# Rock Music Database Design

Connor Sargent

May 1st 2015

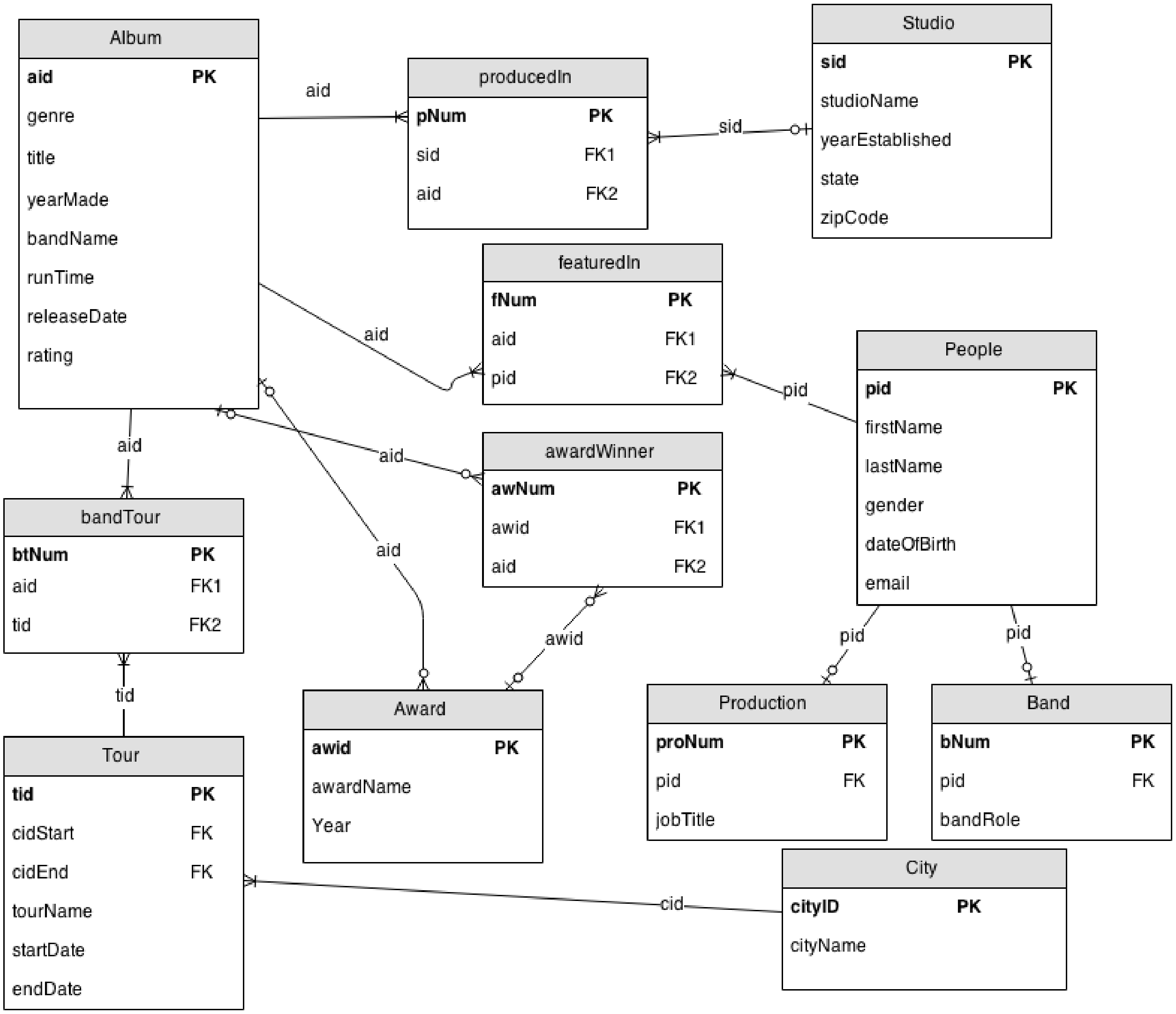
# Table of Contents

# Executive Summary

This document represents the design and implementation of a database for the music in various sub-genres of metal and rock. This database allows the user to access information about the members of the bands, their roles within the band, where they produced their albums, and when and where they will be on tour. This database would be of interest to anyone in need of information about the bands, band members, producers, and awards received by the bands within the database.

First, the entity relationship diagram is shown to demonstrate the relationships within the table. Following the diagram are the Table statements, which includes sample data, SQL code, and functional dependencies. Next shown are the views, stored procedures, and queries for the database. Finally the security features and further information regarding the implementation, known problems, and further enhancements of the database.

# Entity Relationship Diagram



# Table Statements

## Album Table

The Album table contains all of the relevant information about the various albums within the database. From this table we are able to see the albums in the database, as well as the band that wrote them, the genre, the title, run time, and the rating it received by the database. This table is meant to give a general overview of the albums within the music database.

```
CREATE TABLE album

(
  aid text NOT NULL PRIMARY KEY,
  genre character(30),
  title character(35),
  bandName character(30),
  runTime character varying(10),
  releaseDate character varying(15),
  rating integer
);
```

## Functional Dependencies

aid -> genre, title, yearMade, bandName, runTime, releaseDate, rating

## Studio Table

This table shows the information for the studios in which these albums were produced in. This table shows when the studios were established, their names, and their state and zip code information. This will allow the user of the database to know where each album in the database was produced, and where these studios are located to better understand the success of these studios in relation to the albums they produced.

```
CREATE TABLE studio
(
  sid text NOT NULL PRIMARY KEY,
  studioName character(35),
  yearEstablished integer,
  state character(20),
  zipcode integer
);
```

## Functional Dependencies

sid -> studioName, yearEstablished, state, zipCode

## Produced In Table

This table is what actually links the albums with the studio that it was produced in. From this table you will be able to see the album ID's that correspond with the studio ID's, which will show where each one was produced.

```
CREATE TABLE producedIn
(
  pnum integer NOT NULL PRIMARY KEY,
  sid text references studio(sid),
  aid text references album(aid)
);
```

## Functional Dependencies

pnum ->sid, aid

## People Table

The people table holds all the information about the members of each band, as well as producers and managers involved with the creation of the albums listed in the album table. This table holds 50 people that are all contributors to each of the albums in the album table. The people in this table get further broken down to either band members or production members.

```
CREATE TABLE people
(
  pid text NOT NULL PRIMARY KEY,
  firstName character(35),
  lastName character(35),
  gender character(1),
  dateOfBirth character(15),
  email character(50)
);
```

## Functional Dependencies

pid-> firstName, lastName, gender, datOfBirth, email

## Featured In table

This table connects the people in the database to the album in which they worked on. You can see here which people contributed to which album through the use of their people ID's and the album ID's.

```
CREATE TABLE featuredIn
(
  fnum integer NOT NULL PRIMARY KEY,
  aid text references album(aid),
  pid text references people(pid)
);
```

## Functional Dependencies

fNum->aid, pid

## Band

This table takes the people in the people table who are members of the band. It shows their people ID's along with their role in the band. This shows what instrument each member plays in order to see their contribution to the music in the album.

```
CREATE TABLE band
(
  bNum integer NOT NULL PRIMARY KEY,
  pid text references people(pid),
  bandRole character(35)
);
```

## Functional Dependencies

bNum->pid, bandRole

## Production

This table displays the people who took part in the production of the album but were not in the band. For this database it just includes some producers and managers for the bands. Their connection is shown through the people ID's and their job title is described.

```
CREATE TABLE production
(
  proNum integer NOT NULL PRIMARY KEY,
  pid text references people(pid),
  jobTitle character(35)
);
```

## Functional Dependancies

proNum->pid, jobTitle

## Award

This table shows the awards that these albums won. This table has the names of the awards and the year they were won available for display. This table likes to the Award Winner Table to show which album won which award.

```
CREATE TABLE award
(
  awid text NOT NULL PRIMARY KEY,
  awardName character(50),
  awardYear integer
);
```

## Functional Dependencies

awid->awardName, year

## Award Winner

This table links the award ID's with the album ID's to show which albums won which award. Only some of the albums in the database won an award, so the albums that have won can be distinguished from the ones that haven't won any.

```
CREATE TABLE awardWinner
(
  awNum integer NOT NULL PRIMARY KEY,
  awid text references award(awid),
  aid text references album(aid)
);
```

## Functional Dependencies

awNum->awid, aid

## Tour Table

This table shows the different tours that the bands who wrote the albums are currently participating in. This table will show the start and the end city of the tour using the city ID's which references the city table. The table also includes the start and end date of the tournament. This information is useful for venues to know when bands are busy with a tour and when they will be available for other bookings.

```
CREATE TABLE tour
(
  tid text NOT NULL PRIMARY KEY,
  cidStart text references city(cid),
  cidEnd text references city(cid),
  tourName character(35),
  startDate character(15),
  endDate character(15)
);
```

## Functional Dependencies

tid-> cidStart, cidEnd, tourName, startDate, endDate

## Band tour

This table matches the bands through the album ID to their respective tour through the tour ID. This is the table that actually matches the bands with the tours they are on. Several of the bands in the database are on the same tour.

```
CREATE TABLE bandTour
(
  btNum integer NOT NULL PRIMARY KEY,
  aid text references album(aid),
  tid text references tour(tid)
);
```

## Functional Dependancies

btNum->aid, pid

## City

This table holds the cities in which the tours start and end. They all have their own city ID’s and the corresponding city name. This information is relevant to knowing where the band is going with their tour.

