

MyAnimeLite

ANIME DATABASE

ALVARO, Justine Paul | GACUTAN, Alyanna | SANTOS, Kenneth
CS 165 WWW

A. Project Description

There are plenty of anime fans all over the world. As fans, we want to know more details about the anime we watch at the same time find anime we might be interested in. The group chose to develop an anime database similar to MyAnimeList. The idea came from Justine Alvaro and the project is entitled MyAnimeLite. The database contains anime records that are famous among millennials and Generation Z. The project however, does not include details such as voice actors, lists of episode summaries, and character descriptions.

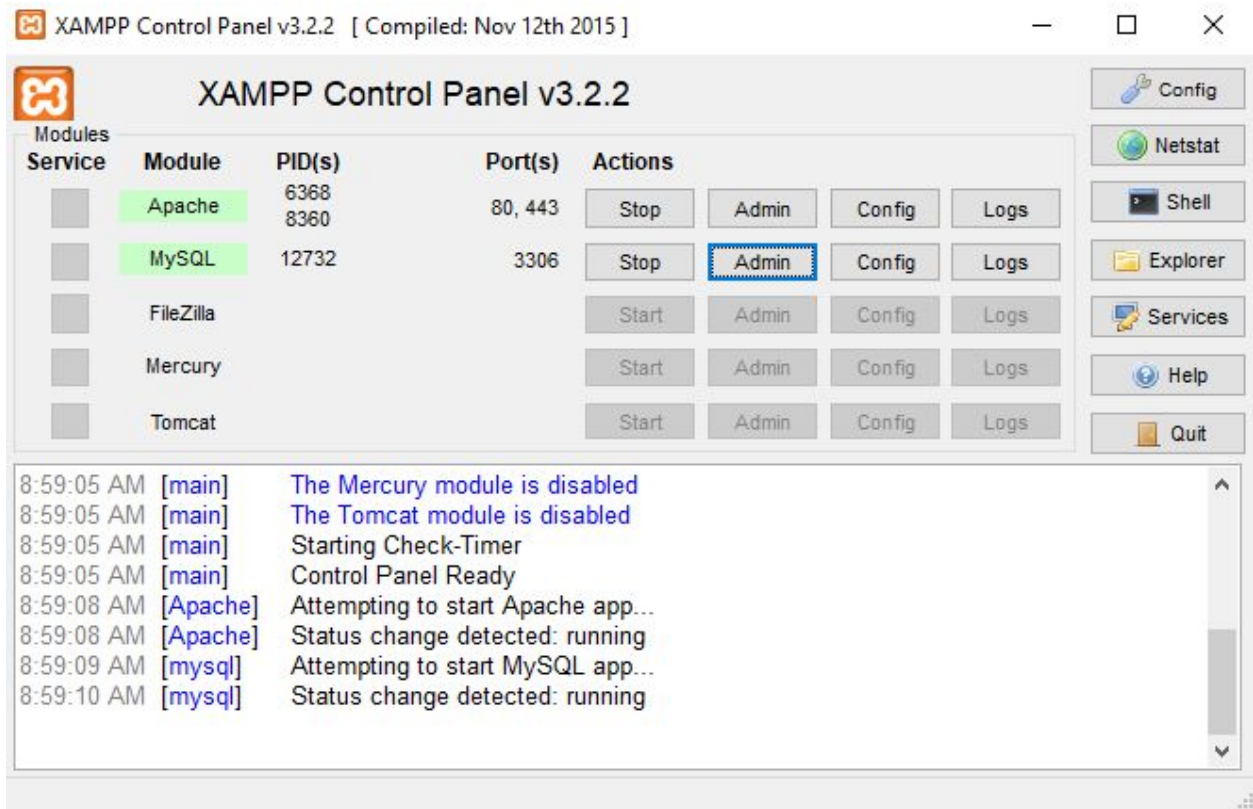
B. Technical Specifications

- **DBMS:** MySQL
 - **Technologies Used:** PHP v7.2.11, HTML 5, CSS
- The group does not plan to deploy the project after development.*

