

Course: IST 659  
Author: Samuel L. Peoples  
Assignment: Database Project : Music  
Date: 15 September 2018

*Music Database*

*Samuel L. Peoples*

*IST 659: Data Administration Concepts  
and Database Management*

## *Contents*

<i>Project Description, Business Case, Stakeholders</i>	<i>3</i>
<i>Data Questions</i>	<i>3</i>
<i>Reflection</i>	<i>3</i>
<i>Entities</i>	<i>4</i>
<i>Conceptual Model</i>	<i>5</i>
<i>Logical Model</i>	<i>6</i>
• <i>Model</i>	<i>6</i>
• <i>Design</i>	<i>7</i>
<i>Implementation &amp; Reporting</i>	<i>11</i>
• <i>Member</i>	<i>11</i>
• <i>Instrument</i>	<i>11</i>
• <i>Studio</i>	<i>12</i>
• <i>Artist</i>	<i>12</i>
• <i>Album</i>	<i>12</i>
• <i>Award</i>	<i>13</i>
• <i>Song</i>	<i>13</i>
• <i>Artist Award List</i>	<i>13</i>
• <i>Album Award List</i>	<i>14</i>
• <i>Instrument List</i>	<i>14</i>
• <i>Member List</i>	<i>15</i>
• <i>Data Question Answers</i>	<i>16</i>
• <i>All Songs</i>	<i>23</i>
<i>Physical Database Design (SQL)</i>	<i>32</i>
• <i>Table Creation</i>	<i>32</i>
• <i>Data Input</i>	<i>34</i>
• <i>Stored Procedures</i>	<i>36</i>
• <i>Views</i>	<i>39</i>
• <i>Data Questions</i>	<i>41</i>

## *Project Description*

Over time, I have gathered a large corpus of music, ranging across many genres. Determining what music I have, and the relevant information about various aspects of the songs and albums often becomes cumbersome, and implementing a database solution for this information can help organize and analyze the music library. The stakeholders in this case include myself and music aficionados, who may want to analyze their own music libraries. Once data is added to the database, many questions may be answered which provide insight to the broad range of musical interests one might have.

## *Data Questions*

1. What is the range of album release dates in my music collection?
2. Which genre is the most popular in my music collection? (By count and plays)
3. Which artist/ album has the most awards in my music collection?
4. Which member hometown is the most popular in my music collection?
5. Which instrument/ instrument type is the most popular in my music collection (By count and plays)
6. What are the most played Songs in my music collection?

## *Reflection*

The scope of the database implementation was reduced to three albums for each of five artists, each associated with at least one award. In a more detailed application, scripts would be used to access the music metadata from within the directories in which the music is stored; play counts would be populated from iTunes metadata, and award information would be scraped from the web on a scheduled basis. Reporting would include visualizations and charts which would facilitate a better understanding of the data within the database.

The implementation of the database was made much simpler with the database design outlined on page seven, while some datatypes required modification. Descriptions were extended to 100 characters, and all datetime datatypes were changed to date, where the time of album release was not necessary.

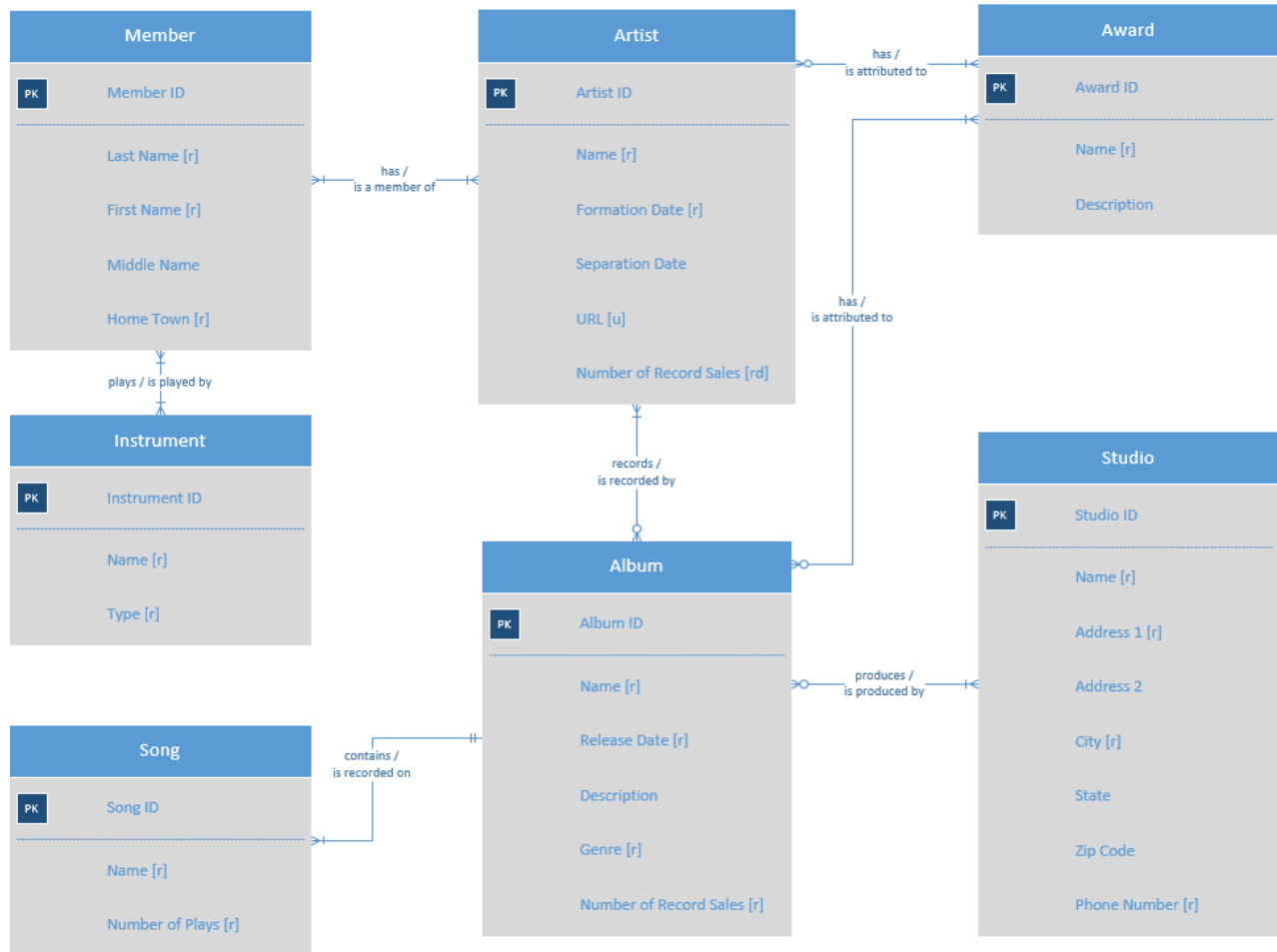
The most beneficial outcome from this project is the generation of playlists based on the number of plays across the database, which is answered in data question six. The graphical interface provided a very simple way of data entry and modification, and allows for fast accessibility.

Going into this project, I would have neglected to create the conceptual and logical models, which would create a gap between the generation of the database objects and their associations. The rigorous use of database modeling made the implementation much easier to understand and prevented mistakes. In the past, I have had to alter tables, views, and stored procedures many times, where the database models made these issues nearly nonexistent.