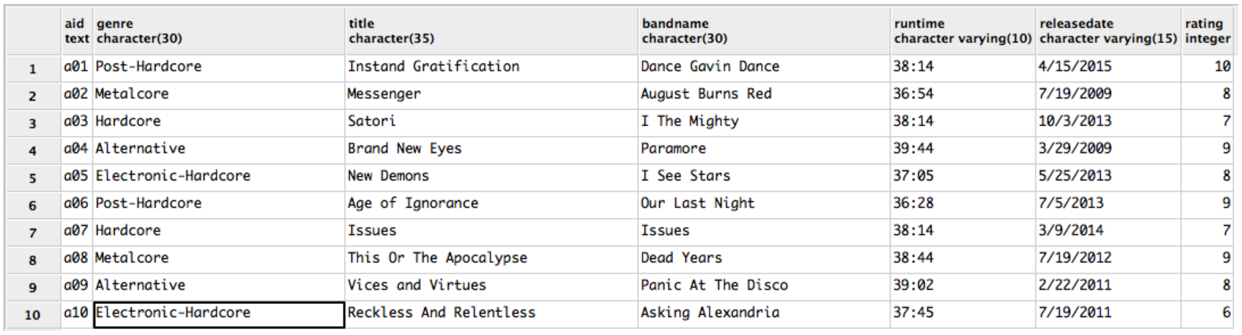
```
CREATE TABLE city
(
cid text NOT NULL PRIMARY KEY,
cityName charachter(50)
);
```

## Functional Dependencies

cid->cityName

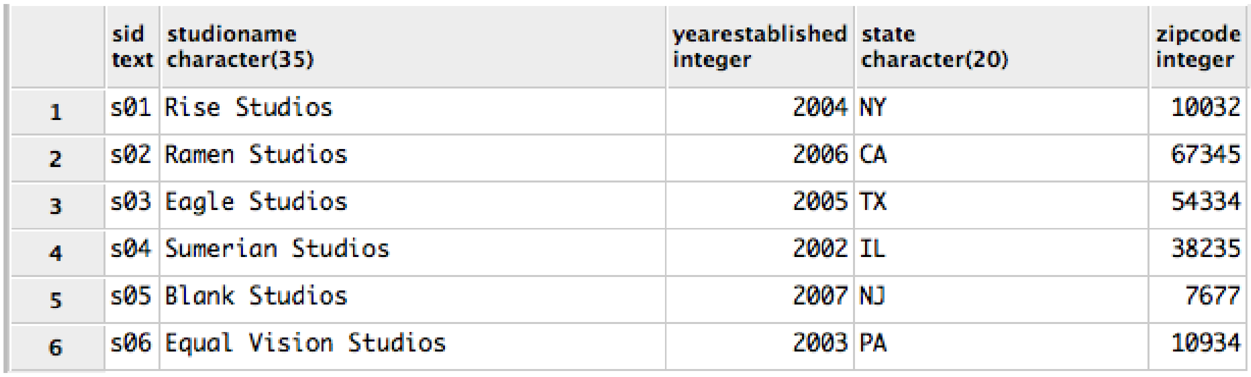
# Sample Data

## Album Table



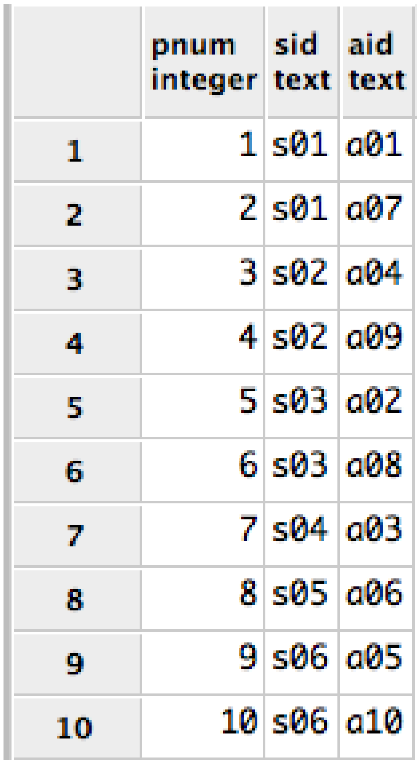
| | aid | genre | title | bandname | runtime | releasedate | rating |
|---|---|---|---|---|---|---|---|
| | text | character(30) | character(35) | character(30) | character varying(10) | character varying(15) | integer |
| 1 | a01 | Post-Hardcore | Instand Gratification | Dance Gavin Dance | 38:14 | 4/15/2015 | 10 |
| 2 | a02 | Metalcore | Messenger | August Burns Red | 36:54 | 7/19/2009 | 8 |
| 3 | a03 | Hardcore | Satori | I The Mighty | 38:14 | 10/3/2013 | 7 |
| 4 | a04 | Alternative | Brand New Eyes | Paramore | 39:44 | 3/29/2009 | 9 |
| 5 | a05 | Electronic-Hardcore | New Demons | I See Stars | 37:05 | 5/25/2013 | 8 |
| 6 | a06 | Post-Hardcore | Age of Ignorance | Our Last Night | 36:28 | 7/5/2013 | 9 |
| 7 | a07 | Hardcore | Issues | Issues | 38:14 | 3/9/2014 | 7 |
| 8 | a08 | Metalcore | This Or The Apocalypse | Dead Years | 38:44 | 7/19/2012 | 9 |
| 9 | a09 | Alternative | Vices and Virtues | Panic At The Disco | 39:02 | 2/22/2011 | 8 |
| 10 | a10 | Electronic-Hardcore | Reckless And Relentless | Asking Alexandria | 37:45 | 7/19/2011 | 6 |

## Studio Table



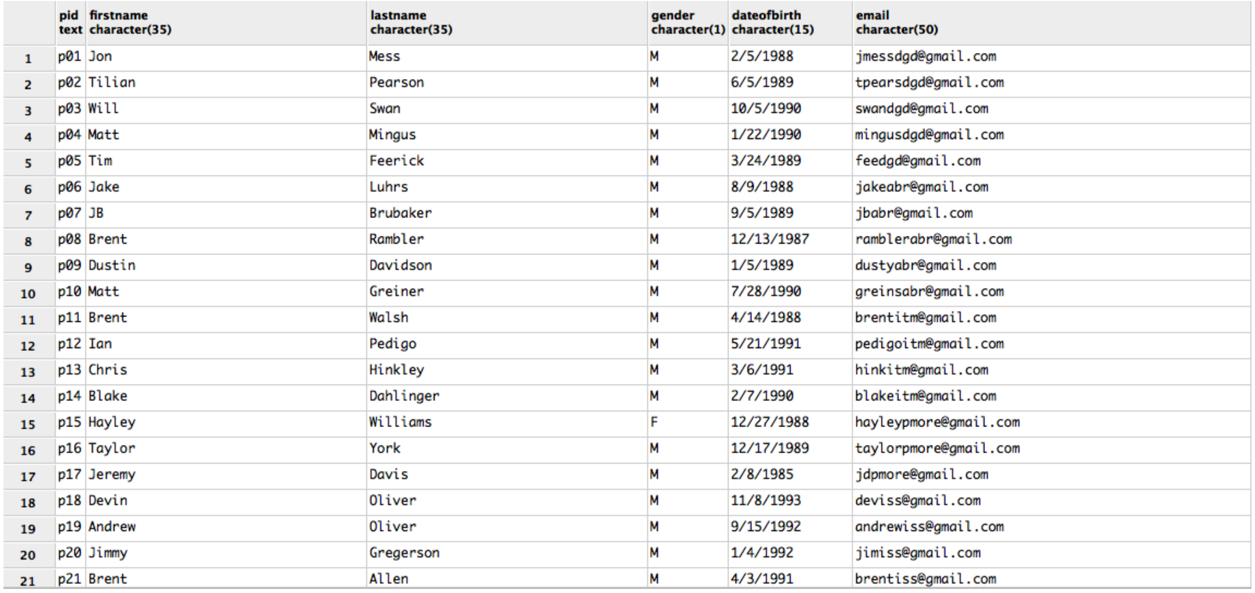
| | sid | studioname | yearestablished | state | zipcode |
|---|---|---|---|---|---|
| | text | character(35) | integer | character(20) | integer |
| 1 | s01 | Rise Studios | 2004 | NY | 10032 |
| 2 | s02 | Ramen Studios | 2006 | CA | 67345 |
| 3 | s03 | Eagle Studios | 2005 | TX | 54334 |
| 4 | s04 | Sumerian Studios | 2002 | IL | 38235 |
| 5 | s05 | Blank Studios | 2007 | NJ | 7677 |
| 6 | s06 | Equal Vision Studios | 2003 | PA | 10934 |

