DBMS MINI PROJECT

Title: Music Database system

Submitted By:

Name:Appini Akhil

SRN : PES1UG20CS074

V Semester

Section : B

Short Description and Scope of the Project

- A database application about storing and managing songs
- Provides user-friendly interface, satisfying the needs of the consumers.
- Employs a new strategy that facilitates easy management of songs.
- It includes managing the songs of artist/writers and their albums.

Scope:

The project scope defines the description of the work that is required in delivering the Music database system.

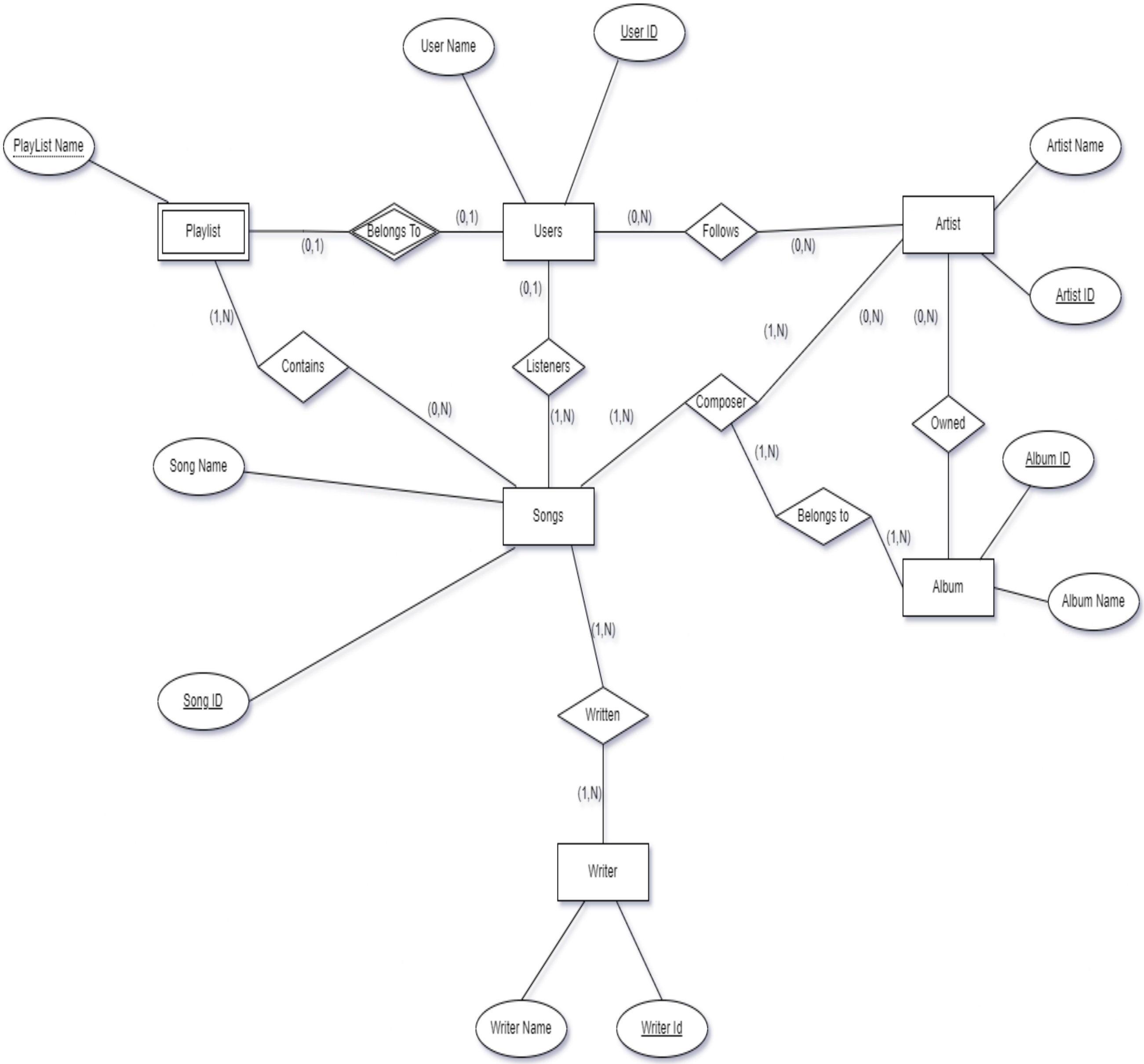
The following are the scopes of work during the course of the project:

- Study and understand the requirement of this project
- ER diagram and relational schema
- A minimalistic UI/front-end
- Creating and populating the database
- Using the SQL queries and show the reflections in the front-end developed

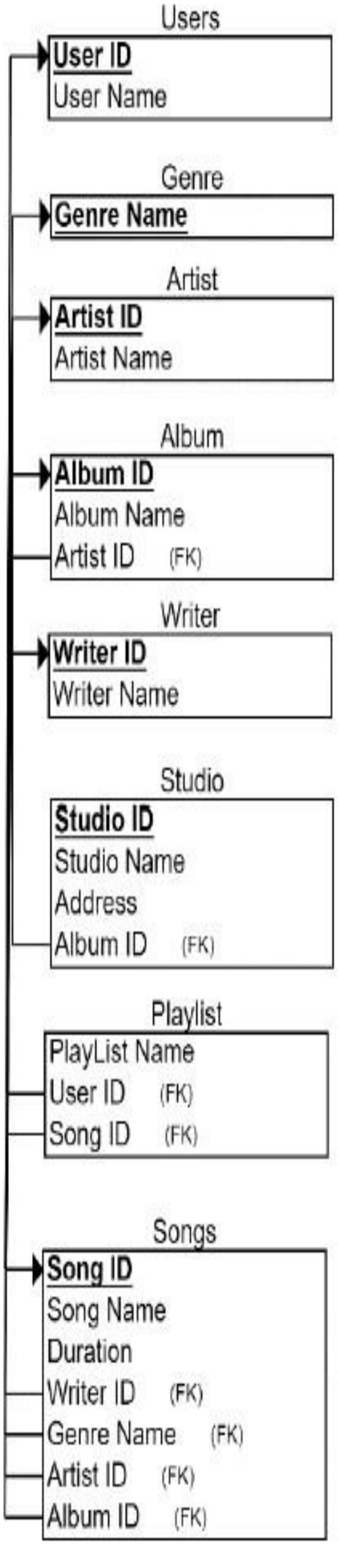
Modules Used:

- Songs: Displays all songs present in the database.
- Playlist: we can add songs to the playlist.
- Artist: Stores the details of Artist and we can update or delete artist details.
- Album: Stores the details of Album and their artists. We can update or delete artist and album details.
- Writers: Stores the details of writer and we can update or delete writer details.

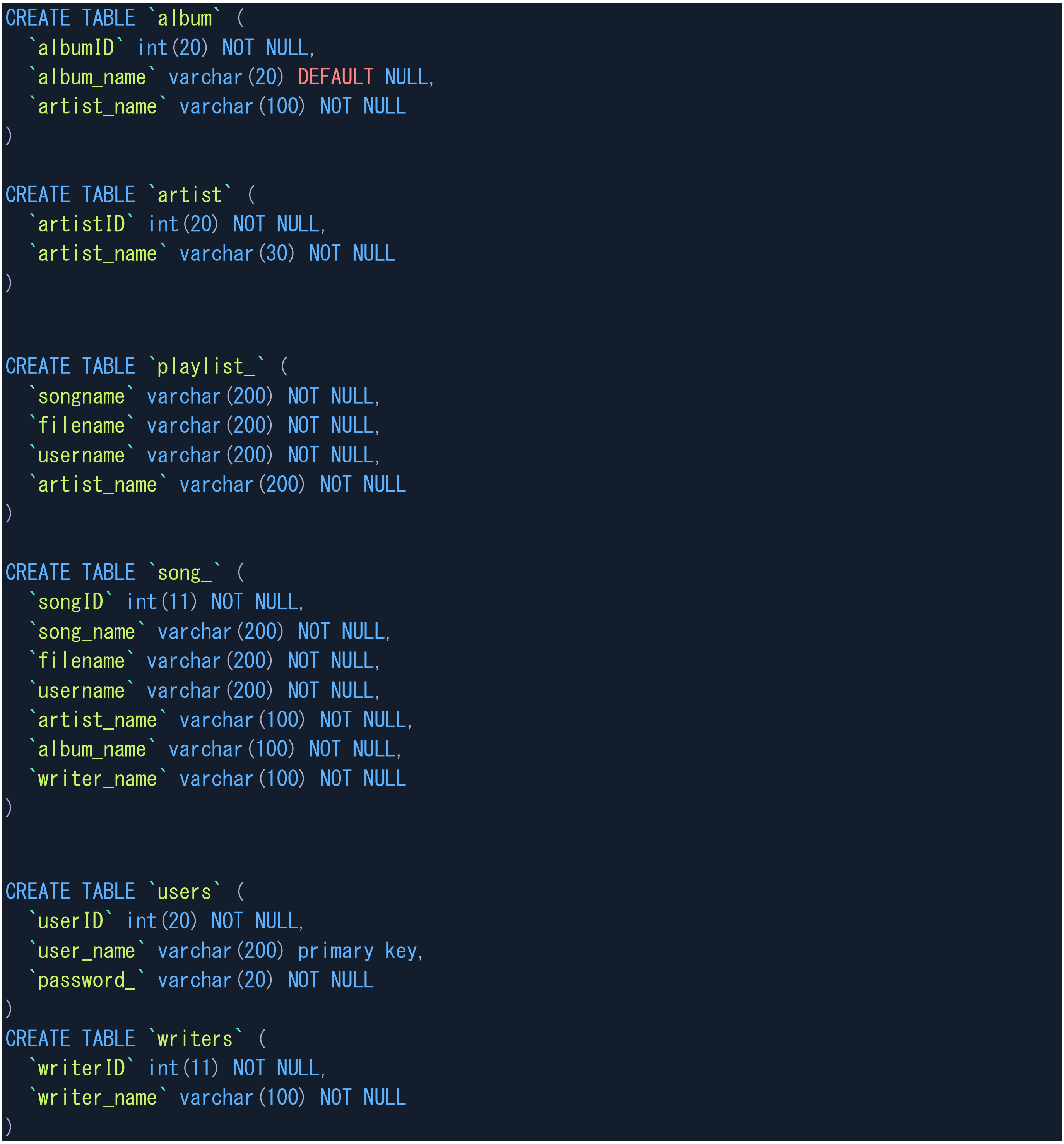
ER Diagram


User Name
User ID
PlayList Name
Artist Name
Playlist
(0,1)
Belongs To
(0,1)
Users
(0,N)
Follows
(0,N)
Artist
(0,1)
Artist ID
(1,N)
(1,N)
(0,N)
(0,N)
Contains
Listeners
Composer
Owned
(0,N)
(1,N)
(1,N)
(1,N)
Album ID
Song Name
Songs
Belongs to
(1,N)
Album
Album Name
(1,N)
Song ID
Written
(1,N)
Writer
Writer Name
Writer Id

Relational Schema



DDL statements - Building the database



```sql
CREATE TABLE `album` (
  `albumID` int(20) NOT NULL,
  `album_name` varchar(20) DEFAULT NULL,
  `artist_name` varchar(100) NOT NULL
)

CREATE TABLE `artist` (
  `artistID` int(20) NOT NULL,
  `artist_name` varchar(30) NOT NULL
)


CREATE TABLE `playlist_` (
  `songname` varchar(200) NOT NULL,
  `filename` varchar(200) NOT NULL,
  `username` varchar(200) NOT NULL,
  `artist_name` varchar(200) NOT NULL
)

CREATE TABLE `song_` (
  `songID` int(11) NOT NULL,
  `song_name` varchar(200) NOT NULL,
  `filename` varchar(200) NOT NULL,
  `username` varchar(200) NOT NULL,
  `artist_name` varchar(100) NOT NULL,
  `album_name` varchar(100) NOT NULL,
  `writer_name` varchar(100) NOT NULL
)


CREATE TABLE `users` (
  `userID` int(20) NOT NULL,
  `user_name` varchar(200) primary key,
  `password_` varchar(20) NOT NULL
)
CREATE TABLE `writers` (
  `writerID` int(11) NOT NULL,
  `writer_name` varchar(100) NOT NULL
)
```

