



Progettazione e sviluppo di una base di dati relazionale per la gestione di una libreria musicale

Davide Maresca
N86003819

Giuseppe Scotto Di Fasano
N86003661

June 2022

Indice

1	DESCRIZIONE DEL PROGETTO	4
1.1	Descrizione e analisi del progetto	4
1.2	Requisiti identificati	4
2	PROGETTAZIONE CONCETTUALE	5
2.1	Introduzione	5
2.2	Class Diagram	5
2.3	Analisi della ristrutturazione del class diagram	6
2.3.1	Analisi delle ridondanze	6
2.3.2	Analisi degli identificativi	6
2.3.3	Rimozione degli attributi multipli	6
2.3.4	Rimozione degli attributi composti	6
2.3.5	Partizione/Accorpamento delle associazioni	6
2.3.6	Rimozione delle gerarchie	7
2.4	Class diagram Ristrutturato	7
2.5	Dizionario delle classi	8
2.6	Dizionario delle associazioni	10
2.7	Dizionario dei vincoli	11
3	SCHEMA LOGICO	12
4	PROGETTAZIONE FISICA	13
4.1	Definizione dei domini	13
4.1.1	Sesso persona	13
4.1.2	Fascia oraria	13
4.2	Definizione delle tabelle	14
4.2.1	Utente	14
4.2.2	Traccia	15
4.2.3	Brani preferiti	15
4.2.4	Album	16
4.2.5	Artista	16
4.2.6	Ascoltatore	16
4.2.7	FormatoTraccia	17
4.2.8	ContieneTracce	17
4.3	Funzioni e trigger per le automazioni	18
4.3.1	Modifica dati dell'album con l'inserimento	18
4.3.2	Modifica dati dell'album con la cancellazione	19
4.3.3	Creazione Brani Preferiti automatica	19
4.3.4	Modifica dati di Brani Preferiti con l'inserimento	20
4.3.5	Modifica dati di Brani Preferiti con la cancellazione	21
4.3.6	Incremento automatico degl numero di ascolti totali	22
4.4	Funzioni e trigger per i vincoli	23
4.4.1	Controllo artista cover diverso dalla traccia originale	23

4.4.2	Controllo artista remastering diverso dalla traccia originale	24
4.4.3	Controllo data pubblicazione del remastering	25
4.4.4	Controllo data pubblicazione cover	26
4.4.5	Controllo della cover all'interno di un album	27
4.4.6	Controllo unico artista in un album	28

5 MANUALE D'USO 29

5.1	Creazione utenti	29
5.1.1	UtenteBase	29
5.1.2	UtenteAdmin	29
5.2	Popolamento	30
5.2.1	Utente	30
5.2.2	Traccia	31
5.2.3	Album	38
5.2.4	Artista_t	39
5.2.5	Ascoltatore	40
5.2.6	FormatoTraccia	43
5.2.7	ContieneTracce	46

DESCRIZIONE DEL PROGETTO

1.1 Descrizione e analisi del progetto

In questo progetto si sviluppa una base di dati utile per la gestione di una **libreria musicale** per un gruppo di utenti. La base conterrà i dati di **utenti, artisti, album e tracce**. Ogni utente avrà accesso ad una propria lista di **brani preferiti** (tipo playlist) che può gestire aggiungendo o rimuovendo delle tracce, inoltre è presente un **admin** che avrà accesso alla gestione del database, con una password dedicata, nel quale può monitorare gli ascolti dei vari utenti e gestire completamente la base di dati, aggiungendo, eliminando o modificando i dati presenti all'interno. Le **tracce** sono suddivise in **tre categorie, originali, remastering** (creata dallo stesso autore della traccia originale) e **cover** (creata da un autore diverso della traccia originale). Gli artisti producono sia tracce originali che cover che remastering. Gli album, vengono creati dagli artisti, contengono le tracce (in un album ci possono essere insieme sia tracce originali che remastering).

1.2 Requisiti identificati

All'interno della traccia proposta dal professore abbiamo individuato le seguenti entità con relative associazioni:

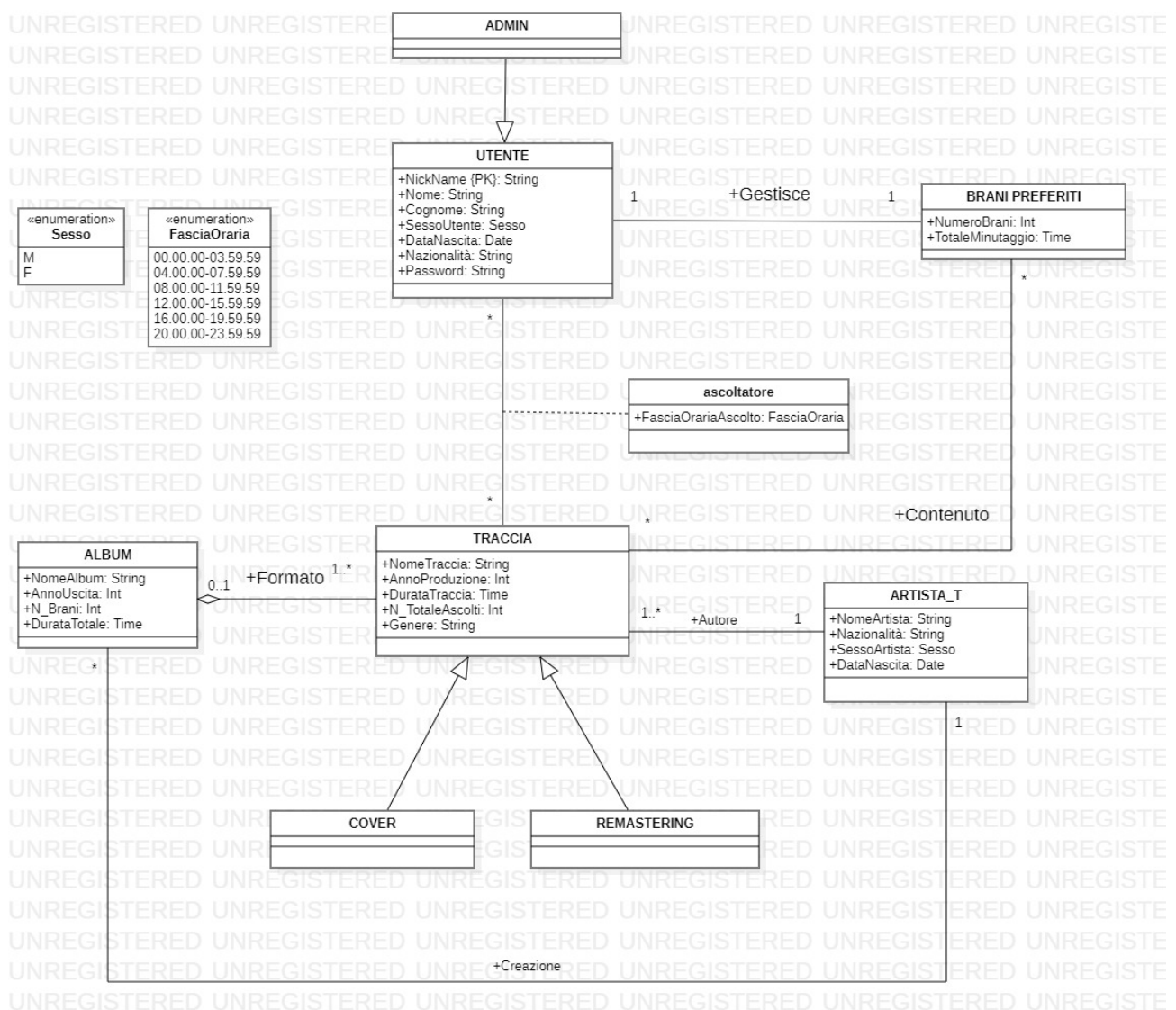
- **Utente**: indica un'entità, l'utente accede alla libreria musicale per ascoltare canzoni. L'utente è in relazione con Brani Preferiti (playlist), ogni utente ha accesso alla propria playlist dove potrà salvare delle nuove tracce. Inoltre è in relazione con traccia, ogni ascolto della singola traccia da parte dell'utente viene registrato e salvato all'interno del database, includendo anche la fascia oraria di ascolto.
- **Tracce**: indica un'entità, è la base della libreria musicale. La traccia può essere di tipo, originale, remastering e cover. La traccia, oltre che con la relazione utente, ha anche diverse relazioni: 1) Con album, la traccia può far parte di un album dedicato (la traccia può essere pubblicata senza album). 2) Con artista, le tracce vengono prodotte da un unico artista senza featuring (due o più artisti per singola traccia). 3) Con Brani Preferiti, le tracce possono essere salvate all'interno di una playlist personale.
- **Artista**: indica un'entità, produce le tracce. L'artista oltre ad avere l'associazione con Tracce, ha anche un'associazione con Album per conoscere chi ha inciso quel determinato album, l'album può essere inciso da un unico artista.
- **Album**: indica un'entità, contiene una lista di tracce l'album può contenere una lista di tracce prodotte dallo stesso artista che ha inciso l'album

PROGETTAZIONE CONCETTUALE

2.1 Introduzione

In questo capitolo introduciamo la progettazione concettuale, nel quale verranno identificati i concetti più importanti. Per la rappresentazione di questi concetti usiamo il **Class Diagram UML**, che ci permette di dare una vista generale del problema con entità, dati delle entità, associazioni, molteplicità delle associazioni tra due entità e in aggiunta delle enumerazioni.

2.2 Class Diagram



2.3 Analisi della ristrutturazione del class diagram

Il class diagram in questione verrà successivamente ristrutturato al fine di **migliorare l'efficienza dell'implementazione**. Verranno eliminati attributi strutturati, attributi multipli e gerarchie di specializzazione.