## ProducedIn Table

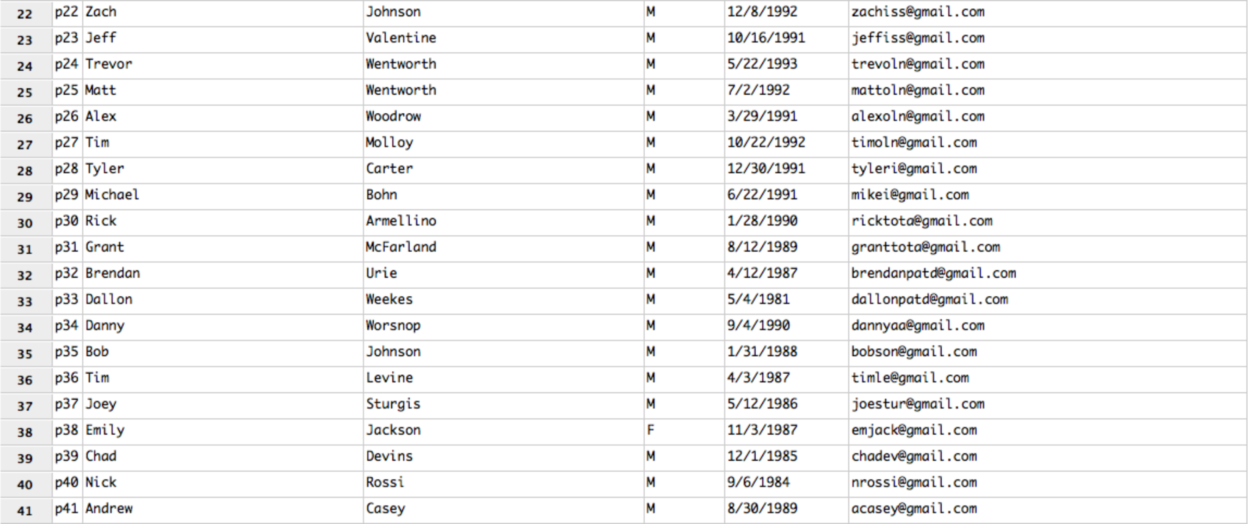


| | pnum<br>integer | sid<br>text | aid<br>text |
|---|---|---|---|
| 1 | 1 | s01 | a01 |
| 2 | 2 | s01 | a07 |
| 3 | 3 | s02 | a04 |
| 4 | 4 | s02 | a09 |
| 5 | 5 | s03 | a02 |
| 6 | 6 | s03 | a08 |
| 7 | 7 | s04 | a03 |
| 8 | 8 | s05 | a06 |
| 9 | 9 | s06 | a05 |
| 10 | 10 | s06 | a10 |

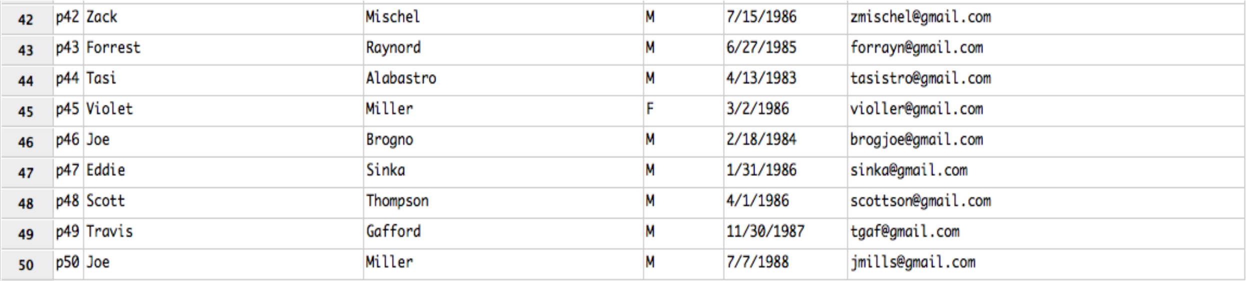
## People Table



| | pid<br>text | firstname<br>character(35) | lastname<br>character(35) | gender<br>character(1) | dateofbirth<br>character(15) | email<br>character(50) |
|---|---|---|---|---|---|---|
| 1 | p01 | Jon | Mess | M | 2/5/1988 | jmessdgd@gmail.com |
| 2 | p02 | Tilian | Pearson | M | 6/5/1989 | tpearsdgd@gmail.com |
| 3 | p03 | Will | Swan | M | 10/5/1990 | swandgd@gmail.com |
| 4 | p04 | Matt | Mingus | M | 1/22/1990 | mingusdgd@gmail.com |
| 5 | p05 | Tim | Feerick | M | 3/24/1989 | feedgd@gmail.com |
| 6 | p06 | Jake | Luhrs | M | 8/9/1988 | jakeabr@gmail.com |
| 7 | p07 | JB | Brubaker | M | 9/5/1989 | jbabr@gmail.com |
| 8 | p08 | Brent | Rambler | M | 12/13/1987 | ramblerabr@gmail.com |
| 9 | p09 | Dustin | Davidson | M | 1/5/1989 | dustyabr@gmail.com |
| 10 | p10 | Matt | Greiner | M | 7/28/1990 | greinsabr@gmail.com |
| 11 | p11 | Brent | Walsh | M | 4/14/1988 | brentitm@gmail.com |
| 12 | p12 | Ian | Pedigo | M | 5/21/1991 | pedigoitm@gmail.com |
| 13 | p13 | Chris | Hinkley | M | 3/6/1991 | hinkitm@gmail.com |
| 14 | p14 | Blake | Dahlinger | M | 2/7/1990 | blakeitm@gmail.com |
| 15 | p15 | Hayley | Williams | F | 12/27/1988 | hayleypmore@gmail.com |
| 16 | p16 | Taylor | York | M | 12/17/1989 | taylorpmore@gmail.com |
| 17 | p17 | Jeremy | Davis | M | 2/8/1985 | jdpmore@gmail.com |
| 18 | p18 | Devin | Oliver | M | 11/8/1993 | deviss@gmail.com |
| 19 | p19 | Andrew | Oliver | M | 9/15/1992 | andrewiss@gmail.com |
| 20 | p20 | Jimmy | Gregerson | M | 1/4/1992 | jimiss@gmail.com |
| 21 | p21 | Brent | Allen | M | 4/3/1991 | brentiss@gmail.com |

