

```
mysql> use music_company;
Database changed
mysql> show tables;
```

```
+-----+
| Tables_in_music_company |
+-----+
| album                    |
| composer                 |
| composergenreview        |
| composes                 |
| contain                  |
| has                      |
| record_label             |
| singer                   |
| songalbumsingerview      |
| songs                    |
| sungs                    |
+-----+
```

11 rows in set (0.22 sec)

=====PROCEDURE to get song details=====

```
mysql> CREATE PROCEDURE GetSongDetails(IN songID INT)
-> BEGIN
->     SELECT s.song_name AS SongName,
->           s.genre AS Genre,
->           a.album_name AS AlbumName,
->           a.release_date AS ReleaseDate,
->           si.singer_name AS SingerName,
->           c.composer_name AS ComposerName
->     FROM songs s
->    JOIN album a ON s.album_id = a.album_id
->    JOIN singer si ON s.singer_id = si.singer_id
->    JOIN composes cp ON s.song_id = cp.song_id
->    JOIN composer c ON cp.composer_id = c.composer_id
->   WHERE s.song_id = songID;
-> END //
```

Query OK, 0 rows affected (0.18 sec)

```
mysql>
mysql> DELIMITER ;
```

```
mysql> CALL GetSongDetails(10);
```

```
+-----+-----+-----+-----+-----+-----+
| SongName | Genre | AlbumName | ReleaseDate | SingerName | ComposerName |
+-----+-----+-----+-----+-----+-----+
| Song 10 | Reggae | Album 10 | 2023-10-10 | David Lee | William Taylor |
+-----+-----+-----+-----+-----+-----+
```

1 row in set (0.01 sec)

Query OK, 0 rows affected (0.01 sec)

=====PROCEDURE to update song genre=====

```
mysql> CREATE PROCEDURE UpdateSongGenre(IN songID INT, IN newGenre VARCHAR(255))
-> BEGIN
->     UPDATE songs
->     SET genre = newGenre
->     WHERE song_id = songID;
-> END //
```

Query OK, 0 rows affected (0.02 sec)

```
mysql> CALL UpdateSongGenre(1, 'Pop/Rock');
```

Query OK, 0 rows affected (0.01 sec)

```
mysql> select*from songs;
```

```
+-----+-----+-----+-----+-----+-----+-----+
| song_id | release_date | song_name | genre | no_of_plays | singer_id | album_id | composer_id |
+-----+-----+-----+-----+-----+-----+-----+
| 2 | 2022-01-02 | Song 2 | Rock | 150 | 1004 | 2 | 2005 |
| 3 | 2022-01-03 | Song 3 | Hip Hop | 120 | 1006 | 3 | 2001 |
| 4 | 2022-01-04 | Song 4 | Electronic | 80 | 1003 | 4 | 2003 |
```

5	2022-01-05	Song 5	Jazz	200	1001	5	2001
6	2022-01-06	Song 6	Blues	90	1007	6	2006
7	2022-01-07	Song 7	Country	110	1008	7	2005
8	2022-01-08	Song 8	Classical	130	1002	8	2010
9	2022-01-09	Song 9	R&B	170	1010	9	2009
10	2022-01-10	Song 10	Reggae	140	1005	10	2008

9 rows in set (0.00 sec)

=====PROCEDURE to get count of songs by genre=====

```
mysql> CREATE PROCEDURE GetSongCountByGenre(IN genreName VARCHAR(255), OUT songCount INT)
-> BEGIN
->     SELECT COUNT(*) INTO songCount
->     FROM songs
->     WHERE genre = genreName;
-> END //
```

Query OK, 0 rows affected (0.02 sec)

```
mysql>
mysql> DELIMITER ;
mysql> CALL GetSongCountByGenre('Pop', @count);
Query OK, 1 row affected (0.02 sec)
```

```
mysql> SELECT @count AS SongCount;
```

SongCount
0

1 row in set (0.00 sec)

```
mysql> CALL GetSongCountByGenre('Rock', @count);
Query OK, 1 row affected (0.00 sec)
```

```
mysql> SELECT @count AS SongCount;
```

SongCount
1

1 row in set (0.00 sec)

=====PROCEDURE to get top songs by plays=====

```
mysql> DELIMITER //
mysql>
mysql> CREATE PROCEDURE GetTopSongsByPlays(IN topN INT)
-> BEGIN
->     SELECT song_id, song_name, genre, no_of_plays
->     FROM songs
->     ORDER BY no_of_plays DESC
->     LIMIT topN;
-> END //
```

Query OK, 0 rows affected (0.02 sec)

```
mysql> DELIMITER ;
mysql> CALL GetTopSongsByPlays(10);
```

song_id	song_name	genre	no_of_plays
5	Song 5	Jazz	200
9	Song 9	R&B	170
2	Song 2	Rock	150
10	Song 10	Reggae	140
8	Song 8	Classical	130
3	Song 3	Hip Hop	120
7	Song 7	Country	110
6	Song 6	Blues	90
4	Song 4	Electronic	80

```
+-----+-----+-----+-----+
9 rows in set (0.00 sec)
```

```
mysql> DELIMITER //
```

```
mysql>
```

```
mysql> CREATE PROCEDURE GetAveragePlaysByGenre(IN genreName VARCHAR(255), OUT avgPlays DECIMAL(10,2))
-> BEGIN
->     SELECT AVG(no_of_plays) INTO avgPlays
->     FROM songs
->     WHERE genre = genreName;
-> END //
```

```
Query OK, 0 rows affected (0.02 sec)
```

```
mysql>
```

```
mysql> DELIMITER ;
```

```
mysql> CALL GetAveragePlaysByGenre('Classical', @avgPlays);
```

```
Query OK, 1 row affected (0.01 sec)
```

```
mysql> SELECT @avgPlays AS AveragePlays;
```

```
+-----+
| AveragePlays |
+-----+
|      130.00 |
+-----+
```

```
1 row in set (0.00 sec)
```