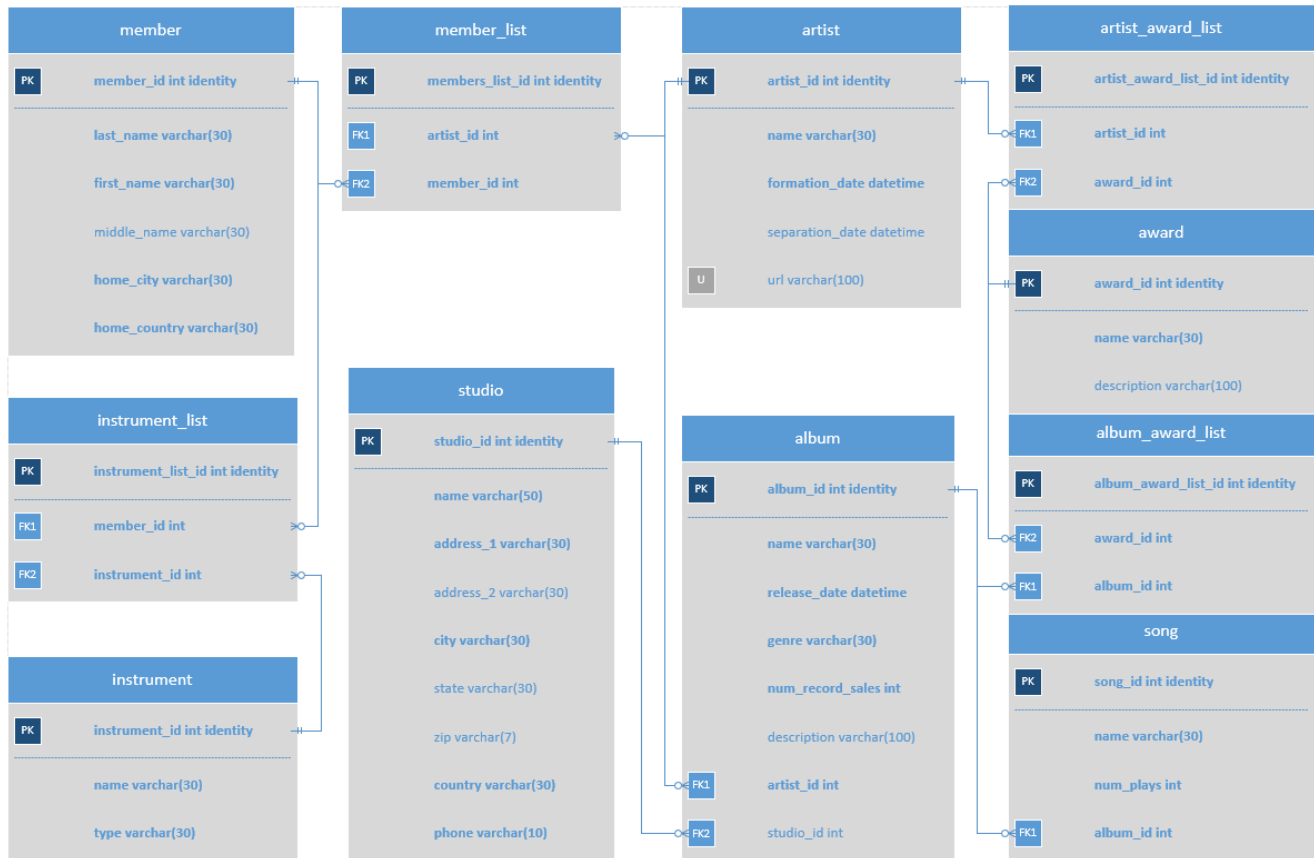
## *Entities*

- Member
  - Each member belongs to at least one artist, and plays at least one instrument. Members have a required first/ last name, and home city/ country. Each member may have an optional middle name. Some members may be the only member in a band, thus an artist may be named after the single individual.
- Instrument
  - Each instrument has a required name and type (woodwind, string, percussion, etc...).
- Studio
  - Studios may produce albums. Each studio has a required name, address\_1, city, country, and phone number. It may optionally have an address\_2, state, and zip code.
- Artist
  - Every artist has at least one member, and may have one or more albums. Artists record albums, but more than one artist cannot record one album. Each artist has a required name and formation\_date, with an optional separation\_date, and unique url.
- Album
  - Each album is created by an artist, and may be produced by a studio; albums contain at least one song. Albums require a name, release\_date, genre, and num\_record\_sales. Albums may have an optional description.
- Award
  - Awards are attributed to either artists or albums. Each award has a required name, and optional description.
- Song
  - Songs have a required song name and number of personal plays. Each song is attributed to one and only one album, being its original release album. LPs and Single-releases are considered single-song albums.

## Conceptual Model



## Logical Model



Each table's relationships are as follows, a default datatype of varchar(30) and singular/ lowercase styling was chosen. The logical model has been transformed from the conceptual model into 3<sup>rd</sup> Normal Form.

Bold represents required entries, underlining represents primary keys, and italics represent foreign keys.

```
member (
    member id int identity,
    last_name varchar(30),
    first_name varchar(30),
    middle_name varchar(30),
    home_city varchar(30),
    home_country varchar(30)
```

- Hometown was converted to two entries, home\_city, and home\_country.

```
instrument_list (
    instrument list id int identity,
    member_id int,
    instrument_id int
)
```

- A new entity was introduced to remove multi-valued fields, as some members play multiple instruments.
- The instrument\_list references the primary keys member\_id and instrument\_id as foreign keys.

```
instrument (
    instrument id int identity,
    name varchar(30),
    type varchar(30)
)
```

- Instruments have a name and type to help categorize instruments.

```
member_list (
    member list id int identity,
    artist_id int,
    member_id int
)
```

- A new entity was introduced to remove multi-valued fields, as most artists have more than one member, and some members may be a part of multiple bands.
- The member\_list references the primary keys artist\_id, and member\_id as foreign keys.

```

studio (
    studio_id int identity,
    name varchar(50),
    address_1 varchar(30),
    address_2 varchar(30),
    city varchar(30),
    state varchar(30),
    zip varchar(7),
    country varchar(30),
    phone varchar(10)
)

```

- Studios were adjusted to account for some which may be outside of the united states. Thus state, and zip were allowed to be optional.
- The studio name was chosen to be longer as some business names are longer than 30 characters.
- The zip code was chosen to be varchar to avoid issues with numeric types, and it is a string of numbers.
- The phone number was set to a required ten characters and was chosen to be varchar to avoid issues with numeric types, and it is a string of numbers.

```

artist (
    artist_id int identity,
    name varchar(30),
    formation_date date,
    separation_date date,
    url varchar(100)
)

```

- The url was chosen to be a 100 character varchar because some urls are longer than 30 characters.
- The formation\_date and separation\_date were chosen to be date to standardize dates across the database, and allow for sorting and searching using dates.
- Separation date was made optional because some bands have not yet separated.
- The derived number of record sales was removed as it was dependent on the album number of record sales, and can be derived at the user level.



```

album (
    album_id int identity,
    name varchar(30),
    release_date date,
    genre varchar(30),
    num_record_sales int,
    description varchar(100),
    artist_id int,
    studio_id int
)