2.3.1 Analisi delle ridondanze

Non sono presenti ridondanze all'interno del diagramma. Tuttavia è possibile ricavare l'età dalle classi che contengono l'attributo 'DataNascita', per eliminare questa derivazione possiamo aggiungere un attributo 'Età'

2.3.2 Analisi degli identificativi

Tranne che per la classe Utente, che viene identificata dal 'NickName' precedentemente alla ristrutturazione, tutte le altre classi non dispongono di chiavi primarie. Quindi per tutte queste classi si è scelto di impiegare sulla base della ristrutturazione chiavi "tecniche". Tali chiavi tecniche saranno identificativi numerici che permetteranno di distinguere con maggiore facilità le istanze. Per convenzione queste chiavi "tecniche" inizieranno con il prefisso ID.

2.3.3 Rimozione degli attributi multipli

Non sono presenti attributi multipli all'interno delle entità

2.3.4 Rimozione degli attributi composti

Non sono presenti attributi composti

2.3.5 Partizione/Accorpamento delle associazioni

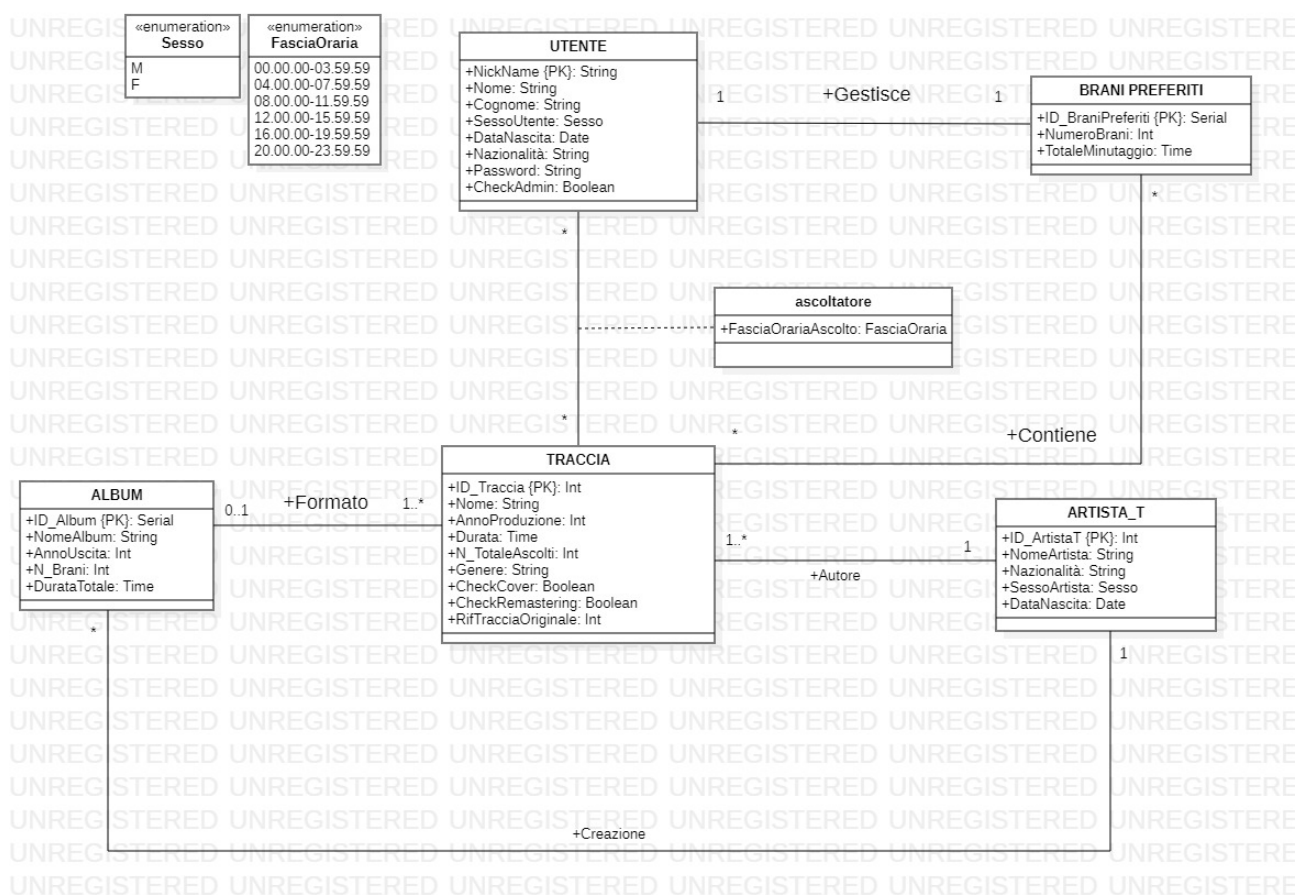
Non ci sono accorpamenti delle associazioni

2.3.6 Rimozione delle gerarchie

Vengono rimosse le seguenti gerarchie:

- **Admin**: questa entità viene incorporata all'interno dell'entità Utente e per farlo abbiamo inserito un ulteriore attributo all'interno di Utente **'checkAdmin'** che identifica l'istanza di un Admin
- **Cover e Remastering**: queste due entità vengono incorporate all'interno dell'entità Traccia, per farlo abbiamo inserito due attributi in Traccia **'CheckCover'** e **'CheckRemastering'** che identificano le istanze di ogni singola cover o remastering, in più un attributo **'RifTracciaOriginale'** per identificare a quale traccia originale fa riferimento la cover oppure il remastering.

2.4 Class diagram Ristrutturato



2.5 Dizionario delle classi

Table 1: Dizionario delle classi.

CLASSE	DESCRIZIONE	ATTRIBUTI
UTENTE	Descrittore di ciascun utente che fa uso della libreria musicale	NickNamePK (String) : Indica il nome univoco dell'utente per accedere alla libreria. Nome (String) : Indica il nome dell'utente. Cognome (String) : Indica il cognome dell'utente. SessoUtente (Sesso) : Indica il sesso dell'utente DataNascita (Date) : Indica la data di nascita dell'utente Nazionalità (String) : Indica la nazionalità dell'utente Password (String) : Indica la password per accedere alla libreria musicale CheckAdmin [0...1] (Boolean) : Indica se l'utente è un admin oppure no
BRANI PREFERITI	Descrittore della lista di brani preferiti di ogni utente	ID_BraniPreferiti PK (Int) : Identifica univocamente la lista dei brani preferiti per ogni utente NumeroBrani (Int) : Indica quanti brani sono presenti all'interno dei brani preferiti TotaleMinutaggio (Time) : Indica il totale tempo di ascolto dipendente dalla durata di ogni singola traccia
TRACCIA	Descrittore di ogni singola traccia	ID_Traccia PK (Int) : Identifica univocamente ogni traccia NomeTraccia (String) : Indica il nome di ogni traccia AnnoProduzione (Int) : Indica l'anno di produzione della traccia DurataTraccia (Time) : Indica la durata della traccia N_TotaleAscolti (Int) : Indica il numero totale di ascolti della traccia Genere (String) : Indica il genere della traccia CheckCover [0...1] (Boolean) : Indica se la traccia è una cover oppure no CheckRemastering [0...1] (Boolean) : Indica se la traccia è un remastering oppure no RifTracciaOriginale (Int) : Indica il riferimento a quale traccia è associata la cover oppure il remastering

CLASSE	DESCRIZIONE	ATTRIBUTI
ARTISTA_T	Descrittore di ogni artista della traccia originale	ID_ArtistaT PK (Int) : Identifica univocamente l'artista NomeArtista (String) : Indica il nome o lo pseudonimo dell'artista Nazionalità (String) : Indica la nazionalità dell'artista SessoArtista (Sesso) : Indica il sesso dell'artista
ALBUM	Descrittore dell'album, che può contenere tracce originali, cover e remastering	ID_Album PK (Int) : Identifica univocamente l'album NomeAlbum (String) : Indica il nome dell'album AnnoUscita (Int) : Indica l'anno di uscita dell'album N_Brani (Int) : Indica il numero di brani presenti all'interno dell'album DurataTotale (Time) : Indica la durata totale dell'album
ASCOLTATORE	Descrittore dell'ascolto di una traccia	NumeroAscoltiUtente (Int) : indica il numero di ascolti della traccia da parte del singolo utente FasciaOrariaAscolto (FasciaOraria) : Indica la fascia oraria in cui un utente ascolta la traccia

2.6 Dizionario delle associazioni

Table 2: Dizionario delle associazioni.

NOME	DESCRIZIONE	CLASSI COINVOLTE
ASCOLTATORE	Esprime l'istanza di ascolto di una traccia da parte di un utente	Utente [0...*] ruolo (ascolta): indica l'utente che ascolta la traccia Traccia [0...*] ruolo (è ascoltata): indica quale traccia è ascoltata dall'utente
GESTIONE	Esprime la gestione della lista di brani preferiti da parte di un utente	Utente [1] ruolo (Gestisce): indica l'utente che gestisce la propria lista di brani preferiti Brani Preferiti [1] ruolo (è gestita): indica la lista di brani preferiti che viene gestita dall'utente
CONTIENETRACCE	Esprime l'aggiunta di una traccia nella lista dei brani preferiti	Brani Preferiti [0...*] ruolo (contiene): indica che la lista dei brani preferiti contiene delle tracce Traccia [0...*] ruolo (è contenuta): indica che la traccia è contenuta nella lista dei brani preferiti
AUTORETRACCE	Esprime l'appartenenza della traccia da parte dell'artista	Traccia [1...*] ruolo (scritta): indica che la traccia viene scritta da un artista Artista_T [1] ruolo (autore): indica che l'artista è autore della traccia
FORMATOTRACCIA	Esprime l'appartenenza di una traccia all'album	Album [0...1] ruolo (è formato): indica che l'album è formato da delle tracce Traccia [1...*] ruolo (forma): indica che le tracce formano un album
CREAZIONE	Esprime l'appartenenza di un artista all'album	Artista[1] ruolo (crea): indica che l'album viene creato da un artista Album[*] ruolo (è creato): indica che l'artista può creare un album

2.7 Dizionario dei vincoli

Table 3: Dizionario dei vincoli.