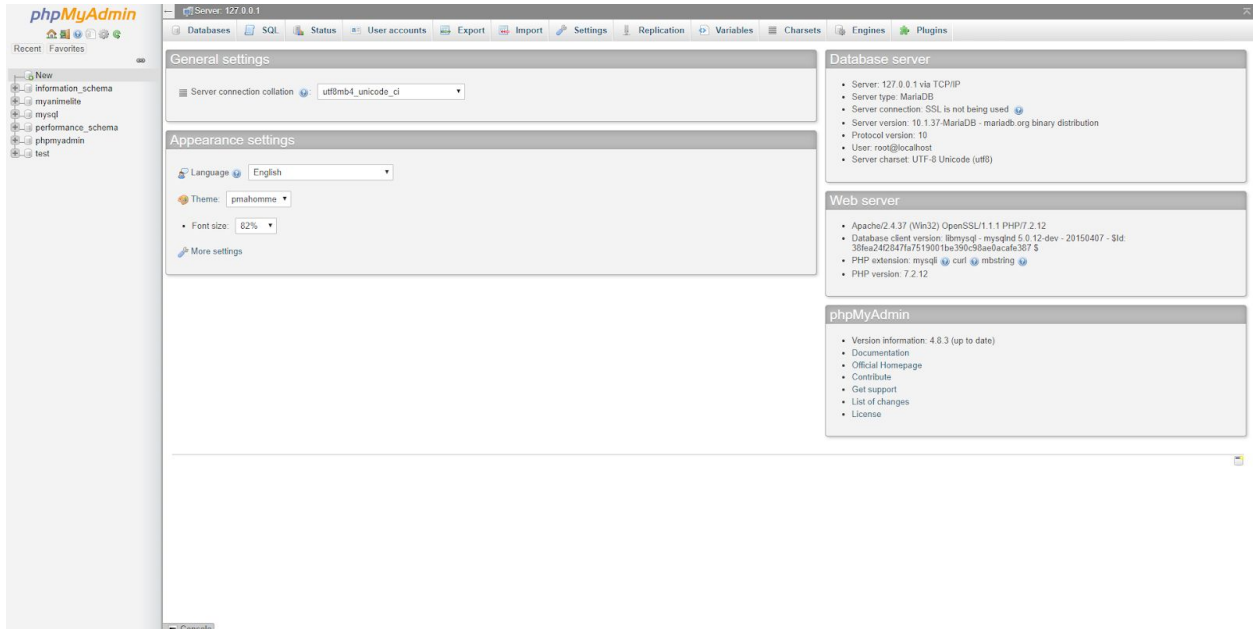
C. Running MyAnimeLite

1. Download and install XAMPP (select Apache, PHP, and MySQL).
2. Download/clone the repository:
<https://github.com/justinealvaro15/my-anime-lite.git>.
3. Move your local file of the repository to C:\xampp\htdocs.

4. Turn on XAMPP's Apache and MySQL.



5. Click MySQL's Admin button and it will redirect you to <http://localhost/phpmyadmin/>.



6. Create a new database named "myanimelite", and select utf8_general_ci as its collation property.

Databases

Create database ⓘ

myanimelite utf8_general_ci Create

Database	Collation	Action
<input type="checkbox"/> information_schema	utf8_general_ci	Check privileges
<input type="checkbox"/> myanimelite	utf8_general_ci	Check privileges
<input type="checkbox"/> mysql	latin1_swedish_ci	Check privileges
<input type="checkbox"/> performance_schema	utf8_general_ci	Check privileges
<input type="checkbox"/> phpmyadmin	utf8_bin	Check privileges
<input type="checkbox"/> test	latin1_swedish_ci	Check privileges
Total: 6	latin1_swedish_ci	

⬆ ☐ Check all With selected: Drop

7. Select the myanimelite database and import the SQL file located in myanimelite/SQL/my-anime-lite.sql.

Importing into the database "myanimelite"

File to import:

File may be compressed (gzip, bzip2, zip) or uncompressed.
A compressed file's name must end in `.[format].[compression]`. Example: `.sql.zip`

Browse your computer: my-anime-lite.sql (Max: 2,048KiB)

You may also drag and drop a file on any page.

