

**This is your practice final exam**

**Step 1 - Create the database and insert the data.** This is a music inventory database..(This is not my collection!) You will have to keep create each table

1. Decide on primary keys and what they should be. (Create a separate unique identifier)
2. Decide if a field is required or options
3. Decide if information should be foreign key or redundant data (Do you really want redundant data?)
4. Decide on what restrictions or checks you should have on data

What you should keep in mind:

1. You are working with a relational database, create foreign keys and relationships where needed.
2. Remember that one of the great facts of relation data is that data is not repeated from table to tale!

Data Set 1: Music Categories

Music Category
Classical
Rock
Jazz
Country
New Age

Data Set 2: Recording Artists

Recording Artist	Notes
The Popular Mechanics	Classic rock sound
Crawdad Stare	Grunge Rock style.
Dog House Rules	Smooth Jazz from Chicago area.
Mary Saveley	Soft soul jase

Data Set 3: Recordings

Recording Title	Recording Artist ID	Music Category ID	Recording Label	Format	Number of Tracks	Notes
Noise in the Garage	Crawdad Stare	Rock	Goes to 11, Inc.	CD	10	1 <sup>st</sup> recording to break top 10 by this artist
Look Both Ways	Crawdad Stare	Rock	Hippo Records	Cassette	12	Artists original release. Underground

Recording Title	Recording Artist ID	Music Category ID	Recording Label	Format	Number of Tracks	Notes
						until Noise in garage was popular.
Outback	Crawdad Stare	Rock		CD	10	
Short Circuit	Dog House Rules	Jazz		Cassette	7	
Meditations	Mary Saveley	New Age		CD	6	
Opus 65	The Popular Mechanics	Classical		CD	5	
Sounds Better Louder	Crawdad Stare	Rock	Goes to 11, Inc.	CD	10	

**After creating your tables and inserting your data, you may proceed to step 2.**

**ASSERTIONS**

A Recording Title must have one and only one Recording Artist.

A Recording Artist may have zero and more Recording Titles.

A Recording Title may have zero or more Media Formats.

A Media Format must have one and only one instance in a Recording Title.

A Recording Title must have one and only one Recording Label.

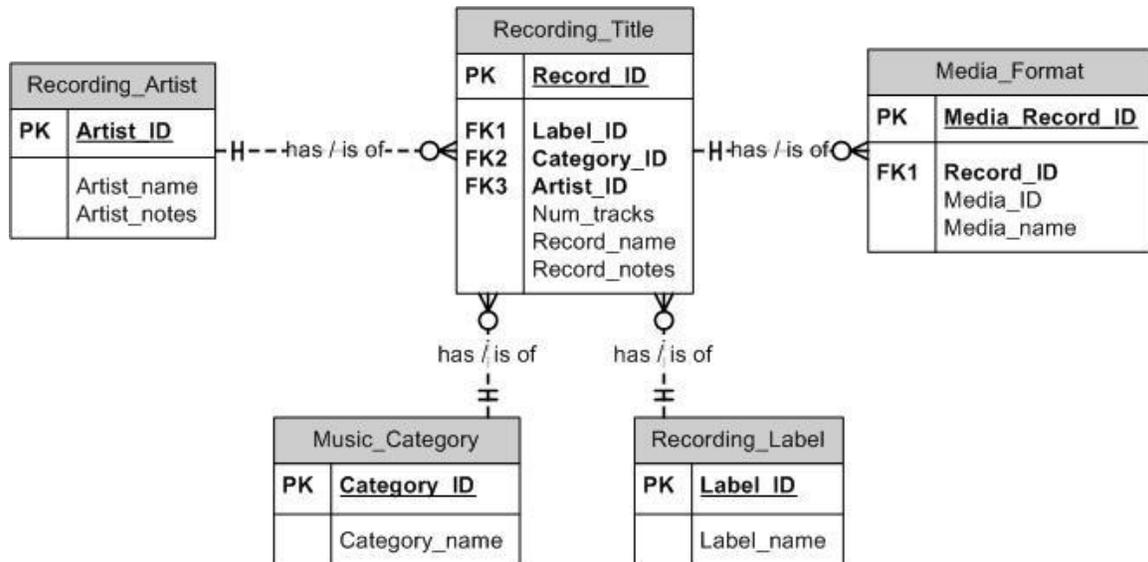
A Recording Label may have zero or more Recording Titles.

A Recording Title must have one and only one Music Category.

A Music Category may have zero or more Recording Titles.