NOME VINCOLO	DESCRIZIONE
SESSO_PERSONA	Questo vincolo permette all'utente di selezionare sesso maschile oppure femminile
CHECKDATA	Questo vincolo controlla la data di nascita dell'utente che si registra alla libreria musicale, nel caso in cui dovesse essere maggiore della data attuale, da errore
CHECKDATA18	Questo è un ulteriore vincolo che controlla la data di nascita dell'utente che si registra alla libreria musicale, abbiamo deciso di permettere la registrazione solo a utenti maggiorenni
CHECKDATAALBUM	Questo vincolo controlla la data di pubblicazione dell'album, se viene pubblicato dopo la data attuale la base di dati da errore
CHECKDATATRACCIA	Questo vincolo controlla la data di pubblicazione della traccia, se viene pubblicata dopo la data attuale la base di dati da errore
TRACCIARIF	Questo vincolo è stato creato per le cover e i remastering, viene indicato a quale traccia originale si riferiscono
UNICALISTABRANIP	Questo vincolo indica che nella base di dati la lista di brani preferiti è unica per ogni singolo utente
UNICOALBUM	Questo vincolo indica che possono esistere due album con lo stesso nome, ma devono appartenere ad artisti diversi
UNICATRACCIACONTENUTA	Questo vincolo indica che la stessa traccia non può essere aggiunta più di una volta all'interno della lista di brani preferiti

SCHEMA LOGICO

In questa fase il class diagram ristrutturato verrà tradotto in uno schema logico, livello di astrazione più basso. Indichiamo con una sottolineatura e in grassetto le **chiavi primarie** di ogni entità e indichiamo con una sottolineatura e un asterisco le **chiavi esterne***. **Da notare:** in album e in traccia abbiamo inserito la chiave primaria di artista, scelta progettuale

Table 4: Schema logico.

ENTITA'/ASSOCIAZIONI	ATTRIBUTI/RIFERIMENTI
Utente	(<u>NickName</u> , Nome, Cognome, SessoUtente, DataNascita, Nazionalità, Password, ID_Admin, TipoUtente, <u>ID_BraniPreferiti</u>) ID_BraniPreferiti —> BraniPreferiti.ID_BraniPreferiti
Ascoltatore	(<u>NickName*</u> , <u>ID_Traccia*</u> , NumeroAscoltiUtente, FasciaOrariaAscolto) NickName —> Utente.ID_BraniPreferiti ID_Traccia —> Traccia.ID_Traccia
BraniPreferiti	(<u>ID_BraniPreferiti</u> , NumeroBrani, TotaleMinutaggio)
ContieneTracce	(<u>ID_BraniPreferiti*</u> , <u>ID_Traccia*</u>) ID_BraniPreferiti —> BraniPreferiti.ID_BraniPreferiti ID_Traccia —> Traccia.ID_Traccia
Album	(<u>ID_Album</u> , NomeAlbum, AnnoUscita, N_Brani, DurataTotale, <u>ID_Artista*</u>) ID_Artista —> Artista.ID_Artista
FormatoTraccia	(<u>ID_Album*</u> , <u>ID_Traccia*</u>) ID_Album —> Album.ID_Album ID_Traccia —> Traccia.ID_Traccia
Traccia	(<u>ID_Traccia</u> , NomeTraccia, AnnoProduzione, DurataTraccia, N_TotaleAscolti, Genere, CheckCover, CheckRemastering, VersioneTraccia, RifTracciaOriginale, <u>ID_Artista*</u>) ID_Artista —> Artista.ID_Artista
Artista_T	(<u>ID_Artista</u> , NomeArtista, Nazionalità, SessoArtista, DataNascita)

PROGETTAZIONE FISICA

In questa sezione verranno elencati domini, entità, associazioni, funzioni e trigger. Il programma usato per la creazione del database è POSTGRESQL, usando come linguaggio PL/PGSQL.

4.1 Definizione dei domini

Seguono le definizioni dei domini in SQL

4.1.1 Sesso persona

```
1 CREATE DOMAIN sesso_persona AS char
2 CHECK(VALUE = 'M' OR VALUE = 'm' OR VALUE = 'F' OR VALUE = 'f');
```

4.1.2 Fascia oraria

```
1 CREATE DOMAIN fascia_oraria AS integer
2 CHECK(
3     VALUE >= 1
4 AND VALUE <= 6
5 );
```

4.2 Definizione delle tabelle

Seguono le definizioni dei domini in SQL

4.2.1 Utente

```
1 CREATE TABLE Utente(  
2 NickName VARCHAR(30) NOT NULL,  
3 Nome VARCHAR(30) NOT NULL,  
4 Cognome VARCHAR(30) NOT NULL,  
5 SessoUtente sesso_persona NOT NULL,  
6 DataNascita DATE NOT NULL,  
7 Nazionalita VARCHAR(30),  
8 Password VARCHAR(20) NOT NULL,  
9 CheckAdmin BOOLEAN DEFAULT false ,  
10  
11 CONSTRAINT pk_Utente PRIMARY KEY(NickName) ,  
12 --Controllo Data Nascita utente  
13 CONSTRAINT checkData CHECK(CURRENT_DATE>DataNascita) ,  
14 --Controllo Eta Utente >= 18 anni  
15 CONSTRAINT checkData18 CHECK(DataNascita <= CURRENT_DATE - interval '18 year')  
16 );
```

4.2.2 Traccia

```
1 CREATE TABLE Traccia(  
2 ID_Traccia SERIAL NOT NULL,  
3 NomeTraccia VARCHAR(50) NOT NULL,  
4 AnnoProduzione INT NOT NULL,  
5 DurataTraccia TIME NOT NULL,  
6 N_TotaleAscolti INT DEFAULT 0,  
7 Genere VARCHAR(20) NOT NULL,  
8 checkCover BOOLEAN DEFAULT false ,  
9 checkRemastering BOOLEAN DEFAULT false ,  
10 RifTracciaOriginale INT,  
11 ID_Artista INT NOT NULL,  
12  
13 CONSTRAINT pk_Traccia PRIMARY KEY(ID_Traccia) ,  
14 --Controllo che l'anno produzione della traccia sia >= dell'anno corrente  
15 CONSTRAINT checkDataTraccia CHECK(EXTRACT(year from CURRENT_DATE)>=  
16 AnnoProduzione) ,  
17 CONSTRAINT fk_RifTraccia FOREIGN KEY(RifTracciaOriginale) REFERENCES Traccia (  
18 ID_Traccia)  
19 ON DELETE CASCADE  
20 ON UPDATE CASCADE,  
21 CONSTRAINT fk_ArtistaT FOREIGN KEY(ID_Artista) REFERENCES Artista_T(ID_Artista)  
22 ON DELETE CASCADE  
23 ON UPDATE CASCADE,  
24 CONSTRAINT tracciarif CHECK((checkcover = false AND checkremastering = false AND  
25 RifTracciaOriginale IS NULL)  
OR (checkcover = true AND checkremastering = false AND RifTracciaOriginale  
IS NOT NULL)  
OR (checkcover = false AND checkremastering = true AND RifTracciaOriginale  
IS NOT NULL))  
);
```

4.2.3 Brani preferiti

```
1 CREATE TABLE BraniPreferiti(  
2 ID_BraniPreferiti SERIAL NOT NULL,  
3 NumeroBrani INT DEFAULT 0,  
4 TotaleMinutaggio TIME DEFAULT '00:00:00' ,  
5 NickName VARCHAR(30) NOT NULL,  
6  
7 CONSTRAINT pk_BraniPreferiti PRIMARY KEY (ID_BraniPreferiti) ,  
8 CONSTRAINT UnicaListaBraniP UNIQUE (NickName) ,  
9 CONSTRAINT fk_Utente FOREIGN KEY (NickName) REFERENCES Utente(NickName)  
10 ON DELETE CASCADE  
11 ON UPDATE CASCADE  
12 );
```

4.2.4 Album

```
1 CREATE TABLE Album(  
2   ID_Album SERIAL NOT NULL,  
3   NomeAlbum VARCHAR(30) NOT NULL,  
4   AnnoUscita INT NOT NULL,  
5   N_Brani INT NOT NULL,  
6   DurataTotale TIME NOT NULL,  
7   ID_Artista INT NOT NULL,  
8  
9   CONSTRAINT pk_Album PRIMARY KEY(ID_Album),  
10  CONSTRAINT unicoAlbum UNIQUE(NomeAlbum, ID_Artista),  
11  CONSTRAINT CheckDataAlbum CHECK(EXTRACT(YEAR from CURRENT_DATE)>=AnnoUscita), --  
    Contollo anno uscita album sia >= dell'anno corrente  
12  CONSTRAINT fk_Artista FOREIGN KEY (ID_Artista) REFERENCES Artista_T(ID_Artista)  
13      ON DELETE CASCADE  
14      ON UPDATE CASCADE  
15 );
```

4.2.5 Artista

```
1 CREATE TABLE Artista_T(  
2   ID_Artista SERIAL NOT NULL,  
3   NomeArtista VARCHAR(30) NOT NULL,  
4   Nazionalita VARCHAR(30) NOT NULL,  
5   SessoArtista sesso_persona NOT NULL,  
6   DataNascita DATE NOT NULL,  
7  
8   CONSTRAINT pk_Artista PRIMARY KEY(ID_Artista)  
9 );
```

4.2.6 Ascoltatore

```
1 CREATE TABLE Ascoltatore(  
2   NickName VARCHAR(20) NOT NULL,  
3   ID_Traccia SERIAL NOT NULL,  
4   FasciaOrariaAscolto fascia_oraria NOT NULL,  
5   CONSTRAINT fk_NickName FOREIGN KEY(NickName) REFERENCES Utente(NickName)  
6       ON DELETE CASCADE  
7       ON UPDATE CASCADE,  
8   CONSTRAINT fk_Traccia FOREIGN KEY(ID_Traccia) REFERENCES Traccia(ID_Traccia)  
9       ON DELETE CASCADE  
10      ON UPDATE CASCADE  
11 );
```