Character set of the file:

8. Finally, go to <http://localhost/my-anime-lite/> and the app should redirect you to a login page.

Sample username-password pairs:

1. databaseislife1
cs165WVW
2. jmendoza1
cs165WVW

Sample screens:

MyAnimeList

Log In

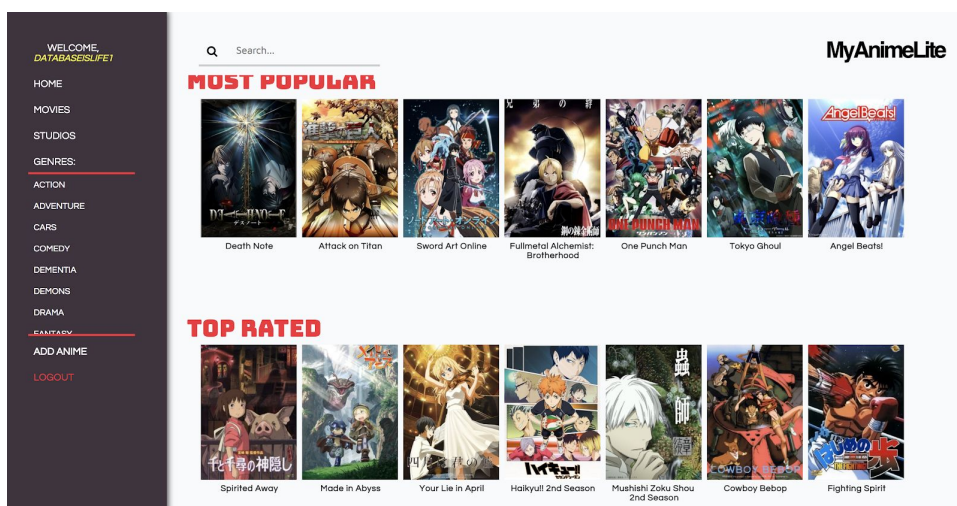
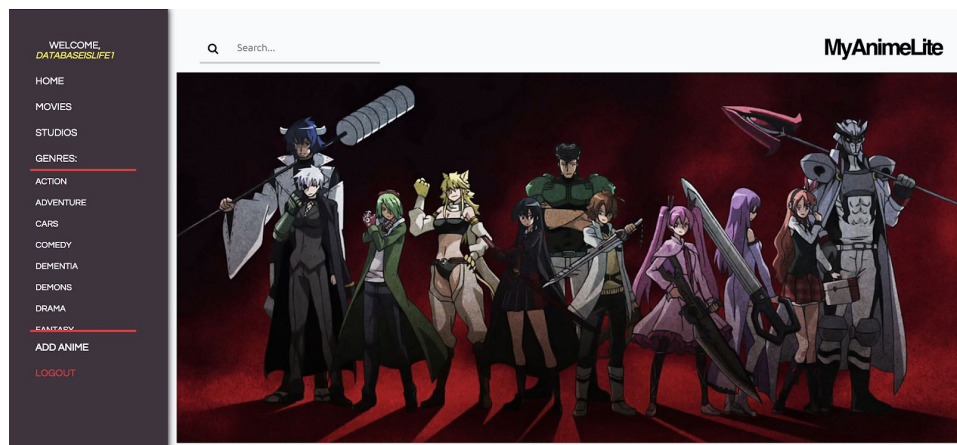
Username

Password

Login

Create an account! [Sign up](#)

LOG-IN PAGE



HOME PAGE

Demonstration video can be found in this link: <https://drive.google.com/file/d/1Jy0oiqCqM5Q2VYsLbM4v1rQXIOuR37tx/view?usp=sharing>