**ASSUMPTIONS**

- A Recording Title will be identified with a single Music Category that defines the overall album or collection of tracks.
- There will be only one Recording Artist or group listed for a Recording Title. The Recording Titles will not be compilations of multiple artists.
- The Recording Label of the Recording Title will be for the compilation of tracks, not the individual tracks themselves as they may have been previously released by another label or on another title.



**Recording Artist**

<u>Artist ID</u>	Artist_name	Artist_notes
1001	The Popular Mechanics	Classic rock sound
1002	Crawdad Stare	Grunge Rock style
1003	Dog House Rules	Smooth Jazz from Chicago area.
1004	Mary Saveley	Soft soul jazz

**Music Category**

<u>Category ID</u>	Category_name
4001	Classical
4002	Rock
4003	Jazz
4004	Country
4005	New Age

**Recording Label**

<u>Label ID</u>	Label-name
5001	Goes to 11, Inc.
5002	Hippo Records

**Media Format**

<b><u>Media_ID</u></b>	<b><u>Record_ID</u></b>	Media_ID	Media_name
3001	2001	6001	CD
3002	2002	6002	Cassette
3003	2003	6001	CD
3004	2004	6002	Cassette
3005	2005	6001	CD
3006	2006	6001	CD
3007	2007	6001	CD

**Recording Title**

<b><u>Record_ID</u></b>	<b><u>Label_ID</u></b>	<b><u>Category_ID</u></b>	<b><u>Artist_ID</u></b>	Num_tracks	Record_name
2001	5001	4002	1002	10	Noise in the Garage
2002	5002	4002	1002	12	Look Both Ways
2003	NULL	4002	1002	10	Outback
2004	NULL	4003	1003	7	Short Circuit
2005	NULL	4005	1004	6	Meditations
2006	NULL	4001	1001	5	Opus 65
2007	5001	4002	1002	10	Sounds Better Louder

<b><u>Record_notes</u></b>
1 <sup>st</sup> recording to break top 10 by this artist
Artists original release. Underground until Noise in the Garage was popular.
NULL

**STEP 1: DATABASE AND TABLE CREATION**

**CREATE DATABASE**

```
CREATE DATABASE MUSIC_INVENTORY
```

\*\*\*\*\*

**CREATE TABLES**

```
CREATE TABLE RECORDING_LABEL (  
  Label_ID Integer(4) NOT NULL,  
  Label_name Char(50),  
  PRIMARY KEY (Label_ID));
```

\*\*\*\*\*

```
CREATE TABLE MUSIC_CATEGORY (  
  Category_ID Integer(4) NOT NULL,  
  Category_name Char(25),  
  PRIMARY KEY (Category_ID));
```

The screenshot shows the Toad for MySQL interface. The SQL editor contains the following code:

```
CREATE TABLE MUSIC_CATEGORY (
  Category_ID Integer(4) NOT NULL,
  Category_name Char(25),
  PRIMARY KEY (Category_ID));
```

The Results pane shows a single message:

Category	Timestamp	Duration	Message	Line	Position
Statement	2/28/2011 7:21:32 PM	0:00:00.046	Executed Successfully	1	0

The message text below the table reads: "CREATE TABLE MUSIC\_CATEGORY ( Category\_ID Integer(4) NOT NULL, Category\_name Char(25), PRIMARY KEY (Category\_ID))".

```
CREATE TABLE RECORDING_ARTIST (
  Artist_ID Integer(4) NOT NULL,
  Artist_name Char(50),
  Artist_notes Text(256),
  PRIMARY KEY (Artist_ID));
```

The screenshot shows the Toad for MySQL interface. The SQL editor contains the following code:

```
CREATE TABLE RECORDING_ARTIST (
  Artist_ID Integer(4) NOT NULL,
  Artist_name Char(50),
  Artist_notes Text(256),
  PRIMARY KEY (Artist_ID));
```

The Results pane shows a single message:

Category	Timestamp	Duration	Message	Line	Position
Statement	2/28/2011 7:22:59 PM	0:00:00.484	Executed Successfully	1	0