4.2.7 FormatoTraccia

```
1 CREATE TABLE FormatoTraccia(  
2 ID_Album SERIAL NOT NULL,  
3 ID_Traccia SERIAL NOT NULL,  
4  
5 CONSTRAINT fk_Album FOREIGN KEY(ID_Album) REFERENCES Album(ID_Album)  
6     ON DELETE CASCADE  
7     ON UPDATE CASCADE,  
8 CONSTRAINT fk_Traccia FOREIGN KEY(ID_Traccia) REFERENCES Traccia(ID_Traccia)  
9     ON DELETE CASCADE  
10    ON UPDATE CASCADE  
11 );
```

4.2.8 ContieneTracce

```
1 CREATE TABLE ContieneTracce(  
2 ID_Traccia SERIAL NOT NULL,  
3 ID_BraniPreferiti SERIAL NOT NULL,  
4  
5 CONSTRAINT UnicaTracciaContenuta UNIQUE(ID_Traccia, ID_BraniPreferiti),  
6 CONSTRAINT fk_Traccia FOREIGN KEY(ID_Traccia) REFERENCES Traccia(ID_Traccia)  
7     ON DELETE CASCADE  
8     ON UPDATE CASCADE,  
9 CONSTRAINT fk_BraniPreferiti FOREIGN KEY(ID_BraniPreferiti) REFERENCES  
10     BraniPreferiti(ID_BraniPreferiti)  
11     ON DELETE CASCADE  
12     ON UPDATE CASCADE  
13 );
```

4.3 Funzioni e trigger per le automazioni

4.3.1 Modifica dati dell'album con l'inserimento

```
1  -- PROCEDURA 1: quando viene aggiunta una traccia bisogna aggiornare il numero
   di tracce e la durata totale dell'album a cui appartiene --
2
3  CREATE FUNCTION Modifica_Colonne_Album_Inserimento()
4  RETURNS TRIGGER AS $STRIGGER_Modifica_Colonne_Album_Inserimento$
5  BEGIN
6      UPDATE Album
7      SET N_Brani = N_Brani+1,
8          DurataTotale = (SELECT SUM(T.duratatraccia)
9                          FROM FormatoTraccia as FT, Traccia AS T
10                         WHERE FT.ID_Traccia = T.ID_Traccia AND NEW.ID_Album = FT.
11                            ID_Album)
12      WHERE NEW.ID_Album = ID_Album;
13      RETURN NULL;
14  END;
15  $STRIGGER_Modifica_Colonne_Album_Inserimento$ LANGUAGE PLPGSQL;
16
17  --TRIGGER 1: modifica il numero delle tracce e la durata totale dell'album
18  quando viene aggiunta una traccia
19  CREATE OR REPLACE TRIGGER TRIGGER_Modifica_Colonne_Album_Inserimento
20  AFTER INSERT ON FormatoTraccia
21  FOR EACH ROW
22  EXECUTE PROCEDURE Modifica_Colonne_Album_Inserimento();
```

4.3.2 Modifica dati dell'album con la cancellazione

```
1  --PROCEDURA 2: Dopo l'eliminazione di una traccia bisogna modificare il numero
   di tracce e la durata totale dell'album a cui apparteneva
2
3  CREATE FUNCTION Modifica_Colonne_Album_Cancellazione()
4  RETURNS TRIGGER AS $TRIGGER_Modifica_Colonne_Album_Cancellazione$
5  BEGIN
6      UPDATE Album
7      SET N_Brani = N_Brani - 1,
8          DurataTotale = (SELECT SUM(T.duratatraccia)
9                          FROM FormatoTraccia AS FT, Traccia AS T
10                         WHERE FT.ID_Traccia = T.ID_Traccia AND OLD.ID_Album = FT.
11                            ID_Album)
12      WHERE OLD.ID_Album = ID_Album;
13  RETURN NULL;
14 END;
15 $TRIGGER_Modifica_Colonne_Album_Cancellazione$ LANGUAGE PLPGSQL;
16
17 --TRIGGER 2: modifica il numero delle tracce e il tempo dell'album a cui
   apparteneva una traccia eliminata
18 CREATE OR REPLACE TRIGGER TRIGGER_Modifica_Colonne_Album_Cancellazione
19 AFTER DELETE ON FormatoTraccia
20 FOR EACH ROW
EXECUTE PROCEDURE Modifica_Colonne_Album_Cancellazione();
```

4.3.3 Creazione Brani Preferiti automatica

```
1  --Funzione creazione branipreferiti
2  --PROCEDURA 3: Creazione della playlist brani preferiti quando viene aggiunto un
   nuovo utente
3
4  CREATE FUNCTION Creazione_BraniPreferiti()
5  RETURNS TRIGGER AS $TRIGGER_Creazione_BraniPreferiti$
6  BEGIN
7      INSERT INTO BraniPreferiti(NickName)
8      VALUES(NEW.Nickname);
9      RETURN NULL;
10 END;
11 $TRIGGER_Creazione_BraniPreferiti$ LANGUAGE PLPGSQL;
12
13 --TRIGGER 3: creazione playlist default
14 CREATE OR REPLACE TRIGGER TRIGGER_Creazione_BraniPreferiti
15 AFTER INSERT ON Utente
16 FOR EACH ROW
17 EXECUTE PROCEDURE Creazione_BraniPreferiti();
```

4.3.4 Modifica dati di Brani Preferiti con l'inserimento

```
1  --PROCEDURA 4: aggiornamento del numero delle tracce e della durata di una
2    playlist quando gli viene aggiunta una traccia
3  CREATE FUNCTION Modifica_Colonne_BraniPreferiti_Inserimento()
4  RETURNS TRIGGER AS $STRIGGER_Modifica_Colonne_BraniPreferiti_Inserimento$
5  BEGIN
6      UPDATE BraniPreferiti
7      SET  numerobrani = numerobrani+1,
8          TotaleMinutaggio = (SELECT SUM(T.duratatraccia)
9                               FROM ContieneTracce AS CT, Traccia AS T
10                              WHERE CT.ID_Traccia = T.ID_Traccia AND NEW.
11                                ID_BraniPreferiti = CT.ID_BraniPreferiti)
12      WHERE NEW.ID_BraniPreferiti = ID_BraniPreferiti;
13  RETURN NULL;
14  END;
15  $STRIGGER_Modifica_Colonne_BraniPreferiti_Inserimento$ LANGUAGE PLPGSQL;
16  -- TRIGGER 4: aggiunta di una Traccia in una Playlist
17  CREATE OR REPLACE TRIGGER TRIGGER_Modifica_Colonne_BraniPreferiti_Inserimento
18  AFTER INSERT ON ContieneTracce
19  FOR EACH ROW
20  EXECUTE PROCEDURE Modifica_Colonne_BraniPreferiti_Inserimento();
```

4.3.5 Modifica dati di Brani Preferiti con la cancellazione

```
1  --PROCEDURA 5: aggiornamento numero tracce e durata alla eliminazione di una
   traccia che contenuta in playlist
2
3  CREATE FUNCTION Modifica_Colonne_BraniPreferiti_Cancellazione()
4  RETURNS TRIGGER AS $STRIGGER_Modifica_Colonne_BraniPreferiti_Cancellazione$
5  BEGIN
6      UPDATE BraniPreferiti
7      SET  numerobrani = numerobrani-1,
8          TotaleMinutaggio = (SELECT SUM(T.duratatraccia)
9                               FROM ContieneTracce AS CT, Traccia AS T
10                              WHERE CT.ID_Traccia = T.ID_Traccia AND OLD.
11                                ID_BraniPreferiti = CT.ID_BraniPreferiti)
12      WHERE OLD.ID_BraniPreferiti = ID_BraniPreferiti;
13      RETURN NULL;
14  END;
15  $STRIGGER_Modifica_Colonne_BraniPreferiti_Cancellazione$ LANGUAGE PLPGSQL;
16
17  -- TRIGGER 5: aggiornamento numero tracce e durata di una playlist dopo che
   stata eliminata una traccia che conteneva
18  CREATE OR REPLACE TRIGGER TRIGGER_Modifica_Colonne_BraniPreferiti_Cancellazione
19  AFTER DELETE ON ContieneTracce
20  FOR EACH ROW
   EXECUTE PROCEDURE Modifica_Colonne_BraniPreferiti_Cancellazione();
```

4.3.6 Incremento automatico degli ascolti totali

```
1  ---TRIGGER 13 Incremento ascolti traccia-----
2
3  CREATE OR REPLACE FUNCTION Incremento_Ascolti()
4  RETURNS TRIGGER AS $TRIGGER_Incremento_Ascolti$
5  BEGIN
6      UPDATE Traccia AS T
7      SET n_totaleascolti = n_totaleascolti + 1
8      WHERE NEW.id_traccia = T.id_traccia;
9      RETURN NULL;
10 END;
11 $TRIGGER_Incremento_Ascolti$ LANGUAGE PLPGSQL;
12
13 --TRIGGER 13:
14 CREATE OR REPLACE TRIGGER TRIGGER_Incremento_Ascolti
15 AFTER INSERT ON Ascoltatore
16 FOR EACH ROW
17 EXECUTE PROCEDURE Incremento_Ascolti();
```

4.4 Funzioni e trigger per i vincoli

4.4.1 Controllo artista cover diverso dalla traccia originale

```
1  --PROCEDURA 7: Controllo artista cover
2
3  CREATE OR REPLACE FUNCTION Controllo_Artista_Cover()
4  RETURNS TRIGGER AS $STRIGGER_Controllo_Artista_Cover$
5  DECLARE
6
7  artisti Traccia.ID_artista%TYPE;
8
9  BEGIN
10     IF(NEW.checkcover = true) THEN
11         SELECT id_artista INTO artisti
12         FROM Traccia
13         WHERE NEW.riftracciaoriginale = id_traccia;
14     IF (artisti = NEW.id_artista) THEN
15         RAISE EXCEPTION USING ERRCODE = 'ARTER';
16     END IF;
17     END IF;
18     RETURN null;
19
20 EXCEPTION
21     WHEN SQLSTATE 'ARTER' THEN
22         RAISE NOTICE 'Nome artista cover uguale al nome dell artista della
23         traccia originale';
24         DELETE FROM Traccia
25         WHERE id_traccia = NEW.id_traccia;
26         RETURN NULL;
27 END;
28 $STRIGGER_Controllo_Artista_Cover$ LANGUAGE PLPGSQL;
29
30 --TRIGGER 7:
31 CREATE OR REPLACE TRIGGER TRIGGER_Controllo_Artista_Cover
32 AFTER INSERT ON Traccia
33 FOR EACH ROW
34 EXECUTE PROCEDURE Controllo_Artista_Cover();
```