Kimi no Na wa. 2016

9.19
SCORE

#2
RANKED

#33
POPULARITY

English Title: **Your Name.**

Type: **Movie**

Source: **Original**

Episode Count: **1**

Duration: **106 min.**

Aired: **Summer 2016**

Studio: **CoMix Wave Films**

Licensor: **Funimation**

Genre: **Drama | Romance | School | Supernatural |**

[Update](#) [Delete](#)

ANIME PAGE

Edit details about this anime

Japanese Title:

Kimi no Na wa.

English Title:

Your Name.

Score:

9.19

Overall Rank:

2

Popularity Rank:

33

Type:

Movie

EDIT ANIME DETAILS

You are about to delete
Gintama.

[Delete](#)

[Cancel](#)

DELETE ANIME

WELCOME
DATABASESLIFE!

HOME

MOVIES

STUDIOS

GENRES:

ACTION

ADVENTURE

CARS

COMEDY

DEMENTIA

DEMONS

DRAMA

FANTASY

ADD ANIME

LOGOUT

Search...

MyAnimeLite

naruto

Boruto: Naruto the Movie (Boruto: Naruto the Movie)

Chunin Exam on Fire! and Naruto vs. Konohamaru! (Naruto: Honoo no Chunin Shiken! Naruto vs. Konohamaru!)

Hidden Leaf Village Grand Sports Festival (Naruto: Dai Katsugeki!! Yuki Hime Shinabu Houjou Dattebayo! Special: Konoha Annual Sports Festival)

Naruto (Naruto)

NARUTO Spin-Off: Rock Lee 6 His Ninja Pals (Naruto SD: Rock Lee no Seishun Full-Power Ninden)

Naruto the Movie 2: Legend of the Stone of Gelel (Naruto Movie 2: Dai Gekitotsu! Maboroshi no Chiteiseki Dattebayo!)

Naruto the Movie 3: Guardians of the Crescent Moon Kingdom (Naruto Movie 3: Dai Kofun! Mikazuki Jima no Animaru Panikku Dattebayo!)

Naruto the Movie: Ninja Clash in the Land of Snow (Naruto Movie 1: Dai Katsugeki!! Yuki Hime Shinabu Houjou Dattebayo!)

Naruto: Shippuden (Naruto: Shippuden)

Naruto: Shippuden the Movie (Naruto: Shippuden Movie 1)

Naruto: Shippuden the Movie 2 - Bonds - (Naruto: Shippuden Movie 2 - Kizuna)

Naruto: The Last Story - Mission: Protect the Waterfall Village (Naruto: Takigakure no Shitau - Ore ga Eiyuu Dattebayo!)

Road to Ninja: Naruto the Movie (Naruto: Shippuden Movie 6 - Road to Ninja)

The Last: Naruto the Movie (The Last: Naruto the Movie)

SEARCH RESULTS

WELCOME
DATABASESLIFE!

HOME

MOVIES

STUDIOS

GENRES:

ACTION

ADVENTURE

CARS

COMEDY

DEMENTIA

DEMONS

DRAMA

FANTASY

ADD ANIME

LOGOUT

Search...

MyAnimeLite

Movies

Cowboy Bebop: The Movie

Neon Genesis Evangelion: The End of Evangelion

Ghost in the Shell

Samurai X: The Motion Picture

AKIRA

Appleseed (Movie)

Mobile Suit Gundam: The 08th MS Team - Killer's Report

Mobile Suit Gundam: Char's Counterattack

X - The Movie

Princess Mononoke

Initial D Third Stage

Spirited Away

MOVIES

WELCOME
DATABASESLIFE!

HOME

MOVIES

STUDIOS

GENRES:

ACTION

ADVENTURE

CARS

COMEDY

DEMENTIA

DEMONS

DRAMA

FANTASY

ADD ANIME

LOGOUT

Search...