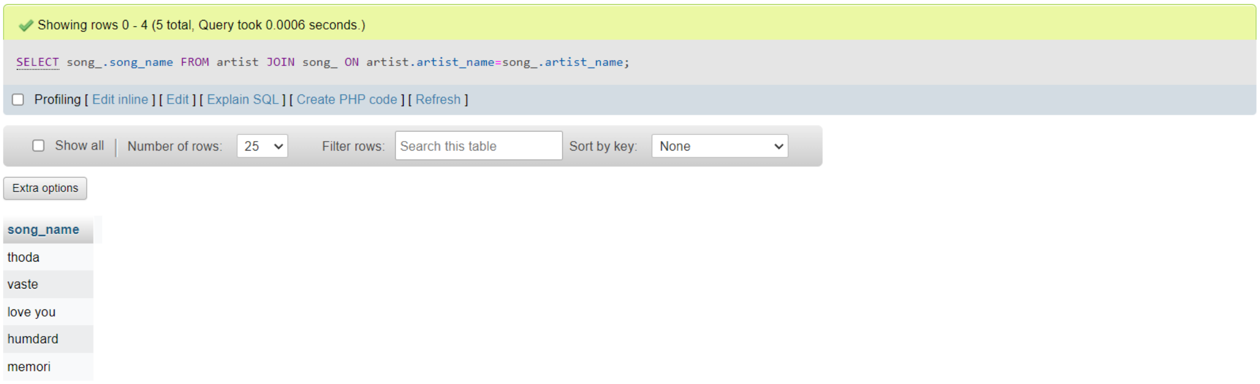
Populating the Database

```
INSERT INTO `album` (`albumID`, `album_name`, `artist_name`) VALUES
(1, 'thoda', 'Ben'),
(2, 'vaste', 'Dhvani'),
(3, 'encore', 'bieber'),
(4, 'Ek villan', 'Arijit'),
(5, 'jordi', 'maron 5');

INSERT INTO `artist` (`artistID`, `artist_name`) VALUES
(1, 'Ben'),
(2, 'Dhvani'),
(3, 'bieber'),
(4, 'Arijit'),
(5, 'maron 5');

INSERT INTO `playlist_` (`songname`, `filename`, `username`, `artist_name`) VALUES
('thoda', 'Thoda-Thoda-Pyar(PaglaSongs).mp3', 'Akhil', 'Ben'),
('vaste', 'vaaste.mp3', 'Akhil', 'Dhvani'),
('love you', 'Let-Me-Love-You_320(PaglaSongs).mp3', 'Akhil', 'bieber'),
('humdard', 'Humdard.mp3', 'Akhil', 'Arijit');

INSERT INTO `users` (`userID`, `user_name`, `password_`) VALUES
(7, 'Akhil', '1234');

INSERT INTO `writers` (`writerID`, `writer_name`) VALUES
(1, 'kumar'),
(2, 'Arafat'),
(3, 'bieber'),
(4, 'mithoon'),
(5, 'levin');


INSERT INTO `song_` (`songID`, `song_name`, `filename`, `username`, `artist_name`,
`album_name`, `writer_name`) VALUES
(1, 'thoda', 'Thoda-Thoda-Pyar(PaglaSongs).mp3', 'Akhil', 'Ben', 'thoda', 'kumar'),
(2, 'vaste', 'vaaste.mp3', 'Akhil', 'Dhvani', 'vaste', 'Arafat'),
(3, 'love you', 'Let-Me-Love-You_320(PaglaSongs).mp3', 'Akhil', 'bieber', 'encore',
'bieber'),
(4, 'humdard', 'Humdard.mp3', 'Akhil', 'Arijit', 'Ek villan', 'mithoon'),
(5, 'memori', 'Memories.mp3', 'Akhil', 'maron 5', 'jordi', 'levin');
```

Join Queries

Write the query in English Language, Show the equivalent SQL statement and also a screenshot of the query and the results

```
SELECT song_.song_name  FROM artist JOIN song_ ON artist.artist_name=song_.artist_name
```



Showing rows 0 - 4 (5 total, Query took 0.0006 seconds.)

```
SELECT song_.song_name FROM artist JOIN song_ ON artist.artist_name=song_.artist_name;
```

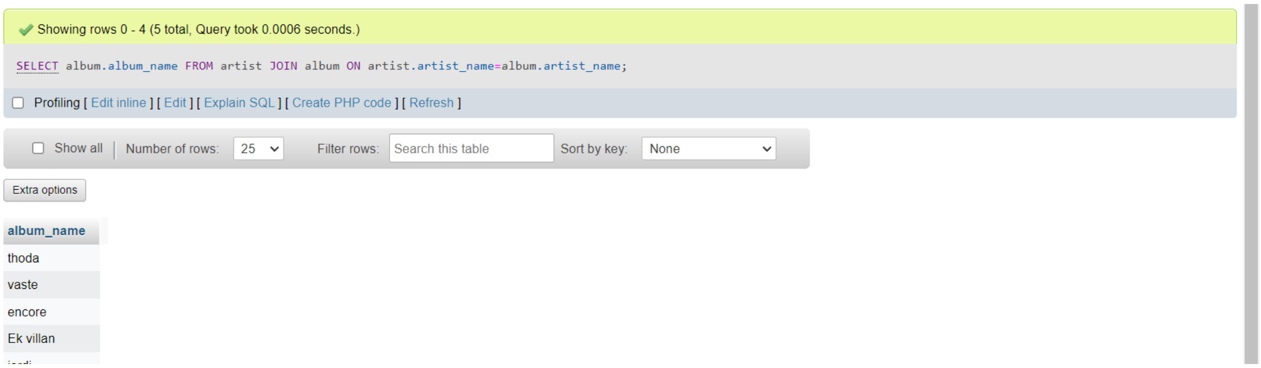
□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

song_name
thoda
vaste
love you
humdard
memori

```
SELECT album.album_name  FROM artist JOIN album ON artist.artist_name=album.artist_name
```



Showing rows 0 - 4 (5 total, Query took 0.0006 seconds.)

```
SELECT album.album_name FROM artist JOIN album ON artist.artist_name=album.artist_name;
```

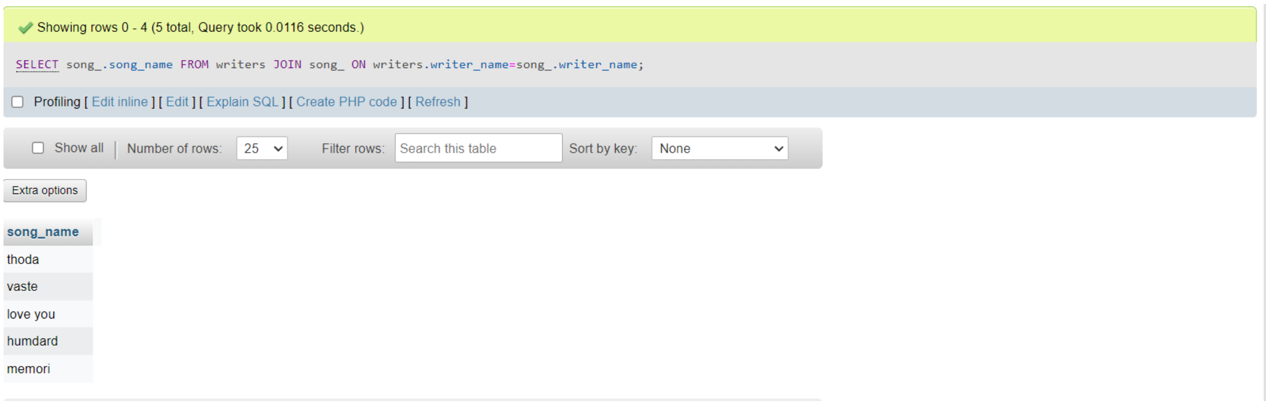
□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

album_name
thoda
vaste
encore
Ek villan

```
SELECT song_.song_name  FROM writers JOIN song_ ON writers.writer_name=song_.writer_name
```



Showing rows 0 - 4 (5 total, Query took 0.0116 seconds.)

```
SELECT song_.song_name FROM writers JOIN song_ ON writers.writer_name=song_.writer_name;
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

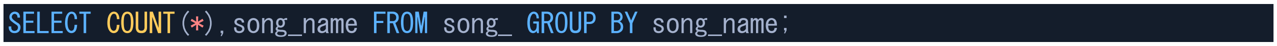
□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

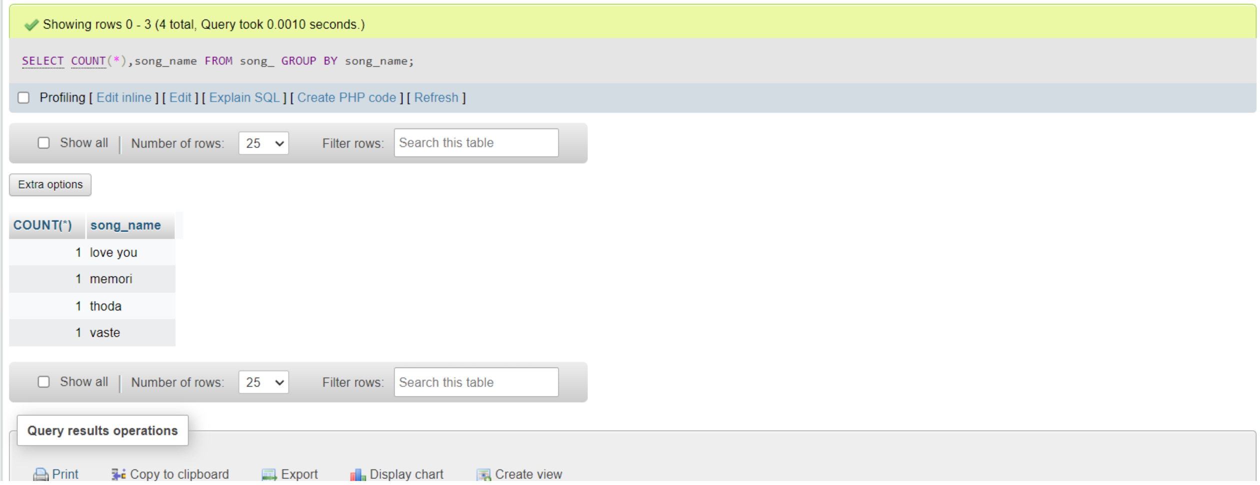
song_name
thoda
vaste
love you
humdard
memori

Aggregate Functions

Write the query in English Language, Show the equivalent SQL statement and also ascreenshot of the query and the results



```
SELECT COUNT(*),song_name FROM song_ GROUP BY song_name;
```



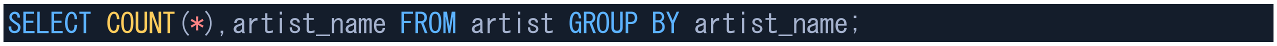
✓ Showing rows 0 - 3 (4 total, Query took 0.0010 seconds.)

```
SELECT COUNT(*),song_name FROM song_ GROUP BY song_name;
```

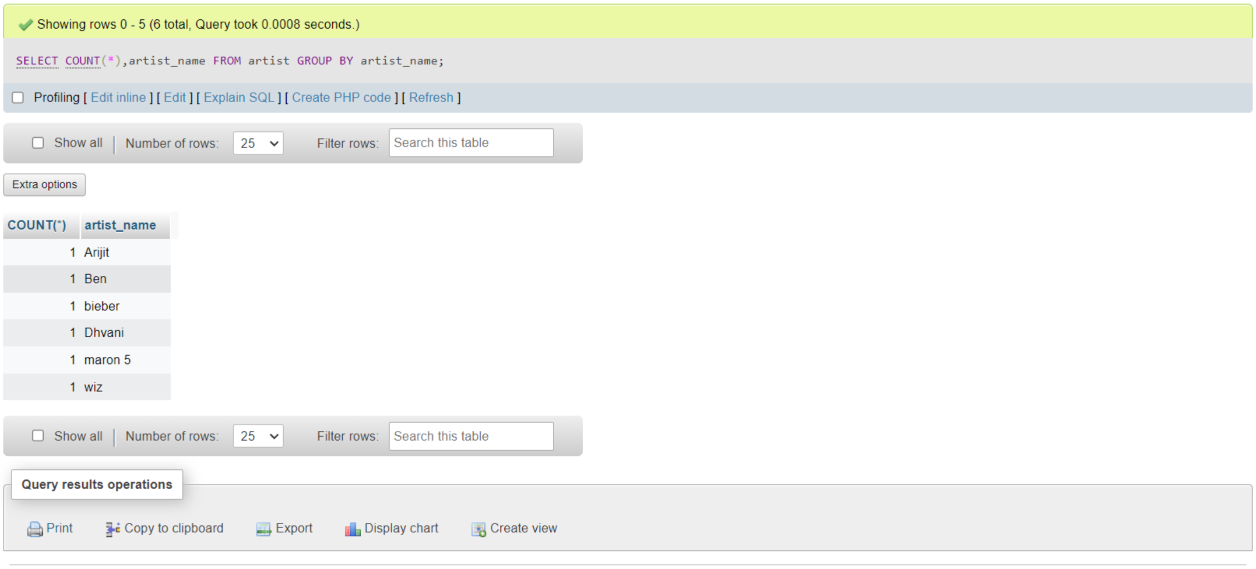
□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
□ Show all | Number of rows: 25 Filter rows: Search this table
Extra options

COUNT(*)	song_name
1	love you
1	memori
1	thoda
1	vaste

□ Show all | Number of rows: 25 Filter rows: Search this table
Query results operations
Print
Copy to clipboard
Export
Display chart
Create view



```
SELECT COUNT(*),artist_name FROM artist GROUP BY artist_name;
```



✓ Showing rows 0 - 5 (6 total, Query took 0.0008 seconds.)