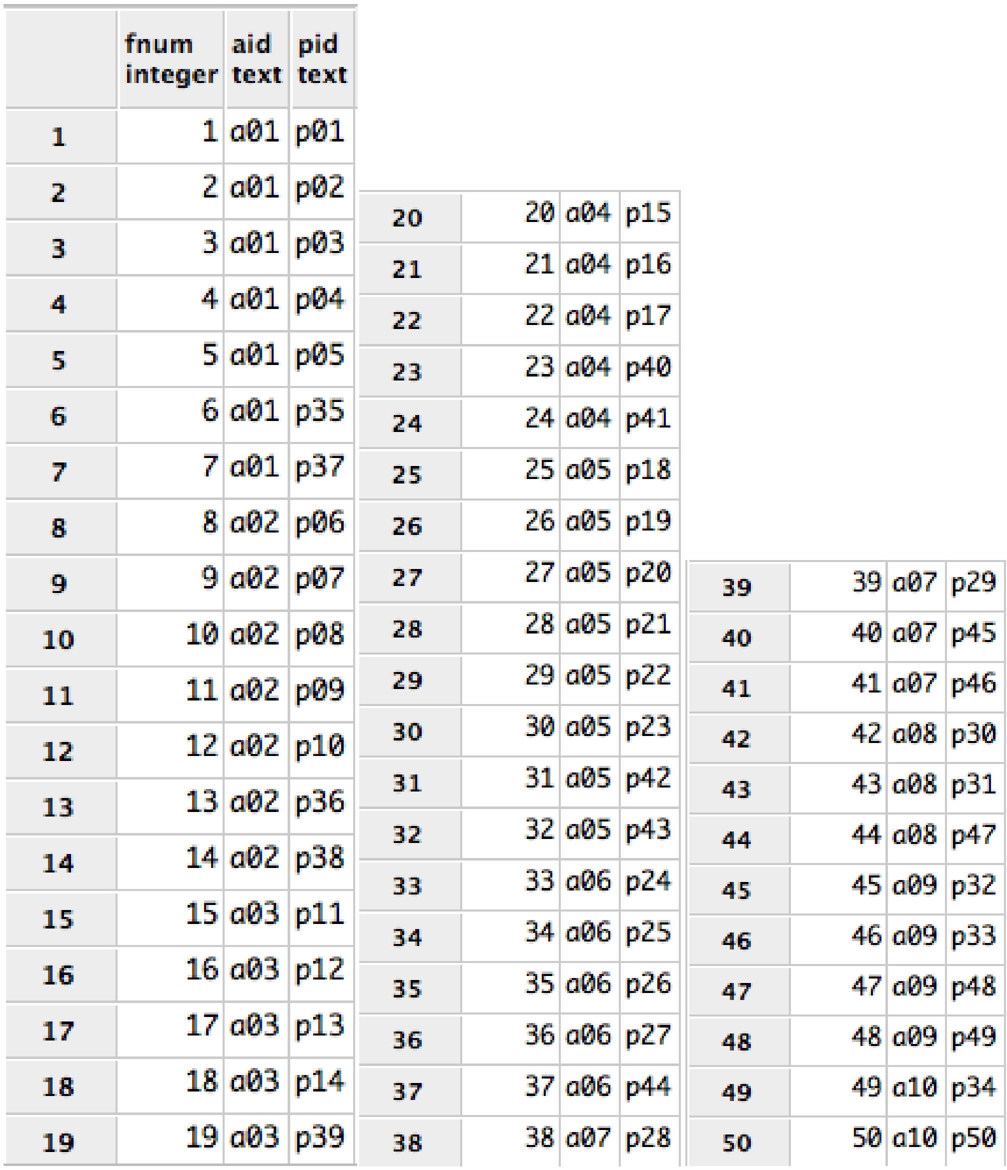


| | | | | | | |
|---|---|---|---|---|---|---|
| 22 | p22 | Zach | Johnson | M | 12/8/1992 | zachiss@gmail.com |
| 23 | p23 | Jeff | Valentine | M | 10/16/1991 | jeffiss@gmail.com |
| 24 | p24 | Trevor | Wentworth | M | 5/22/1993 | trevoln@gmail.com |
| 25 | p25 | Matt | Wentworth | M | 7/2/1992 | mattoln@gmail.com |
| 26 | p26 | Alex | Woodrow | M | 3/29/1991 | alexoln@gmail.com |
| 27 | p27 | Tim | Molloy | M | 10/22/1992 | timoln@gmail.com |
| 28 | p28 | Tyler | Carter | M | 12/30/1991 | tyleri@gmail.com |
| 29 | p29 | Michael | Bohn | M | 6/22/1991 | mikei@gmail.com |
| 30 | p30 | Rick | Armellino | M | 1/28/1990 | ricktota@gmail.com |
| 31 | p31 | Grant | McFarland | M | 8/12/1989 | granttota@gmail.com |
| 32 | p32 | Brendan | Urie | M | 4/12/1987 | brendanpatd@gmail.com |
| 33 | p33 | Dallon | Weekes | M | 5/4/1981 | dallonpatd@gmail.com |
| 34 | p34 | Danny | Worsnop | M | 9/4/1990 | dannyaa@gmail.com |
| 35 | p35 | Bob | Johnson | M | 1/31/1988 | bobson@gmail.com |
| 36 | p36 | Tim | Levine | M | 4/3/1987 | timle@gmail.com |
| 37 | p37 | Joey | Sturgis | M | 5/12/1986 | joestur@gmail.com |
| 38 | p38 | Emily | Jackson | F | 11/3/1987 | emjack@gmail.com |
| 39 | p39 | Chad | Devins | M | 12/1/1985 | chadev@gmail.com |
| 40 | p40 | Nick | Rossi | M | 9/6/1984 | nrossi@gmail.com |
| 41 | p41 | Andrew | Casey | M | 8/30/1989 | acasey@gmail.com |



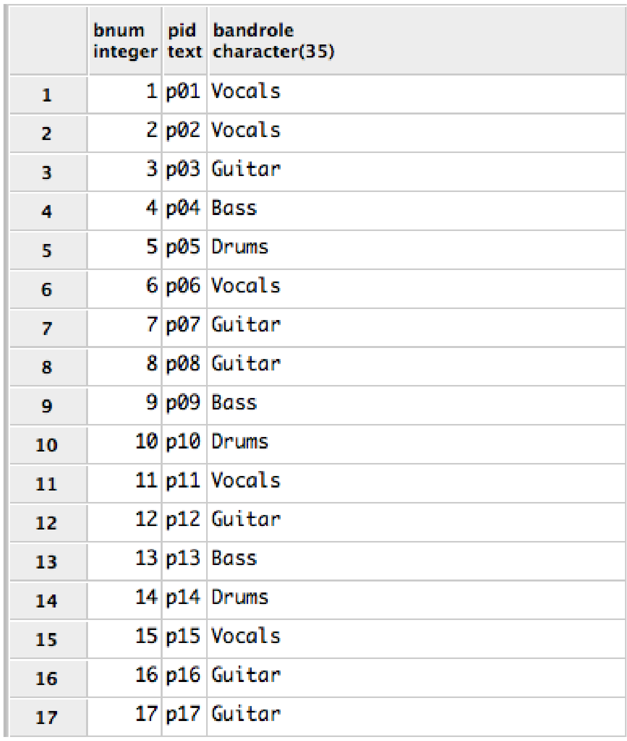
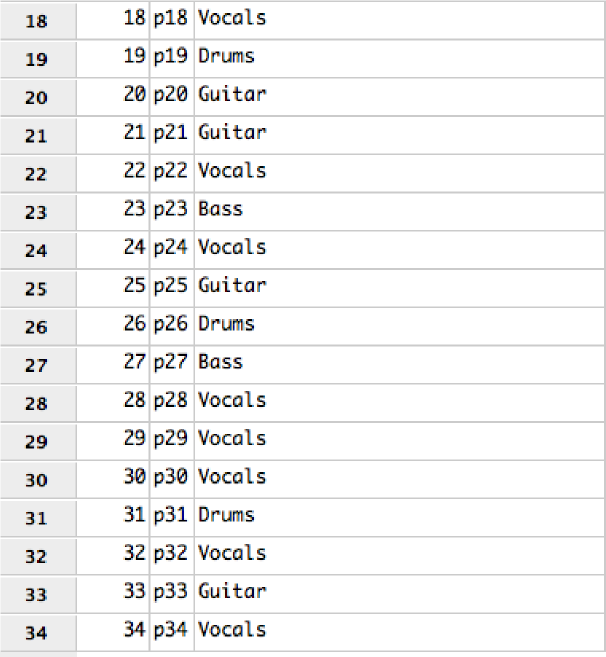
| | | | | | | |
|---|---|---|---|---|---|---|
| 42 | p42 | Zack | Mischel | M | 7/15/1986 | zmischel@gmail.com |
| 43 | p43 | Forrest | Raynord | M | 6/27/1985 | forrayn@gmail.com |
| 44 | p44 | Tasi | Alabastro | M | 4/13/1983 | tasistro@gmail.com |
| 45 | p45 | Violet | Miller | F | 3/2/1986 | violler@gmail.com |
| 46 | p46 | Joe | Brogno | M | 2/18/1984 | brogjoe@gmail.com |
| 47 | p47 | Eddie | Sinka | M | 1/31/1986 | sinka@gmail.com |
| 48 | p48 | Scott | Thompson | M | 4/1/1986 | scottson@gmail.com |
| 49 | p49 | Travis | Gafford | M | 11/30/1987 | tgaf@gmail.com |
| 50 | p50 | Joe | Miller | M | 7/7/1988 | jmills@gmail.com |

