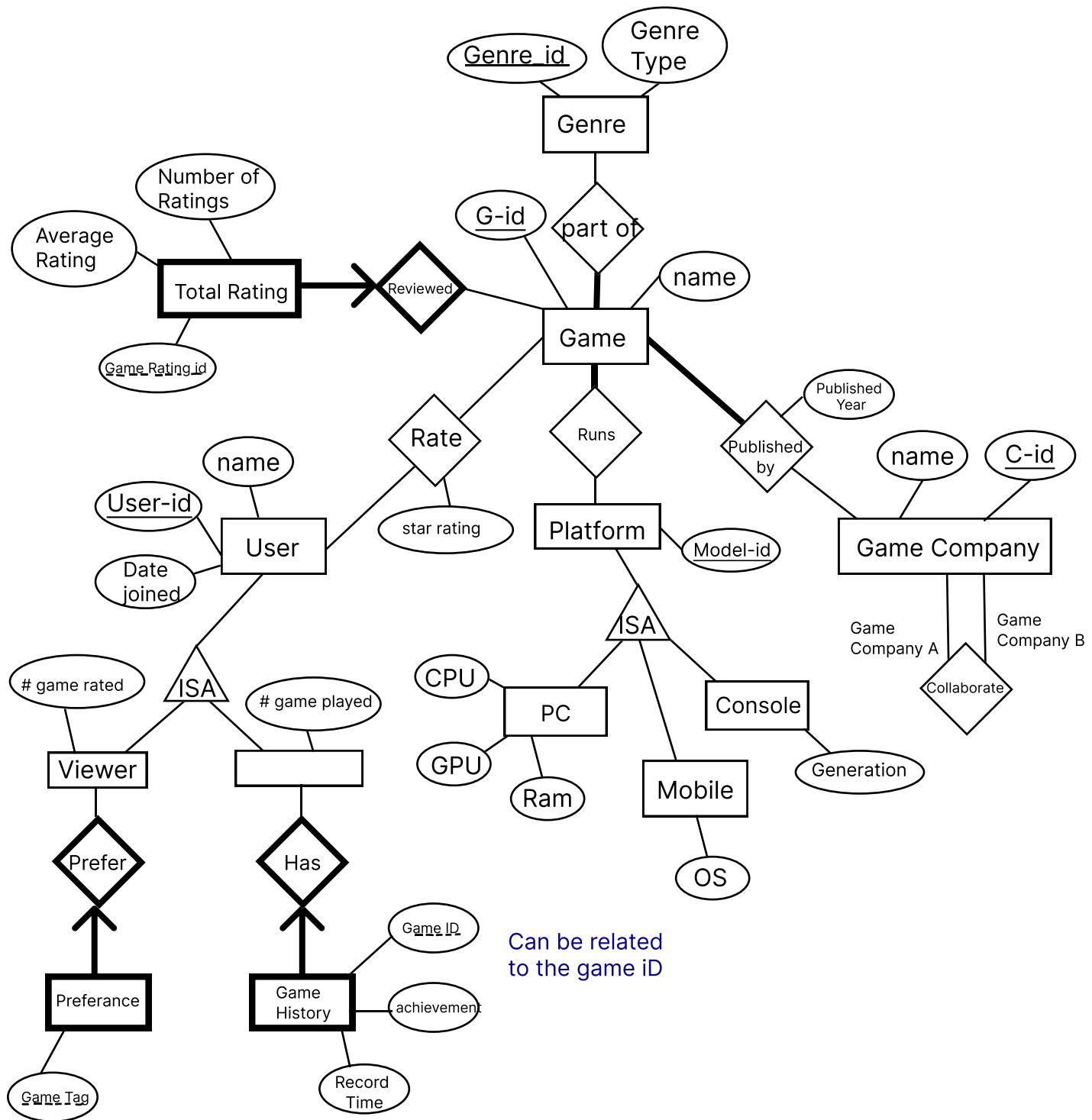


ERD OF Video Game Archive and Profile Database



The Relational Schema derived from the ER diagram

Game(<u>G-id</u> , name, Published_Year)	Runs(G-id REFERENCES Game, Model-id REFERENCES Platform)
Genre(<u>Genre_id</u> , Genre_Type)	Part_of(Genre-id REFERENCES Genre, G-id REFERENCES Game)
Platform(<u>Model-id</u>)	Collaborate(A_C-id REFERENCES Game_Company, B_C-id REFERENCES Game_Company)
PC(Model-id REFERENCE Platform, CPU, GPU, RAM)	Rate(G-id REFERENCES Game, User-id REFERENCES User, star-rating)
Console (Model-id REFERENCE Platform, Generation)	Published_by(G-id REFERENCES Game, C-id REFERENCES Game_Company)
Mobile (Model-id REFERENCE Platform, OS)	Reviewed_Total_Rating(G-id REFERENCES Game, Game_Rating_id, Average_Rating, Number_of_Ratings)
Game_Company(<u>C-id</u> , name)	Has_Game_History(User-id REFERENCE User, Game_ID, Achievement, Record Time)
User(<u>User-id</u> , <u>name</u> , Date_joined)	Prefer_Preference(User-id REFERENCE User, Game_Tag)
Viewer(User-id REFERENCE User, num_game_rated)	
Player(User-id REFERENCE User, num_game_played)	

Function Dependency

Game(G_id, Name, Published_year)	Runs(<u>G_id</u> REFERENCES Game, <u>Model_id</u> REFERENCES Platform)
$G_id \rightarrow Name$	Trivial Dependency
$G_id \rightarrow Published_year$	$G_id, Model_id \rightarrow G_id, Model_id$
Genre(Genre_id, Genre_type)	Part_of(<u>Genre_id</u> REFERENCES Genre, <u>G_id</u> REFERENCES Game)
$Genre_id \rightarrow Genre_type$	Trivial Dependency
Platform(Model_id)	$Genre_id, G_id \rightarrow Genre_id, G_id$
NO Dependency	
