APPLICAZIONI TECNICO SCIENTIFICHE

Una etichetta discografica indipendente chiede che sia realizzata una base di dati utile all'archiviazione ed alla gestione delle proprie produzioni musicali, anche al fine di renderle eventualmente disponibili su Internet.
Le scelte editoriali dell'etichetta sono relative a diversi generi musicali ciascuno comprendente diversi sottogeneri come, ad esempio:

1. **Genere Popolare**
   Sottogenere
   a) Avanguardia
   b) Blues
   c) Elettronica
   d) Folk
   e) Jazz
   f) New Age
   g) Rap
   h) Rock
   i) ……

2. **Genere classico**
   Sottogenere
   a) Musica da camera
   b) Concerto
   c) Opera
   d) Sinfonia
   e) Musica corale
   f) ……..

L'organizzazione della casa discografica prevede che:

- un artista o un gruppo musicale o un'orchestra possano registrare musica di generi diversi o, nell'ambito dello stesso genere, di diversi sottogeneri;
- possano essere rilasciate più edizioni di uno stesso album ciascuna con una diversa copertina;
- un artista o un gruppo o un'orchestra possano non essere più attivi sul mercato discografico.

Si chiede, in particolare, che si possano archiviare i seguenti dati:

a) il nome dell'artista e/o del gruppo musicale e/o dell'orchestra;
b) il periodo (in anni) di attività sul mercato discografico;
c) il titolo dell'album;
d) il nome del direttore d'orchestra;
e) la data (giorno - mese – anno) di pubblicazione della/e edizione/i;
f) il titolo dei brani musicali presenti nell'album;
g) la durata (minuti – secondi) dell'album;
h) la durata (minuti – secondi) dei singoli brani dell'album;
i) il nome dell'autore o degli autori dei brani;

edutecnica.altervista.org

M070 - ESAME DI STATO DI ISTITUTO TECNICO INDUSTRIALE

CORSO DI ORDINAMENTO

Indirizzo: INFORMATICA

Tema di: INFORMATICA GENERALE E APPLICAZIONI TECNICO SCIENTIFICHE

j) brevi notizie biografiche dell'artista e/o del gruppo musicale e/o del direttore d'orchestra;

k) l'immagine della copertina dell'album.

Il candidato, fatte le opportune ipotesi aggiuntive, progetti una base di dati utile alla realizzazione dell'archivio richiesto dall'etichetta indipendente, fornendo:

1. uno schema concettuale della base di dati;
2. uno schema logico della base di dati;
3. la definizione delle relazioni della base di dati in linguaggio SQL;

ed inoltre:

4. implementi in linguaggio SQL la seguente interrogazione:
   - Dato il titolo di un brano, quali sono le date di pubblicazione, la durata, l'artista e/o il gruppo musicale e/o l'orchestra con il relativo periodo di attività nonché gli autori e l'album di appartenenza.

Opzionalmente, supponendo che la base di dati sia disponibile su di un server on line, il candidato implementi una directory per il web che consenta, a partire da una pagina con l'elenco dei generi musicali, di ottenere:

- con un click sul nome di un genere musicale, una pagina con l'elenco dei nomi dei sottogeneri musicali;
- con un click sul nome di un sottogenere musicale, una pagina con l'elenco dei nomi degli artisti;
- con un click sul nome di un artista, l'elenco degli album di quell'artista con le immagini di copertina, la durata, l'anno o gli anni di edizione, le notizie bibliografiche sull'artista ed il periodo di attività.

edutecnica.altervista.org

SOLUZIONE AL TEMA DI INFORMATICA (PROGETTO ABACUS)

