



Record Label Database Design

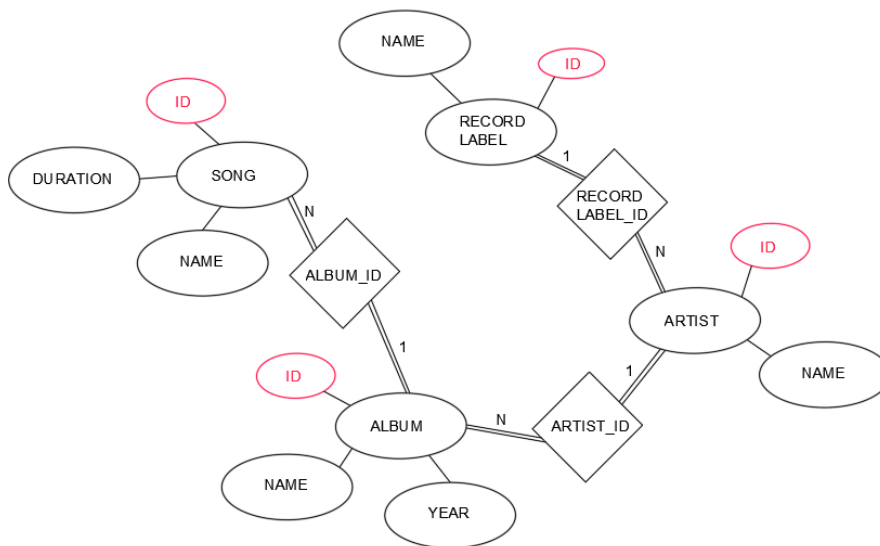
ID: 4815127

Name: PATRICIA ATTAH

NOTE: PROJECT WRITTEN WITH MYSQL WORKBENCH DUE TO ISSUES USING ORACLE LIVE:

This data base was created for a song streaming app similar to Spotify to keep track of current playlist for each of the subscriber's playlist. The Database keeps track of the artists that performs the songs, their record label they were on while those albums were being produced. Also, all albums that were produced for each artist and the year that album was released. All songs produced by those artists and the lengths of time of each song.

ER Diagram:



DDL Statements (With primary and foreign keys):

```
CREATE TABLE record_label (  
  id int unsigned not null,  
  name varchar(50) not null,  
  PRIMARY KEY (id),  
  UNIQUE KEY uk_name_in_record_label (name)  
);
```

```
CREATE TABLE artist (  
  id int unsigned not null,  
  record_label_id int unsigned not null,  
  name varchar(50) not null,  
  PRIMARY KEY (id),  
  KEY fk_record_label_in_artist (record_label_id),
```



```
CONSTRAINT fk_record_label_in_artist FOREIGN KEY (record_label_id) REFERENCES record_label (id),  
UNIQUE KEY uk_name_in_artist (record_label_id, name)  
);
```

```
CREATE TABLE album (  
  id      int unsigned not null,  
  artist_id int unsigned not null,  
  name    varchar(50) not null,  
  year    int unsigned not null,  
  PRIMARY KEY (id),  
  KEY fk_artist_in_album (artist_id),  
  CONSTRAINT fk_artist_in_album FOREIGN KEY (artist_id) REFERENCES artist (id),  
  UNIQUE KEY uk_name_in_album (artist_id, name)  
);
```

```
CREATE TABLE song (  
  id int unsigned not null,  
  album_id int unsigned not null,  
  name varchar(50) not null,  
  duration real not null,  
  PRIMARY KEY (id),  
  KEY fk_album_in_song (album_id),  
  CONSTRAINT fk_album_in_song FOREIGN KEY (album_id) REFERENCES album (id),  
  UNIQUE KEY uk_name_in_song (album_id, name)  
);
```

```
INSERT INTO record_label VALUES(1,'Rutgers');  
INSERT INTO record_label VALUES(2,'Warner Bros');  
INSERT INTO record_label VALUES(3,'Universal');  
INSERT INTO record_label VALUES(4,'MCA');
```

```
INSERT INTO Artist VALUES(1, 1,'Metallica');  
INSERT INTO Artist VALUES(2, 1,'Megadeth');  
INSERT INTO Artist VALUES(3, 1,'Anthrax');  
INSERT INTO Artist VALUES(4, 2,'Eric Clapton');
```

```
INSERT INTO album VALUES(20, 16, 'The Rains',2004);  
INSERT INTO album VALUES(21, 17, 'Golden Years',1975);  
INSERT INTO album VALUES(22, 18, 'Greatest Love',2003);  
INSERT INTO album VALUES(23, 19, 'Lonely Hearts', 1967);
```

```
INSERT INTO song VALUES(44,17,'Lovely', 2.7);  
INSERT INTO song VALUES(45,17,'Good Morning', 2.6833);  
INSERT INTO song VALUES(46,18,'Sgt. Pepper's Lonely Hearts (Reprise)', 1.3166);  
INSERT INTO song VALUES(47,18,'A Day in the Life', 5.65);
```

Complex queries:

correlated inner query: the name of songs in each album that that were longer than the average duration for that album.

```
Select a.name, s.name, l.year  
From artist a join album l  
on (a.id = l.artist_id)  
join song s on (s.album_id = l.id)
```



Where s.duration > (Select Avg(duration)
From song g
where s.album_id = g.album_id);

set operation: the song name and album name of music that was produced in year 1988 and 2009

Select s.name, l.name From song s left Join album l on (s.album_id = l.id)
Where year = 1988 Union
Select s.name, l.name From song s left Join album l on (s.album_id = l.id)
Where year = 2009;

inline query: The difference between the average song and other songs.

Select s.name, s.duration, s.duration - M.a Diff From song s, (select Avg(duration) as a
from song) M;

aggregate function: The number of albums produced by year

Select l.year, Count(*)
From album l
group by l.year;

join operations: All albums produced over the year 2000

Select l.name, a.name, l.year
FROM album l inner join artist a
on a.id = l.artist_id
and year > 2000;

Stored Function:

Function returns the duration of a song

DELIMITER \$\$
CREATE function get_bal(acc_no int)
RETURNS int

READS SQL DATA
DETERMINISTIC

BEGIN
DECLARE acc_bal int;
SELECT duration
INTO acc_bal
FROM song
WHERE id = acc_no;
RETURN(acc_bal);
END\$\$
DELIMITER ;

Trigger:

Add an additional year to the year that is entered to the insert statement
DELIMITER \$\$
CREATE TRIGGER TR_INS



SMU

World Changers
Shaped Here

MSDS 7330

```
before INSERT ON album
for each row
begin
    set new.year = new.year + 1;
end$$
DELIMITER ;
```