```

- The release date was chosen to be date to standardize dates across the database, and allow for sorting and searching using dates.
- The num\_record\_sales was chosen to be int because it is a whole number, and calculations may be conducted on this data.
- The primary key artist\_id is a required reference as a foreign key.
- The primary key studio\_id is an optional reference as a foreign key because some albums are independently produced.
- Each album has an optional 100 character description to elaborate on any specific features.

```

artist_award_list (
    artist_award_list_id int identity,
    artist_id int,
    award_id int
)

```

- A new entity was introduced to remove multi-valued fields, as some artists have multiple awards.
- The artist\_award\_list references the primary keys artist\_id and award\_id as foreign keys.

```

award (
    award_id int identity,
    name varchar(30),
    description varchar(100)
)

```

- Each award as an optional 100 character description, allowing for a reduction in ambiguity and more information when analyzing different awards.

```
album_award_list (  
    album award list id int identity,  
    award_id int,  
    album_id int  
)
```

- A new entity was introduced to remove multi-valued fields, as some albums have multiple awards.
- The album\_award\_list references the primary keys album\_id and award\_id as foreign keys.

```
song (  
    song id int identity,  
    name varchar(30),  
    num_plays int,  
    album_id int  
)
```

- The num\_plays was assigned as an integer because it is a whole-number and will be used in calculations.
- The primary key album\_id is a required reference as a foreign key.

## Implementation

A front-end was developed to modify and add data using tables by connecting Microsoft Access to the database, which facilitated the addition of **five** artists, of which each have **three** albums, and various **awards** from Wikipedia entries. Access allows users to access and modify data with minimal work required to connect to the data source. Utilizing tables allows the user to quickly complete data entry, as well as maintain various aspects, such as the number of plays for a song, and newly awarded accolades.

- Member

member_id	last_name	first_name	middle_name	home_city	home_county
1	Daltrey	Roger	Harry	London	England
2	Townshend	Peter	Dennis Blandford	Middlesex	England
3	Entwistle	John	Alec	London	England
4	Moon	Keith	John	Middlesex	England
5	Jones	Kenney	Thomas	London	England
7	Plant	Robert	Anthony	Staffordshire	England
8	Page	James	Patrick	Middlesex	England
9	Baldwin	John	Richard	Kent	England
10	Bonham	John	Henry	Worcestershire	England
11	Fleetwood	Michael	John	Cornwall	England
12	McVie	John	Grahm	Middlesex	England
13	McVie	Christine	Perfect	Lancashire	England
14	Nicks	Stephanie	Lynn	Phoenix	USA
15	Campbell	Michael	Wayne	Jacksonville	USA
16	Finn	Neil	Mullane	Te Awamutu	New Zealand
17	Kiedis	Anthony		Grand Rapids	USA
18	Balzary	Muchael	Peter	Melbourne	Australia
19	Smith	Chadwick	Gaylord	Saint Paul	USA
20	Klinghoffer	Josh	Adam	Snata Monica	USA
21	Folds	Benjamin	Scott	Winston-Salem	USA
*	(New)				

- Instrument

instrument_id	name	type
2	Backup Vocals	Vocals
7	Bass	String
5	Drums	Percussion
9	General Horn	Horn
8	General Percussion	Percussion
3	Guitar	String
4	Harmonica	Wind
6	Keyboard	Percussion
1	Lead Vocals	Vocals
11	Mandolin	String
10	Piano	Percussion
12	Trumpet	Horn
*	(New)	

- Studio

studio_id	name	address_1	address_2	city	state	zip	country	phone
1	IBC Studios	35 Portland Pla		London			England	
2	Atlantic Recorc	1841 Broadway		New York	New York	10104	USA	2127072000
3	Blue Horizon	Defunct		London			England	
6	Reprise	1290 6th Ave		New York	New York	10104	USA	2127073200
7	Warner Bros.	3300 Warner Bl		Burbank	California	91505	USA	8188469090
8	Epic Records	25 Madison Av		New York	New York	10010	USA	2128338800
10	Sony Music Ent	25 Madison Av		New York	New York	10010	USA	2128338800
*	(New)							

- Artist

artist_id	name	formation_d	separation_d	url
1	The Who	1964-01-01		<a href="https://www.thewho.com/">https://www.thewho.com/</a>
2	Led Zeppelin	1968-01-01	1980-12-04	<a href="http://www.ledzeppelin.com/">http://www.ledzeppelin.com/</a>
3	Fleetwood Mac	1967-01-01		<a href="https://www.fleetwoodmac.com/">https://www.fleetwoodmac.com/</a>
4	Red Hot Chili Peppers	1983-01-01		<a href="https://redhotchilipeppers.com/">https://redhotchilipeppers.com/</a>
5	Ben Folds	1988-01-01		<a href="https://www.benfolds.com/">https://www.benfolds.com/</a>
*	(New)			

- Album

album_id	name	release_date	genre	peak_chart	description	artist_id	studio_id
1	My Generation	1965-12-03	Classic Rock	5	My Generation	1	1
2	Quadrophenia	1973-10-26	Classic Rock	2	Quadrophenia	1	1
5	Tommy	1968-05-23	Classic Rock	2	Tommy is the f	1	1
7	Led Zeppelin	1969-01-12	Classic Rock	7	Led Zeppelin is	2	2
8	Led Zeppelin II	1969-10-22	Classic Rock	1	Led Zeppelin II	2	2
10	Led Zeppelin II	1970-10-05	Classic Rock	1	Led Zeppelin II	2	2
12	Fleetwood Ma	1968-02-24	Classic Rock	4	Fleetwood Ma	3	3
14	The White Albu	1975-07-01	Classic Rock	1	The White Albu	3	6
15	Rumours	1977-02-04	Classic Rock	1	Rumours is the	3	7
16	Blood Sugar Se	1991-09-24	Hard Rock	1	Blood Sugar Se	4	7
17	Californication	1999-06-08	Hard Rock	3	Californication	4	7
18	Stadium Arcadi	2006-05-09	Hard Rock	1	Stadium Arcadi	4	7
21	Rockin' the Suk	2001-09-11	Alternative Ro	42	Rockin' the Suk	5	8
22	Songs for Silve	2005-04-11	Alternative Ro	13	Songs for Silve	5	10
23	Way to Normal	2008-09-17	Alternative Ro	11	Way to Normal	5	10
*	(New)						

- Award

award_id	name	description
1	UK Gold Certification	100k Certified units sold
2	Lifetime Achievement Award	1988 British Phographic Industry
3	American Music Award	2005 American Music Awards
4	Outstanding Contribution Music	British Phonographic Industry Awards
5	American Music Awards	Three wins, seven nominations
6	Billboard Music Awards	One win, five nominations
7	Grammy Awards	Six wins, sixteen nominations
8	RIAA Gold Record	2002 Win for Whatever and Ever Amen
9	Music City Walk of Fame	2018 Nashville TN
10	Gold Record	Tommy earned over 500k Sales
11	Grammy Hall of Fame	2004 induction
12	Best Recording Package	Grammy Award 1970
13	Album of the Year	Grammy Award 1978
14	Grammy Award	Grammy Awards- Song 2000
15	Best Rock Album	Grammy Awards 2006
16	Best Limited Edition Package	Grammy Awards 2006
*(New)		