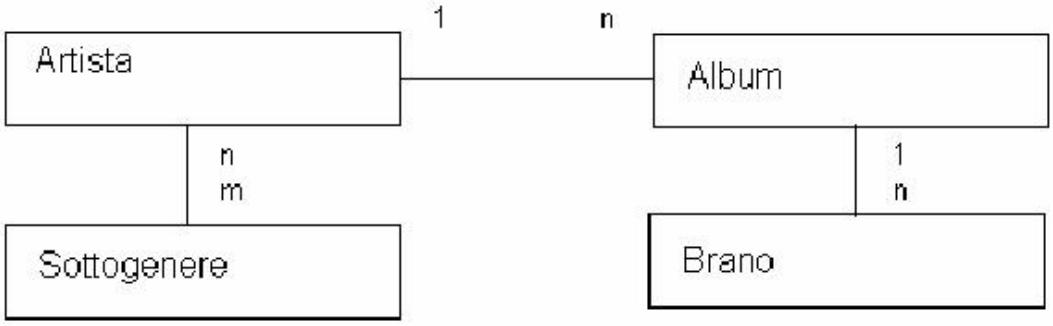
Punto 1

Da una prima analisi del testo appare evidente che per la costruzione della base di dati si possono individuare le seguenti tre entità fondamentali:

1. GENERE
   che definisce le caratteristiche che possono avere le produzioni musicali
2. ARTISTA
   che caratterizza il cantante, gruppo musicale o orchestra che esegue i brani
3. ALBUM
   che caratterizza il singolo album musicale

Osservando però che i due insiemi di sottogeneri sono tra loro disgiunti è possibile sostituire direttamente l'entità GENERE con l'entità SOTTOGENERE definendo tra le proprietà il GENERE di appartenenza. Inoltre, visto che per ogni brano devono essere registrati altri dati relativi alla durata e agli autori risulta conveniente creare una quarta entità denominata BRANO.
Anche se in linea teorica un album potrebbe essere stato inciso da più artisti il testo evidenzia in modo chiaro ed esplicito che ad ogni album è associato un unico cantante e/o un unico gruppo musicale e/o un'unica orchestra. In modo analogo il quesito presente nella parte 4 della traccia evidenzia che un brano musicale più essere presente in un unico album.

In base a tutto ciò possiamo fornire lo schema E/R della nostra base di dati:



Analisi delle entità.

Osserviamo innanzitutto che per tutte le entità conviene definire tra le proprietà un codice univoco di identificazione da usare come chiave primaria nella definizione delle relazioni.
L'entità SOTTOGENERE risulta caratterizzata dalle seguenti proprietà: CODICE (chiave primaria), Nome, Genere di appartenenza.

L'entità ARTISTA risulta caratterizzata dalle seguenti proprietà: CODICE (chiave primaria), Nome, Data inizio attività, Data fine attività (eventualmente mancante), Note biografiche o storiche.

L'entità ALBUM risulta caratterizzata dalle seguenti proprietà: CODICE (chiave primaria), Codice cantante, Codice gruppo, Codice orchestra, Codice direttore d'orchestra, Titolo, Date di pubblicazione delle singole edizioni, Durata, Immagine copertina.

L'entità BRANO risulta caratterizzata dalle seguenti proprietà: CODICE (chiave primaria), Codice album, Titolo, Durata, Nomi degli autori.