```
SELECT COUNT(*),artist_name FROM artist GROUP BY artist_name;
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
□ Show all | Number of rows: 25 Filter rows: Search this table
Extra options

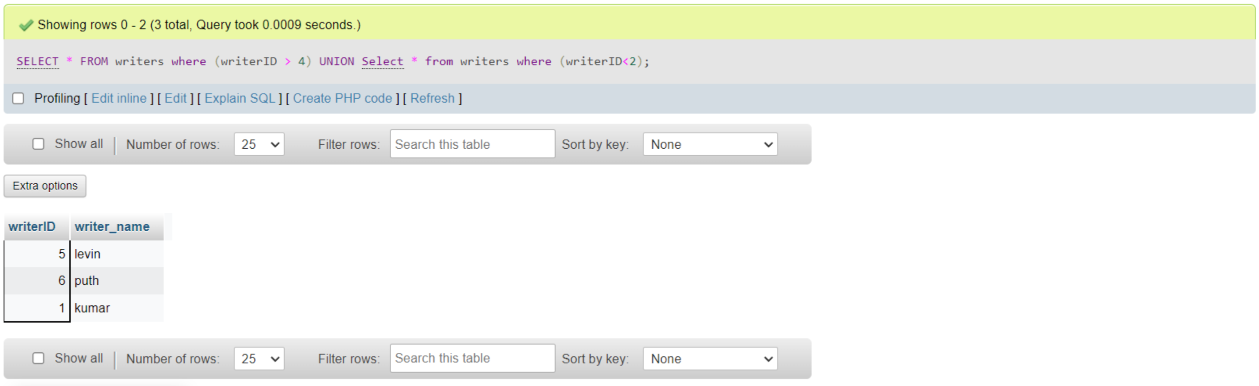
COUNT(*)	artist_name
1	Arijit
1	Ben
1	bieber
1	Dhvani
1	maron 5
1	wiz

□ Show all | Number of rows: 25 Filter rows: Search this table
Query results operations
Print
Copy to clipboard
Export
Display chart
Create view

Set Operations

Write the query in English Language, Show the equivalent SQL statement and also a screenshot of the query and the results

```
SELECT * FROM writers where (writerID > 4) UNION Select * from writers where (writerID<2);
```



✓ Showing rows 0 - 2 (3 total, Query took 0.0009 seconds.)

```
SELECT * FROM writers where (writerID > 4) UNION Select * from writers where (writerID<2);
```

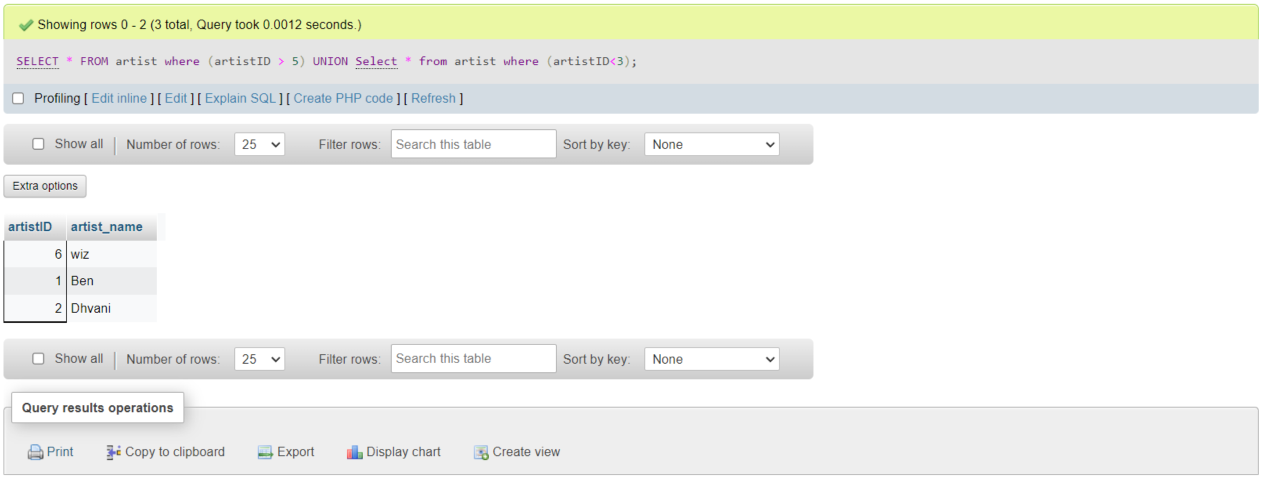
□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None
Extra options

writerID	writer_name
5	levin
6	puth
1	kumar

□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

```
SELECT * FROM artist  where (artistID > 5) UNION Select * from artist where (artistID<3);
```



✓ Showing rows 0 - 2 (3 total, Query took 0.0012 seconds.)

```
SELECT * FROM artist where (artistID > 5) UNION Select * from artist where (artistID<3);
```

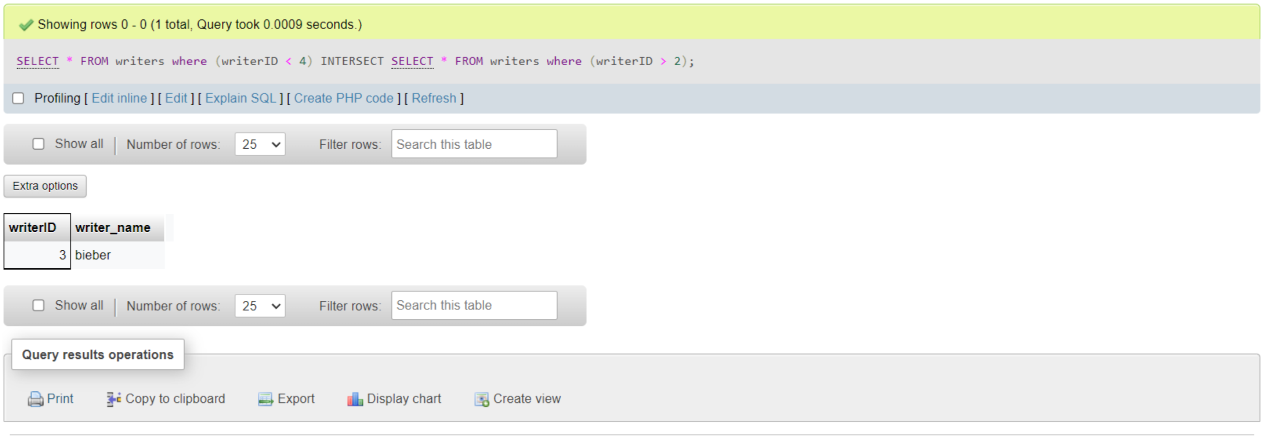
□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None
Extra options

artistID	artist_name
6	wiz
1	Ben
2	Dhvani

□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None
Query results operations
Print
Copy to clipboard
Export
Display chart
Create view

```
SELECT * FROM writers where (writerID < 4) INTERSECT SELECT * FROM writers where (writerID
> 2);
```



✓ Showing rows 0 - 0 (1 total, Query took 0.0009 seconds.)

```
SELECT * FROM writers where (writerID < 4) INTERSECT SELECT * FROM writers where (writerID > 2);
```

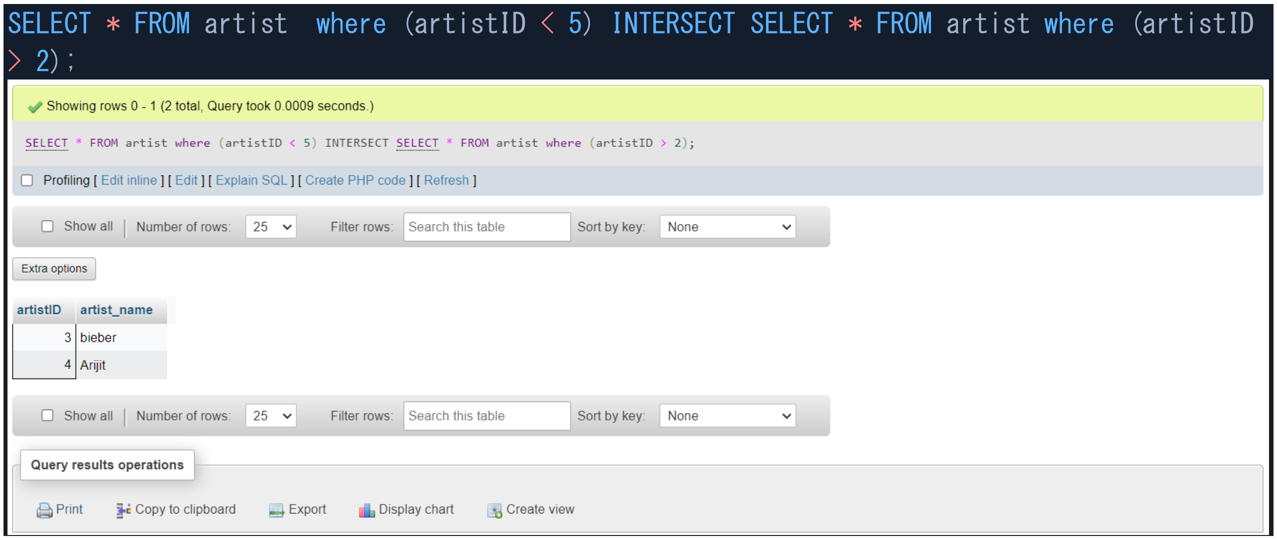
□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

□ Show all | Number of rows: 25 Filter rows: Search this table

Extra options

writerID	writer_name
3	bieber

□ Show all | Number of rows: 25 Filter rows: Search this table



```
SELECT * FROM artist  where (artistID < 5) INTERSECT SELECT * FROM artist where (artistID > 2);
```

✓ Showing rows 0 - 1 (2 total, Query took 0.0009 seconds.)

```
SELECT * FROM artist where (artistID < 5) INTERSECT SELECT * FROM artist where (artistID > 2);
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

artistID	artist_name
3	bieber
4	Arijit

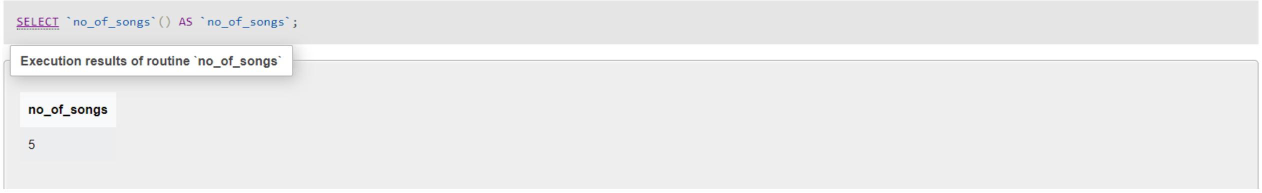
□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Functions and Procedures

Create a Function and Procedure. State the objective of the function / Procedure. Run and display the results.

Functions:

```
DELIMITER $$
CREATE FUNCTION no_of_songs()
RETURNS int
begin
declare num int;
SET num=(SELECT count(*) FROM song_);
RETURN num;
end
$$
DELIMITER ;
```

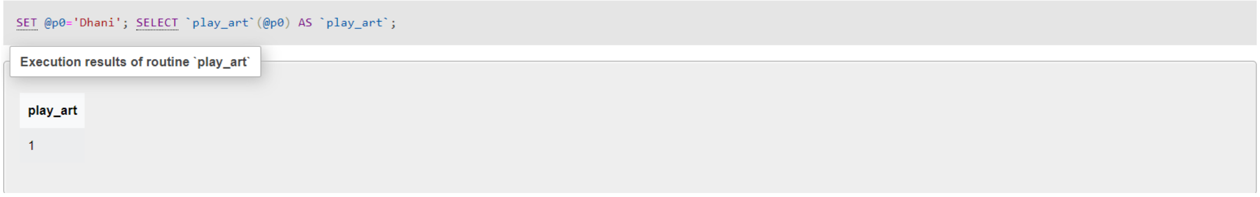


```
SELECT `no_of_songs`() AS `no_of_songs`;
```

Execution results of routine `no_of_songs`

no_of_songs
5

```
DELIMITER $$
CREATE FUNCTION play_art(artistname text) RETURNS int
begin
DROP TEMPORARY TABLE IF EXISTS playlist_artist;
CREATE TEMPORARY TABLE playlist_artist SELECT playlist_.songname  FROM artist JOIN
playlist_ ON artist.artist_name=playlist_.artist_name WHERE artist.artist_name =
artistname;
return 1;
end
$$
DELIMITER ;
```

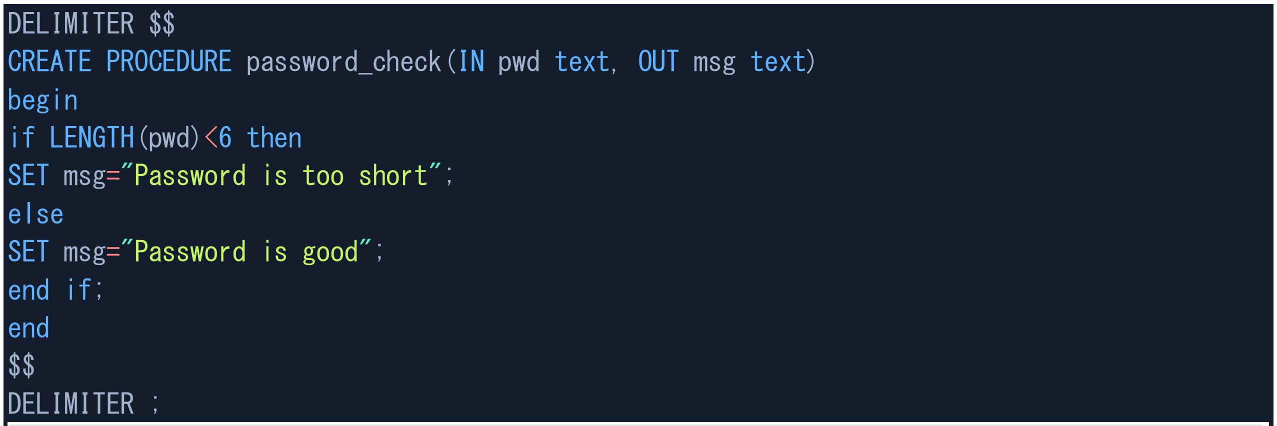


```
SET @p0='Dhani'; SELECT `play_art`(@p0) AS `play_art`;
```

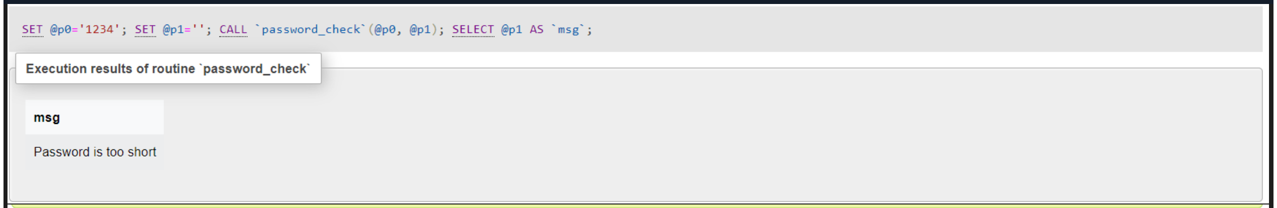
Execution results of routine `play_art`

play_art
1

Procedures:



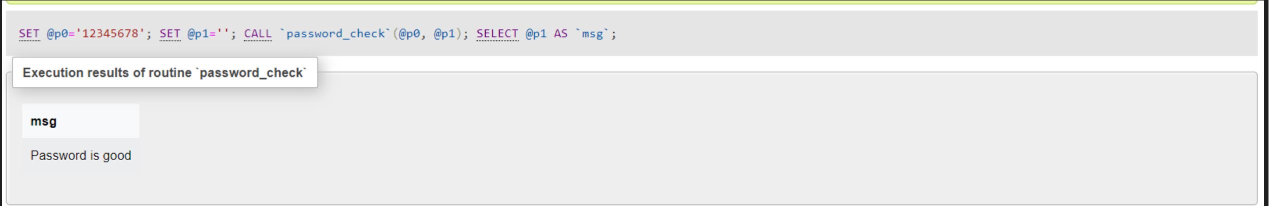
```
DELIMITER $$
CREATE PROCEDURE password_check(IN pwd text, OUT msg text)
begin
if LENGTH(pwd)<6 then
SET msg=”Password is too short”;
else
SET msg=”Password is good”;
end if;
end
$$
DELIMITER ;
```