- Song (219 Total)

song_id	name	num_plays	album_id
1	Out in the Stre	1651	1
2	I dont Mind	231	1
3	The Goods Gon	321	1
4	La-La-La-Lies	658	1
5	Much too Much	31	1
6	My Generation	1213	1
7	The Kids are Al	234	1
8	Please, Please,	627	1
9	Its Not True	1876	1
10	Im a Man	682	1

- Artist Award List

artist_awarc	artist_id	award_id
1	1	2
2	2	3
3	3	4
4	4	5
5	4	6
6	4	7
7	5	8
8	5	9
*(New)		

- Album Award List

album_awar ▾	award_id ▾	album_id ▾
1	1	1
2	10	5
7	11	7
8	12	8
9	13	15
10	14	17
11	15	18
12	16	18
* (New)		

- Instrument List (64 total)

instrument_ ▾	member_id ▾	instrument_ ▾
1	1	1
2	1	2
3	1	3
4	1	4
5	1	9
7	2	1
8	2	2
6	2	3
9	2	6
12	3	1
13	3	2
10	3	7

- Member List

	member_lis ▾	artist_id ▾	member_id ▾
	1	1	1
	2	1	2
	3	1	3
	4	1	4
	5	1	5
	6	2	7
	8	2	8
	9	2	9
	10	2	10
	11	3	11
	12	3	12
	13	3	13
	14	3	14
	19	3	15
	20	3	16
	21	4	17
	22	4	18
	23	4	19
	24	4	20
	25	5	21
*	(New)		

Reports were generated from the data questions and their respective views:

1. What is the range of album release dates in my music collection?

```
SELECT * FROM release_range
```

	earliest_release_date	latest_release_date
1	1965-12-03	2008-09-17

## Album Release Range

earliest_release_date	latest_release_date
1965-12-03	2008-09-17

Thursday, September 13, 2018

Page 1 of 1

Knowing the earliest and latest release dates provides insight to the different styles of music within the database, in this instance, the earliest album release date is 1965, where the latest is 2008.

2. Which genre is the most popular in my music collection? (By count and plays)

```
SELECT * FROM popular_genre_plays
```

	genre	total_plays
1	Classic Rock	54709
2	Hard Rock	21008
3	Alternative Rock	18175



## Popular Genres by Song Plays

Thursday, September 13, 2018

4:09:08 PM

genre	total_plays
Classic Rock	54709
Hard Rock	21008
Alternative Rock	18175
3	

Page 1 of 1



```
SELECT * FROM popular_genre_count
```

	genre	album_count
1	Classic Rock	9
2	Hard Rock	3
3	Alternative Rock	3

Popular Genres by Number of Albums		Thursday, September 13, 2018 4:08:19 PM
genre	album_count	
Classic Rock	9	
Hard Rock	3	
Alternative Rock	3	
	3	

Page 1 of 1

Knowing the popular genres by the number of song-plays is useful for knowing the user's preferences for genres, while their music corpus may contain more albums from different genres. In answering the question from the perspective of album counts and play counts, the user can gain insight into the contents and preferences of the music available. In this instance, the contents of the database suggest that my preferences are more towards Classic Rock.

3. Which artist/ album has the most awards in my music collection?

```
SELECT * FROM most_artist_awards
```

	artist_name	count
1	Red Hot Chili Peppers	3
2	Ben Folds	2
3	The Who	1
4	Led Zeppelin	1
5	Fleetwood Mac	1

Most Awards by Artist		Thursday, September 13, 2018 4:11:05 PM
artist_name	count	
Red Hot Chili Peppers	3	
Ben Folds	2	
The Who	1	
Led Zeppelin	1	
Fleetwood Mac	1	
	5	

Page 1 of 1

```
SELECT * FROM most_album_awards
```

	album_name	artist_name	count
1	Stadium Arcadium	Red Hot Chili Peppers	2
2	Tommy	The Who	1
3	My Generation	The Who	1
4	Californication	Red Hot Chili Peppers	1
5	Led Zeppelin II	Led Zeppelin	1
6	Led Zeppelin	Led Zeppelin	1
7	Rumours	Fleetwood Mac	1

Most Awards by Album			Thursday, September 13, 2018
			4:11:43 PM
album_name	artist_name	count	
Stadium Arcadium	Red Hot Chili Peppers	2	
Tommy	The Who	1	
My Generation	The Who	1	
Californication	Red Hot Chili Peppers	1	
Led Zeppelin II	Led Zeppelin	1	
Led Zeppelin	Led Zeppelin	1	
Rumours	Fleetwood Mac	1	
		7	

Page 1 of 1

Red Hot Chili Peppers have the most awards in the database, and by querying the artist awards separately from the album awards, different perspectives can be taken to find the most successful or critically acclaimed artists.

4. Which member hometown is the most popular in my music collection?

```
SELECT * FROM most_popular_hometown
```

	home_city	home_country	count
1	Middlesex	England	4
2	London	England	3
3	Jacksonville	USA	1
4	Grand Rapids	USA	1
5	Te Awamutu	New Zealand	1
6	Worcestershire	England	1
7	Staffordshire	England	1
8	Lancashire	England	1
9	Kent	England	1
10	Cornwall	England	1



## Most Popular Artist Hometown

Thursday, September 13, 2018

4:12:27 PM

home_city	home_country	count
Middlesex	England	4
London	England	3
Jacksonville	USA	1
Grand Rapids	USA	1
Te Awamutu	New Zealand	1
Worcestershire	England	1
Staffordshire	England	1
Lancashire	England	1
Kent	England	1
Cornwall	England	1

ho

Page 1 of 1

The most popular hometown is Middlesex England. This view allows the user to gain insight into their musical influences, which could lead them to more artists which they would enjoy.

5. Which instrument/ instrument type is the most popular in my music collection (By count and plays)

`SELECT * FROM most_popular_instrument`

	instrument_name	count
1	Backup Vocals	11
2	Lead Vocals	11
3	Guitar	8
4	Drums	6
5	Bass	6
6	General Percussion	6
7	Keyboard	6
8	Piano	4
9	Hamonica	2
10	Mandolin	1



## Most Popular Instrument

Thursday, September 13, 2018

4:12:51 PM

instrument_name	count
Backup Vocals	11
Lead Vocals	11
Guitar	8
Drums	6
Bass	6
General Percussion	6
Keyboard	6
Piano	4
Hamonica	2
Mandolin	1

10

```
SELECT * FROM most_popular_type
```

	instrument_type	count
1	Percussion	22
2	Vocals	22
3	String	15
4	Wind	2
5	Horn	2

Most Popular Instrument Type		Thursday, September 13, 2018 4:13:09 PM
instrument_type	count	
Percussion	22	
Vocals	22	
String	15	
Wind	2	
Horn	2	
		5

Page 1 of 1

The most popular instrument in the database is vocals, while the most popular type is percussion. In separating the question to instruments and their types, different levels of granulation can be analyzed, allowing the user to gain insight into the different preferred instruments in their music libraries.