MyAnimeLite

Adventure

Update Delete

Exploring new places, environments or situations. This is often associated with people on long journeys to places far away encountering amazing things, usually not in an epic but in a rather gripping and interesting way.

Fullmetal Alchemist: Brotherhood

Hunter x Hunter

Spirited Away

Made in Abyss

Cowboy Bebop

Princess Mononoke

Fullmetal Alchemist: Brotherhood

Hunter x Hunter

Spirited Away

Made in Abyss

Cowboy Bebop

Princess Mononoke

GENRE

Edit genre description of Drama

Description:

Anime that often show life or characters through conflict and emotions. In general, the different parts of the story tend to form a whole that is greater than the sum of the parts. In other words, the story has a message that is bigger than just the story line itself.

[Cancel](#)

UPDATE GENRE

You are about to delete Drama.

[Cancel](#)

DELETE GENRE

WELCOME DATABASESLIFE!

HOME

MOVIES

STUDIOS

GENRES:

ACTION

ADVENTURE

CARS

COMEDY

DEMENTIA

DEMONS

DRAMA

SANTARY

ADD ANIME

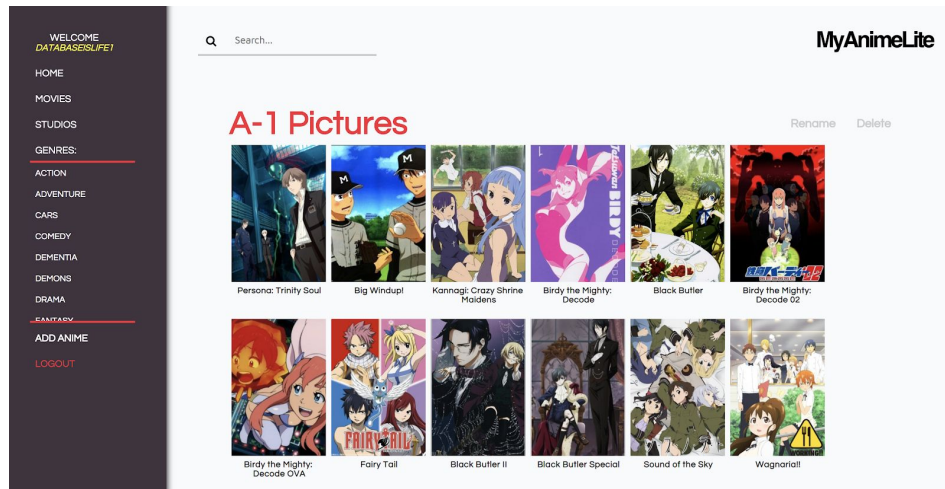
LOGOUT

Search...

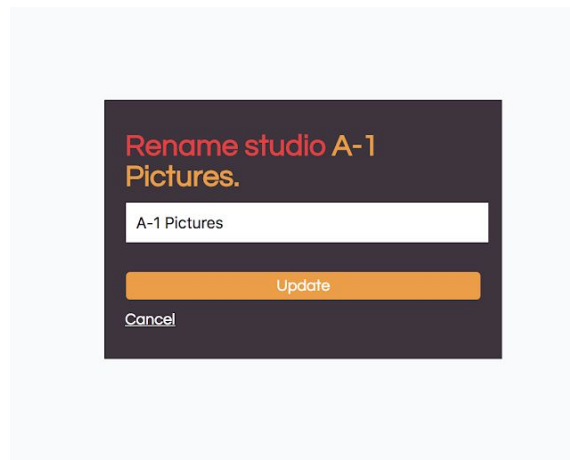
- 3xCube
- 8bit
- A-1 Pictures
- A-Real
- A.C.G.T.
- Actas
- Agent 21
- AIC
- AIC A.S.T.A.
- AIC Build
- AIC Classic
- AIC Plus+
- AIC Spirits
- Ajia-Do
- Animate Film
- Animation Do