```
SET @p0='1234'; SET @p1=''; CALL `password_check`(@p0, @p1); SELECT @p1 AS `msg`;
```

Execution results of routine `password_check`

msg
Password is too short



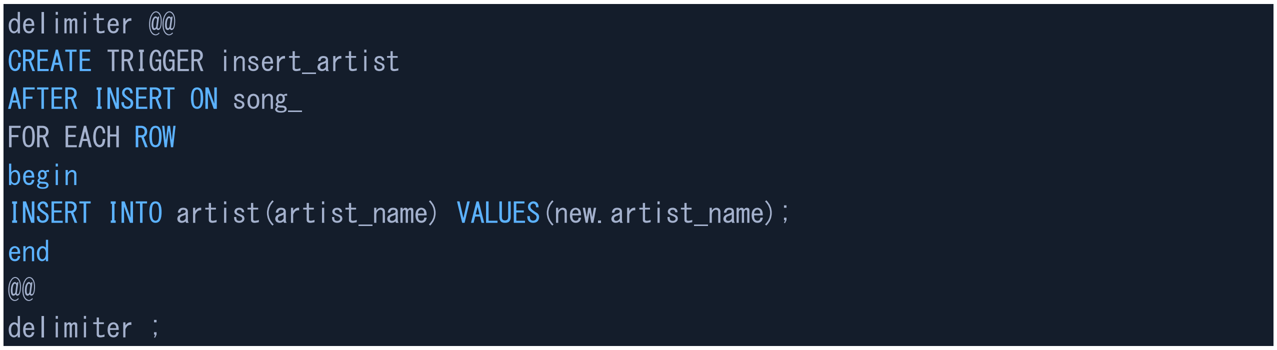
```
SET @p0='12345678'; SET @p1=''; CALL `password_check`(@p0, @p1); SELECT @p1 AS `msg`;
```

Execution results of routine `password_check`

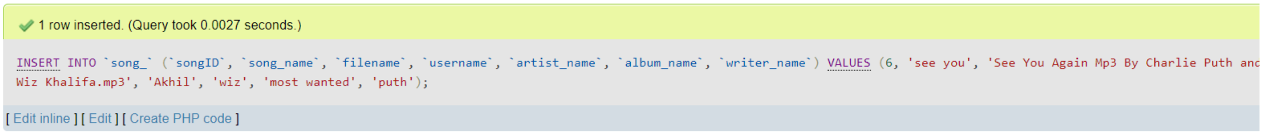
msg
Password is good

Triggers

Trigger:

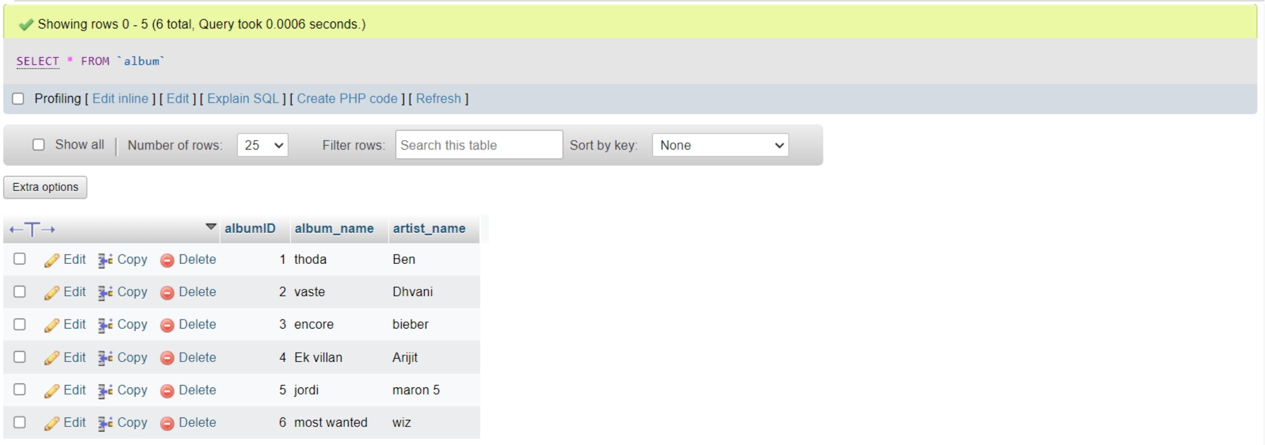


```
delimiter @@
CREATE TRIGGER insert_artist
AFTER INSERT ON song_
FOR EACH ROW
begin
INSERT INTO artist(artist_name) VALUES(new.artist_name);
end
@@
delimiter ;
```



✓ 1 row inserted. (Query took 0.0027 seconds.)

```
INSERT INTO `song_` (`songID`, `song_name`, `filename`, `username`, `artist_name`, `album_name`, `writer_name`) VALUES (6, 'see you', 'See You Again Mp3 By Charlie Puth and
Wiz Khalifa.mp3', 'Akhil', 'wiz', 'most wanted', 'puth');
```



✓ Showing rows 0 - 5 (6 total, Query took 0.0006 seconds.)

```
SELECT * FROM `album`
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
□ Show all
Filter rows: Search this table
Extra options
←T→

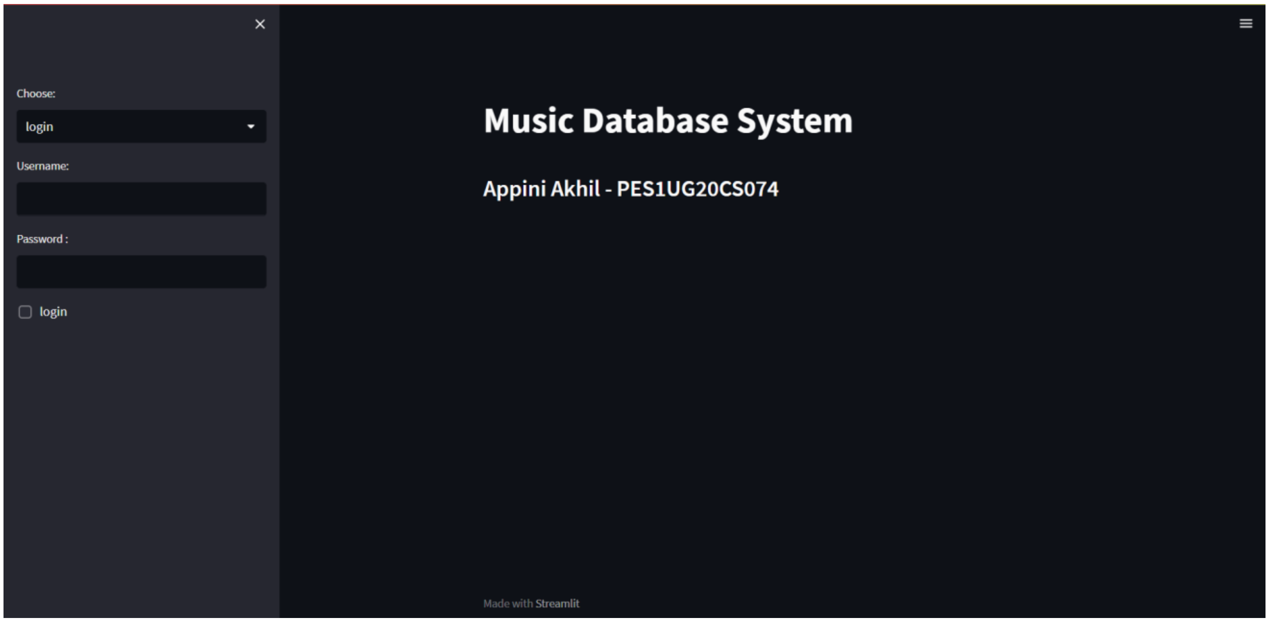
Number of rows: 25

Sort by key: None

	albumID	album_name	artist_name
□ Edit Copy Delete	1	thoda	Ben
□ Edit Copy Delete	2	vaste	Dhvani
□ Edit Copy Delete	3	encore	bieber
□ Edit Copy Delete	4	Ek villan	Arijit
□ Edit Copy Delete	5	jordi	maron 5
□ Edit Copy Delete	6	most wanted	wiz

Developing a Frontend

FrontEnd



Choose: login

Username:

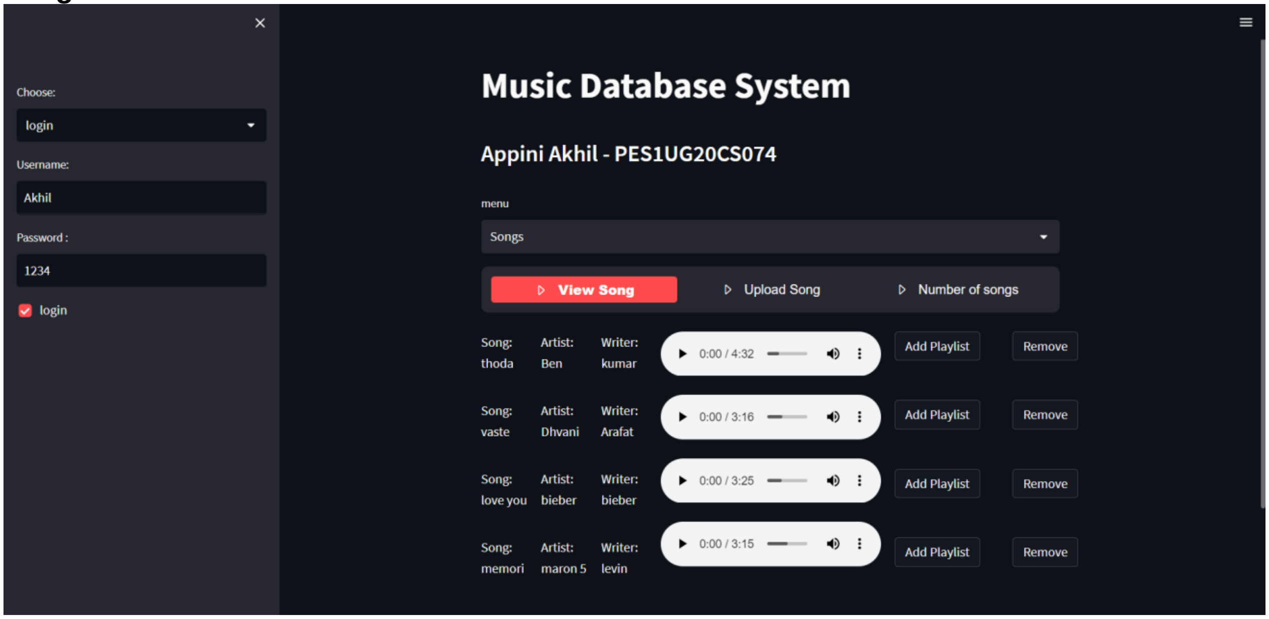
Password :

☐ login

Music Database System

Appini Akhil - PES1UG20CS074

songs



Choose: login

Username: Akhil

Password : 1234

☑ login

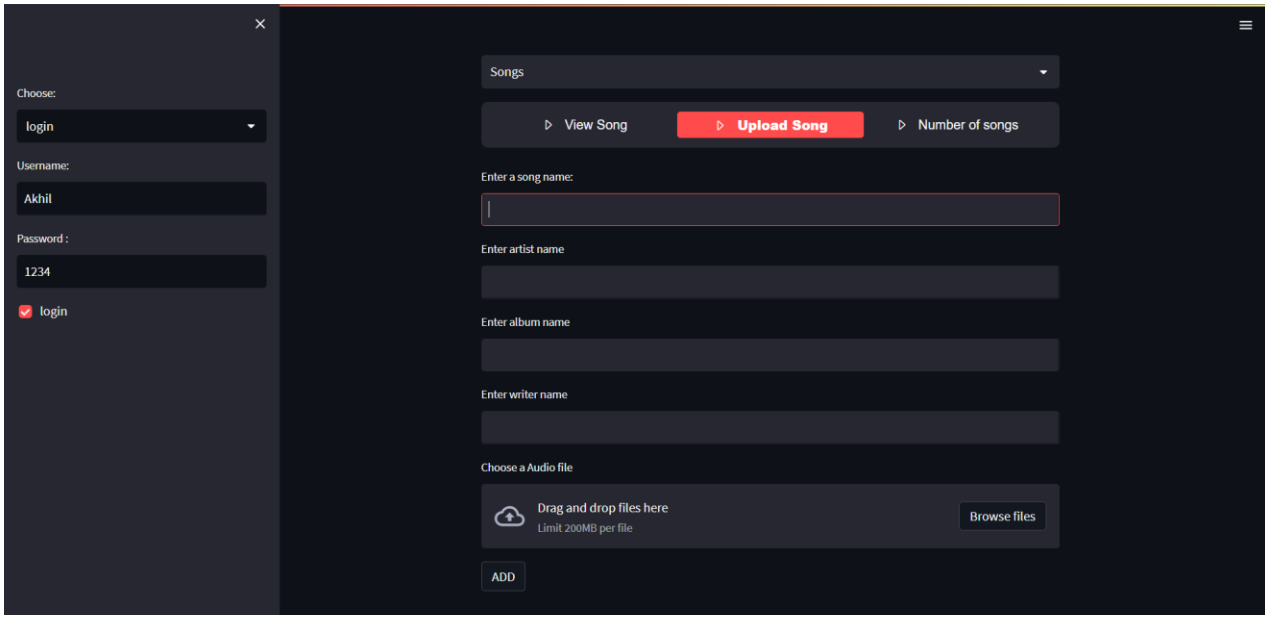
Music Database System

Appini Akhil - PES1UG20CS074

menu: Songs

Song:	Artist:	Writer:			
thoda	Ben	kumar	0:00 / 4:32	Add Playlist	Remove
vaste	Dhvani	Arafat	0:00 / 3:16	Add Playlist	Remove
love you	bieber	bieber	0:00 / 3:25	Add Playlist	Remove
memori	maron 5	levin	0:00 / 3:15	Add Playlist	Remove