The message text below the table reads: "CREATE TABLE RECORDING\_ARTIST ( Artist\_ID Integer(4) NOT NULL, Artist\_name Char(50), Artist\_notes Text(256), PRIMARY KEY (Artist\_ID))".

```
CREATE TABLE RECORDING_TITLE (
  Record_ID Integer(4) NOT NULL,
  Label_ID Integer(4),
  Category_ID Integer(4),
  Artist_ID Integer(4),
  Num_tracks Integer(4),
  Record_name Char(50),
  Record_notes Text(256),
  PRIMARY KEY (Record_ID),
  FOREIGN KEY (Label_ID)
  REFERENCES RECORDING_LABEL (Label_ID),
  FOREIGN KEY (Category_ID)
  REFERENCES MUSIC_CATEGORY (Category_ID),
  FOREIGN KEY (Artist_ID)
  REFERENCES RECORDING_ARTIST (Artist_ID));
```

```

CREATE TABLE RECORDING_TITLE (
  Record_ID      Integer(4) NOT NULL,
  Label_ID       Integer(4),
  Category_ID    Integer(4),
  Artist_ID      Integer(4),
  Num_tracks     Integer(4),
  Record_name    Char(50),
  Record_notes   Text(256),
  PRIMARY KEY   (Record_ID),
  FOREIGN KEY   (Label_ID)
    REFERENCES RECORDING_LABEL (Label_ID),
  FOREIGN KEY   (Category_ID)
    REFERENCES MUSIC_CATEGORY (Category_ID),
  FOREIGN KEY   (Artist_ID)
    REFERENCES RECORDING_ARTIST (Artist_ID));

```

Category	Timestamp	Duration	Message	Line	Position
Statement	2/28/2011 7:25:22 PM	0:00:00.062	Executed Successfully	1	0

```

CREATE TABLE MEDIA_FORMAT (
  Media_Record_ID Integer(4) NOT NULL,
  Record_ID       Integer(4),
  Media_ID        Integer(4),
  Media_name      Char(10),
  PRIMARY KEY    (Media_Record_ID),
  FOREIGN KEY    (Record_ID)
    REFERENCES RECORDING_TITLE (Record_ID));

```

```

CREATE TABLE MEDIA_FORMAT (
  Media_Record_ID Integer(4) NOT NULL,
  Record_ID       Integer(4),
  Media_ID        Integer(4),
  Media_name      Char(10),
  PRIMARY KEY    (Media_Record_ID),
  FOREIGN KEY    (Record_ID)
    REFERENCES RECORDING_TITLE (Record_ID));

```

Category	Timestamp	Duration	Message	Line	Position
Statement	2/28/2011 7:26:33 PM	0:00:00.062	Executed Successfully	1	0

## INSERT INTO TABLES

```

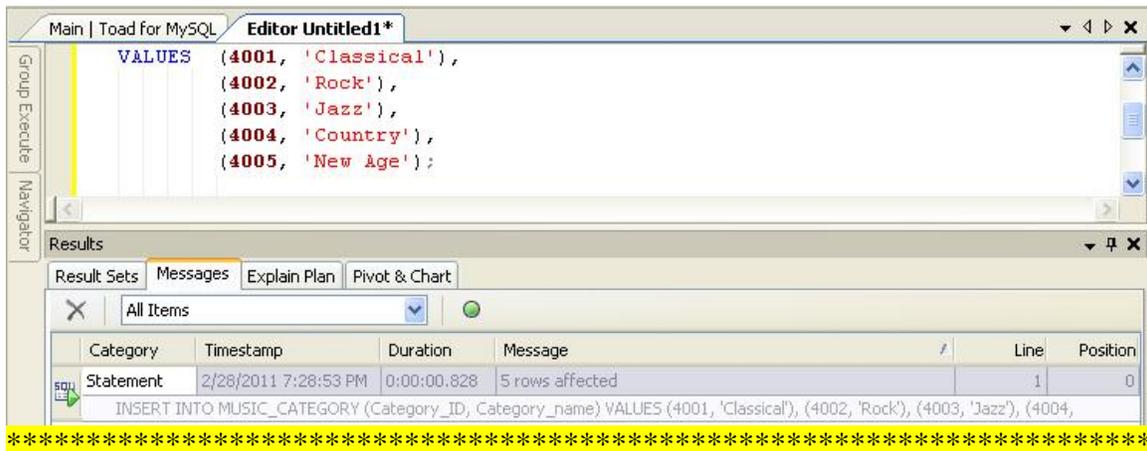
INSERT INTO RECORDING_ARTIST
  (Artist_ID, Artist_name, Artist_notes)