4.4.2 Controllo artista remastering diverso dalla traccia originale

```
1  --PROCEDURA 8: Controllo artista remastering
2
3  CREATE OR REPLACE FUNCTION Controllo_Artista_Remastering()
4  RETURNS TRIGGER AS $TRIGGER_Controllo_Artista_Remastering$
5  DECLARE
6
7  artisti Traccia.ID_artista%TYPE;
8
9  BEGIN
10     IF(NEW.checkremastering = true) THEN
11         SELECT id_artista INTO artisti
12         FROM Traccia
13         WHERE NEW.riftracciaoriginale = id_traccia;
14     IF (artisti <> NEW.id_artista) THEN
15         RAISE EXCEPTION USING ERRCODE = 'ARTRE';
16     END IF;
17     END IF;
18     RETURN null;
19
20 EXCEPTION
21     WHEN SQLSTATE 'ARTRE' THEN
22         RAISE NOTICE 'Nome artista remastering diverso dal nome dell artista della
23         traccia originale';
24         DELETE FROM Traccia
25         WHERE id_traccia = NEW.id_traccia;
26         RETURN NULL;
27 END;
28 $TRIGGER_Controllo_Artista_Remastering$ LANGUAGE PLPGSQL;
29
30 --TRIGGER 8:
31 CREATE OR REPLACE TRIGGER TRIGGER_Controllo_Artista_Remastering
32 AFTER INSERT ON Traccia
33 FOR EACH ROW
34 EXECUTE PROCEDURE Controllo_Artista_Remastering();
```


4.4.3 Controllo data pubblicazione del remastering

```
1  --PROCEDURA 9: Controllo data Remastering
2
3  CREATE OR REPLACE FUNCTION Controllo_Data_Remastering()
4  RETURNS TRIGGER AS $TRIGGER_Controllo_Data_Remastering$
5  DECLARE
6
7  annoprod Traccia.annoproduzione%TYPE;
8
9  BEGIN
10     IF(NEW.checkremastering = true) THEN
11         SELECT annoproduzione INTO annoprod
12         FROM Traccia
13         WHERE NEW.riftracciaoriginale = id_traccia;
14     IF (annoprod > NEW.annoproduzione) THEN
15         RAISE EXCEPTION USING ERRCODE = 'DATRE';
16     END IF;
17     END IF;
18     RETURN null;
19
20 EXCEPTION
21     WHEN SQLSTATE 'DATRE' THEN
22         RAISE NOTICE 'Anno di produzione della remastering errato, inserire anno
23         successivo o uguale alla traccia originale';
24         DELETE FROM Traccia
25         WHERE id_traccia = NEW.id_traccia;
26         RETURN NULL;
27 END;
28 $TRIGGER_Controllo_Data_Remastering$ LANGUAGE PLPGSQL;
29
30 --TRIGGER 9:
31 CREATE OR REPLACE TRIGGER TRIGGER_Controllo_Data_Remastering
32 AFTER INSERT ON Traccia
33 FOR EACH ROW
34 EXECUTE PROCEDURE Controllo_Data_Remastering();
```

4.4.4 Controllo data pubblicazione cover

```
1  --PROCEDURA 10: Controllo data Cover
2
3  CREATE OR REPLACE FUNCTION Controllo_Data_Cover()
4  RETURNS TRIGGER AS $TRIGGER_Controllo_Data_Cover$
5  DECLARE
6
7  annoprod Traccia.annoproduzione%TYPE;
8
9  BEGIN
10     IF(NEW.checkcover = true) THEN
11         SELECT annoproduzione INTO annoprod
12         FROM Traccia
13         WHERE NEW.riftracciaoriginale = id_traccia;
14     IF (annoprod > NEW.annoproduzione) THEN
15         RAISE EXCEPTION USING ERRCODE = 'DATER';
16     END IF;
17     END IF;
18     RETURN null;
19 EXCEPTION
20     WHEN SQLSTATE 'DATER' THEN
21         RAISE NOTICE 'Anno di produzione della cover errato, inserire anno
22         successivo o uguale alla traccia originale';
23         DELETE FROM Traccia
24         WHERE id_traccia = NEW.id_traccia;
25         RETURN NULL;
26 END;
27 $TRIGGER_Controllo_Data_Cover$ LANGUAGE PLPGSQL;
28
29 --TRIGGER 10:
30 CREATE OR REPLACE TRIGGER TRIGGER_Controllo_Data_Cover
31 AFTER INSERT ON Traccia
32 FOR EACH ROW
33 EXECUTE PROCEDURE Controllo_Data_Cover();
```

4.4.5 Controllo della cover all'interno di un album

```
1  --TRIGGER 11: Controllo cover in un album
2
3  CREATE OR REPLACE FUNCTION Controllo_Cover_Album()
4  RETURNS TRIGGER AS $TRIGGER_Controllo_Cover_Album$
5
6  DECLARE
7
8  checkc Traccia.checkcover%TYPE;
9  riforig Traccia.riftracciaoriginale%TYPE;
10 rifalbum formatotraccia.id_album%TYPE;
11
12 BEGIN
13     SELECT T.checkcover INTO checkc
14     FROM Traccia AS T
15     WHERE NEW.id_traccia = T.id_traccia;
16
17     IF(checkc = true) THEN
18         SELECT T.riftracciaoriginale INTO riforig
19         FROM Traccia AS T
20         WHERE NEW.id_traccia = T.id_traccia;
21
22         SELECT FT.id_album INTO rifalbum
23         FROM formatotraccia AS FT
24         WHERE FT.id_traccia = riforig;
25
26         IF(NEW.id_album = rifalbum) THEN
27             RAISE EXCEPTION USING ERRCODE = 'ALBER';
28         END IF;
29     END IF;
30     RETURN null;
31 EXCEPTION
32     WHEN SQLSTATE 'ALBER' THEN
33         RAISE NOTICE 'La cover non puo essere inserita nello stesso album di
34 appartenenza della sua traccia originale';
35         DELETE FROM formatotraccia
36         WHERE id_traccia = NEW.id_traccia;
37         RETURN NULL;
38 END;
39 $TRIGGER_Controllo_Cover_Album$ LANGUAGE PLPGSQL;
40
41 --TRIGGER 11:
42 CREATE OR REPLACE TRIGGER TRIGGER_Controllo_Cover_Album
43 AFTER INSERT ON formatotraccia
44 FOR EACH ROW
45 EXECUTE PROCEDURE Controllo_Cover_Album();
```

4.4.6 Controllo unico artista in un album

```
1  --TRIGGER 12: Controllo artista in album
2
3  CREATE OR REPLACE FUNCTION Controllo_Artista_Album()
4  RETURNS TRIGGER AS $TRIGGER_Controllo_Artista_Album$
5
6  DECLARE
7
8  art_album album.id_artista%TYPE;
9  art_traccia traccia.id_artista%TYPE;
10
11 BEGIN
12     SELECT A.ID_Artista INTO art_album
13     FROM Album AS A
14     WHERE NEW.id_Album = A.id_Album;
15
16     SELECT T.id_artista INTO art_traccia
17     FROM Traccia AS T
18     WHERE NEW.id_Traccia = T.id_Traccia;
19
20
21     IF (art_album <> art_traccia) THEN
22         RAISE EXCEPTION USING ERRCODE = 'ARTDV';
23     END IF;
24     RETURN NULL;
25 EXCEPTION
26     WHEN SQLSTATE 'ARTDV' THEN
27         RAISE NOTICE 'L album non puo contenere due artisti diversi';
28         DELETE FROM formatotraccia
29         WHERE id_traccia = NEW.id_traccia AND id_album = NEW.id_Album;
30     RETURN NULL;
31 END;
32 $TRIGGER_Controllo_Artista_Album$ LANGUAGE PLPGSQL;
33
34 --TRIGGER 12:
35 CREATE OR REPLACE TRIGGER TRIGGER_Controllo_Artista_Album
36 AFTER INSERT ON FormatoTraccia
37 FOR EACH ROW
38 EXECUTE PROCEDURE Controllo_Artista_Album();
```

MANUALE D'USO

5.1 Creazione utenti

5.1.1 UtenteBase

```
1 -----PERMESSI UTENTI E AMMINISTRATORE-----
2
3 CREATE USER UtenteBase PASSWORD '12345';
4 GRANT SELECT on Traccia to UtenteBase ;
5 GRANT SELECT on Album to UtenteBase ;
6 GRANT SELECT on Artista_t to UtenteBase ;
7 GRANT SELECT on BraniPreferiti to UtenteBase ;
8 GRANT SELECT on ContieneTracce to UtenteBase ;
9 GRANT SELECT on FormatoTraccia to UtenteBase ;
10 GRANT SELECT on Utente to UtenteBase ;
11 GRANT INSERT on ContieneTracce to UtenteBase ;
12 GRANT UPDATE on ContieneTracce to UtenteBase ;
13 GRANT DELETE on ContieneTracce to UtenteBase ;
14 GRANT INSERT on Ascoltatore to UtenteBase ;
```

5.1.2 UtenteAdmin

```
1 CREATE USER UtenteAdmin PASSWORD 'admin';
2 GRANT ALL on Traccia to UtenteAdmin ;
3 GRANT ALL on Album to UtenteAdmin ;
4 GRANT ALL on Artista_t to UtenteAdmin ;
5 GRANT ALL on Ascoltatore to UtenteAdmin ;
6 GRANT ALL on BraniPreferiti to UtenteAdmin ;
7 GRANT ALL on Utente to UtenteAdmin ;
8 GRANT ALL on ContieneTracce to UtenteAdmin ;
9 GRANT ALL on FormatoTraccia to UtenteAdmin ;
```