## FeaturedIn Table



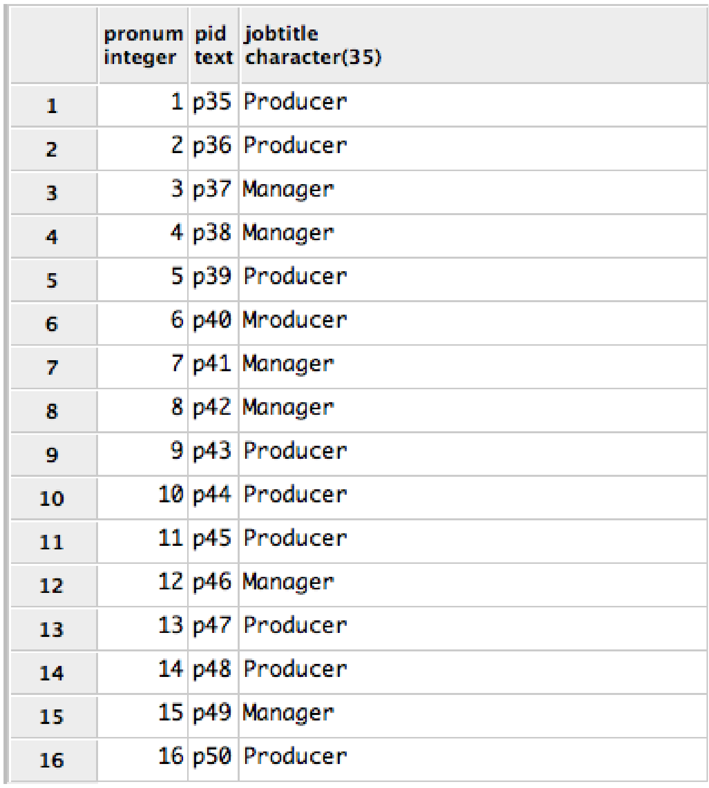
| | fnum | aid | pid |
|---|---|---|---|
| | integer | text | text |
| 1 | 1 | a01 | p01 |
| 2 | 2 | a01 | p02 |
| 3 | 3 | a01 | p03 |
| 4 | 4 | a01 | p04 |
| 5 | 5 | a01 | p05 |
| 6 | 6 | a01 | p35 |
| 7 | 7 | a01 | p37 |
| 8 | 8 | a02 | p06 |
| 9 | 9 | a02 | p07 |
| 10 | 10 | a02 | p08 |
| 11 | 11 | a02 | p09 |
| 12 | 12 | a02 | p10 |
| 13 | 13 | a02 | p36 |
| 14 | 14 | a02 | p38 |
| 15 | 15 | a03 | p11 |
| 16 | 16 | a03 | p12 |
| 17 | 17 | a03 | p13 |
| 18 | 18 | a03 | p14 |
| 19 | 19 | a03 | p39 |
| 20 | 20 | a04 | p15 |
| 21 | 21 | a04 | p16 |
| 22 | 22 | a04 | p17 |
| 23 | 23 | a04 | p40 |
| 24 | 24 | a04 | p41 |
| 25 | 25 | a05 | p18 |
| 26 | 26 | a05 | p19 |
| 27 | 27 | a05 | p20 |
| 28 | 28 | a05 | p21 |
| 29 | 29 | a05 | p22 |
| 30 | 30 | a05 | p23 |
| 31 | 31 | a05 | p42 |
| 32 | 32 | a05 | p43 |
| 33 | 33 | a06 | p24 |
| 34 | 34 | a06 | p25 |
| 35 | 35 | a06 | p26 |
| 36 | 36 | a06 | p27 |
| 37 | 37 | a06 | p44 |
| 38 | 38 | a07 | p28 |
| 39 | 39 | a07 | p29 |
| 40 | 40 | a07 | p45 |
| 41 | 41 | a07 | p46 |
| 42 | 42 | a08 | p30 |
| 43 | 43 | a08 | p31 |
| 44 | 44 | a08 | p47 |
| 45 | 45 | a09 | p32 |
| 46 | 46 | a09 | p33 |
| 47 | 47 | a09 | p48 |
| 48 | 48 | a09 | p49 |
| 49 | 49 | a10 | p34 |
| 50 | 50 | a10 | p50 |

## Band

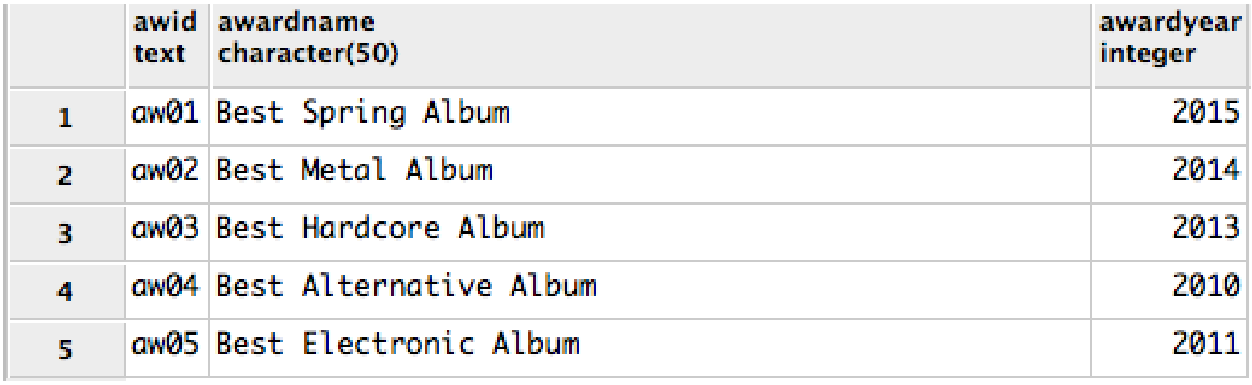
| | bnum | pid | bandrole |
|---|---|---|---|
| | integer | text | character(35) |
| 1 | 1 | p01 | Vocals |
| 2 | 2 | p02 | Vocals |
| 3 | 3 | p03 | Guitar |
| 4 | 4 | p04 | Bass |
| 5 | 5 | p05 | Drums |
| 6 | 6 | p06 | Vocals |
| 7 | 7 | p07 | Guitar |
| 8 | 8 | p08 | Guitar |
| 9 | 9 | p09 | Bass |
| 10 | 10 | p10 | Drums |
| 11 | 11 | p11 | Vocals |
| 12 | 12 | p12 | Guitar |
| 13 | 13 | p13 | Bass |
| 14 | 14 | p14 | Drums |
| 15 | 15 | p15 | Vocals |
| 16 | 16 | p16 | Guitar |
| 17 | 17 | p17 | Guitar |
| 18 | 18 | p18 | Vocals |
| 19 | 19 | p19 | Drums |
| 20 | 20 | p20 | Guitar |
| 21 | 21 | p21 | Guitar |
| 22 | 22 | p22 | Vocals |
| 23 | 23 | p23 | Bass |
| 24 | 24 | p24 | Vocals |
| 25 | 25 | p25 | Guitar |
| 26 | 26 | p26 | Drums |
| 27 | 27 | p27 | Bass |
| 28 | 28 | p28 | Vocals |
| 29 | 29 | p29 | Vocals |
| 30 | 30 | p30 | Vocals |
| 31 | 31 | p31 | Drums |
| 32 | 32 | p32 | Vocals |
| 33 | 33 | p33 | Guitar |
| 34 | 34 | p34 | Vocals |

## Production



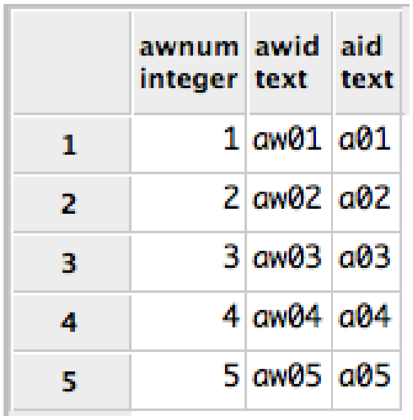
| | pronum | pid | jobtitle |
|---|---|---|---|
| | integer | text | character(35) |
| 1 | 1 | p35 | Producer |
| 2 | 2 | p36 | Producer |
| 3 | 3 | p37 | Manager |
| 4 | 4 | p38 | Manager |
| 5 | 5 | p39 | Producer |
| 6 | 6 | p40 | Mroducer |
| 7 | 7 | p41 | Manager |
| 8 | 8 | p42 | Manager |
| 9 | 9 | p43 | Producer |
| 10 | 10 | p44 | Producer |
| 11 | 11 | p45 | Producer |
| 12 | 12 | p46 | Manager |
| 13 | 13 | p47 | Producer |
| 14 | 14 | p48 | Producer |
| 15 | 15 | p49 | Manager |
| 16 | 16 | p50 | Producer |

## Award



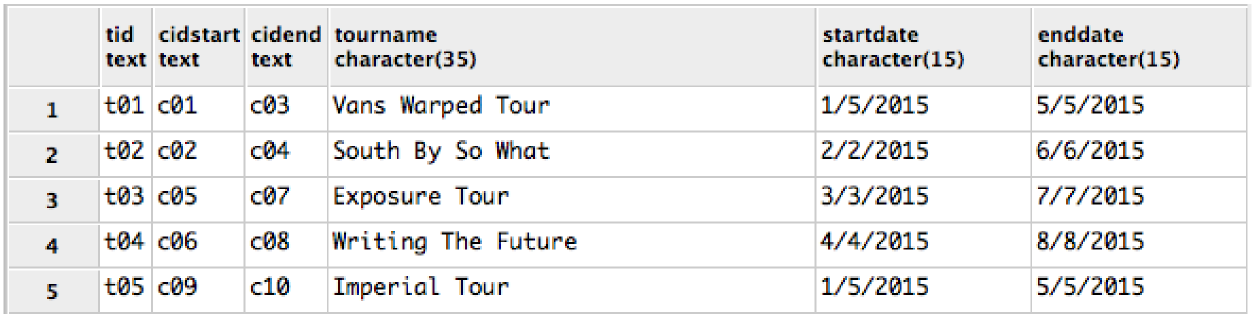
| | awid<br>text | awardname<br>character(50) | awardyear<br>integer |
|---|---|---|---|
| 1 | aw01 | Best Spring Album | 2015 |
| 2 | aw02 | Best Metal Album | 2014 |
| 3 | aw03 | Best Hardcore Album | 2013 |
| 4 | aw04 | Best Alternative Album | 2010 |
| 5 | aw05 | Best Electronic Album | 2011 |