```

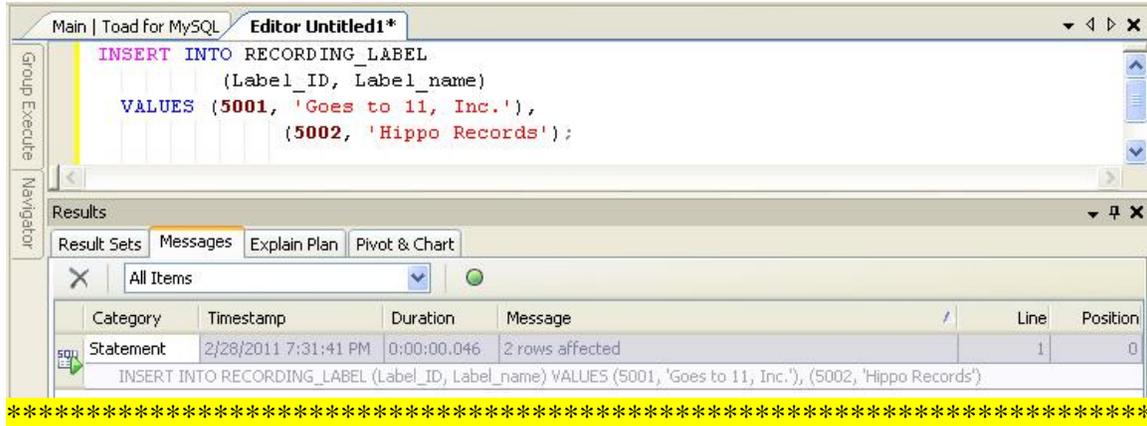
```
VALUES (1001, 'The Popular Mechanics', 'Classic rock sound'),  
(1002, 'Crawdada Stare', 'Grunge rock style.'),  
(1003, 'Dog House Rules', 'Smooth Jazz from Chicago area'),  
(1004, 'Mary Savely', 'Soft soul jazz');
```



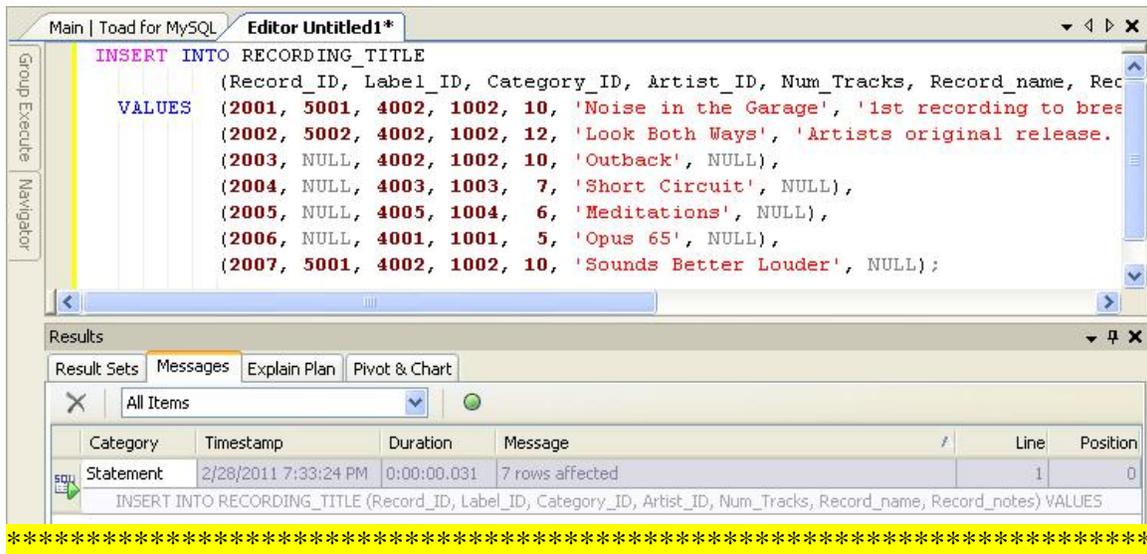
```
INSERT INTO MUSIC_CATEGORY  
(Category_ID, Category_name)  
VALUES (4001, 'Classical'),  
(4002, 'Rock'),  
(4003, 'Jazz'),  
(4004, 'Country'),  
(4005, 'New Age');
```