6. What are the most played Songs in my music collection?

`SELECT * FROM best_songs`

	song_name	num_plays	artist_name	album_name
1	Love, Reign O'er Me	1988	The Who	Quadrophenia
2	The Luckiest	1940	Ben Folds	Rockin' the Suburbs
3	Landslide	1884	Fleetwood Mac	The White Album
4	Its Not True	1876	The Who	My Generation
5	Babe I'm Gonna Leave You	1864	Led Zeppelin	Led Zeppelin
6	The Chain	1844	Fleetwood Mac	Rumours
7	Whole Lotta Love	1664	Led Zeppelin	Led Zeppelin II
8	Out in the Street	1651	The Who	My Generation
9	Dani California	1646	Red Hot Chili Peppers	Stadium Arcadium
10	Scar Tissue	1646	Red Hot Chili Peppers	Californication

Most Played Songs				Thursday, September 13, 2018	
				4:13:28 PM	
song_name	num_plays	artist_name	album_name		
Love, Reign O'er Me	1988	The Who	Quadrophenia		
The Luckiest	1940	Ben Folds	Rockin' the Suburbs		
Landslide	1884	Fleetwood Mac	The White Album		
Its Not True	1876	The Who	My Generation		
Babe I'm Gonna Leave You	1864	Led Zeppelin	Led Zeppelin		
The Chain	1844	Fleetwood Mac	Rumours		
Whole Lotta Love	1664	Led Zeppelin	Led Zeppelin II		
Out in the Street	1651	The Who	My Generation		
Dani California	1646	Red Hot Chili Peppers	Stadium Arcadium		
Scar Tissue	1646	Red Hot Chili Peppers	Californication		

Page 1 of 1

This question is the most useful in the data questions. It allows the user to view the most popular songs in their music collection, and generate ten-song playlists from their favorite songs. In this instance, no song play-count exceeded 2000 plays, and the top ten songs included songs from each of the five artists in this implementation.

## All Songs

All Songs			
genre	Alternative Rock		
artist_name	Ben Folds		
album_name	Rockin' the Suburbs		
	song_id	song_name	num_plays
	184	Annie Walks	325
	185	Zak and Sara	132
	186	Still Fighting It	1213
	187	Gone	321
	188	Fred Jones Pt 2	112
	189	The Ascent of Stand	799
	190	Losing Lisa	776
	191	Carrying Cathy	332
	192	Not the Same	778
	193	Rockin the Suburbs	1424
	194	Fired	184
	195	The Luckiest	1940
	196	Hiro's Song	321
album_name	Songs for Silverman		
	song_id	song_name	num_plays
	197	Bastard	12
	198	You to Thank	14
	199	Jesusland	654
	200	Landed	321
	201	Gracie	46
	202	Trusted	787
	203	Give Judy my Notice	323
	204	Late	774

205	Sentimental Guy	32
206	Prison Food	779
207	Bitches Aint Shit	1615
<div> <div>album_name</div> <div>Way to Normal</div> </div> <div> <div>song_id</div> <div>song_name</div> <div>num_plays</div> </div>		
208	Hiroshima	44
209	Dr. Yang	1
210	The Frown Song	13
211	You Don't Know Me	1346
212	Before Cologne	6
213	Cologne	76
214	Errant Dog	665
215	Free Coffee	13
216	Bitch Went Nuts	654
217	Brainwascht	798
218	Effington	332
219	Kylie from Connecticut	213
genre	Classic Rock	

artist_name		Fleetwood Mac
<div> <div>album_name</div> <div>Fleetwood Mac</div> </div> <div> <div>song_id</div> <div>song_name</div> <div>num_plays</div> </div>		
89	My Heart Beat Like a Hammer	664
90	Merry Go Round	72
91	Long Grey Mare	87
92	Hellhound on my Trail	79
93	Shake Your Moneymaker	84
94	Looking for Somebody	91
95	No Place to Go	221
96	My Baby's Good to Me	433



97	I Loved Another Woman	220
98	Cold Black Night	87
99	The World Keep on Turning	1277
100	Got to Move	1
<div> <div>album_name</div> <div>Rumours</div> <div> <div>song_id</div> <div>song_name</div> <div>num_plays</div> </div> </div>		
112	Second Hand News	55
113	Dreams	78
114	Never Going Back Again	998
115	Don't Stop	135
116	Go Your Own Way	665
117	Songbird	789
118	The Chain	1844
119	You Make Loving Fun	1213
120	I Don't Want to Know	232
121	Oh Daddy	541
122	Gold Dust Woman	221
<div> <div>album_name</div> <div>The White Album</div> <div> <div>song_id</div> <div>song_name</div> <div>num_plays</div> </div> </div>		
101	Monday Morning	77
102	Warm Ways	135
103	Blue Letter	278
104	Rhiannon	335
105	Over my Head	887
106	Crystal	135
107	Say You Love Me	465
108	Landslide	1884
109	World Turning	98
110	Sugar Daddy	7
111	I'm So Afraid	98

artist_name	Led Zeppelin		
album_name	Led Zeppelin		
	song_id	song_name	num_plays
	59	Good Times Bad Times	416
	60	Babe I'm Gonna Leave You	1864
	62	You Shook Me	187
	63	Dazed and Confused	1246
	64	Your Time is Gonna Come	876
	65	Black Mountain Side	216
	66	Communication Breakdown	46
	67	I Can't Quit You Baby	87
	68	How Many More Times	998
album_name	Led Zeppelin II		
	song_id	song_name	num_plays
	69	Whole Lotta Love	1664
	70	What Is & What Should Never Be	264
	71	The Lemon Song	374
	72	Thank You	76
	73	Heartbreaker	621
	74	Living Loving Maid	112
	75	Ramble On	664
	76	Moby Dick	4
	77	Bring it on Home	26
album_name	Led Zeppelin III		
	song_id	song_name	num_plays
	78	Immigrant Song	921
	79	Friends	14
	80	Celebration Day	18
	81	Since I've Been Loving You	245

82	Out on the Tiles	874
83	Gallows Pole	346
84	Tangerine	1246
86	That's the Way	442
87	Bron-Y-Aur Stomp	330
88	Hats Off the Harper	423

artist_name	The Who	
album_name	My Generation	
song_id	song_name	num_plays
1	Out in the Street	1651
2	I dont Mind	231
3	The Goods Gone	321
4	La-La-La-Lies	658
5	Much too Much	31
6	My Generation	1213
7	The Kids are Alright	234
8	Please, Please, Please	627
9	Its Not True	1876
10	Im a Man	682
11	A Legal Matter	27
12	The Ox	1246
album_name	Quadrophenia	
song_id	song_name	num_plays
14	I Am the Sea	354
15	The Real Me	1584
17	Quadrophenia	1386
18	Cut My Hair	894
21	The Punk and the Godfather	132
22	I'm One	983
23	The Dirty Jobs	19

24	Helpless Dancer	181
25	Is it in My Head?	193
26	I've Had Enough	91
27	5:15	98
28	Sea and Sand	1134
29	Drowned	122
30	Bell Boy	89
31	Doctor Jimmy	45
32	The Rock	98
33	Love, Reign O'er Me	1988
<div> <div>album_name</div> <div>Tommy</div> <div>song_id</div> <div>song_name</div> <div>num_plays</div> </div>		
34	Overture	899
35	It's a Boy	26
36	1921	88
37	Amazing Journey	4
38	Sparks	68
39	The Hawker	78
40	Christmas	44
41	Cousin Kevin	89
43	The Acid Queen	220
44	Underture	465
45	Do You Think It's Alright	98
46	Fiddle About	668
47	Pinball Wizard	1530
48	There's a Doctor	789
49	Go to the Mirror!	223
50	Tommy Can You Hear Me?	78
51	Smash the Mirror	32
52	Sensation	156
53	Miracle Cure	122