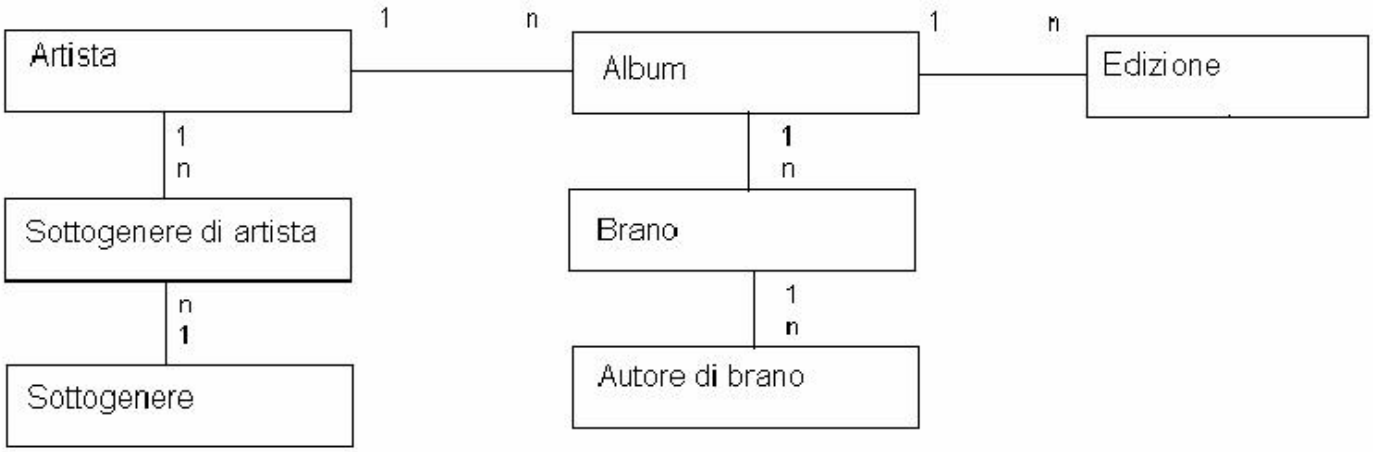
Prima di passare alla stesura definitiva del modello relazionale conviene procedere ad un raffinamento del modello concettuale in modo da eliminare le relazioni n a m e i campi ripetuti. In questo modo saremo sicuri che il nostro modello logico sia espresso almeno in prima forma normale (1NF).

Per fare questo inseriamo le seguenti nuove entità:

EDIZIONE, caratterizzata da CODICE (chiave primaria), Data, Codice album

AUTORE DI BRANO, caratterizzata da NOME e CODICE BRANO che formano la chiave primaria.

SOTTOGENERE DI ARTISTA, caratterizzata da CODICE ARTISTA e CODICE SOTTOGENERE, che formano la chiave primaria.
Il modello concettuale a questo punto può essere descritto come segue:



Punto 2

Utilizzando come modello logico il modello relazionale sarà sufficiente implementare le tabelle relative alle singole entità.

Per quanto riguarda le note biografiche è possibile, qualora il gestore di database lo consenta, definire un campo testo di dimensioni elevate oppure inserire un link a un file testo in cui le stesse sono definite.

In modo analogo la gestione delle immagini può essere fatta tramite link ai file specifici oppure attraverso la gestione di oggetti OLE correlati.

A titolo esemplificativo presentiamo la definizione delle tabelle in ambiente Microsoft Access, gestore diffuso nelle scuole.

ARTISTA : Tabella

Nome campo	Tipo dati
Codice	Numerico
Nome	Testo
Data inizio	Data/ora
Data fine	Data/ora
Biografia	Memo

SOTTOGENERE : Tabella

Nome campo	Tipo dati
Codice	Numerico
Nome	Testo
Genere	Testo

ALBUM : Tabella

Nome campo	Tipo dati	Descrizione
Codice	Numerico	
Codice cantante	Numerico	opzionale (chiave esterna)
Codice gruppo	Numerico	opzionale (chiave esterna)
Codice orchestra	Numerico	opzionale (chiave esterna)
Codice direttore	Numerico	opzionale (chiave esterna)
Titolo	Testo	
Durata	Data/ora	formato minuti:secondi
Copertina	Oggetto OLE	riferimento adimmagine JPG (chiave esterna)

BRANO : Tabella

Nome campo	Tipo dati	Descrizione
Codice	Numerico	
Codice album	Numerico	chiave esterna
Titolo	Testo	
Durata	Data/ora	formato minuti:secondi