MyAnimeLite

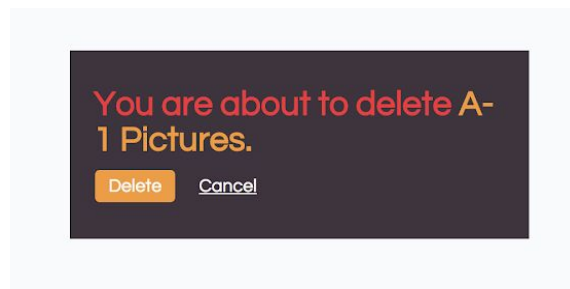
LIST OF ALL STUDIOS



STUDIOS

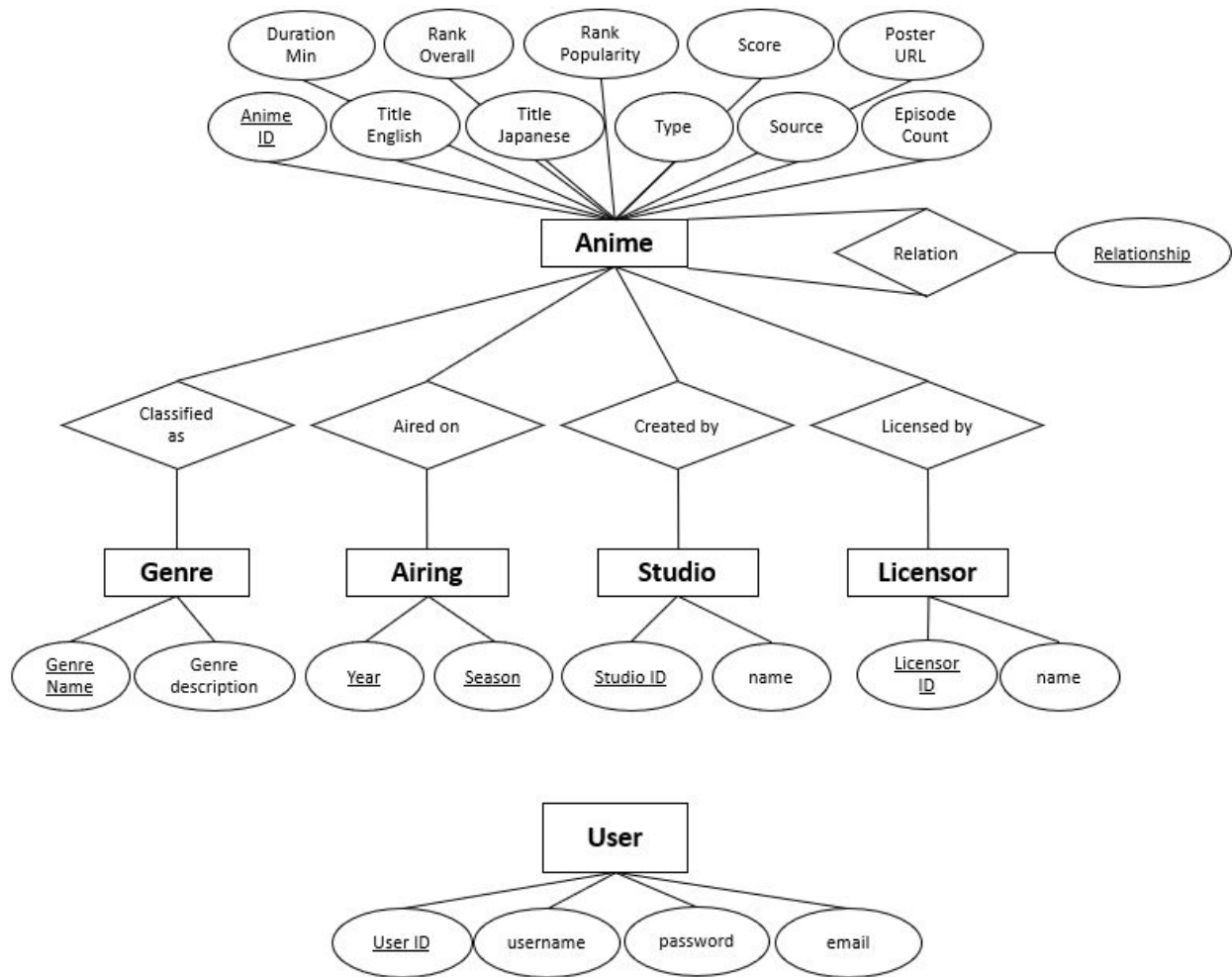


RENAME STUDIO



DELETE STUDIO

D. ER Diagram + Schema



Schema

- Anime(anime_id INTEGER,
title_jap TEXT,
title_eng TEXT,
type TEXT,
source TEXT,
ep_count INT,
duration_min INT,
rank_overall INT,
rank_popularity INT,
score REAL,
poster_link TEXT)
- Genre(genre_name TEXT,
genre_desc TEXT)

- Studio(studio_id INTEGER,
studio_name TEXT)
- Licensor(lic_id INTEGER,
lic_name TEXT)
- Airing(year INT,
season TEXT)
- Users(id INT,
username TEXT,
email TEXT,
password TEXT)

Relationships

- Relationship(anime1_id INT,
anime2_id INT,
relation INT)
- Classification(anime_id INT,
genre TEXT)
- Created(studio_id INT,
anime_id INT)
- Licensed(lic_id INT,
anime_ID INT)
- Aired(year INT,
season TEXT,
anime_ID INT)

E. Functionalities

CREATE = INSERT

<i>Description of Functionality</i>	<i>MySQL Statement</i>