5.2 Popolamento

5.2.1 Utente

```
1 INSERT INTO utente (nickname, nome, cognome, sessoutente, datanascita, "
   nazionalita", password, CheckAdmin)
2 VALUES ('kale', 'Matteo', 'Monna', 'M', '2003-04-05', 'Italia', 'hdet!45', false
   );
3
4 INSERT INTO utente (nickname, nome, cognome, sessoutente, datanascita, "
   nazionalita", password, CheckAdmin)
5 VALUES ('Root', 'Fabrizio', 'Minino', 'M', '2002-03-04', 'Italia', 'vsaset235',
   false);
6
7 INSERT INTO utente (nickname, nome, cognome, sessoutente, datanascita, "
   nazionalita", password, CheckAdmin)
8 VALUES ('Bonazzoli97', 'Federico', 'Bonazzoli', 'M', '21-05-1997', 'Italia', '
   ForzaSalernitana', false);
9
10 INSERT INTO utente (nickname, nome, cognome, sessoutente, datanascita, "
   nazionalita", password, CheckAdmin)
11 VALUES ('Lordong', 'Lorenzo', 'D Orio', 'M', '20-04-1999', 'Italia', 'Odino',
   false);
12
13 INSERT INTO utente (nickname, nome, cognome, sessoutente, datanascita, "
   nazionalita", password, CheckAdmin)
14 VALUES('PincoPallino', 'Pinco', 'Pallino', 'M', '10-10-2000', 'Francia', '
   IIPallino', false);
15
16 INSERT INTO utente (nickname, nome, cognome, sessoutente, datanascita, "
   nazionalita", password, CheckAdmin)
17 VALUES('Ilpazzo', 'Pippo', 'Pimpante', 'M', '09-12-1980', 'Italia', '12345678',
   false);
18
19 INSERT INTO utente (nickname, nome, cognome, sessoutente, datanascita, "
   nazionalita", password, CheckAdmin)
20 VALUES('Principessa78', 'Francesca', 'Marini', 'F', '24-04-1978', 'Italia', '
   princess78', false);
21
22 INSERT INTO utente (nickname, nome, cognome, sessoutente, datanascita, "
   nazionalita", password, CheckAdmin)
23 VALUES('amministratore', 'Silvio', 'Barra', 'M', '24-04-1990', 'Italia', 'admin'
   , true);
```

5.2.2 Traccia

```
1 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
  n_totaleascolti, genere, checkcover, checkremastering, rifTracciaOriginale,
  id_artista)
2 VALUES ('1', 'Emanuele', '2021', '00:03:50', '0', 'Rap', false, false, NULL, '1'
  );
3
4 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
  n_totaleascolti, genere, checkcover, checkremastering, rifTracciaOriginale,
  id_artista)
5 VALUES ('2', 'Italiano', '2021', '00:03:10', '0', 'Rap', false, false, NULL, '2'
  );
6
7 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
  n_totaleascolti, genere, checkcover, checkremastering, rifTracciaOriginale,
  id_artista)
8 VALUES ('3', 'Rivoluzione', '2021', '00:02:20', '0', 'Rap', false, false, NULL,
  '3');
9
10 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
  n_totaleascolti, genere, checkcover, checkremastering, rifTracciaOriginale,
  id_artista)
11 VALUES ('4', 'Red Barz RMX', '2020', '00:03:20', '0', 'Rap', false, false, NULL,
  '4');
12
13 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
  n_totaleascolti, genere, checkcover, checkremastering, rifTracciaOriginale,
  id_artista)
14 VALUES ('5', 'Veleno', '2010', '00:03:00', '0', 'Rap', false, false, NULL, '5');
15
16 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
  n_totaleascolti, genere, checkcover, checkremastering, rifTracciaOriginale,
  id_artista)
17 VALUES ('6', 'LOVE', '2021', '00:03:41', '0', 'Rap', false, false, NULL, '6');
18
19 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
  n_totaleascolti, genere, checkcover, checkremastering, rifTracciaOriginale,
  id_artista)
20 VALUES('7', 'Infinity 2008', '2008', '00:04:25', '0', 'EDM', false, false, NULL
  , '7');
21
22 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
  n_totaleascolti, genere, checkcover, checkremastering, rifTracciaOriginale,
  id_artista)
23 VALUES('8', 'Stan', '2006', '00:04:00', '0', 'Rap', false, false, NULL, '8');
24
25 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
  n_totaleascolti, genere, checkcover, checkremastering, rifTracciaOriginale,
  id_artista)
26 VALUES('9', 'Break my heart', '2019', '00:03:51', '0', 'Pop', false, false, NULL
  , '9');
27
```

```

28 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, rifTracciaOriginale,
    id_artista)
29 VALUES('10', 'Il ragazzo della via', '1966', '00:04:30', '0', 'Pop', false,
    false, NULL, '10');
30
31 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
32 VALUES ('11', 'Veleno (Remastering)', '2015', '00:03:00', '0', 'rap', false,
    true, '5', '5');
33
34 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
35 VALUES ('12', 'Rivoluzione(Cover)', '2022', '00:02:50', '0', 'rap', true, false,
    '3', '11');
36
37 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
38 VALUES ('13', 'Mamma mia', '2021', '00:03:00', '0', 'rap', false, false, NULL, '
    2');
39
40 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
41 VALUES ('14', 'Chicano', '2019', '00:03:00', '0', 'Rap', false, false, NULL, '1'
    );
42
43 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
44 VALUES ('15', 'Senz E Me', '2019', '00:02:25', '0', 'Rap', false, false, NULL, '
    1');
45
46 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
47 VALUES ('16', 'Na Catena', '2019', '00:02:45', '0', 'Rap', false, false, NULL, '
    1');
48
49 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
50 VALUES ('17', 'Provino', '2019', '00:03:15', '0', 'Rap', false, false, NULL, '1'
    );
51
52 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
53 VALUES ('18', 'Easy', '2022', '00:03:26', '0', 'Rap', false, false, NULL, '2');
54

```



```

55 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
56 VALUES ('19', 'Sola', '2022', '00:02:27', '0', 'Rap', false, false, NULL, '2');
57
58 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
59 VALUES ('20', 'Popstar', '2018', '00:03:03', '0', 'Trap', false, false, NULL, '2');
60
61 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
62 VALUES ('21', 'Happy Birthday', '2018', '00:02:50', '0', 'Trap', false, false,
    NULL, '2');
63
64 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
65 VALUES ('22', 'Cupido', '2018', '00:03:31', '0', 'Trap', false, false, NULL, '2');
66
67 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
68 VALUES ('23', 'Ricchi x Sempre', '2018', '00:03:43', '0', 'Trap', false, false,
    NULL, '2');
69
70 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
71 VALUES ('24', 'Solido', '2021', '00:02:51', '0', 'Rap', false, false, NULL, '3');
72
73 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
74 VALUES ('25', 'Fantastica', '2021', '00:03:07', '0', 'Rap', false, false, NULL,
    '3');
75
76 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
77 VALUES ('26', 'L' urdemo vase', '2021', '00:03:08', '0', 'Pop', false, false,
    NULL, '3');
78
79 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
80 VALUES ('27', 'L' urdemo vase (Rap Version)', '2021', '00:03:50', '0', 'Rap',
    false, true, '26', '3');
81

```

```

82 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
83 VALUES ('28', 'Pensando a lei', '2020', '00:02:56', '0', 'Rap', false, false,
    NULL, '4');
84
85 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
86 VALUES ('29', 'Niente da perdere', '2020', '00:02:46', '0', 'Rap', false, false,
    NULL, '4');
87
88 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
89 VALUES ('30', 'Hugh Guefner', '2018', '00:01:19', '0', 'Rap', false, false, NULL,
    '5');
90
91 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
92 VALUES ('31', 'Trap Phone', '2018', '00:03:20', '0', 'Rap', false, false, NULL,
    '5');
93
94 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
95 VALUES ('32', 'Borsello', '2018', '00:03:22', '0', 'Rap', false, false, NULL, '5');
96
97 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
98 VALUES ('33', 'Gangster Of Love', '2021', '00:02:45', '0', 'Rap', false, false,
    NULL, '5');
99
100 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
101 VALUES ('34', 'Piango Sulla Lambo', '2021', '00:03:21', '0', 'Rap', false, false,
    NULL, '5');
102
103 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
104 VALUES ('35', 'Blitz!', '2021', '00:03:00', '0', 'Rap', false, false, NULL, '5');
105
106 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
107 VALUES ('36', 'Daytona', '2021', '00:02:42', '0', 'Rap', false, false, NULL, '5');

```

```

108
109 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
      n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
      id_artista)
110 VALUES ('37', 'Piango Sulla Lambo (EDM Remix)', '2022', '00:04:00', '0', 'EDM',
      false, true, '34', '5');
111
112 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
      n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
      id_artista)
113 VALUES ('38', 'Crazy Love', '2021', '00:03:21', '0', 'Rap', false, false, NULL,
      '6');
114
115 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
      n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
      id_artista)
116 VALUES ('39', 'Cosplayer', '2021', '00:03:41', '0', 'Rap', false, false, NULL, '
      6');
117
118 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
      n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
      id_artista)
119 VALUES ('40', 'Dubbi', '2021', '00:03:54', '0', 'Rap', false, false, NULL, '6');
120
121 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
      n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
      id_artista)
122 VALUES ('41', 'Noi', '2021', '00:04:32', '0', 'Rap', false, false, NULL, '6');
123
124 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
      n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
      id_artista)
125 VALUES ('42', 'NEON – Le Ali', '2020', '00:03:27', '0', 'Commerciale', false,
      false, NULL, '6');
126
127 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
      n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
      id_artista)
128 VALUES ('43', 'CRUDELIA – I nervi', '2020', '00:03:50', '0', 'Commerciale',
      false, false, NULL, '6');
129
130 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
      n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
      id_artista)
131 VALUES ('44', 'SUPREME – L'ego', '2020', '00:03:43', '0', 'Rap', false, false,
      NULL, '6');
132
133 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
      n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
      id_artista)
134 VALUES ('45', 'Love to Infinity', '2020', '00:02:48', '0', 'EDM', false, false,
      NULL, '7');
135