EDIZIONE : Tabella

Nome campo	Tipo dati
Numero	Numerico
Codice album	Numerico
Data	Data/ora

AUTORE DI BRANO : Tabella

Nome campo	Tipo dati
Nome	Testo
Codice brano	Numerico

SOTTOGENERE DI ARTISTA : Tabella

Nome campo	Tipo dati
Codice artista	Numerico
Codice sottogenere	Numerico

Punto 3

Prima di presentare una soluzione al quesito vorremmo sottolineare che questo punto della traccia risulta essenzialmente nozionistico e non rispecchia le modalità di lavoro degli studenti.
Infatti oggi la quasi totalità dei gestori di database relazionali consente di definire le tabelle e le relazioni in modo grafico senza ricorrere all'uso del linguaggio SQL.
Inoltre, dato che in un modello relazionale le relazioni sono definite direttamente tramite le tabelle e la flessibilità di SQL viene utilizzata proprio per definire le correlazioni tra i dati solo al momento di sviluppare le query mediante le clausole JOIN in modo da poter costruire viste diverse e separate dello stesso schema concettuale, appare quanto meno curioso che venga richiesto di definire tali relazioni in modo del dutto distaccato dalle interrogazioni riducendo così la soluzione ad un mero elenco di CREATE TABLE.
In ogni caso, qualunque sia la soluzione prescelta occorre prestare attenzione ad inserire nella definizione delle tabelle tutte le colonne che consentono di definire le chiavi esterne di correlazione, come da noi fatto nell'esemplificazione di cui al punto precedente.

Punto 4

La soluzione alla traccia può essere fornita in diversi modi, di cui il più semplice è sicuramente quella che fa riferimento a una serie di query correlate che vengono richiamate in sequenza:

Query 1: individua la durata del brano e il codice dell'album di appartenenza

```
SELECT BRANO.Codice, BRANO.[Codice album], BRANO.Titolo, BRANO.Durata
FROM BRANO
WHERE (((BRANO.Titolo)=[inserire il titolo del brano]));
```

Query 2: aggiunge i dati relativi all'album

```
SELECT Query1.Codice, Query1.[Codice album], Query1.Titolo AS Query1_Titolo, Query1.Durata, ALBUM.[Codice cantante], ALBUM.[Codice gruppo], ALBUM.[Codice orchestra], ALBUM.[Codice direttore], ALBUM.Titolo AS ALBUM_Titolo
FROM ALBUM INNER JOIN Query1 ON ALBUM.Codice = Query1.[Codice album];
```

Query 3: aggiunge i dati relativi al cantante

```
SELECT [Query 2].Codice, [Query 2].[Codice album], [Query 2].Query1_Titolo, [Query 2].Durata, [Query 2].[Codice cantante], [Query 2].[Codice gruppo], [Query 2].[Codice orchestra], [Query 2].[Codice direttore], [Query 2].ALBUM_Titolo, ARTISTA.Nome, ARTISTA.[Data inizio], ARTISTA.[Data fine]
FROM ARTISTA INNER JOIN [Query 2] ON ARTISTA.Codice = [Query 2].[Codice cantante];
```

Query 4: aggiunge i dati relativi all'eventuale gruppo

```
SELECT [Query 3].[Codice] AS [Query 3_Codice], [Query 3].[Codice album], [Query 3].[Query1_Titolo], [Query 3].[Durata], [Query 3].[Codice cantante], [Query 3].[Codice gruppo], [Query 3].[Codice orchestra], [Query 3].[Codice direttore], [Query 3].[ALBUM_Titolo], [Query 3].[Nome] AS [Query 3_Nome], [Query 3].[Data inizio] AS [Query 3_Data inizio], [Query 3].[Data fine] AS [Query 3_Data fine], [ARTISTA].[Codice] AS ARTISTA_Codice, [ARTISTA].[Nome] AS ARTISTA_Nome, [ARTISTA].[Data inizio] AS [ARTISTA_Data inizio], [ARTISTA].[Data fine] AS [ARTISTA_Data fine]
FROM ARTISTA INNER JOIN [Query 3] ON [ARTISTA].[Codice]=[Query 3].[Codice gruppo];
```