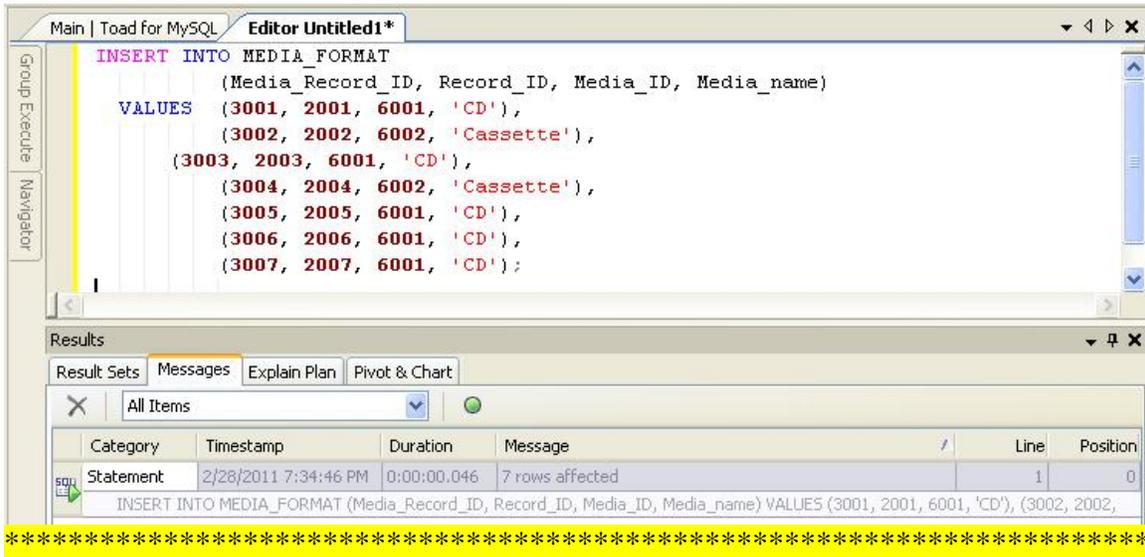
```
INSERT INTO RECORDING_LABEL  
(Label_ID, Label_name)  
VALUES (5001, 'Goes to 11, Inc.'),  
(5002, 'Hippo Records');
```



```
INSERT INTO RECORDING_TITLE
  (Record_ID, Label_ID, Category_ID, Artist_ID, Num_Tracks,
  Record_name, Record_notes)
VALUES (2001, 5001, 4002, 1002, 10, 'Noise in the Garage', '1st
recording to break top 10 by this artist'),
       (2002, 5002, 4002, 1002, 12, 'Look Both Ways', 'Artists
original release. Underground until Noise in the Garage was popular.'),
       (2003, NULL, 4002, 1002, 10, 'Outback', NULL),
       (2004, NULL, 4003, 1003, 7, 'Short Circuit', NULL),
       (2005, NULL, 4005, 1004, 6, 'Meditations', NULL),
       (2006, NULL, 4001, 1001, 5, 'Opus 65', NULL),
       (2007, 5001, 4002, 1002, 10, 'Sounds Better Louder', NULL);
```



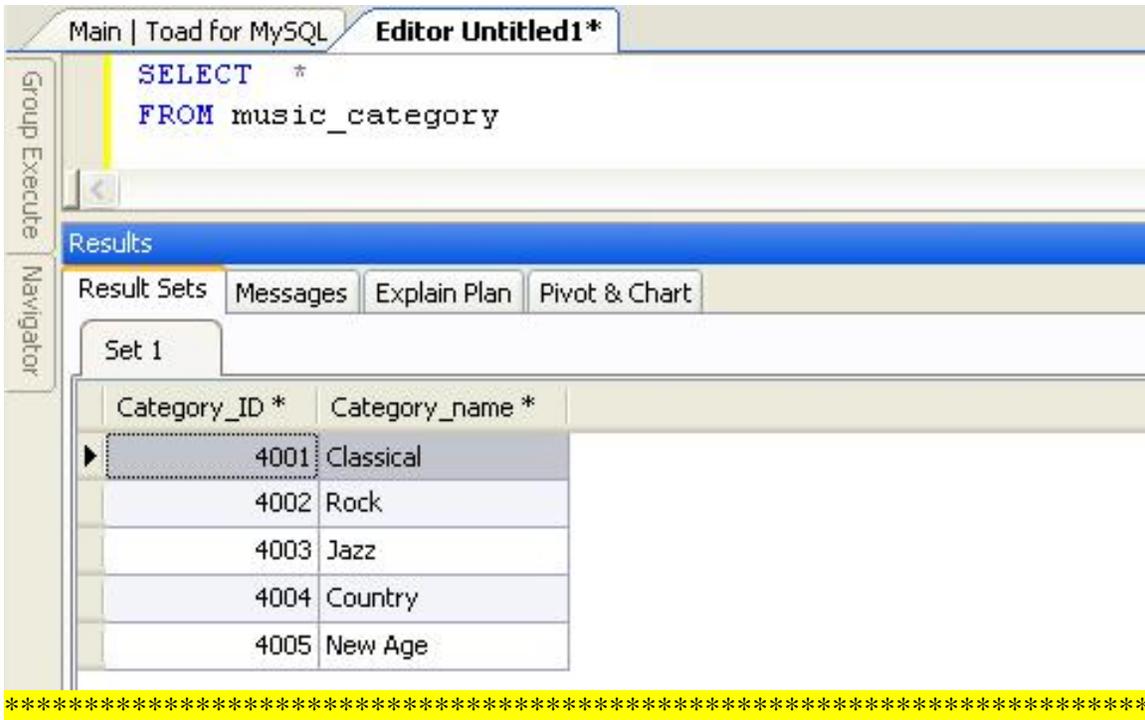
```
INSERT INTO MEDIA_FORMAT
  (Media_Record_ID, Record_ID, Media_ID, Media_name)
VALUES (3001, 2001, 6001, 'CD'),
       (3002, 2002, 6002, 'Cassette'),
       (3003, 2003, 6001, 'CD'),
       (3004, 2004, 6002, 'Cassette'),
       (3005, 2005, 6001, 'CD'),
       (3006, 2006, 6001, 'CD'),
       (3007, 2007, 6001, 'CD');
```