PC(<u>Model_id</u> REFERENCE Platform, CPU, GPU, RAM)	Collaborate(<u>A_C_id</u> REFERENCES Game_Company, <u>B_C_id</u> REFERENCES Game_Company)
$Model_id \rightarrow CPU, GPU, RAM$	Trivial Dependency
$A_C_id, B_C_id \rightarrow A_C_id, B_C_id$	
Console (<u>Model_id</u> REFERENCE Platform, Generation)	Rate(<u>G_id</u> REFERENCES Game, <u>User_id</u> REFERENCES User, User_rating)
$Model_id \rightarrow Generation$	$G_id, User_id \rightarrow User_rating$
Mobile (<u>Model_id</u> REFERENCE Platform, OS)	Published_by(<u>G_id</u> REFERENCES Game, <u>C_id</u> REFERENCES Game_Company)
$Model_id \rightarrow OS$	Trivial Dependency
$G_id, C_id \rightarrow G_id, C_id$	
Game_Company(C_id, Name)	Reviewed_Total_Rating(<u>G_id</u> REFERENCES Game, Game_Rating_id, Average_Rating, Star_rating)
$C_id \rightarrow Name$	$G_id, Game_rating_id \rightarrow Average_Rating, Star_rating$
$Average_rating \rightarrow Star_rating$	
User(User_id, Name, Year_joined, Account_age)	Has_Game_History(<u>User_id</u> REFERENCE User, Game_id, Achievement, Record_time, Badge)
$User_id \rightarrow Name, Year_joined, Account_age$	$User_id, Game_id \rightarrow Achievement, Record_time$
$Year_joined \rightarrow Account_age$	$User_id, Game_id \rightarrow Badge$
$Record_Time \rightarrow Achievement, Badge$	
Viewer(<u>User_id</u> REFERENCE User, Num_game_rated)	Achievement \rightarrow Badge
$User_id \rightarrow Num_game_rated$	
Player(<u>User_id</u> REFERENCE User, Num_game_played, Award)	Prefer_Preference(<u>User_id</u> REFERENCE User, Game_Tag_1, Game_Tag_2, Game_Tag_3)
$User_id \rightarrow Num_game_played, Award$	$User_id \rightarrow Game_Tag_1, Game_Tag_2, Game_Tag_3$
$Num_game_played \rightarrow Award$	