## AwardWinner



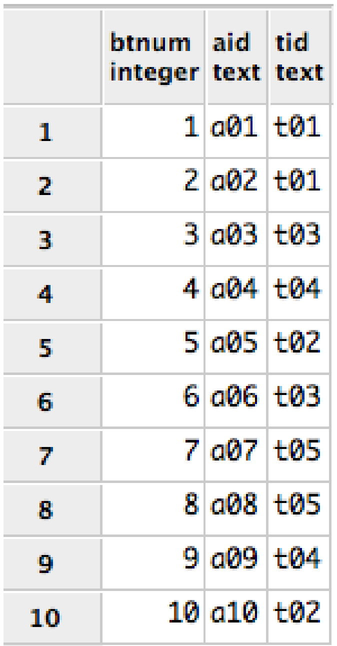
| | awnum<br>integer | awid<br>text | aid<br>text |
|---|---|---|---|
| 1 | 1 | aw01 | a01 |
| 2 | 2 | aw02 | a02 |
| 3 | 3 | aw03 | a03 |
| 4 | 4 | aw04 | a04 |
| 5 | 5 | aw05 | a05 |

## Tour



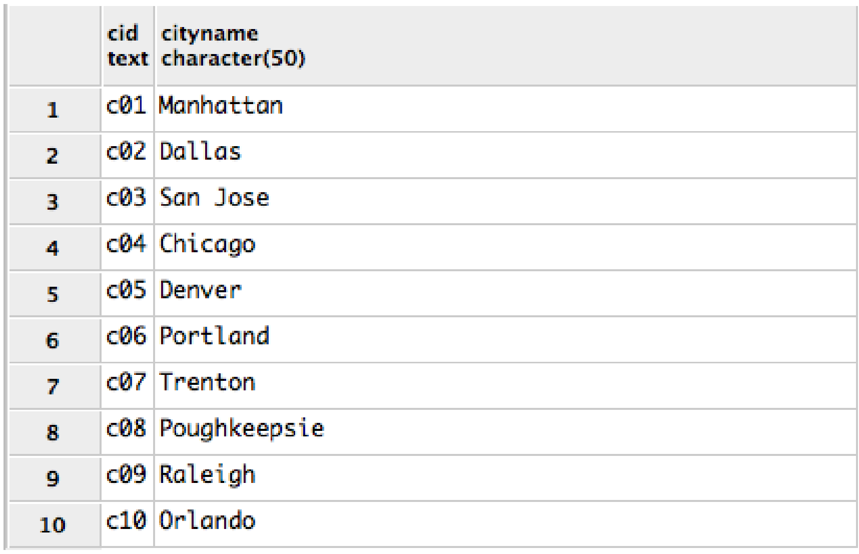
| | tid<br>text | cidstart<br>text | cidend<br>text | tourname<br>character(35) | startdate<br>character(15) | enddate<br>character(15) |
|---|---|---|---|---|---|---|
| 1 | t01 | c01 | c03 | Vans Warped Tour | 1/5/2015 | 5/5/2015 |
| 2 | t02 | c02 | c04 | South By So What | 2/2/2015 | 6/6/2015 |
| 3 | t03 | c05 | c07 | Exposure Tour | 3/3/2015 | 7/7/2015 |
| 4 | t04 | c06 | c08 | Writing The Future | 4/4/2015 | 8/8/2015 |
| 5 | t05 | c09 | c10 | Imperial Tour | 1/5/2015 | 5/5/2015 |

## BandTour



| | btnum<br>integer | aid<br>text | tid<br>text |
|---|---|---|---|
| 1 | 1 | a01 | t01 |
| 2 | 2 | a02 | t01 |
| 3 | 3 | a03 | t03 |
| 4 | 4 | a04 | t04 |
| 5 | 5 | a05 | t02 |
| 6 | 6 | a06 | t03 |
| 7 | 7 | a07 | t05 |
| 8 | 8 | a08 | t05 |
| 9 | 9 | a09 | t04 |
| 10 | 10 | a10 | t02 |

## City



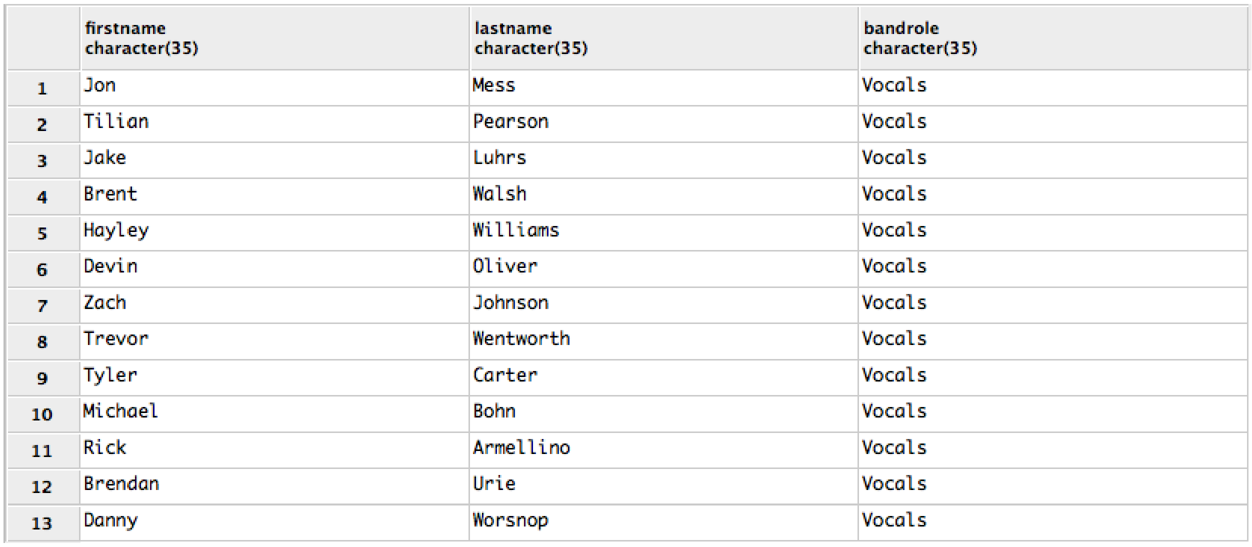
| | cid<br>text | cityname<br>character(50) |
|---|---|---|
| 1 | c01 | Manhattan |
| 2 | c02 | Dallas |
| 3 | c03 | San Jose |
| 4 | c04 | Chicago |
| 5 | c05 | Denver |
| 6 | c06 | Portland |
| 7 | c07 | Trenton |
| 8 | c08 | Poughkeepsie |
| 9 | c09 | Raleigh |
| 10 | c10 | Orlando |

# Views

## Vocalist View

```
create view vocals as
       select a.firstName, a.lastName, b.bandRole
       from people a, band b
       where a.pid = b.pid
       and(bandRole = 'Vocals');

select *
from vocals
```



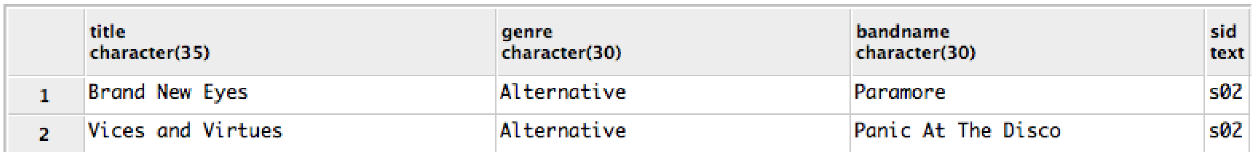
| | firstname<br>character(35) | lastname<br>character(35) | bandrole<br>character(35) |
|---|---|---|---|
| 1 | Jon | Mess | Vocals |
| 2 | Tilian | Pearson | Vocals |
| 3 | Jake | Luhrs | Vocals |
| 4 | Brent | Walsh | Vocals |
| 5 | Hayley | Williams | Vocals |
| 6 | Devin | Oliver | Vocals |
| 7 | Zach | Johnson | Vocals |
| 8 | Trevor | Wentworth | Vocals |
| 9 | Tyler | Carter | Vocals |
| 10 | Michael | Bohn | Vocals |
| 11 | Rick | Armellino | Vocals |
| 12 | Brendan | Urie | Vocals |
| 13 | Danny | Worsnop | Vocals |

There are several different roles within a band between all the different instruments and vocalists. This view is used to select the names of all the vocalists within the database for the viewer's use. The same could be done with the various instruments within a band in order to see a list of the people who play a certain instrument. This view's main purpose is for the user to be able to locate the names of the instrument of their choosing.