54	Sally Simpson	1343
55	I'm Free	135
56	Welcome	36
57	Tommy's Holiday Camp	123
58	We're Not Going to Take It	121

genre                      Hard Rock

artist\_name                Red Hot Chili Peppers

album_name	Blood Sugar Sex Magik	
song_id	song_name	num_plays
123	The Power of Equality	354
124	If You Have to Ask	223
125	Breaking the Girl	0
126	Funky Monks	221
127	Suck My Kiss	1416
128	I Could Have Lied	111
129	Mellowship Slinky in B Major	79
130	The Righteous & the Wicked	99
131	Give it Away	7
132	Blood Sugar Sex Magick	17
133	Under the Bridge	64
134	Naked in the Rain	77
135	Apache Rose Peacock	16
136	The Greeting Song	77
137	My Lovely Man	8
138	Sir Psycho Sexy	16
139	They're Red Hot	225
album_name	Californication	
song_id	song_name	num_plays
140	Around the World	19

141	Parallel Universe	465
142	Scar Tissue	1646
143	Otherside	779
144	Get on Top	4
145	Californication	921
146	Easily	143
147	Porcelain	34
148	Emit Remmus	65
149	I Like Dirt	98
150	The Velvet Glove	134
151	Savior	468
152	Purple Stain	357
153	Right on Time	898
154	Road Trippin	445

album_name	Stadium Arcadium	
song_id	song_name	num_plays
155	Dani California	1646
156	Snow	65
157	Charlie	521
158	Stadium Arcadium	654
159	Hump de Bump	54
160	She's Only 18	524
161	Slow Cheetah	321
162	Torture Me	181
163	Strip my Mind	87
164	Especially in Michigan	798
165	Warlocks	125
166	Cmon Girl	321
167	Wet Sand	6
168	Hey	78
169	Desecration Smile	99

170	Tell me Baby	132
171	Hard to Concentrate	88
172	21st Century	984
173	She Looks to Me	651
174	Readymade	798
175	If	112
176	Make you Feel Better	332
177	Animal Bar	498
178	So Much I	1321
179	Storm in a Teacup	321
180	We Believe	13
181	Turn it Again	13
182	Death of a Martian	445
183	Havana Affair	334

```

/*
    Samuel L. Peoples
    IST 659
    Music Database : Create Tables
*/

-- Create the member table
CREATE TABLE member(
    member_id int identity PRIMARY KEY,
    last_name varchar(30) NOT NULL,
    first_name varchar(30) NOT NULL,
    middle_name varchar(30),
    home_city varchar(30) NOT NULL,
    home_country varchar(30) NOT NULL
);

-- Create the instrument table
CREATE TABLE instrument(
    instrument_id int identity PRIMARY KEY,
    name varchar(30) UNIQUE NOT NULL,
    type varchar(30) NOT NULL
);

-- Create the studio table
CREATE TABLE studio(
    studio_id int identity PRIMARY KEY,
    name varchar(50) NOT NULL,
    address_1 varchar(30) NOT NULL,
    address_2 varchar(30),
    city varchar(30) NOT NULL,
    state varchar(30),
    zip varchar(7),
    country varchar(30) NOT NULL,
    phone varchar(10)
);

-- Create the artist table
CREATE TABLE artist(
    artist_id int identity PRIMARY KEY,
    name varchar(30) NOT NULL,
    formation_date date NOT NULL,
    separation_date date,
    url varchar(100)
);

-- Create the album table
CREATE TABLE album(
    album_id int identity PRIMARY KEY,
    name varchar(30) NOT NULL,
    release_date date NOT NULL,
    genre varchar(30) NOT NULL,
    peak_chart_position int NOT NULL,
    description varchar(100),
    artist_id int FOREIGN KEY REFERENCES artist(artist_id) NOT NULL,
    studio_id int FOREIGN KEY REFERENCES studio(studio_id) NOT NULL
);

```



```

-- Create the award table
CREATE TABLE award(
    award_id int identity PRIMARY KEY,
    name varchar(30) NOT NULL,
    description varchar(100)
);

-- Create the song table
CREATE TABLE song(
    song_id int identity PRIMARY KEY,
    name varchar(30) NOT NULL,
    num_plays int NOT NULL,
    album_id int FOREIGN KEY REFERENCES album(album_id) NOT NULL
);

-- Create the artist_award_list
CREATE TABLE artist_award_list(
    artist_award_list_id int identity PRIMARY KEY,
    artist_id int FOREIGN KEY REFERENCES artist(artist_id) NOT NULL,
    award_id int FOREIGN KEY REFERENCES award(award_id) NOT NULL,
    CONSTRAINT award_list_info UNIQUE(artist_id, award_id)
);

-- Create the album_award_list
CREATE TABLE album_award_list(
    album_award_list_id int identity PRIMARY KEY,
    award_id int FOREIGN KEY REFERENCES award(award_ID) NOT NULL,
    album_id int FOREIGN KEY REFERENCES album(album_id) NOT NULL
    CONSTRAINT album_list_info UNIQUE(album_id, award_id)
);

-- Create the instrument list table
CREATE TABLE instrument_list(
    instrument_list_id int identity PRIMARY KEY,
    member_id int FOREIGN KEY REFERENCES member(member_id) NOT NULL,
    instrument_id int FOREIGN KEY REFERENCES instrument(instrument_id) NOT NULL
    CONSTRAINT instrument_list_info UNIQUE(member_id, instrument_id)
);

-- Create the member list table
CREATE TABLE member_list(
    member_list_id int identity PRIMARY KEY,
    artist_id int FOREIGN KEY REFERENCES artist(artist_id) NOT NULL,
    member_id int FOREIGN KEY REFERENCES member(member_id) NOT NULL
    CONSTRAINT list_info UNIQUE(artist_id, member_id)
);