Candidate/Primary Keys

Game(<u>G_id</u> , Name, Published_year)	Runs(<u>G_id</u> REFERENCES Game, <u>Model_id</u> REFERENCES Platform) • G_id, Model_id
Genre(<u>Genre_id</u> , Genre_type)	Part_of(<u>Genre_id</u> REFERENCES Genre, <u>G_id</u> REFERENCES Game) • Genre_id, G_id
Platform(<u>Model_id</u>)	Collaborate(<u>A_C_id</u> REFERENCES Game_Company, <u>B_C_id</u> REFERENCES Game_Company) • A_C_id, B_C_id
PC(<u>Model_id</u> REFERENCE Platform, CPU, GPU, RAM)	Rate(<u>G_id</u> REFERENCES Game, <u>User_id</u> REFERENCES User, User_rating) • G_id, User_id
Console (<u>Model_id</u> REFERENCE Platform, Generation)	Published_by(<u>G_id</u> REFERENCES Game, <u>C_id</u> REFERENCES Game_Company) • G_id, C_id
Mobile (<u>Model_id</u> REFERENCE Platform, OS)	Reviewed_Total_Rating(<u>G_id</u> REFERENCES Game, Game_Rating_id, Average_Rating, Star_rating) • G_id, Game_rating_id
Game_Company(<u>C_id</u> , Name)	Has_Game_History(<u>User_id</u> REFERENCE User, Game_id, Achievement, Record_time, Badge) • User_id, Game_id
User(<u>User_id</u> , Name, Year_joined, Account_age)	Prefer_Preference(<u>User_id</u> REFERENCE User, Game_Tag_1, Game_Tag_2, Game_Tag_3) • User_id
Viewer(<u>User_id</u> REFERENCE User, Num_game_rated)	
Player(<u>User_id</u> REFERENCE User, Num_game_played, Award)	
	<h3>Additional Candidate Keys from the Normalization</h3> <p>User_age(<u>Year_joined</u>, Account_age): • Year_joined User(<u>User_id</u>, Name, <u>Year_joined</u> REFERENCE User_age) • User_id</p> <p>Game_played_award(<u>Num_game_played</u> , Award) • Num_game_played Player(<u>User_id</u> REFERENCE User, <u>Num_game_played</u> REFERENCE Game_played_award) • User_id</p> <p>Reviewed_Total_Rating(<u>G_id</u> REFERENCES Game, Game_rating_id, Average_rating REFERENCE Average_to_star) • G_id, Game_rating_id Average_to_star(<u>Average_rating</u>, Star_rating) • Average_rating</p> <p>Record_achievement(<u>Record_time</u>, Achievement, Badge) • Record_time Game_history(<u>User_id</u> REFERENCE User, <u>Game_id</u>, <u>Record_time</u> REFERENCE Record_achievement) • User_id, Game_id</p>

Normalization

Relationships that need to be normalized to 3NF or BCNF:

User(User_id, Name, Year_joined, Account_age)

Minum cover & optimized:

User_id → Name, Year_joined, Account_age

Year_joined → Account_age

Normalized to 3NF(BCNF) in order to preserve function dependency

Normalized Decomposition:

User_age(Year_joined, Account_age)

User(User_id, Name, **Year_joined** REFERENCE User_age)

- Primary Key is the same as candidate key (Underlined)

Player(User_id REFERENCE User, Num_game_played, Award)

Minum cover & optimized:

User_id → Num_game_played, Award

Num_game_played → Award

Normalized to 3NF (BCNF) in order to preserve function dependency.

Normalized Decomposition:

Game_played_award(Num_game_played , Award)

Player(User_id REFERENCE User, **Num_game_played** REFERENCE Game_played_award)

- Primary Key is the same as candidate key (Underlined)

Reviewed_Total_Rating(G_id REFERENCES Game, Game_rating_id, Average_rating, Star_rating)

Minum cover & optimized:

G_id, Game_rating_id → Average_Rating, Star_rating

Average_rating → Star_rating

Normalized to 3NF in order to preserve function dependency.