**STEP 2: STATEMENTS**

1. Show all information available on Music Categories

```
SELECT *
FROM music_category
```



2. Show all the information available on Recordings.

```
SELECT *  
FROM recording_title;
```

The screenshot shows the Toad for MySQL interface. The SQL editor contains the query: `SELECT * FROM recording_title;`. The Results pane displays a table with the following data:

Record_ID*	Label_ID	Category_ID	Artist_ID	Num_tracks	Record_name	Record_notes
2001	5001	4002	1002	10	Noise in the Garage	1st recording to break top 10 by this artist
2002	5002	4002	1002	12	Look Both Ways	Artists original release. Underground until Noi
2003		4002	1002	10	Outback	{null}
2004		4003	1003	7	Short Circuit	{null}
2005		4005	1004	6	Meditations	{null}
2006		4001	1001	5	Opus 65	{null}
2007	5001	4002	1002	10	Sounds Better Louder	{null}

**3. Show all the information available on Recording Artists.**

```
SELECT *  
FROM recording_artist;
```

The screenshot shows the Toad for MySQL interface. The SQL editor contains the query: `SELECT * FROM recording_artist;`. The Results pane displays a table with the following data:

Artist_ID*	Artist_name	Artist_notes
1001	The Popular Mechanics	Classic rock sound
1002	Crawdad Stare	Grunge rock style.
1003	Dog House Rules	Smooth Jazz from Chicago area
1004	Mary Savely	Soft soul jazz

**4. For all Recording Artists, show the artist's name, recording title, recording format and the number of tracks available on the recording.**

```
SELECT Artist_name, Record_name, Media_name, Num_Tracks  
FROM recording_artist, recording_title, media_format  
WHERE recording_artist.Artist_ID = recording_title.Artist_ID  
AND recording_title.Record_ID = media_format.Record_ID  
ORDER BY Artist_name, Record_name;
```

The screenshot shows the Toad for MySQL interface. The SQL editor contains the following query:

```
SELECT Artist_name, Record_name, Media_name, Num_Tracks
FROM recording_artist, recording_title, media_format
WHERE recording_artist.Artist_ID = recording_title.Artist_ID
AND recording_title.Record_ID = media_format.Record_ID
ORDER BY Artist_name, Record_name;
```

The Results pane shows the following data:

Artist_name	Record_name	Media_name	Num_Tracks
Crawdad Stare	Look Both Ways	Cassette	12
Crawdad Stare	Noise in the Garage	CD	10
Crawdad Stare	Outback	CD	10
Crawdad Stare	Sounds Better Louder	CD	10
Dog House Rules	Short Circuit	Cassette	7
Mary Savely	Meditations	CD	6
The Popular Mechanics	Opus 65	CD	5

**5. For all Music Categories, show the category and recording title.**

```
SELECT Category_name, Record_name
FROM music_category, recording_title
WHERE recording_title.Category_ID = music_category.Category_ID
```

The screenshot shows the Toad for MySQL interface. The SQL editor contains the following query:

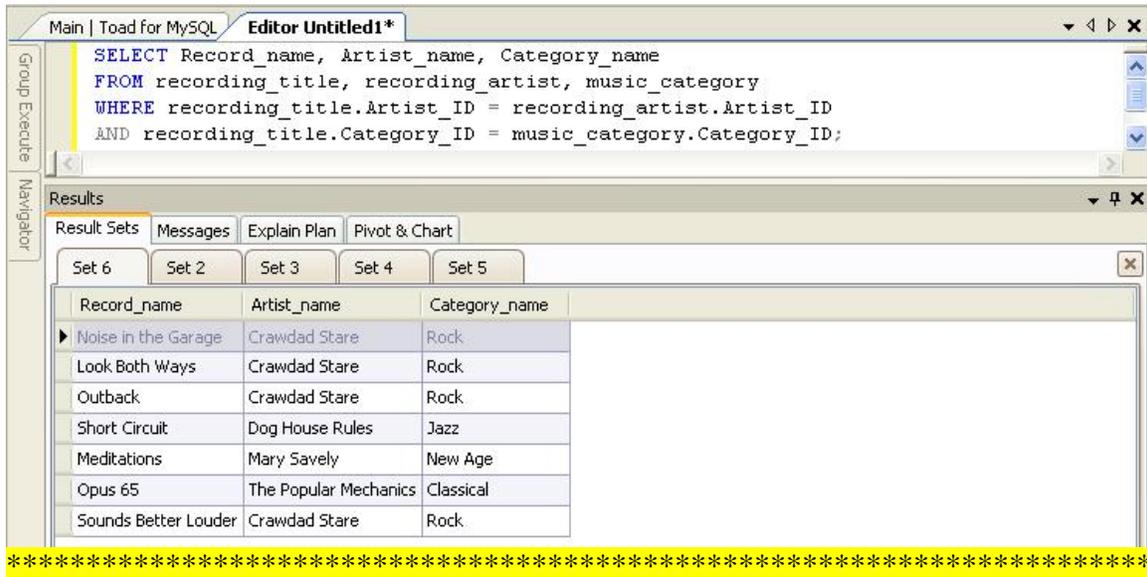
```
SELECT Category_name, Record_name
FROM music_category, recording_title
WHERE recording_title.Category_ID = music_category.Category_ID
```

The Results pane shows the following data:

Category_name	Record_name
Classical	Opus 65
Rock	Noise in the Garage
Rock	Look Both Ways
Rock	Outback
Rock	Sounds Better Louder
Jazz	Short Circuit
New Age	Meditations

6. For all Recordings, show the recording title, artist name, and music category.

```
SELECT Record_name, Artist_name, Category_name
FROM recording_title, recording_artist, music_category
WHERE recording_title.Artist_ID = recording_artist.Artist_ID
AND recording_title.Category_ID = music_category.Category_ID;
```



The screenshot shows the Toad for MySQL interface. The editor window contains the following SQL query:

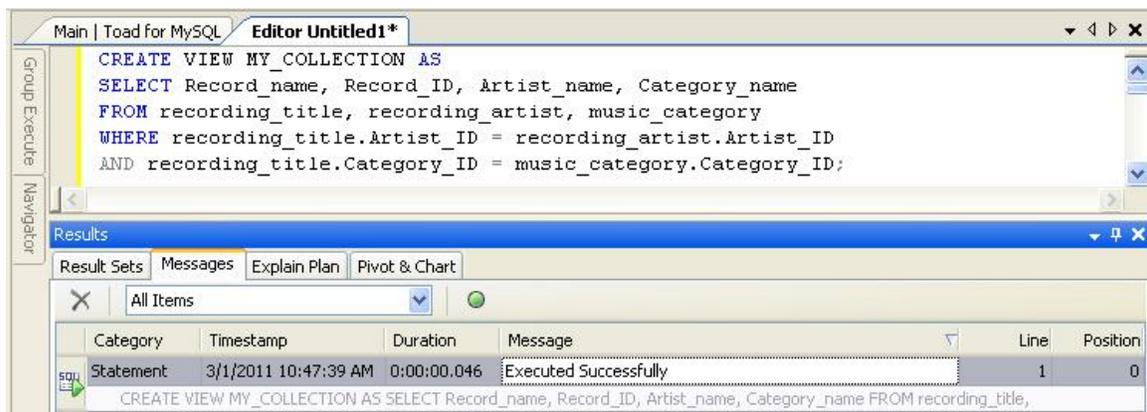
```
SELECT Record_name, Artist_name, Category_name
FROM recording_title, recording_artist, music_category
WHERE recording_title.Artist_ID = recording_artist.Artist_ID
AND recording_title.Category_ID = music_category.Category_ID;
```

The Results pane shows the following data:

Record_name	Artist_name	Category_name
Noise in the Garage	Crawdad Stare	Rock
Look Both Ways	Crawdad Stare	Rock
Outback	Crawdad Stare	Rock
Short Circuit	Dog House Rules	Jazz
Meditations	Mary Savely	New Age
Opus 65	The Popular Mechanics	Classical
Sounds Better Louder	Crawdad Stare	Rock

7. You have found that you are asking for the information in question 6 often as your collection grows. Create a view entitled: "my\_collection" AND show all information available from the view.

```
CREATE VIEW MY_COLLECTION AS
SELECT Record_name, Record_ID, Artist_name, Category_name
FROM recording_title, recording_artist, music_category
WHERE recording_title.Artist_ID = recording_artist.Artist_ID
AND recording_title.Category_ID = music_category.Category_ID;
```