Query 5: aggiunge i dati relativi all'eventuale orchestra

```
SELECT [Query 4].[Query 3_Codice], [Query 4].[Codice album], [Query 4].[Query1_Titolo], [Query 4].[Durata], [Query 4].[Codice cantante], [Query 4].[Codice gruppo], [Query 4].[Codice orchestra], [Query 4].[Codice direttore], [Query 4].[ALBUM_Titolo], [Query 4].[Query 3_Nome], [Query 4].[Query 3_Data inizio], [Query 4].[Query 3_Data fine], [Query 4].[ARTISTA_Codice], [Query 4].[ARTISTA_Nome], [Query 4].[ARTISTA_Data inizio], [Query 4].[ARTISTA_Data fine], [ARTISTA].[Codice], [ARTISTA].[Nome]
FROM [Query 4] INNER JOIN ARTISTA ON [Query 4].[Codice orchestra]=[ARTISTA].[Codice];
```

Query 6: aggiunge i dati relativi agli autori dei brani

SELECT [Query 5].[Query 3_Codice], [Query 5].[Codice album], [Query 5].Query1_Titolo, [Query 5].Durata, [Query 5].[Codice cantante], [Query 5].[Codice gruppo], [Query 5].[Codice orchestra], [Query 5].[Codice direttore], [Query 5].ALBUM_Titolo, [Query 5].[Query 3_Nome], [Query 5].[Query 3_Data inizio], [Query 5].[Query 3_Data fine], [Query 5].ARTISTA_Codice, [Query 5].ARTISTA_Nome, [Query 5].[ARTISTA_Data inizio], [Query 5].[ARTISTA_Data fine], [Query 5].Codice, [Query 5].Nome AS [Query 5_Nome], [AUTORE DI BRANO].Nome AS [AUTORE DI BRANO_Nome]
FROM [Query 5] INNER JOIN [AUTORE DI BRANO] ON [Query 5].[Query 3_Codice] = [AUTORE DI BRANO].[Codice brano];

Query 7: aggiunge i dati mancanti

SELECT [Query 6].[Query 3_Codice], [Query 6].[Codice album], [Query 6].Query1_Titolo, [Query 6].Durata, [Query 6].[Codice cantante], [Query 6].[Codice gruppo], [Query 6].[Codice orchestra], [Query 6].[Codice direttore], [Query 6].ALBUM_Titolo, [Query 6].[Query 3_Nome], [Query 6].[Query 3_Data inizio], [Query 6].[Query 3_Data fine], [Query 6].ARTISTA_Codice, [Query 6].ARTISTA_Nome, [Query 6].[ARTISTA_Data inizio], [Query 6].[ARTISTA_Data fine], [Query 6].Codice, [Query 6].[Query 5_Nome], [Query 6].[AUTORE DI BRANO_Nome], EDIZIONE.Numero, EDIZIONE.Data
FROM (ALBUM INNER JOIN [Query 6] ON ALBUM.Codice = [Query 6].[Codice album]) INNER JOIN EDIZIONE ON ALBUM.Codice = EDIZIONE.[Codice album];

Query 8: visualizza i dati nel formato richiesto dalla traccia

SELECT [Query 7].[Data] AS Data_pubblicazione, [Query 7].[Durata] AS Durata, [Query 7].[Query 3_Nome] AS Cantante, [Query 7].[Query 3_Data inizio] AS cantante_inizio, [Query 7].[Query 3_Data fine] AS Cantante_fine, [Query 7].[ARTISTA_Nome] AS gruppo, [Query 7].[ARTISTA_Data inizio] AS gruppo_inizio, [Query 7].[ARTISTA_Data fine] AS gruppo_fine, [Query 7].[Query 5_Nome] AS Orchestra, [Query 7].[AUTORE DI BRANO_Nome] AS Autore, [Query 7].[ALBUM_Titolo] AS Titolo_album
FROM [Query 7];