```

```

/*
    Samuel L. Peoples
    IST 659
    Music Database : Add Data
*/

-- Populate each table with some initial data

-- Add The Who to artist
INSERT INTO artist(name, formation_date, url)
VALUES('The Who', '1964-01-01', 'https://www.thewho.com/');

-- Add Led Zeppelin to artist
INSERT INTO artist(name, formation_date, separation_date, url)
VALUES('Led Zeppelin', '1968-01-01', '1980-12-04', 'http://www.ledzeppelin.com/');

-- Add Fleetwood Mac to artist
INSERT INTO artist(name, formation_date, url)
VALUES('Fleetwood Mac', '1967-01-01', 'https://www.fleetwoodmac.com/');

-- Add Red Hot Chili Peppers to artist
INSERT INTO artist(name, formation_date, url)
VALUES('Red Hot Chili Peppers', '1983-01-01', 'https://redhotchilipeppers.com/');

-- Add Ben Folds to artist
INSERT INTO artist(name, formation_date, url)
VALUES('Ben Folds', '1988-01-01', 'https://www.benfolds.com/');

-- Add the members from The Who
INSERT INTO member(last_name, first_name, middle_name, home_city, home_country)
VALUES('Daltrey', 'Roger', 'Harry', 'London', 'England');
INSERT INTO member(last_name, first_name, middle_name, home_city, home_country)
VALUES('Townshend', 'Peter', 'Dennis Blandford', 'Middlesex', 'England');
INSERT INTO member(last_name, first_name, middle_name, home_city, home_country)
VALUES('Entwistle', 'John', 'Alec', 'London', 'England');
INSERT INTO member(last_name, first_name, middle_name, home_city, home_country)
VALUES('Moon', 'Keith', 'John', 'Middlesex', 'England');
INSERT INTO member(last_name, first_name, middle_name, home_city, home_country)
VALUES('Jones', 'Kenney', 'Thomas', 'London', 'England');

-- Associate the members with The Who

INSERT INTO member_list(artist_id, member_id)
VALUES(1,1);
INSERT INTO member_list(artist_id, member_id)
VALUES(1,2);
INSERT INTO member_list(artist_id, member_id)
VALUES(1,3);
INSERT INTO member_list(artist_id, member_id)
VALUES(1,4);
INSERT INTO member_list(artist_id, member_id)
VALUES(1,5);

-- Add the instruments the Members play
INSERT INTO instrument(name, type)
VALUES('Lead Vocals', 'Vocals');
INSERT INTO instrument(name, type)

```

```

VALUES('Backup Vocals', 'Vocals');
INSERT INTO instrument(name, type)
VALUES('Guitar', 'String');
INSERT INTO instrument(name, type)
VALUES('Harmonica', 'Wind');
INSERT INTO instrument(name, type)
VALUES('Drums', 'Percussion');
INSERT INTO instrument(name, type)
VALUES('Keyboard', 'Percussion');
INSERT INTO instrument(name, type)
VALUES('Bass', 'String');
INSERT INTO instrument(name, type)
VALUES('General Percussion', 'Percussion');
INSERT INTO instrument(name, type)
VALUES('General Horn', 'Horn');

-- Associate the instruments with members
INSERT INTO instrument_list(member_id, instrument_id)
VALUES('1', '1');
INSERT INTO instrument_list(member_id, instrument_id)
VALUES('1', '2');
INSERT INTO instrument_list(member_id, instrument_id)
VALUES('1', '3');
INSERT INTO instrument_list(member_id, instrument_id)
VALUES('1', '4');
INSERT INTO instrument_list(member_id, instrument_id)
VALUES('1', '9');
INSERT INTO instrument_list(member_id, instrument_id)
VALUES('2', '3');
INSERT INTO instrument_list(member_id, instrument_id)
VALUES('2', '1');
INSERT INTO instrument_list(member_id, instrument_id)
VALUES('2', '2');
INSERT INTO instrument_list(member_id, instrument_id)
VALUES('2', '6');
INSERT INTO instrument_list(member_id, instrument_id)
VALUES('3', '7');
INSERT INTO instrument_list(member_id, instrument_id)
VALUES('3', '8');
INSERT INTO instrument_list(member_id, instrument_id)
VALUES('3', '1');
INSERT INTO instrument_list(member_id, instrument_id)
VALUES('3', '2');
INSERT INTO instrument_list(member_id, instrument_id)
VALUES('4', '5');
INSERT INTO instrument_list(member_id, instrument_id)
VALUES('4', '1');
INSERT INTO instrument_list(member_id, instrument_id)
VALUES('4', '2');
INSERT INTO instrument_list(member_id, instrument_id)
VALUES('5', '5');

-- Add an award
INSERT INTO award(name, description)
VALUES('UK Gold Certification', '100k Certified units sold');
INSERT INTO award(name, description)
VALUES('Lifetime Achievement Award', '1988 British Phographic Industry');

```

```

-- Associate the award with The Who
INSERT INTO artist_award_list(award_id, artist_id)
VALUES(2,1);

-- Add The Who's first album's studio
INSERT INTO studio(name, address_1, city, country)
VALUES('IBC Studios','35 Portland Place','London','England');

-- Add the first album from The Who
INSERT INTO album(name, release_date, genre, peak_chart_position, artist_id, studio_id)
VALUES ('My Generation','1965-12-03','Classic Rock',5,1,1);

-- Associate the award with The Who's First Album
INSERT INTO album_award_list(award_id, album_id)
VALUES(1,1);

-- Add the songs from the first album (My Generation)
INSERT INTO song(name, num_plays, album_id)
VALUES('Out in the Street',3651,1);
INSERT INTO song(name, num_plays, album_id)
VALUES('I dont Mind',231,1);
INSERT INTO song(name, num_plays, album_id)
VALUES('The Goods Gone',321,1);
INSERT INTO song(name, num_plays, album_id)
VALUES('La-La-La-Lies',658,1);
INSERT INTO song(name, num_plays, album_id)
VALUES('Much too Much',31,1);
INSERT INTO song(name, num_plays, album_id)
VALUES('My Generation',3213,1);
INSERT INTO song(name, num_plays, album_id)
VALUES('The Kids are Alright',234,1);
INSERT INTO song(name, num_plays, album_id)
VALUES('Please, Please, Please',627,1);
INSERT INTO song(name, num_plays, album_id)
VALUES('Its Not True',3876,1);
INSERT INTO song(name, num_plays, album_id)
VALUES('Im a Man',682,1);
INSERT INTO song(name, num_plays, album_id)
VALUES('A Legal Matter',27,1);
INSERT INTO song(name, num_plays, album_id)
VALUES('The Ox',1246,1);

```

```

/*
    Samuel L. Peoples
    IST 659
    Music Database : Stored Procedures
*/

-- Create a stored procedure to drop albums and their dependencies
GO
CREATE PROCEDURE drop_album(@name varchar(30)) AS
BEGIN
    DECLARE @id int
    SELECT @id = album_id FROM album WHERE album.name = @name
    DELETE FROM album_award_list WHERE album_id = @id
    DELETE FROM song WHERE album_id = @id
    DELETE FROM album WHERE album_id = @id
END
GO;

-- Test the removal of an album
EXEC drop_album 'My Generation'

-- Create a stored procedure to drop artists and their dependencies
GO
CREATE PROCEDURE drop_artist(@name varchar(30)) AS
BEGIN
    DECLARE @id int
    SELECT @id = artist_id FROM artist WHERE artist.name = @name
    DELETE FROM artist_award_list WHERE artist_id = @id
    DELETE FROM album WHERE artist_id = @id
    DELETE FROM member_list WHERE artist_id = @id
    DELETE FROM artist WHERE artist_id = @id
END
GO;

-- Test the removal of an artist
EXEC drop_artist 'Ben Folds'

-- Create a stored procedure to drop awards and their dependencies
GO
CREATE PROCEDURE drop_award(@name varchar(30)) AS
BEGIN
    DECLARE @id int
    SELECT @id = award_id FROM award WHERE award.name = @name
    DELETE FROM album_award_list WHERE award_id = @id
    DELETE FROM artist_award_list WHERE award_id = @id
    DELETE FROM award WHERE award_id = @id
END
GO;

-- Test the removal of an award
EXEC drop_award 'UK Gold Certification'

-- Create a stored procedure to drop instruments and their dependencies
GO
CREATE PROCEDURE drop_instrument(@name varchar(30)) AS
BEGIN
    DECLARE @id int
    SELECT @id = instrument_id FROM instrument WHERE instrument.name = @name