Add a new anime in the database.	INSERT INTO anime (anime_id, title_jap, title_eng, score, rank_overall, rank_popularity, `type`, source, ep_count, duration_min, poster_link) VALUES ('\$new_anime_id', '\$new_title_jap', '\$new_title_eng', '\$new_score', '\$new_rank_overall', '\$new_rank_popularity', '\$new_type', '\$new_source', '\$new_ep_count', '\$new_duration_min', '\$new_poster_link')" INSERT INTO `licensed` (lic_id, anime_id) VALUES ('\$new_lic_id', '\$new_anime_id') INSERT INTO `created` (studio_id, anime_id) VALUES ('\$new_studio_id', '\$new_anime_id') INSERT INTO `aired` (year, season, anime_id) VALUES ('\$new_year', '\$new_season', '\$new_anime_id') (loop) INSERT INTO `classification` (anime_id, genre) VALUES ('\$new_anime_id', '\$genre')
Add a new studio in the database. (done if studio does not exist when creating a new anime)	INSERT INTO `studio` (studio_id, studio_name) VALUES ('\$new_studio_id', '\$new_studio_name')
Add a new licensor in the database. (done if licensor does not exist when creating a new anime)	INSERT INTO `licensor` (lic_id, lic_name) VALUES ('\$new_lic_id', '\$new_lic_name')
Register new user	INSERT INTO users (username, email, password) VALUES ('\$username', '\$email', '\$password')