The screenshot shows the Toad for MySQL interface. The editor window contains the following SQL query:

```
CREATE VIEW MY_COLLECTION AS
SELECT Record_name, Record_ID, Artist_name, Category_name
FROM recording_title, recording_artist, music_category
WHERE recording_title.Artist_ID = recording_artist.Artist_ID
AND recording_title.Category_ID = music_category.Category_ID;
```

The Results pane shows a message indicating successful execution:

Category	Timestamp	Duration	Message	Line	Position
Statement	3/1/2011 10:47:39 AM	0:00:00.046	Executed Successfully	1	0

```
SELECT *
FROM my_collection
ORDER BY Record_ID;
```

The screenshot shows the Toad for MySQL interface. The SQL editor contains the query: `SELECT * FROM my_collection ORDER BY Record_ID;`. The Results pane shows a table with the following data:

Record_name	Record_ID *	Artist_name	Category_name
Noise in the Garage	2001	Crawdad Stare	Rock
Look Both Ways	2002	Crawdad Stare	Rock
Outback	2003	Crawdad Stare	Rock
Short Circuit	2004	Dog House Rules	Jazz
Meditations	2005	Mary Savely	New Age
Opus 65	2006	The Popular Mechanics	Classical
Sounds Better Louder	2007	Crawdad Stare	Rock

**8. Using your newly created view, show all the information from "my\_collection" and the comments about the recordings;**

```
SELECT my_collection.Record_name AS Record_Name,
       my_collection.Artist_name AS Artist_Name,
       my_collection.Category_name AS Category_Name,
       my_collection.Record_ID AS Record_Id,
       Record_notes
FROM my_collection JOIN recording_title
WHERE my_collection.Record_ID = recording_title.Record_ID
ORDER BY Record_ID;
```

The screenshot shows the Toad for MySQL interface with the following query: `SELECT my_collection.Record_name AS Record_Name, my_collection.Artist_name AS Artist_Name, my_collection.Category_name AS Category_Name, my_collection.Record_ID AS Record_Id, Record_notes FROM my_collection JOIN recording_title WHERE my_collection.Record_ID = recording_title.Record_ID ORDER BY Record_ID;`. The Results pane shows a table with the following data:

Record_Name	Artist_Name	Category_Name	Record_Id *	Record_notes
Noise in the Garage	Crawdad Stare	Rock	2001	1st recording to break top 10 by this artist
Look Both Ways	Crawdad Stare	Rock	2002	Artists original release. Underground until Noise in the Garage was popular.
Outback	Crawdad Stare	Rock	2003	{null}
Short Circuit	Dog House Rules	Jazz	2004	{null}
Meditations	Mary Savely	New Age	2005	{null}
Opus 65	The Popular Mechanics	Classical	2006	{null}
Sounds Better Louder	Crawdad Stare	Rock	2007	{null}

\*\*\*\*\*

**9. Using your newly created view, show all the information from "my\_collection" and the comments about the recording artists.**

```
SELECT my_collection.Record_name AS Record_Name,  
       my_collection.Artist_name AS Artist_Name,  
       my_collection.Category_name AS Category_Name,  
       my_collection.Record_ID AS Record_Id,  
       Artist_notes  
FROM my_collection JOIN recording_artist  
WHERE my_collection.Artist_name = recording_artist.Artist_name  
ORDER BY Record_ID;
```

The screenshot shows the Toad for MySQL interface. The top pane displays the SQL query from the previous block. The bottom pane shows the results of the query in a table format. The table has five columns: Record\_Name, Artist\_Name, Category\_Name, Record\_Id \*, and Artist\_notes. There are seven rows of data.

Record_Name	Artist_Name	Category_Name	Record_Id *	Artist_notes
Noise in the Garage	Crawdad Stare	Rock	2001	Grunge rock style.
Look Both Ways	Crawdad Stare	Rock	2002	Grunge rock style.
Outback	Crawdad Stare	Rock	2003	Grunge rock style.
Short Circuit	Dog House Rules	Jazz	2004	Smooth Jazz from Chicago area
Meditations	Mary Savely	New Age	2005	Soft soul jazz
Opus 65	The Popular Mechanics	Classical	2006	Classic rock sound
Sounds Better Louder	Crawdad Stare	Rock	2007	Grunge rock style.

\*\*\*\*\*

**10. Delete your database and all information**

```
DROP DATABASE MUSIC_INVENTORY
```

The screenshot shows the Toad for MySQL interface. The top pane displays the SQL statement: DROP DATABASE MUSIC\_INVENTORY. The bottom pane shows the results of the execution, which is a message indicating that 6 rows were affected.

Category	Timestamp	Duration	Message	Line	Position
Statement	3/1/2011 11:44:02 AM	0:00:00.156	6 rows affected	1	0