```

```

        DELETE FROM instrument_list WHERE instrument_id = @id
        DELETE FROM instrument WHERE instrument_id = @id
END
GO;

-- Test the removal of an instrument
EXEC drop_instrument 'Guitar'

-- Create a stored procedure to drop awards and their dependencies
GO
CREATE PROCEDURE drop_member(@first_name varchar(30), @last_name varchar(30)) AS
BEGIN
    DECLARE @id int
    SELECT @id = member_id FROM member WHERE member.last_name = @last_name AND
member.first_name = @first_name
    DELETE FROM instrument_list WHERE member_id = @id
    DELETE FROM member_list WHERE member_id = @id
END
GO;

-- Test the removal of a member
EXEC drop_member 'Roger', 'Daltrey'

-- Create a stored procedure to drop studios and their dependencies
GO
CREATE PROCEDURE drop_studio(@name varchar(30)) AS
BEGIN
    DECLARE @id int
    SELECT @id = studio_id FROM studio WHERE studio.name = @name
    DELETE FROM album WHERE studio_id = @id
    DELETE FROM studio WHERE studio_id = @id
END
GO;

-- Test the removal of a studio
EXEC drop_studio 'IBC Studios'

```

```

/*
    Samuel L. Peoples
    IST 659
    Music Database : Views
*/

-- Create a view for albums
GO
CREATE VIEW album_report AS
    SELECT
        album_id,
        album.name AS album_name,
        release_date,
        genre,
        peak_chart_position,
        artist.name AS artist_name,
        studio.name AS studio_name
    FROM album
    JOIN artist ON album.artist_id = artist.artist_id
    JOIN studio ON album.studio_id = studio.studio_id
GO;

-- Test the album_report view
SELECT * FROM album_report

-- Create a view for album awards
GO
CREATE VIEW album_award_report AS
    SELECT
        album_award_list_id,
        award.name AS award_name,
        award.description AS award_description,
        artist.name AS artist_name,
        album.name AS album_name
    FROM album_award_list
    JOIN award ON album_award_list.award_id = award.award_id
    JOIN album ON album_award_list.album_id = album.album_id
    JOIN artist ON album_award_list.artist_id = artist.artist_id
GO;

-- Test the album_award_report
SELECT * FROM album_award_report

-- Create a view for artist awards
GO
CREATE VIEW artist_award_report AS
    SELECT
        artist_award_list_id,
        award.name AS award_name,
        award.description AS award_description,
        artist.name AS artist_name
    FROM artist_award_list
    JOIN award ON artist_award_list.award_id = award.award_id
    JOIN artist ON artist_award_list.artist_id = artist.artist_id
GO;

-- Test the artist_award_report
SELECT * FROM artist_award_report

```

```

-- Create a view for the instrument list
GO
CREATE VIEW instrument_report AS
    SELECT
        instrument_list_id,
        member.last_name AS last_name,
        member.first_name AS first_name,
        artist.name AS artist_name,
        instrument.name AS instrument_name,
        instrument.type AS instrument_type
    FROM instrument_list
    JOIN member ON instrument_list.member_id = member.member_id
    JOIN instrument ON instrument_list.instrument_id = instrument.instrument_id
    JOIN member_list ON member.member_id = member_list.member_id
    JOIN artist ON member_list.artist_id = artist.artist_id
GO;

-- Test the instrument_report
SELECT * FROM instrument_report

-- Create a view for the member list
GO
CREATE VIEW member_report AS
    SELECT
        member_list_id,
        member.last_name,
        member.first_name,
        member.home_city,
        member.home_country,
        artist.name AS artist_name
    FROM member_list
    JOIN member ON member_list.member_id = member.member_id
    JOIN artist ON member_list.artist_id = artist.artist_id
GO;

-- Test the member_report
SELECT * FROM member_report

-- Create a view for the songs
CREATE VIEW song_report AS
    SELECT
        song_id,
        song.name AS song_name,
        num_plays,
        album.name AS album_name,
        artist.name AS artist_name,
        genre
    FROM song
    JOIN album ON song.album_id = album.album_id
    JOIN artist ON album.artist_id = artist.artist_id
GO;

-- Test the song_report
SELECT * FROM song_report

```



```

/*
    Samuel L. Peoples
    IST 659
    Music Database : Data Questions
*/

-- 1. What is the range of album release dates in my music collection?
CREATE VIEW release_range AS
    SELECT
        MIN(release_date) AS earliest_release_date,
        MAX(release_date) AS latest_release_date
    FROM album
-- Answer:
SELECT * FROM release_range

-- 2. Which genre is the most popular in my music collection? (By count and plays)
CREATE VIEW popular_genre_count AS
    SELECT TOP 10
        genre,
        COUNT(genre) AS album_count
    FROM album
    GROUP BY genre
    ORDER BY album_count DESC
-- Answer:
SELECT * FROM popular_genre_count

CREATE VIEW popular_genre_plays AS
    SELECT TOP 10
        genre,
        SUM(num_plays) AS total_plays
    FROM song_report
    GROUP BY genre
    ORDER BY total_plays DESC
-- Answer
SELECT * FROM popular_genre_plays

-- 3. Which artist/ album has the most awards in my music collection?
CREATE VIEW most_artist_awards AS
    SELECT TOP 10
        artist_name,
        COUNT(artist_name) AS count
    FROM artist_award_report
    GROUP BY artist_name
    ORDER BY count DESC
-- Answer
SELECT * FROM most_artist_awards

CREATE VIEW most_album_awards AS
    SELECT TOP 10
        album_name,
        artist_name,
        COUNT(album_name) AS count
    FROM album_award_report
    GROUP BY album_name, artist_name
    ORDER BY count DESC
-- Answer
SELECT * FROM most_album_awards

```

```

-- 4. Which member hometown is the most popular in my music collection?
CREATE VIEW most_popular_hometown AS
    SELECT TOP 10
        home_city,
        home_country,
        COUNT(home_city) AS count
    FROM member_report
    GROUP BY home_city, home_country
    ORDER BY count DESC
-- Answer
SELECT * FROM most_popular_hometown

-- 5. Which instrument/ instrument type is the most popular in my music collection
CREATE VIEW most_popular_instrument AS
    SELECT TOP 10
        instrument_name,
        COUNT(instrument_name) AS count
    FROM instrument_report
    GROUP BY instrument_name
    ORDER BY count DESC
-- Answer
SELECT * FROM most_popular_instrument

CREATE VIEW most_popular_type AS
    SELECT TOP 10
        instrument_type,
        COUNT(instrument_type) AS count
    FROM instrument_report
    GROUP BY instrument_type
    ORDER BY count DESC
-- Answer
SELECT * FROM most_popular_type

-- 6. What are the most played songs in my music collection?
-- We're making a playlist of ten songs
CREATE VIEW best_songs AS
    SELECT TOP 10
        song_name,
        num_plays,
        artist_name,
        album_name
    FROM song_report
    GROUP BY num_plays, song_name, album_name, artist_name
    ORDER BY num_plays DESC
-- Answer:
SELECT * FROM best_songs

```