Upload songs page



Choose:

login

Username:

Akhil

Password :

1234

☑ login

Songs

Enter a song name:

Enter artist name

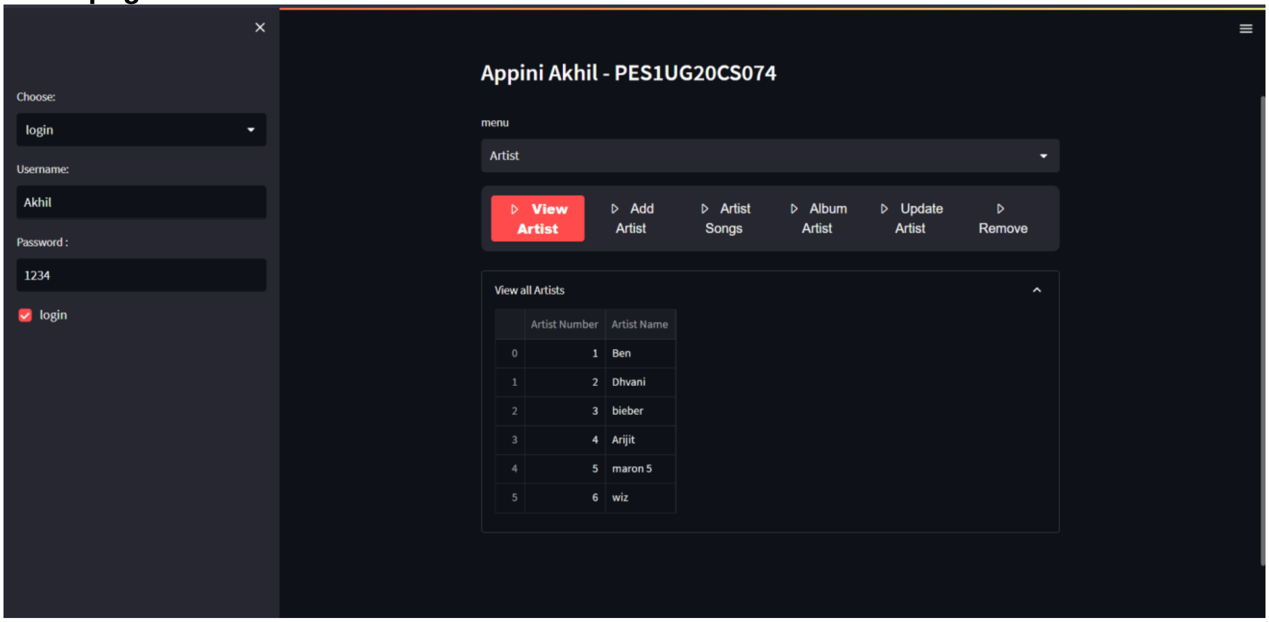
Enter album name

Enter writer name

Choose a Audio file

Drag and drop files here
Limit 200MB per file

Artist page



Choose:

login

Username:

Akhil

Password :

1234

☑ login

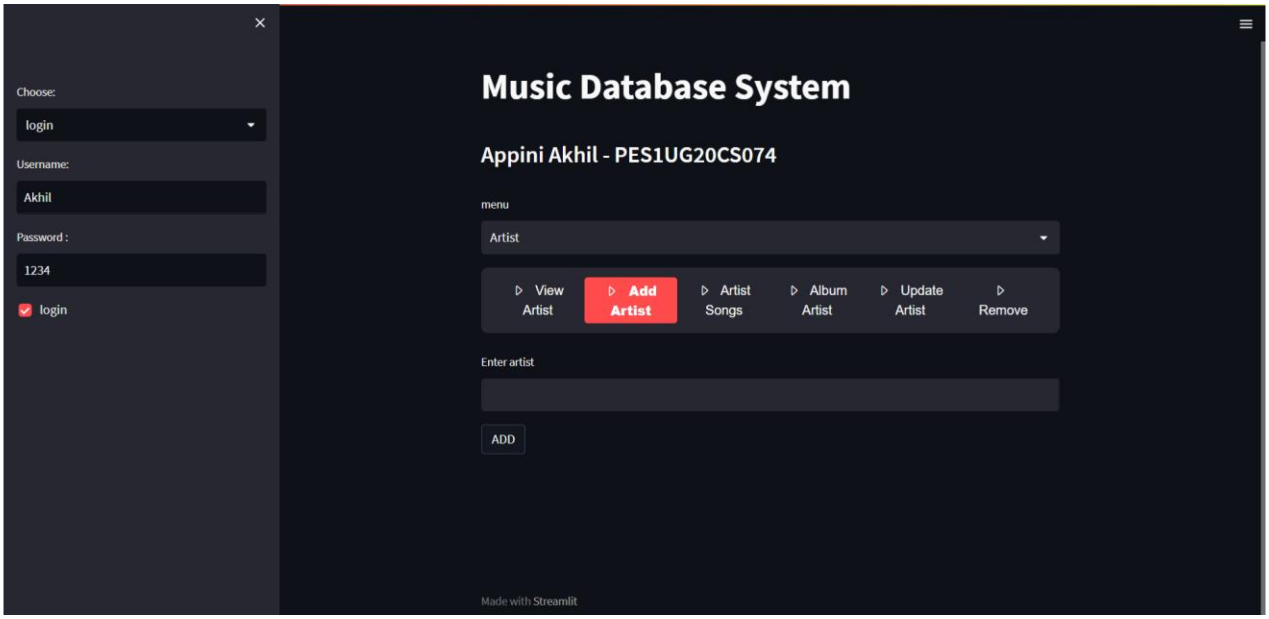
Appini Akhil - PES1UG20CS074

menu

Artist

View all Artists

	Artist Number	Artist Name
0	1	Ben
1	2	Dhvani
2	3	bieber
3	4	Arijit
4	5	maron 5
5	6	wiz



Choose: login

Username: Akhil

Password : 1234

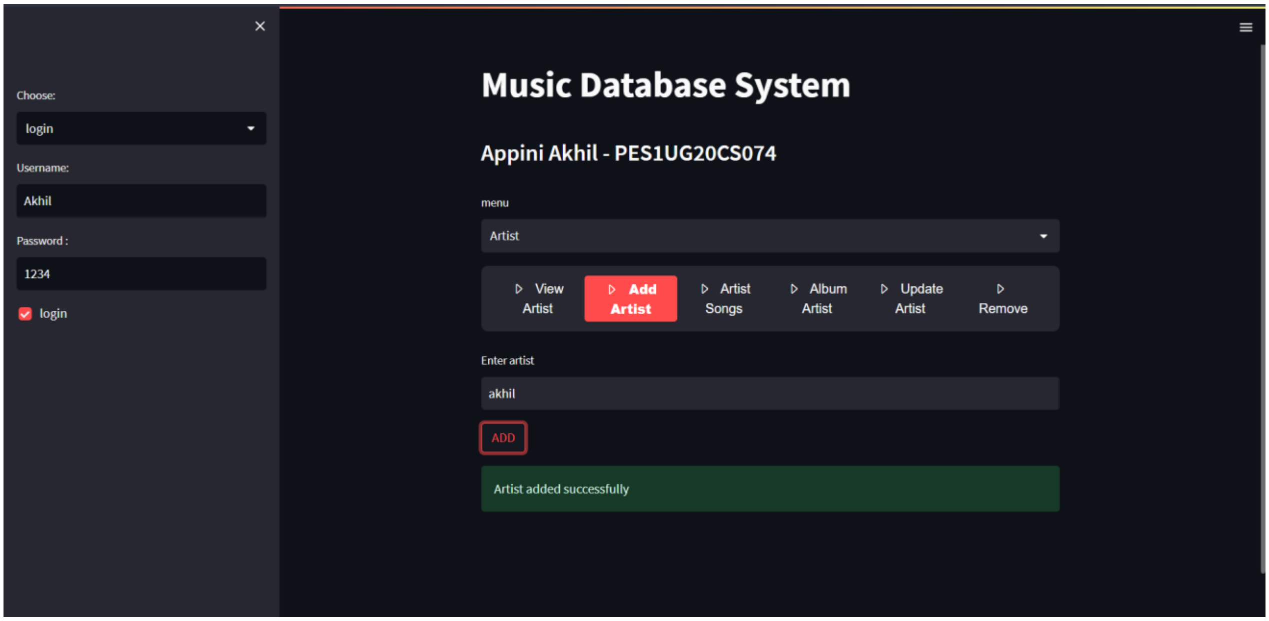
☑ login

Music Database System

Appini Akhil - PES1UG20CS074

menu: Artist

Enter artist



Choose: login

Username: Akhil

Password : 1234

☑ login

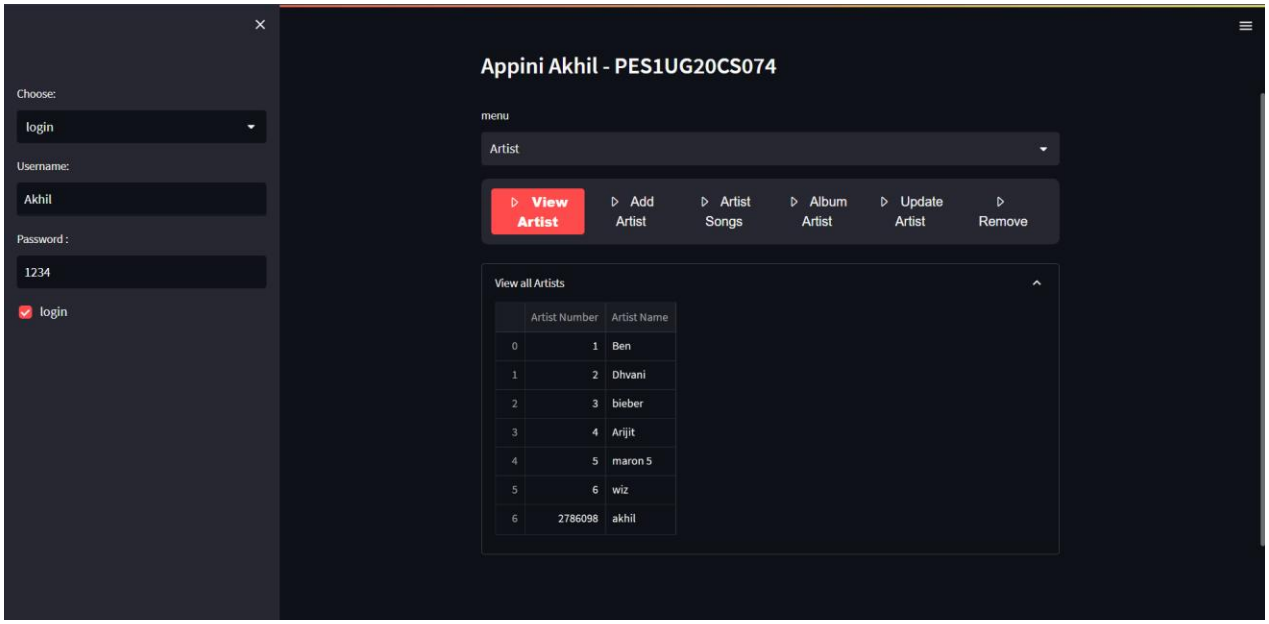
Music Database System

Appini Akhil - PES1UG20CS074

menu: Artist

Enter artist: akhil

Artist added successfully



Choose: login

Username: Akhil

Password : 1234

☑ login

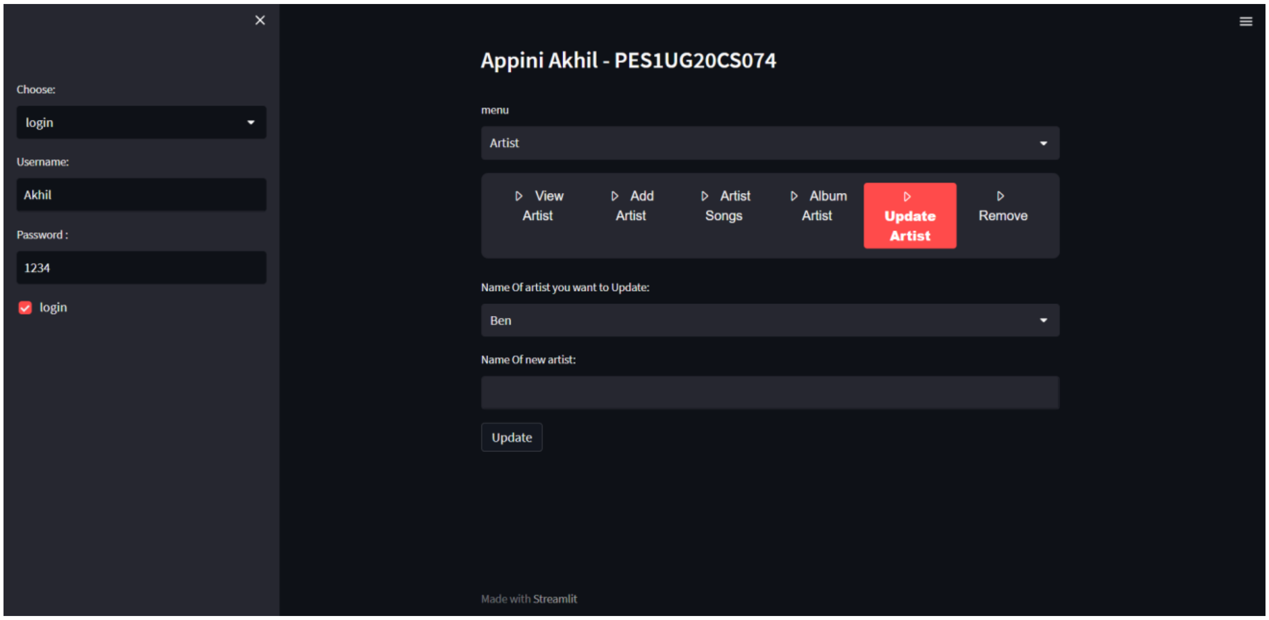
Appini Akhil - PES1UG20CS074

menu: Artist

View all Artists

	Artist Number	Artist Name
0	1	Ben
1	2	Dhvani
2	3	bieber
3	4	Arijit
4	5	maron 5
5	6	wiz
6	2786098	akhil