## Genre View

```
create view altAlbum as
       select c.title, c.genre, c.bandName, d.sid
       from album c, producedIn d
       where c.aid = d.aid
       and(genre = 'Alternative');
```

```
select *
from altAlbum
```



| | title<br>character(35) | genre<br>character(30) | bandname<br>character(30) | sid<br>text |
|---|---|---|---|---|
| 1 | Brand New Eyes | Alternative | Paramore | s02 |
| 2 | Vices and Virtues | Alternative | Panic At The Disco | s02 |

Users of the database may want to find bands via genre. This view shows the table of the albums in the database under the “Alternative” genre. As a result, the user is able to locate the names of the album and band that is under the Alternative genre, as well as the corresponding studio ID to show where these albums were produced.

# Stored Procedures

## Top Rated Albums

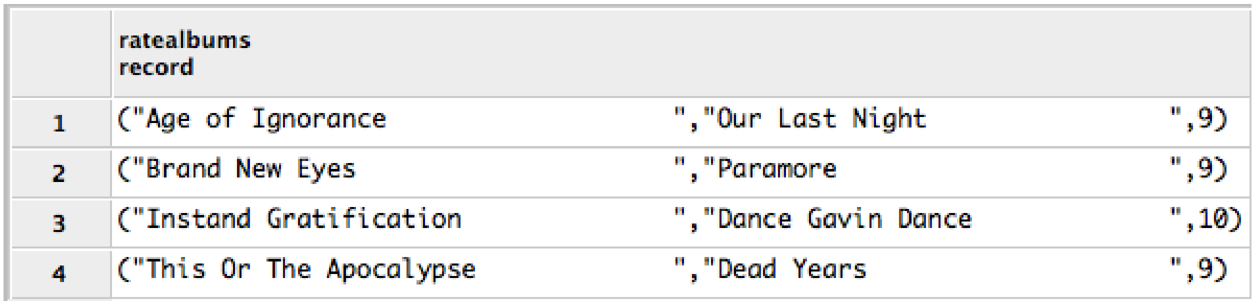
```
create or replace function rateAlbums(rating int)
returns table("Title" char(30), "Band Name" char(30), "Rating" int) AS $$

begin
	RETURN QUERY SELECT a.title AS "title", a.bandName AS "bandName", a.rating AS 
"rating"
	FROM album a
	WHERE a.rating > 8
	ORDER BY a.title, a.bandName, a.rating;

end;
$$
language plpgsql;
```

The following stored procedure rateAlbums will display the albums with a rating of 9 or higher. This procedure is included for the user who is interested in viewing the top rated albums to get a table which shows the albums with the highest ratings.

Sample Data: `select rateAlbums(9)`



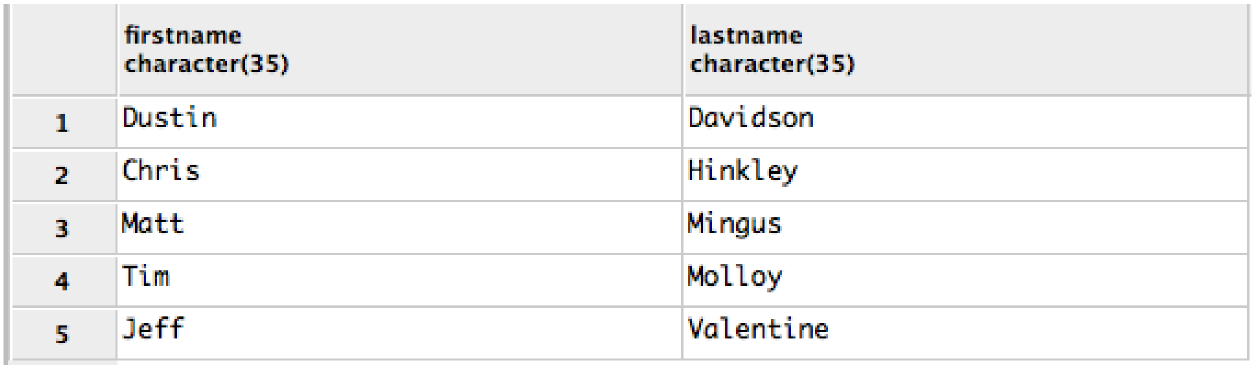
| | ratealbums<br>record |
|---|---|
| 1 | ("Age of Ignorance                    ","Our Last Night                  ",9) |
| 2 | ("Brand New Eyes                      ","Paramore                        ",9) |
| 3 | ("Instand Gratification               ","Dance Gavin Dance               ",10) |
| 4 | ("This Or The Apocalypse              ","Dead Years                      ",9) |

# Queries

## Bass Players

As a bass player myself, I know that sometimes their hard work and talent is underappreciated by many. This query highlights all the names of the bassists within the database in alphabetical order by last name.

```
select firstName, lastName
from people
where pid in
(select pid from band where bandRole = 'Bass')
order by lastName asc
```



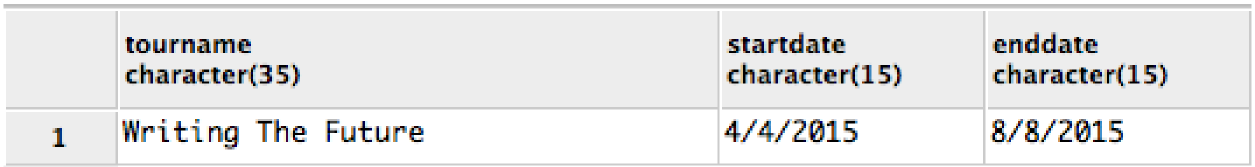
| | firstname<br>character(35) | lastname<br>character(35) |
|---|---|---|
| 1 | Dustin | Davidson |
| 2 | Chris | Hinkley |
| 3 | Matt | Mingus |
| 4 | Tim | Molloy |
| 5 | Jeff | Valentine |

## Find a tour that ends in your city

This Query allows the user to search for a tour that passes through the city in which they input. For the purpose of this example, the table will show the tours which end in Poughkeepsie, as well as information of the tour such as the start and end date.

```
select tourName, startDate, endDate
```

```
from tour
where cidEnd in
(select cid from city where cityName = 'Poughkeepsie')
```

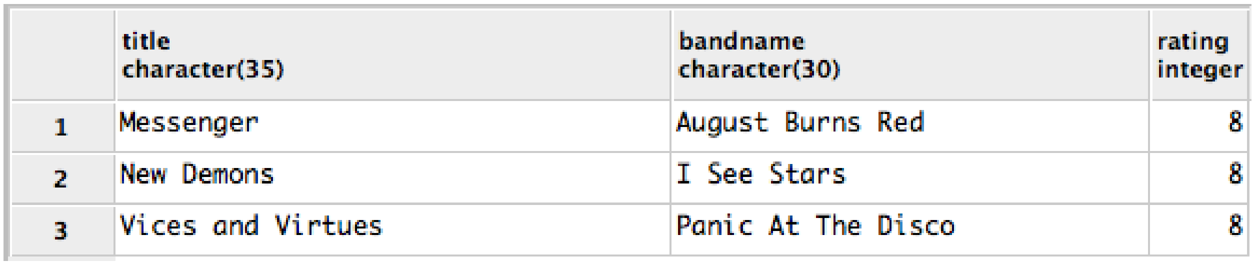


| | tourname<br>character(35) | startdate<br>character(15) | enddate<br>character(15) |
|---|---|---|---|
| 1 | Writing The Future | 4/4/2015 | 8/8/2015 |

## Find albums through rating

This Query allows the user to input a rating of their choice to gather the albums in the database that were given the rating in which the user input. For example, this is the result of a user searching for albums that received an 8 rating.

```
select title, bandName, rating
from album
where aid in
(select aid from album where rating = 8)
```