READ = SELECT

<i>Description of Functionality</i>	<i>MySQL Statement</i>
View all anime with the specified genre.	(Display array of anime) SELECT Anime.poster_link, Anime.title_eng, Anime.title_jap, Anime.anime_id AS anime_id FROM Anime, Classification WHERE genre="\$_SESSION[link]". AND Classification.anime_id= Anime.anime_id GROUP BY rank_overall (Display genre name and description) SELECT genre, genre_desc FROM Genre WHERE genre_name="\$_SESSION[link]".
View top 20 anime with from a specified year and season, ranked by popularity (displayed in Home page)	SELECT poster_link, title_eng, title_jap FROM Anime, Aired WHERE year=2018 AND season='Winter' AND Aired.anime_id = Anime.anime_id ORDER BY Anime.rank_popularity ASC LIMIT 20
View top 20 anime ranked by overall rankings (displayed in Home page)	SELECT poster_link, title_eng, title_jap, anime_id FROM Anime ORDER BY score DESC LIMIT 20
View top 20 anime ranked by popularity rankings (displayed in Home page)	SELECT poster_link, title_eng, title_jap, anime_id FROM Anime ORDER BY rank_popularity ASC LIMIT 20
View an anime (displayed in Anime page)	(display anime details) SELECT * FROM Anime t1 INNER JOIN Aired t2 ON t1.anime_id = t2.anime_id INNER JOIN Created t3 ON t1.anime_id = t3.anime_id INNER JOIN Licensed t4 ON t1.anime_id = t4.anime_id INNER JOIN Studio r1 ON t3.studio_id = r1.studio_id INNER JOIN Licensor r2 ON t4.lic_id = r2.lic_id WHERE t1.anime_id = {\$_SESSION[link]} (display genre)

	SELECT genre FROM classification WHERE anime_id = {\$_SESSION['link']}
User login	SELECT * FROM users WHERE username='\$username' AND password='\$password'
Display movies (displayed in Movies page)	SELECT poster_link, title_eng, title_jap, anime_id FROM Anime WHERE type='Movie'
Search anime given a substring of its name	SELECT title_eng, title_jap, poster_link, anime_id FROM Anime WHERE title_eng LIKE '%'. \$_SESSION['searchkey']. '%' OR title_jap LIKE '%'. \$_SESSION['searchkey']. '%' ORDER BY title_eng ASC
Display list of genres in sidenav	SELECT genre_name FROM Genre
Display list of studios in StudiosPage	SELECT studio_name FROM Studio
View all anime with the specified studio.	SELECT * FROM Anime INNER JOIN Created ON Anime.anime_id = Created.anime_id INNER JOIN Studio ON Created.studio_id = Studio.studio_id WHERE Studio.studio_name = ". \$_SESSION['link']. "

UPDATE

<i>Description of Functionality</i>	<i>MySQL Statement</i>
Update anime details	UPDATE Anime t1 INNER JOIN Aired t2 ON t1.anime_id = t2.anime_id INNER JOIN Created t3 ON t1.anime_id = t3.anime_id INNER JOIN Licensed t4 ON t1.anime_id = t4.anime_id INNER JOIN Studio r1 ON t3.studio_id = r1.studio_id INNER JOIN Licensor r2 ON t4.lic_id = r2.lic_id SET title_jap = '\$new_title_jap', title_eng = '\$new_title_eng', score = '\$new_score', rank_overall = '\$new_rank_overall', rank_popularity = '\$new_rank_popularity', `type` = '\$new_type', source = '\$new_source', ep_count = '\$new_ep_count', duration_min = '\$new_duration_min', season = '\$new_season', year = '\$new_year', studio_name = '\$new_studio_name', lic_name = '\$new_lic_name', poster_link = '\$new_poster_link' WHERE t1.anime_id = '\$anime_id'
Update genre description	UPDATE Genre SET genre_desc = '\$new_genre_desc' WHERE genre_name = '\$genre_name'
Update studio name	UPDATE Studio r1 SET studio_name = '\$new_studio_name' WHERE r1.studio_id = '\$studio_id'

DELETE

<i>Description of Functionality</i>	<i>MySQL Statement</i>
Delete an anime entry.	DELETE FROM aired WHERE anime_id='\$anime_id' DELETE FROM created WHERE anime_id='\$anime_id' DELETE FROM licensed WHERE anime_id='\$anime_id' DELETE FROM classification WHERE anime_id='\$anime_id' DELETE FROM anime WHERE anime_id='\$anime_id'
Delete a genre entry.	DELETE FROM Studio WHERE studio_id='\$studio_id'
Delete a studio entry.	DELETE FROM Classification WHERE genre='\$genre_name' DELETE FROM Genre WHERE genre_name='\$genre_name'