Update artist



Choose: login

Username: Akhil

Password : 1234

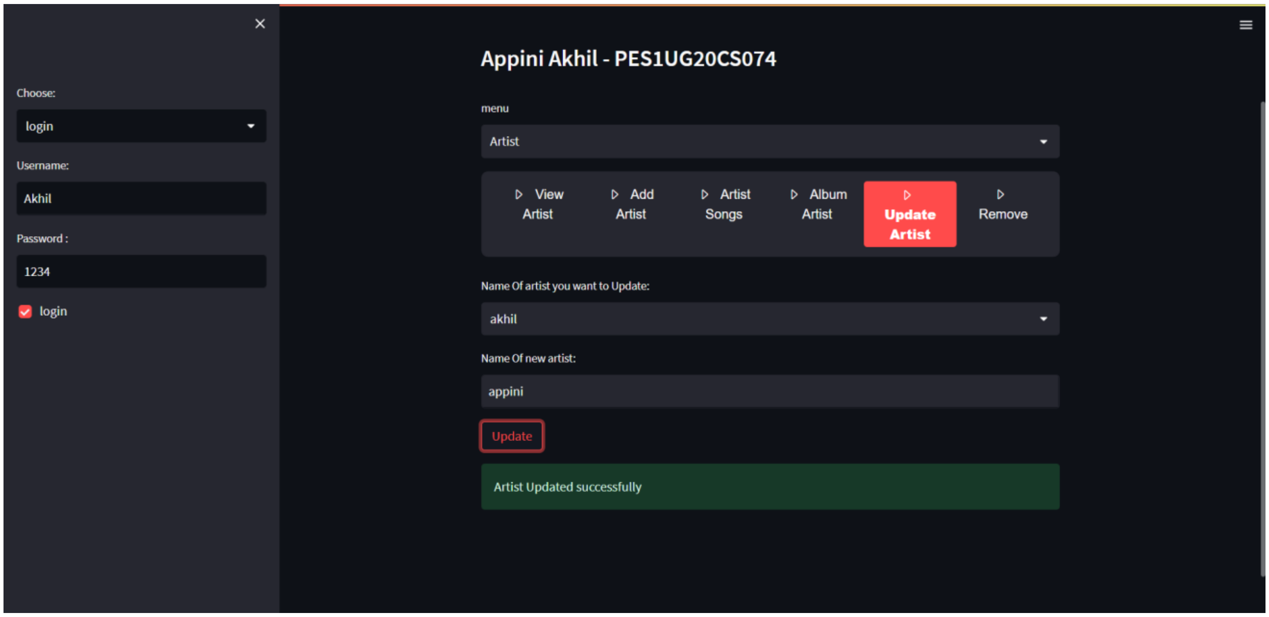
☑ login

Appini Akhil - PES1UG20CS074

menu: Artist

Name Of artist you want to Update: Ben

Name Of new artist:



Choose: login

Username: Akhil

Password : 1234

☑ login

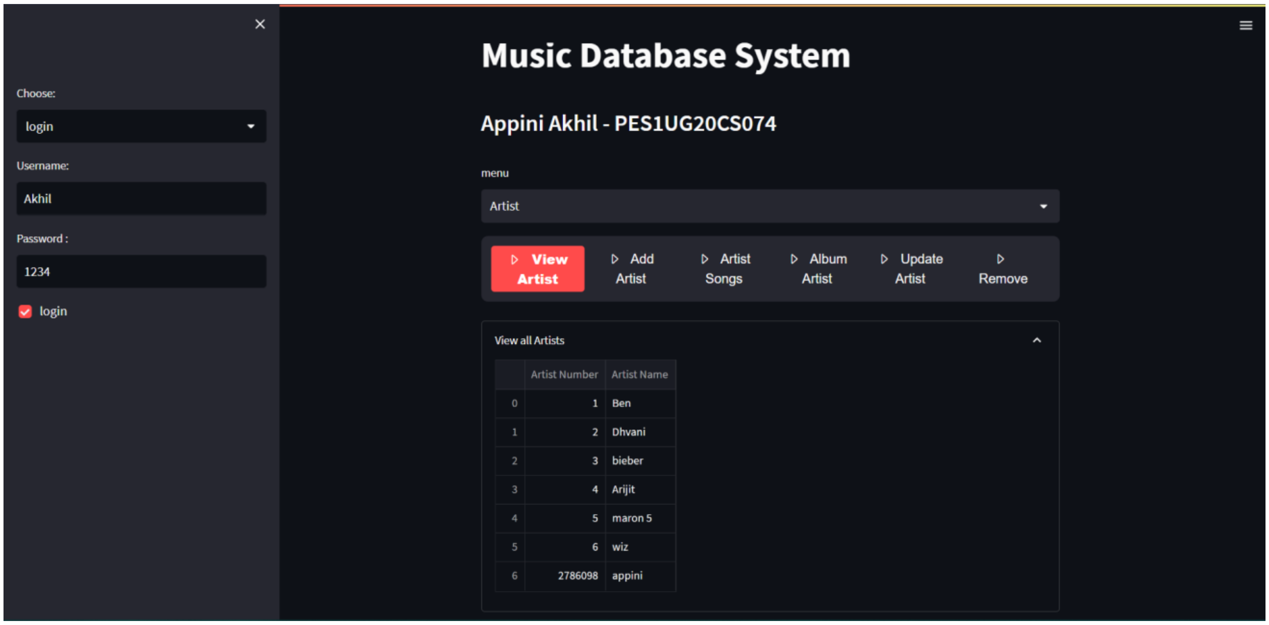
Appini Akhil - PES1UG20CS074

menu: Artist

Name Of artist you want to Update: akhil

Name Of new artist: appini

Artist Updated successfully



Choose: login

Username: Akhil

Password : 1234

☑ login

Music Database System

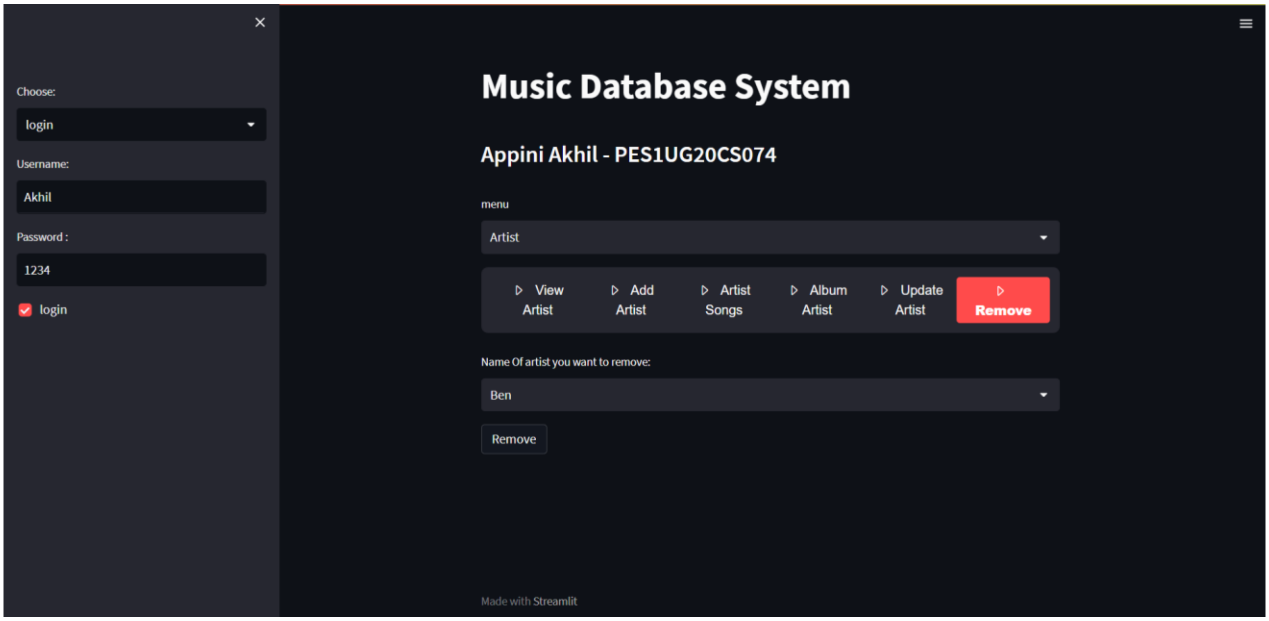
Appini Akhil - PES1UG20CS074

menu: Artist

View all Artists

	Artist Number	Artist Name
0	1	Ben
1	2	Dhvani
2	3	bieber
3	4	Arijit
4	5	maron 5
5	6	wiz
6	2786098	appini

Delete artist:



Choose: login

Username: Akhil

Password : 1234

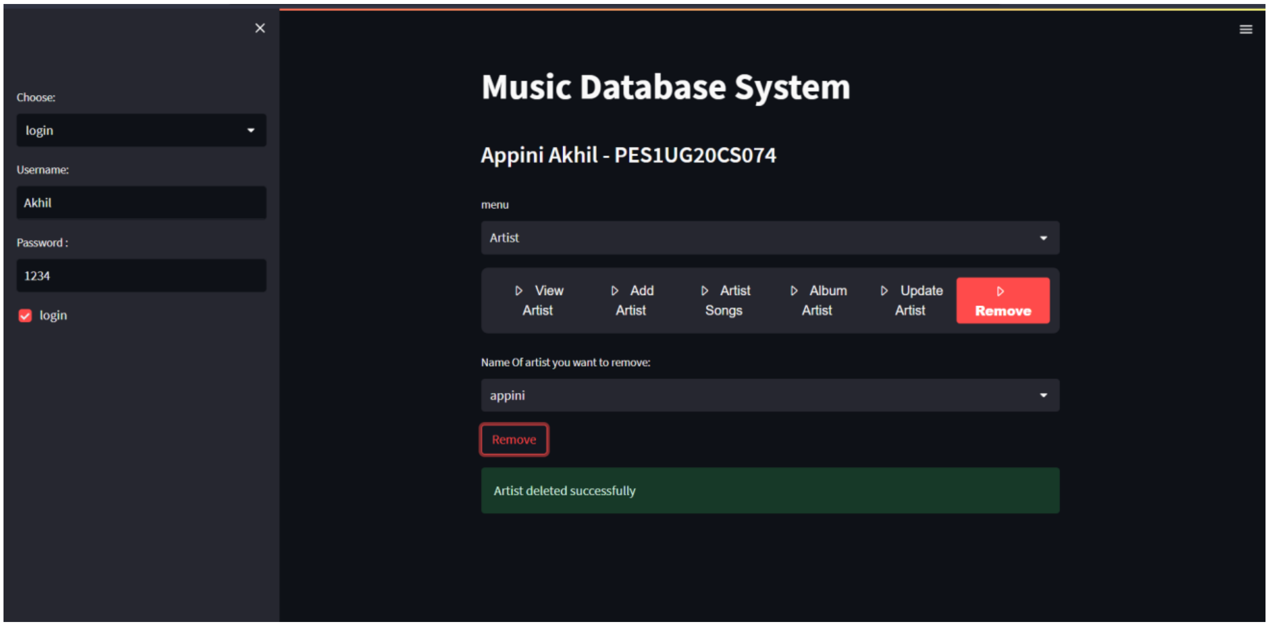
☑ login

Music Database System

Appini Akhil - PES1UG20CS074

menu: Artist

Name Of artist you want to remove: Ben



Choose: login

Username: Akhil

Password : 1234

☑ login

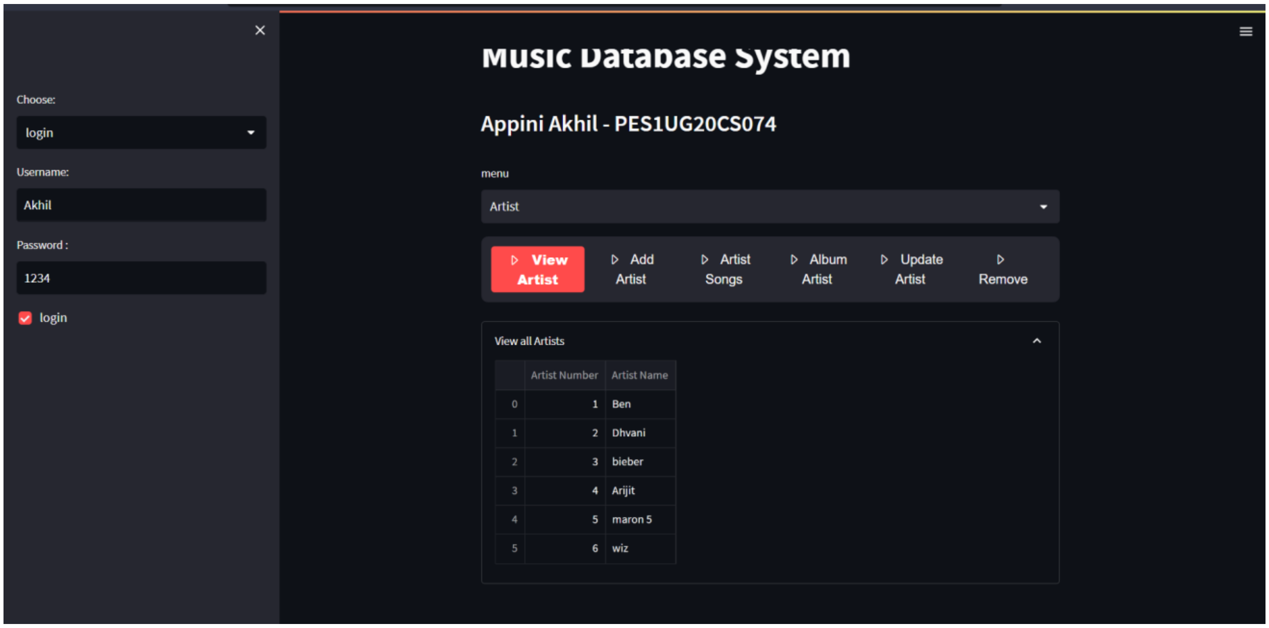
Music Database System

Appini Akhil - PES1UG20CS074

menu: Artist

Name Of artist you want to remove: appini

Artist deleted successfully



Choose: login

Username: Akhil

Password : 1234

☑ login

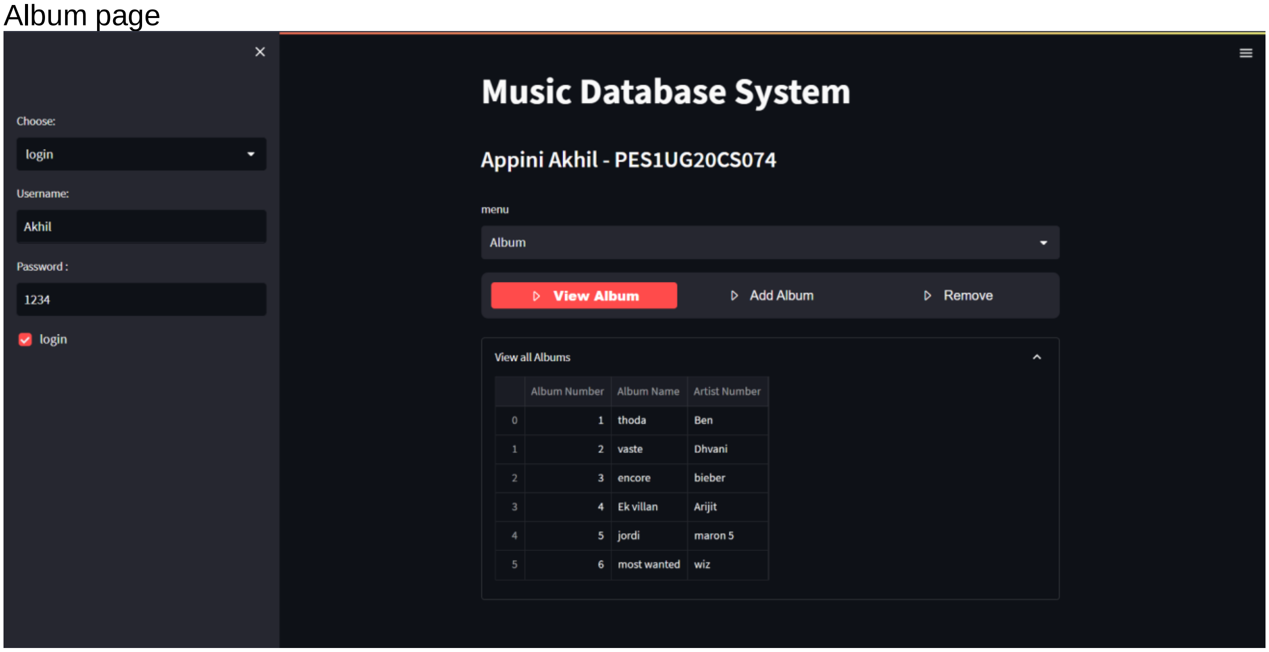
Music Database System

Appini Akhil - PES1UG20CS074

menu: Artist

	Artist Number	Artist Name
0	1	Ben
1	2	Dhvani
2	3	bieber
3	4	Arijit
4	5	maron 5
5	6	wiz

Album page



Choose: login

Username: Akhil

Password : 1234

☑ login

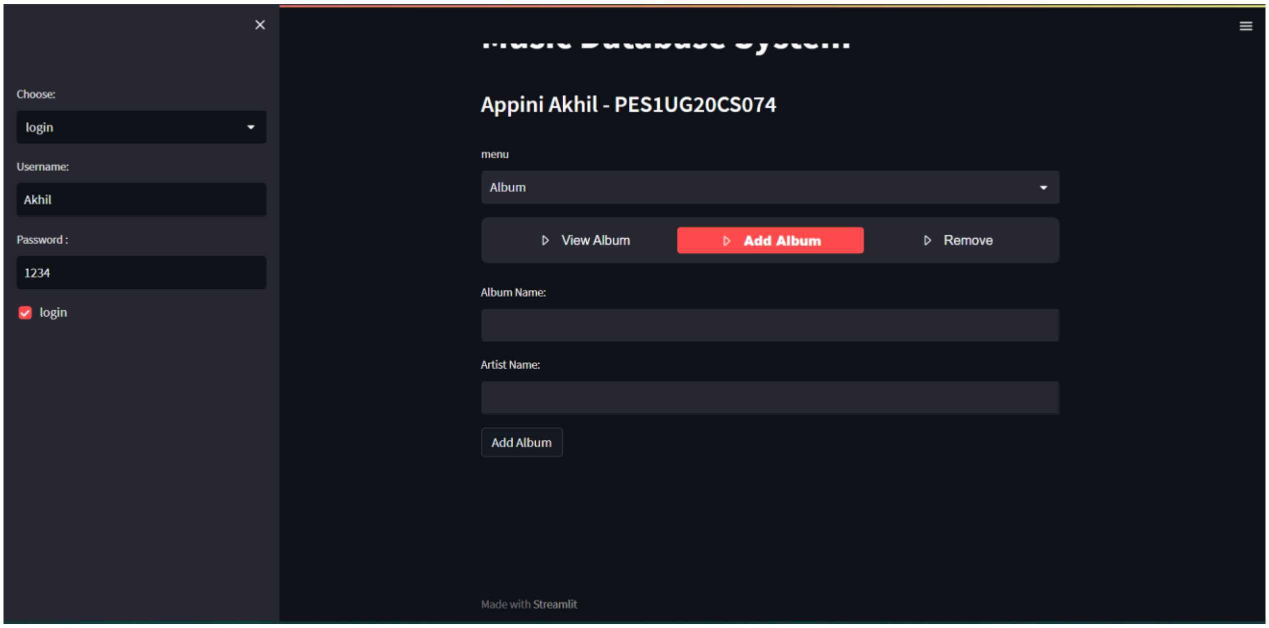
Music Database System

Appini Akhil - PES1UG20CS074

menu: Album

	Album Number	Album Name	Artist Number
0	1	thoda	Ben
1	2	vaste	Dhvani
2	3	encore	bieber
3	4	Ek villan	Arijit
4	5	jordi	maron 5
5	6	most wanted	wiz

Add album



Choose:
login
Username:
Akhil
Password :
1234
☑ login

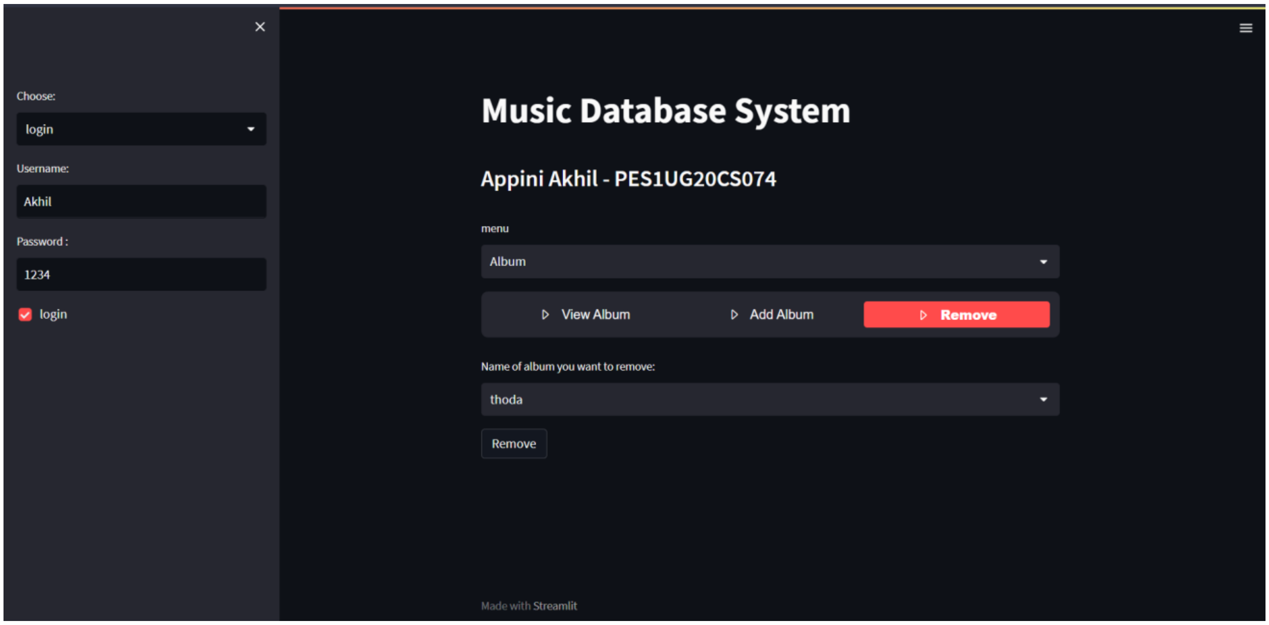
Appini Akhil - PES1UG20CS074

menu
Album

Album Name:

Artist Name:

Remove album:



Choose:
login
Username:
Akhil
Password :
1234
☑ login

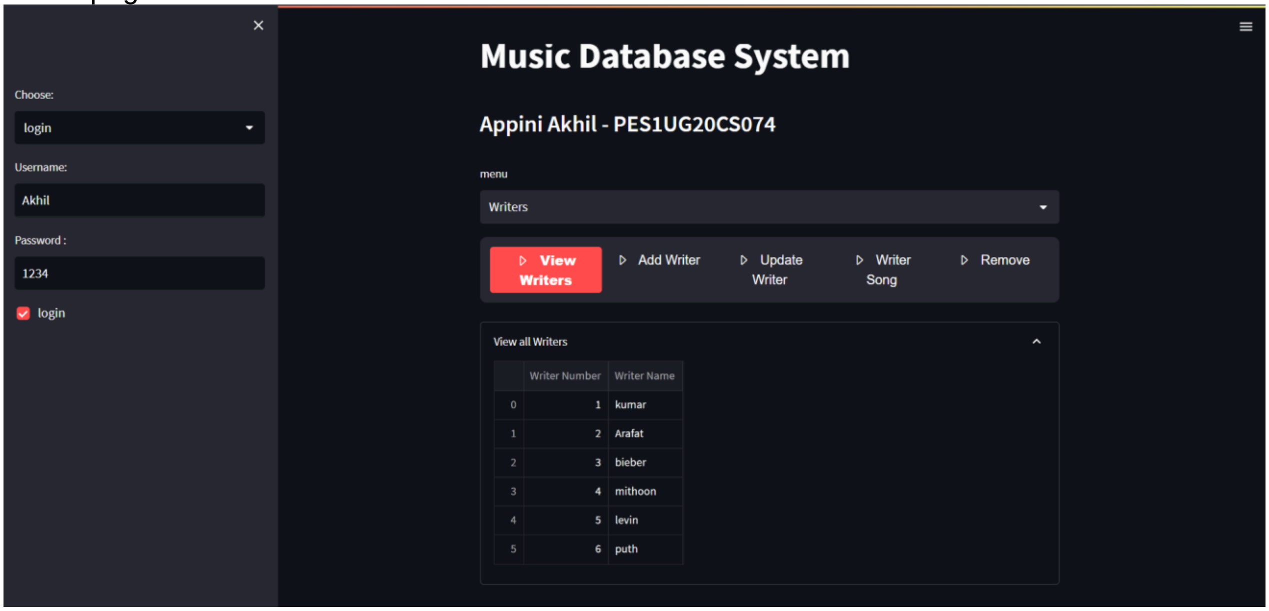
Music Database System

Appini Akhil - PES1UG20CS074

menu
Album

Name of album you want to remove:
thoda

Writer page:



Music Database System

Appini Akhil - PES1UG20CS074

Choose: login

Username: Akhil

Password : 1234

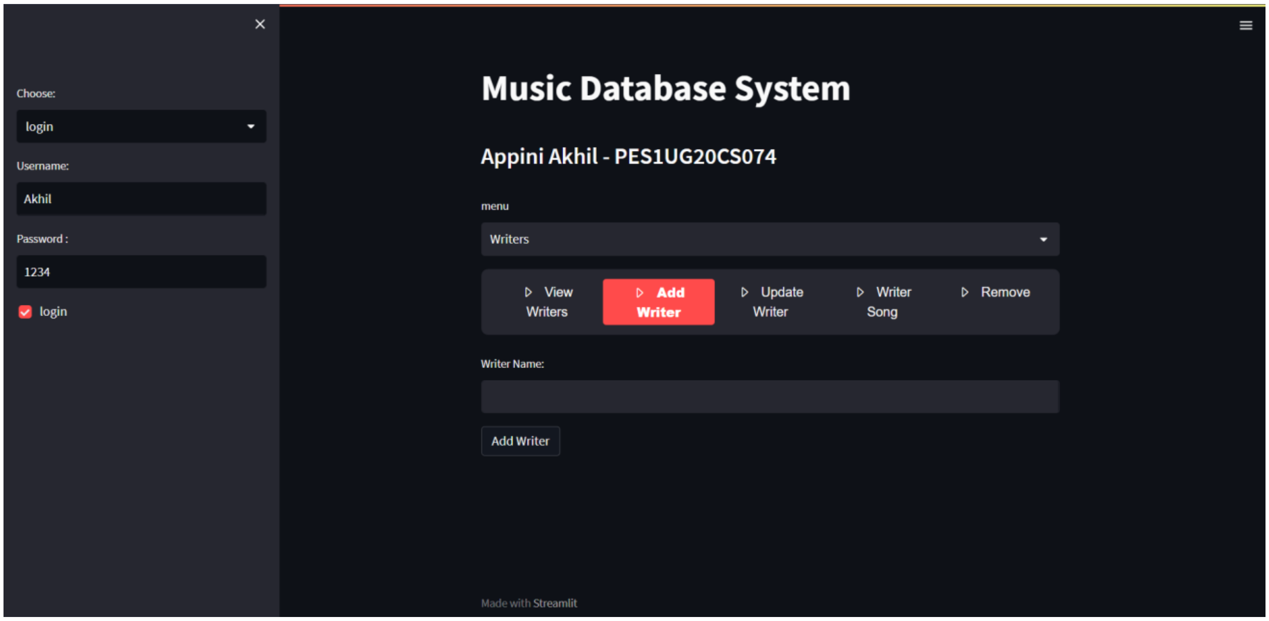
☑ login

menu: Writers

View all Writers

	Writer Number	Writer Name
0	1	kumar
1	2	Arafat
2	3	bieber
3	4	mithoon
4	5	levin
5	6	puth

Add writer



Music Database System

Appini Akhil - PES1UG20CS074

Choose: login

Username: Akhil

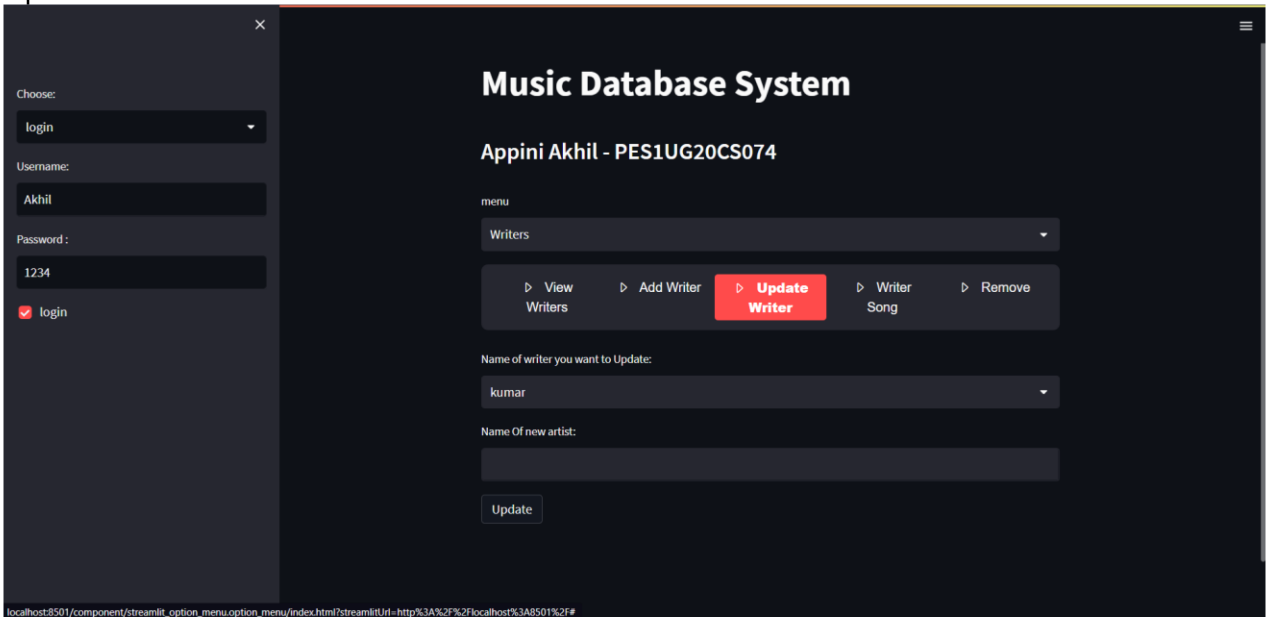
Password : 1234

☑ login

menu: Writers

Writer Name:

Update writer



Music Database System

Appini Akhil - PES1UG20CS074

Choose: login

Username: Akhil

Password : 1234

☑ login

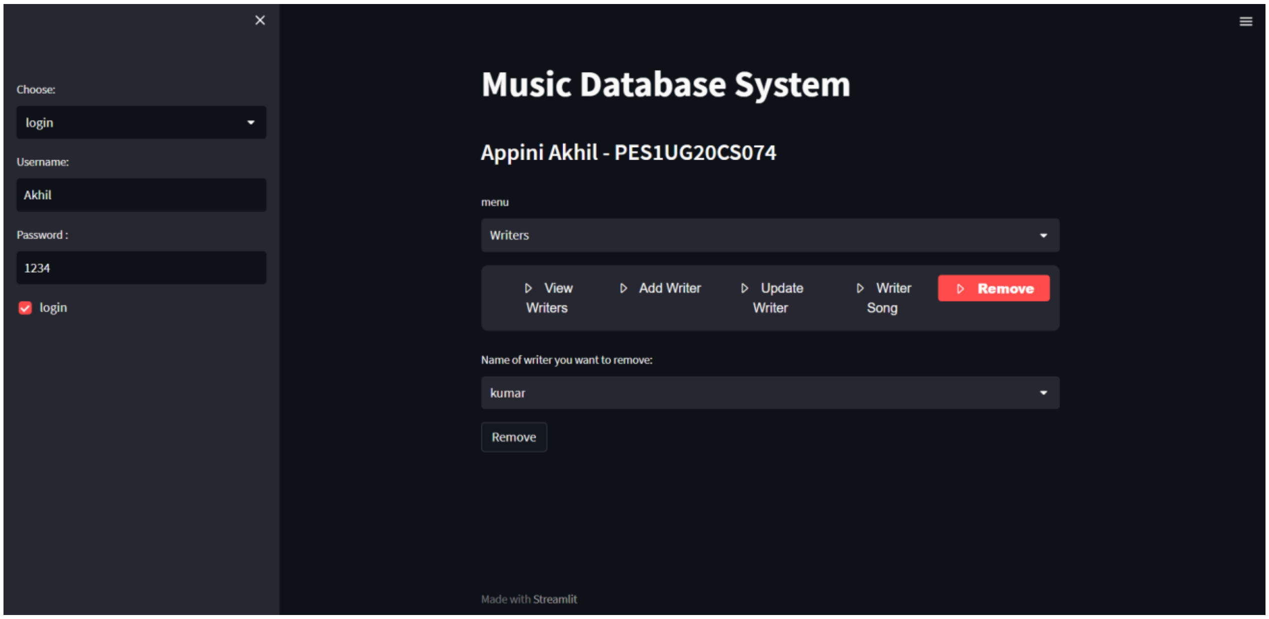
menu: Writers

Name of writer you want to Update: kumar

Name Of new artist:

localhost:8501/component/streamlit_option_menu.option_menu/index.html?streamlitUrl=http%3A%2F%2Flocalhost%3A8501%2F#

Remove writer



Music Database System

Appini Akhil - PES1UG20CS074

Choose: login

Username: Akhil

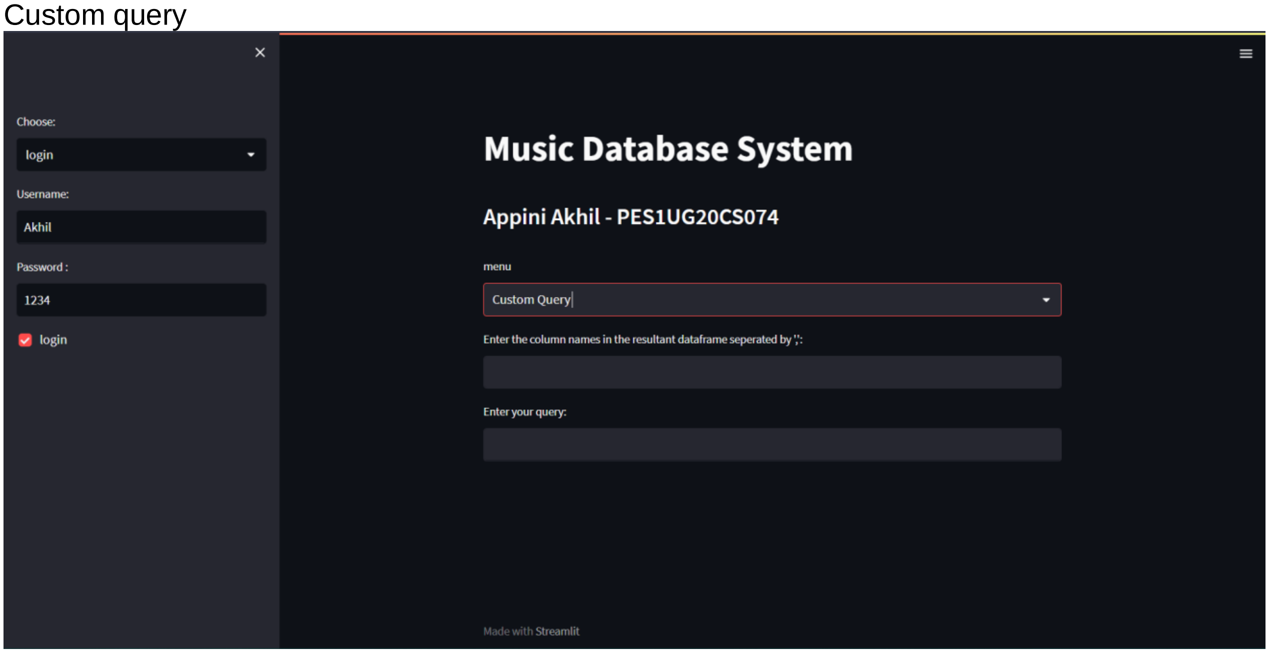
Password : 1234

☑ login

menu: Writers

Name of writer you want to remove: kumar

Custom query



Choose:

login

Username:

Akhil

Password :

1234

☑ login

Music Database System

Appini Akhil - PES1UG20CS074

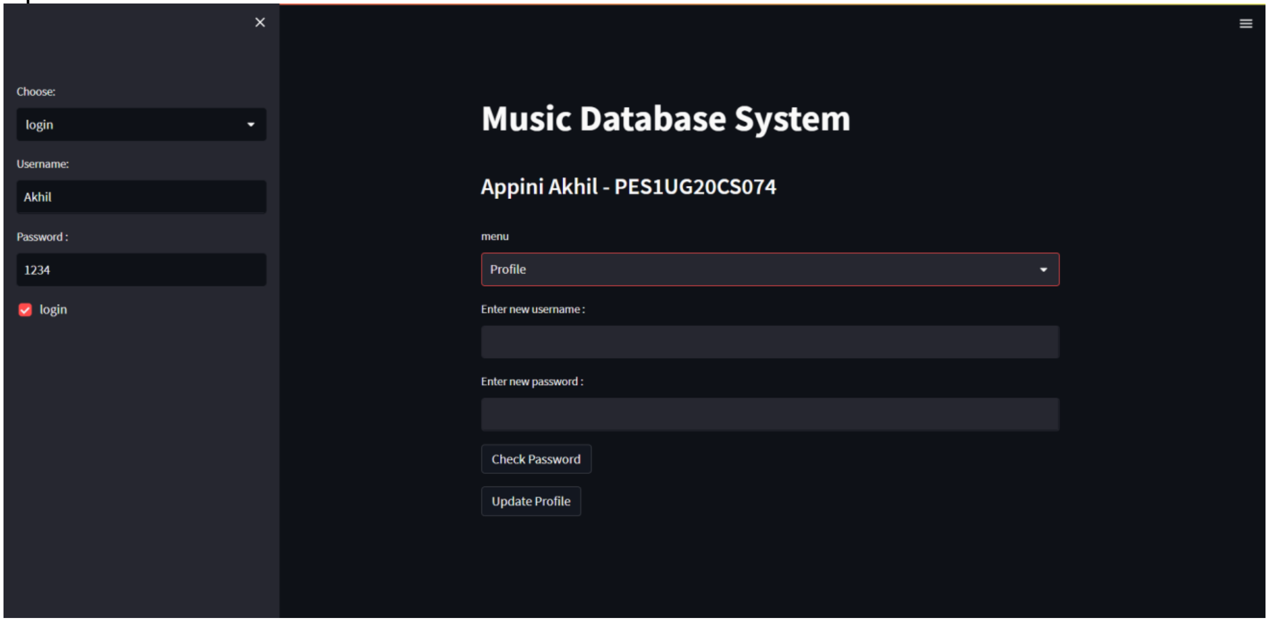
menu

Custom Query

Enter the column names in the resultant dataframe seperated by ',':

Enter your query:

Update Profile



Choose:

login

Username:

Akhil

Password :

1234

☑ login

Music Database System

Appini Akhil - PES1UG20CS074

menu

Profile

Enter new username :

Enter new password :