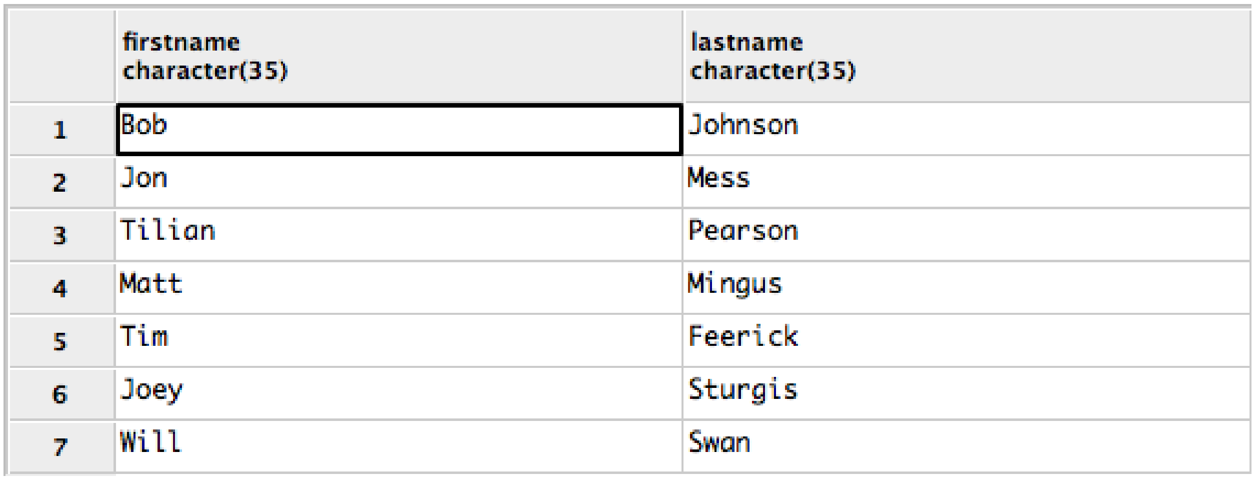


| | title<br>character(35) | bandname<br>character(30) | rating<br>integer |
|---|---|---|---|
| 1 | Messenger | August Burns Red | 8 |
| 2 | New Demons | I See Stars | 8 |
| 3 | Vices and Virtues | Panic At The Disco | 8 |

## Find album contributors

This Query lets the user input an album ID and find the names of the contributors of the album whether they are band members, producers, or managers. For this query the contributors shown are under the album ID 1.

```
select firstName, lastName
from people
where pid in
(select pid from featuredIn where aid = 'a01')
```

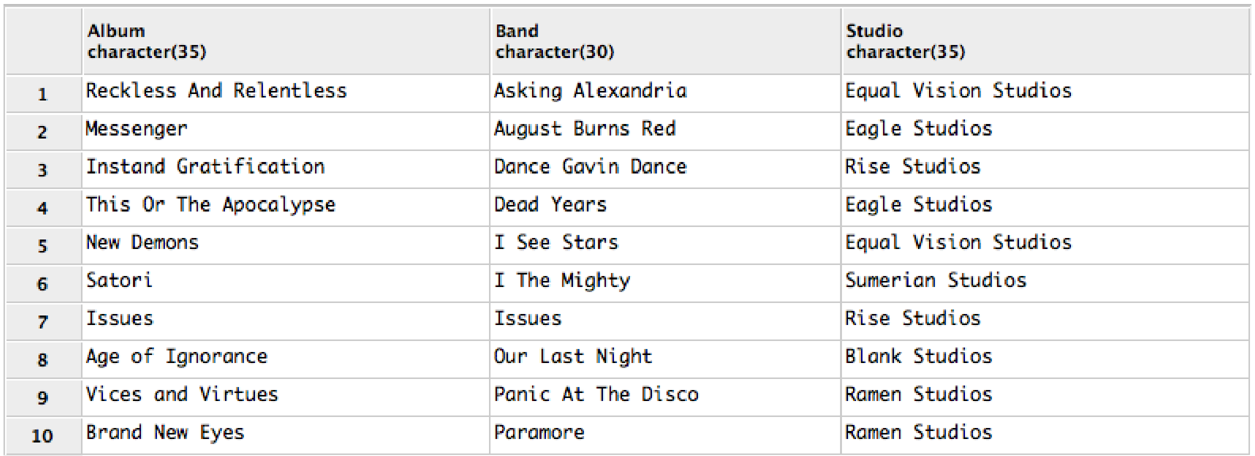


| | firstname<br>character(35) | lastname<br>character(35) |
|---|---|---|
| 1 | Bob | Johnson |
| 2 | Jon | Mess |
| 3 | Tilian | Pearson |
| 4 | Matt | Mingus |
| 5 | Tim | Feerick |
| 6 | Joey | Sturgis |
| 7 | Will | Swan |

## Where the album was produced

This table shows the band who wrote the album, the name of the album, and what studio it was produced in.

```
select a.title AS "Album", a.bandName AS "Band", b.studioName AS "Studio"
from album a, studio b, producedIn c
where a.aid = c.aid AND b.sid = c.sid
order by a.bandName
```

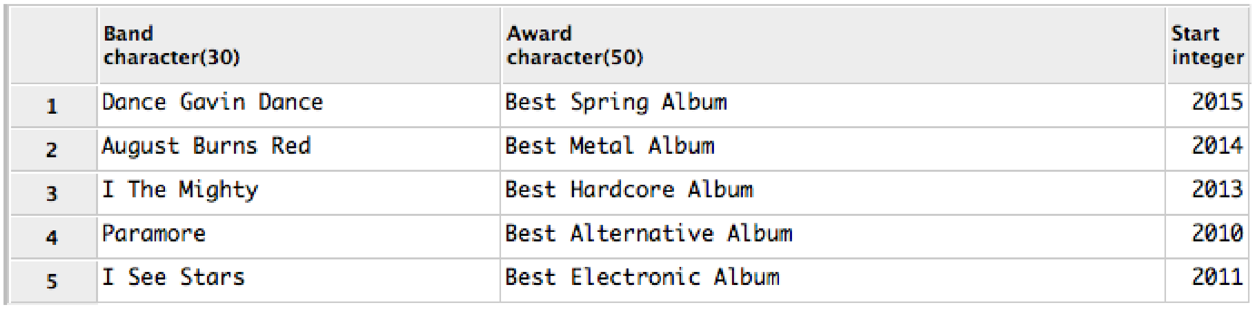


| | Album<br>character(35) | Band<br>character(30) | Studio<br>character(35) |
|---|---|---|---|
| 1 | Reckless And Relentless | Asking Alexandria | Equal Vision Studios |
| 2 | Messenger | August Burns Red | Eagle Studios |
| 3 | Instand Gratification | Dance Gavin Dance | Rise Studios |
| 4 | This Or The Apocalypse | Dead Years | Eagle Studios |
| 5 | New Demons | I See Stars | Equal Vision Studios |
| 6 | Satori | I The Mighty | Sumerian Studios |
| 7 | Issues | Issues | Rise Studios |
| 8 | Age of Ignorance | Our Last Night | Blank Studios |
| 9 | Vices and Virtues | Panic At The Disco | Ramen Studios |
| 10 | Brand New Eyes | Paramore | Ramen Studios |

## Award Winning Albums

This table shows the albums within the database that have received awards. It shows the name of the award, which band received it, and the year they received it.

```
select a.bandname AS "Band", b.awardName AS "Award", b.awardYear AS "Start"
from album a, award b, awardWinner c
where a.aid = c.aid AND c.awid = b.awid
```



| | Band<br>character(30) | Award<br>character(50) | Start<br>integer |
|---|---|---|---|
| 1 | Dance Gavin Dance | Best Spring Album | 2015 |
| 2 | August Burns Red | Best Metal Album | 2014 |
| 3 | I The Mighty | Best Hardcore Album | 2013 |
| 4 | Paramore | Best Alternative Album | 2010 |
| 5 | I See Stars | Best Electronic Album | 2011 |

# Security

Because this database is meant to be for online viewing purposes by an end user, there are only two main users to this database, the Administrator and the End User. The End Users for this database include, but are not limited to, venue managers, booking agents, record labels, talent agencies, and anyone who wishes to research information on the bands within the database.

## Admin

```
CREATE ROLE admin

GRANT SELECT, INSERT, UPDATE, ALTER

ON ALL TABLES IN SCHEMA PUBLIC

TO admin
```

## End User

```
CREATE ROLE enduser

GRANT SELECT

ON ALL TABLES IN SCHEMA PUBLIC

TO enduser
```

# Implementation Notes, Known Problems, and Future Enhancements.

The implementation of this database went fairly smooth with minimal issues. The main issue with the structure of the whole database is the large amount of tables that may not be entirely necessary, or could be better organized for less clutter and more efficiency in the data. There are many tables that exist simply to link two pairs of ID's together, which made things difficult to keep track of.

Along with the excess of tables, there is also the issue of insufficient data within the database. This is a known problem that could further be expanded upon as more bands, albums, and studios get added to the database. There is enough data to serve its current purpose, but to be more useful to an end user this database would require exponentially more information to be relevant in the real world.