```

```

136 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkremastering, checkcover, riftracciaoriginale,
    id_artista)
137 VALUES ('46', 'Love to Infinity (CLUB MIX)', '2021', '00:03:48', '0', 'EDM',
    true, false, '45', '7');
138
139 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
140 VALUES ('47', 'Arabian Nights', '2020', '00:03:58', '0', 'EDM', false, false,
    NULL, '7');
141
142 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkremastering, checkcover, riftracciaoriginale,
    id_artista)
143 VALUES ('48', 'Arabian Nights (CLUB MIX)', '2021', '00:04:00', '0', 'EDM', true,
    false, '47', '7');
144
145 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
146 VALUES ('49', 'W.T.P.', '2010', '00:03:58', '0', 'Rap', false, false, NULL, '8')
    ;
147
148 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
149 VALUES ('50', 'Not Afraid', '2010', '00:04:08', '0', 'Rap', false, false, NULL,
    '8');
150
151 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
152 VALUES ('51', 'Levitating', '2020', '00:03:23', '0', 'Pop', false, false, NULL,
    '9');
153
154 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
155 VALUES ('52', 'Don't Start Now', '2020', '00:03:03', '0', 'Pop', false, false,
    NULL, '9');
156
157 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
158 VALUES ('53', 'Sabato Triste', '1966', '00:03:22', '0', 'Pop', false, false,
    NULL, '10');
159
160 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
161 VALUES ('54', 'Chi Ce L'ha Con Me', '1966', '00:03:18', '0', 'Pop', false,
    false, NULL, '10');

```

```

162
163 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
164 VALUES ('55', 'Il Ragazzo Della Via Gluck – Remastered', '2011', '00:03:56', '0'
    , 'Pop', false, true, '10', '10');
165
166 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
167 VALUES ('56', 'Sabato Triste–Remastered', '2011', '00:04:01', '0', 'Pop', false,
    true, '53', '10');
168
169 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
170 VALUES ('57', 'Chi Ce L'ha Con Me – Remastered', '2011', '00:04:00', '0', 'Pop'
    , false, true, '54', '10');
171
172 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
173 VALUES ('58', 'A Volo', '2022', '00:03:59', '0', 'Commerciale', false, false,
    NULL, '11');
174
175 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
176 VALUES ('59', 'A Volo (Hard Remix)', '2022', '00:05:00', '0', 'EDM', false, true
    , '58', '11');
177
178 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
179 VALUES ('60', 'The King and I', '2022', '00:03:00', '0', 'Rap', false, false,
    NULL, '8');
180
181 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
182 VALUES ('61', 'SaraEsse', '2022', '00:04:51', '0', 'Reggeaton', false, false,
    NULL, '12');
183 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
184 VALUES ('62', 'SaraEsse (Rap Version)', '2022', '00:02:59', '0', 'Rap', false,
    true, '61', '12');
185 INSERT INTO traccia (id_traccia, nometraccia, annoproduzione, duratatraccia,
    n_totaleascolti, genere, checkcover, checkremastering, riftracciaoriginale,
    id_artista)
186 VALUES ('63', 'Stan (Cover)', '2022', '00:03:52', '0', 'Reggeaton', true, false,
    '8', '12');

```

5.2.3 Album

```
1  INSERT INTO album (id_album, n_brani, nomealbum, annouscita, duratatotale,
2  id_artista)
3  VALUES ('1', '0', 'Emanuele', '2019', '00:00:00', '1');
4
5  INSERT INTO album (id_album, n_brani, nomealbum, annouscita, duratatotale,
6  id_artista)
7  VALUES ('2', '0', 'Italiano', '2022', '00:00:00', '2');
8
9  INSERT INTO album (id_album, n_brani, nomealbum, annouscita, duratatotale,
10 id_artista)
11 VALUES ('3', '0', 'Rivoluzione', '2010', '00:00:00', '3');
12
13 INSERT INTO album (id_album, n_brani, nomealbum, annouscita, duratatotale,
14 id_artista)
15 VALUES ('4', '0', 'RedBarz', '2020', '00:00:00', '4');
16
17 INSERT INTO album (id_album, n_brani, nomealbum, annouscita, duratatotale,
18 id_artista)
19 VALUES ('5', '0', 'Sinatra', '2021', '00:00:00', '5');
20
21 INSERT INTO album (id_album, n_brani, nomealbum, annouscita, duratatotale,
22 id_artista)
23 VALUES ('6', '0', 'Noi loro gli altri', '2021', '00:00:00', '6');
24
25 INSERT INTO album (id_album, n_brani, nomealbum, annouscita, duratatotale,
26 id_artista)
27 VALUES ('7', '0', 'Guru Josh Album', '2003', '00:00:00', '7');
28
29 INSERT INTO album (id_album, n_brani, nomealbum, annouscita, duratatotale,
30 id_artista)
31 VALUES ('8', '0', 'Recovery', '2010', '00:00:00', '8');
32
33 INSERT INTO album (id_album, n_brani, nomealbum, annouscita, duratatotale,
34 id_artista)
35 VALUES ('9', '0', 'Future Nostalgia', '2020', '00:00:00', '9');
36
37 INSERT INTO album (id_album, n_brani, nomealbum, annouscita, duratatotale,
38 id_artista)
39 VALUES ('10', '0', 'La Festa', '1966', '00:00:00', '10');
40
41 INSERT INTO album (id_album, n_brani, nomealbum, annouscita, duratatotale,
42 id_artista)
43 VALUES ('11', '0', 'Rockstar', '2018', '00:00:00', '2');
44
45 INSERT INTO album (id_album, n_brani, nomealbum, annouscita, duratatotale,
46 id_artista)
47 VALUES ('12', '0', 'Persona', '2019', '00:00:00', '6');
48
49 INSERT INTO album (id_album, n_brani, nomealbum, annouscita, duratatotale,
50 id_artista)
51 VALUES ('13', '0', 'GVESVS', '2021', '00:00:00', '5');
52
53 INSERT INTO album (id_album, n_brani, nomealbum, annouscita, duratatotale,
54 id_artista)
55 VALUES ('14', '0', 'La Festa (2011 REMASTERED)', '2011', '00:00:00', '10');
```

5.2.4 Artista_t

```
1  Insert INTO artista_t(id_artista , nomeartista , nazionalit , sessoartista ,
   datanascita)
2  VALUES ( '1', 'Geolier', 'Italiano', 'm', '2010-02-03');
3
4  INSERT INTO artista_t (id_artista , nomeartista , "nazionalit ", sessoartista ,
   datanascita)
5  VALUES ( '2', 'Sferaebbasta', 'Italia', 'M', '1998-01-01');
6
7  INSERT INTO artista_t (id_artista , nomeartista , "nazionalit ", sessoartista ,
   datanascita)
8  VALUES ( '3', 'Rocco Hunt', 'Italia', 'M', '1997-02-02');
9
10 INSERT INTO artista_t (id_artista , nomeartista , "nazionalit ", sessoartista ,
   datanascita)
11 VALUES ( '4', 'Shiva', 'Italia', 'M', '1999-02-02');
12
13 INSERT INTO artista_t(id_artista , nomeartista , "nazionalit ", sessoartista ,
   datanascita)
14 VALUES ( '5', 'Gu ', 'Italia', 'M', '1995-02-02');
15
16 INSERT INTO artista_t(id_artista , nomeartista , nazionalit , sessoartista ,
   datanascita)
17 VALUES( '6', 'Marracash', 'italia', 'M', '25-08-1992');
18
19 INSERT INTO artista_t(id_artista , nomeartista , nazionalit , sessoartista ,
   datanascita)
20 VALUES( '7', 'Guru Josh', 'Inghilterra', 'M', '06-06-1964');
21
22 INSERT INTO artista_t(id_artista , nomeartista , nazionalit , sessoartista ,
   datanascita)
23 VALUES( '8', 'Eminem', 'USA', 'M', '17-10-1972');
24
25 INSERT INTO artista_t(id_artista , nomeartista , nazionalit , sessoartista ,
   datanascita)
26 VALUES( '9', 'Dua Lipa', 'USA', 'F', '22-08-1995');
27
28 INSERT INTO artista_t(id_artista , nomeartista , nazionalit , sessoartista ,
   datanascita)
29 VALUES( '10', 'Adriano Celentano', 'Italia', 'M', '06-01-1938');
30
31 INSERT INTO artista_t(id_artista , nomeartista , nazionalit , sessoartista ,
   datanascita)
32 VALUES( '11', 'Davide Maresca', 'Italia', 'M', '01-01-1999');
33
34 INSERT INTO artista_t(id_artista , nomeartista , nazionalit , sessoartista ,
   datanascita)
35 VALUES( '12', 'PeppeSc ', 'Uzbekistan', 'M', '1999-11-04');
```

5.2.5 Ascoltatore

```
1  --POPOLAMENTO ASCOLTATORE
2  INSERT INTO Ascoltatore
3  VALUES ('kale', '1', '1');
4
5  INSERT INTO Ascoltatore
6  VALUES ('kale', '2', '1');
7
8  INSERT INTO Ascoltatore
9  VALUES ('kale', '3', '1');
10
11 INSERT INTO Ascoltatore
12 VALUES ('kale', '4', '2');
13
14 INSERT INTO Ascoltatore
15 VALUES ('kale', '5', '6');
16
17 INSERT INTO Ascoltatore
18 VALUES ('kale', '6', '6');
19
20 INSERT INTO Ascoltatore
21 VALUES ('kale', '7', '1');
22
23 INSERT INTO Ascoltatore
24 VALUES ('kale', '8', '5');
25
26 INSERT INTO Ascoltatore
27 VALUES ('kale', '9', '1');
28
29 INSERT INTO Ascoltatore
30 VALUES ('kale', '10', '1');
31
32 INSERT INTO Ascoltatore
33 VALUES ('kale', '1', '1');
34
35 INSERT INTO Ascoltatore
36 VALUES ('kale', '5', '4');
37
38 INSERT INTO Ascoltatore
39 VALUES ('kale', '10', '4');
40
41 INSERT INTO Ascoltatore
42 VALUES ('Root', '3', '4');
43
44 INSERT INTO Ascoltatore
45 VALUES ('Root', '4', '5');
46
47 INSERT INTO Ascoltatore
48 VALUES ('Root', '60', '6');
49
50 INSERT INTO Ascoltatore
51 VALUES ('Root', '22', '5');
```



```

52
53 INSERT INTO Ascoltatore
54 VALUES ( 'Root', '11', '5');
55
56 INSERT INTO Ascoltatore
57 VALUES ( 'Root', '19', '5');
58
59 INSERT INTO Ascoltatore
60 VALUES ( 'Bonazzoli97', '2', '5');
61
62 INSERT INTO Ascoltatore
63 VALUES ( 'Bonazzoli97', '5', '3');
64
65 INSERT INTO Ascoltatore
66 VALUES ( 'Bonazzoli97', '2', '3');
67
68 INSERT INTO Ascoltatore
69 VALUES ( 'Bonazzoli97', '5', '3');
70
71 INSERT INTO Ascoltatore
72 VALUES ( 'Bonazzoli97', '5', '4');
73
74 INSERT INTO Ascoltatore
75 VALUES ( 'Bonazzoli97', '35', '5');
76
77 INSERT INTO Ascoltatore
78 VALUES ( 'Bonazzoli97', '50', '6');
79
80 INSERT INTO Ascoltatore
81 VALUES ( 'Bonazzoli97', '44', '5');
82
83 INSERT INTO Ascoltatore
84 VALUES ( 'Bonazzoli97', '22', '5');
85
86 INSERT INTO Ascoltatore
87 VALUES ( 'Bonazzoli97', '32', '5');
88
89 INSERT INTO Ascoltatore
90 VALUES ( 'Lordong', '1', '4');
91
92 INSERT INTO Ascoltatore
93 VALUES ( 'Lordong', '44', '4');
94
95 INSERT INTO Ascoltatore
96 VALUES ( 'Lordong', '45', '4');
97
98 INSERT INTO Ascoltatore
99 VALUES ( 'Lordong', '55', '4');
100
101 INSERT INTO Ascoltatore
102 VALUES ( 'Lordong', '22', '4');
103
104 INSERT INTO Ascoltatore

```

```
105 VALUES ( 'Lordong', '10', '1' );
106
107 INSERT INTO Ascoltatore
108 VALUES ( 'Lordong', '7', '1' );
109
110 INSERT INTO Ascoltatore
111 VALUES ( 'PincoPallino', '1', '1' );
112
113 INSERT INTO Ascoltatore
114 VALUES ( 'PincoPallino', '61', '1' );
115
116 INSERT INTO Ascoltatore
117 VALUES ( 'PincoPallino', '41', '1' );
118
119 INSERT INTO Ascoltatore
120 VALUES ( 'PincoPallino', '50', '4' );
121
122 INSERT INTO Ascoltatore
123 VALUES ( 'Ilpazzo', '1', '4' );
124
125 INSERT INTO Ascoltatore
126 VALUES ( 'Ilpazzo', '51', '4' );
127
128 INSERT INTO Ascoltatore
129 VALUES ( 'Ilpazzo', '31', '4' );
130
131 INSERT INTO Ascoltatore
132 VALUES ( 'Principessa78', '1', '4' );
```

5.2.6 FormatoTraccia

```
1  --POPOLAMENTO FORMATO TRACCIA
2  INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
3  VALUES( '1', '1');
4
5  INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
6  VALUES( '2', '2');
7
8  INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
9  VALUES( '2', '13');
10
11 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
12 VALUES( '3', '3');
13
14 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
15 VALUES( '4', '4');
16
17 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
18 VALUES( '5', '5');
19
20 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
21 VALUES( '6', '6');
22
23 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
24 VALUES( '7', '7');
25
26 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
27 VALUES( '8', '8');
28
29 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
30 VALUES( '9', '9');
31
32 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
33 VALUES( '10', '10');
34
35 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
36 VALUES( '1', '14');
37
38 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
39 VALUES( '1', '15');
40
41 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
42 VALUES( '1', '16');
43
44 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
45 VALUES( '1', '17');
46
47 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
48 VALUES( '2', '18');
49
50 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
51 VALUES( '2', '19');
```

```

52
53 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
54 VALUES( '11', '20');
55
56 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
57 VALUES( '11', '21');
58
59 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
60 VALUES( '11', '22');
61
62 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
63 VALUES( '11', '23');
64
65 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
66 VALUES( '3', '24');
67
68 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
69 VALUES( '3', '25');
70
71 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
72 VALUES( '3', '26');
73
74 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
75 VALUES( '3', '27');
76
77 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
78 VALUES( '4', '28');
79
80 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
81 VALUES( '4', '29');
82
83 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
84 VALUES( '5', '30');
85
86 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
87 VALUES( '5', '31');
88
89 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
90 VALUES( '5', '32');
91
92 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
93 VALUES( '13', '33');
94
95 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
96 VALUES( '13', '34');
97
98 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
99 VALUES( '13', '35');
100
101 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
102 VALUES( '13', '36');
103
104 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)

```

```

105 VALUES( '13', '37' );
106
107 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
108 VALUES( '6', '38' );
109
110 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
111 VALUES( '6', '39' );
112
113 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
114 VALUES( '6', '40' );
115
116 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
117 VALUES( '6', '41' );
118
119 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
120 VALUES( '12', '42' );
121
122 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
123 VALUES( '12', '43' );
124
125 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
126 VALUES( '12', '44' );
127
128 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
129 VALUES( '7', '45' );
130
131 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
132 VALUES( '7', '46' );
133
134 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
135 VALUES( '7', '47' );
136
137 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
138 VALUES( '7', '48' );
139
140 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
141 VALUES( '8', '49' );
142
143 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
144 VALUES( '8', '50' );
145
146 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
147 VALUES( '9', '51' );
148
149 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
150 VALUES( '9', '52' );
151
152 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
153 VALUES( '10', '53' );
154
155 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
156 VALUES( '10', '54' );
157

```

```

158 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
159 VALUES ( '14', '55' );
160
161 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
162 VALUES ( '14', '56' );
163
164 INSERT INTO FormatoTraccia(ID_Album, ID_Traccia)
165 VALUES ( '14', '57' );

```

5.2.7 ContieneTracce

```

1  --POPOLAMENTO CONTIENETRACCIA
2
3  INSERT INTO ContieneTracce(ID_Traccia, ID_BraniPreferiti)
4  VALUES ( '1', '1' );
5
6  INSERT INTO ContieneTracce(ID_Traccia, ID_BraniPreferiti)
7  VALUES ( '2', '1' );
8
9  INSERT INTO ContieneTracce(ID_Traccia, ID_BraniPreferiti)
10 VALUES ( '3', '1' );
11
12 INSERT INTO ContieneTracce(ID_Traccia, ID_BraniPreferiti)
13 VALUES ( '4', '1' );
14
15 INSERT INTO ContieneTracce(ID_Traccia, ID_BraniPreferiti)
16 VALUES ( '51', '1' );
17
18 INSERT INTO ContieneTracce(ID_Traccia, ID_BraniPreferiti)
19 VALUES ( '31', '1' );
20
21 INSERT INTO ContieneTracce(ID_Traccia, ID_BraniPreferiti)
22 VALUES ( '39', '1' );
23
24 INSERT INTO ContieneTracce(ID_Traccia, ID_BraniPreferiti)
25 VALUES ( '1', '2' );
26
27 INSERT INTO ContieneTracce(ID_Traccia, ID_BraniPreferiti)
28 VALUES ( '3', '2' );
29
30 INSERT INTO ContieneTracce(ID_Traccia, ID_BraniPreferiti)
31 VALUES ( '1', '3' );
32
33 INSERT INTO ContieneTracce(ID_Traccia, ID_BraniPreferiti)
34 VALUES ( '2', '3' );
35
36 INSERT INTO ContieneTracce(ID_Traccia, ID_BraniPreferiti)
37 VALUES ( '4', '3' );
38
39 INSERT INTO ContieneTracce(ID_Traccia, ID_BraniPreferiti)

```

```

40 VALUES ( '24' , '3' );
41
42 INSERT INTO ContieneTracce(ID_Traccia , ID_BraniPreferiti)
43 VALUES ( '17' , '3' );
44
45 INSERT INTO ContieneTracce(ID_Traccia , ID_BraniPreferiti)
46 VALUES ( '1' , '4' );
47
48 INSERT INTO ContieneTracce(ID_Traccia , ID_BraniPreferiti)
49 VALUES ( '10' , '5' );
50
51 INSERT INTO ContieneTracce(ID_Traccia , ID_BraniPreferiti)
52 VALUES ( '5' , '5' );
53
54 INSERT INTO ContieneTracce(ID_Traccia , ID_BraniPreferiti)
55 VALUES ( '6' , '5' );
56
57 INSERT INTO ContieneTracce(ID_Traccia , ID_BraniPreferiti)
58 VALUES ( '3' , '6' );
59
60 INSERT INTO ContieneTracce(ID_Traccia , ID_BraniPreferiti)
61 VALUES ( '9' , '6' );
62
63 INSERT INTO ContieneTracce(ID_Traccia , ID_BraniPreferiti)
64 VALUES ( '7' , '7' );
65
66 INSERT INTO ContieneTracce(ID_Traccia , ID_BraniPreferiti)
67 VALUES ( '9' , '7' );
68
69 INSERT INTO ContieneTracce(ID_Traccia , ID_BraniPreferiti)
70 VALUES ( '10' , '7' );
71
72 INSERT INTO ContieneTracce(ID_Traccia , ID_BraniPreferiti)
73 VALUES ( '2' , '7' );
74
75 INSERT INTO ContieneTracce(ID_Traccia , ID_BraniPreferiti)
76 VALUES ( '1' , '8' );
77
78 INSERT INTO ContieneTracce(ID_Traccia , ID_BraniPreferiti)
79 VALUES ( '10' , '8' );

```