Normalized Decomposition:

Reviewed_Total_Rating(G_id REFERENCES Game, Game_rating_id, Average_rating

REFERENCE Average_to_star)

Average_to_star(Average_rating, Star_rating)

- Primary Key is the same as candidate key (Underlined)

Has_Game_History(User_id REFERENCE User, Game_id, Achievement, Record_time, Badge)

Minum Cover & Optimized:

User_id → Achievement, Record_time, Badge

Record_time → Achievement, Badge

Achievement → Badge

Normalized to BCNF:

Normalized Decomposition:

Record_achievement(Record_time, Achievement, Badge)

Game_history(User_id REFERENCE User, Game_id, **Record_time** REFERENCE

Record_achievement)

- Primary Key is the same as candidate key (Underlined)

SQL- DDL

```
CREATE TABLE Game(  
G_id INT AUTO_INCREMENT,  
name CHAR(25) NOT NULL,  
Published_Year INT,  
PRIMARY KEY (G_id));
```

```
CREATE TABLE Genre(  
Genre_id INT AUTO_INCREMENT,  
Genre_Type CHAR(25) NOT NULL,  
PRIMARY KEY (Genre_id));
```

```
CREATE TABLE Platform(  
Model_id INT AUTO_INCREMENT,  
PRIMARY KEY (Model_id));
```

```
CREATE TABLE PC(  
Model_id INT PRIMARY KEY,  
CPU CHAR(15), GPU CHAR(15),  
RAM CHAR(15),  
FOREIGN KEY(Model_id) REFERENCES  
platform(Model_id));
```

```
CREATE TABLE Console(  
Model_id INT PRIMARY KEY,  
Generation CHAR(15) NOT NULL,  
FOREIGN KEY(Model_id) REFERENCES  
platform(Model_id));
```

```
CREATE TABLE Mobile(  
Model_id INT PRIMARY KEY,  
OS CHAR(20) NOT NULL,  
FOREIGN KEY(Model_id) REFERENCES  
platform(Model_id));
```

```
CREATE TABLE Game_Company(  
C_id INT AUTO_INCREMENT,  
name CHAR(20) NOT NULL,  
PRIMARY KEY (C_id));
```

```
CREATE TABLE Users(  
User_id INT AUTO_INCREMENT,  
name CHAR(25),  
Year_joined INT,  
PRIMARY KEY (User_id));
```

```
ALTER TABLE Users AUTO_INCREMENT=1000;
```

```
CREATE TABLE User_age(  
Year_joined INT PRIMARY KEY,  
Account_age INT);
```

```
CREATE TABLE Viewer(  
User_id INT PRIMARY KEY,  
Num_game_rated INT,  
FOREIGN KEY(User_id) REFERENCES  
Users(User_id));
```

```
CREATE TABLE Game_played_award(  
Num_game_played INT PRIMARY KEY,  
Award CHAR(25));
```

```
CREATE TABLE Player(  
User_id INT PRIMARY KEY,  
Num_game_played INT,  
FOREIGN KEY(User_id) REFERENCES  
Users(User_id),  
FOREIGN KEY(Num_game_played) REFERENCES  
Game_played_award(Num_game_played) );
```

```
CREATE TABLE Runs(  
G_id INT NOT NULL,  
Model_id INT,  
counts INT,  
PRIMARY KEY (G_id, Model_id),  
FOREIGN KEY(G_id) REFERENCES  
Game(G_id),  
FOREIGN KEY(Model_id) REFERENCES  
Platform(Model_id));
```

```
CREATE TABLE Part_of(  
Genre_id INT,  
G_id INT NOT NULL,  
PRIMARY KEY (Genre_id, G_id),  
FOREIGN KEY(Genre_id) REFERENCES  
Genre(Genre_id),  
FOREIGN KEY(G_id) REFERENCES  
Game(G_id));
```

SQL- DDL

```
CREATE TABLE Collaborate(  
A_C_id INT,  
B_C_id INT,  
PRIMARY KEY (A_C_id, B_C_id),  
FOREIGN KEY(A_C_id) REFERENCES  
Game_Company(C_id),  
FOREIGN KEY(B_C_id) REFERENCES  
Game_Company(C_id));
```

```
CREATE TABLE Rate(  
G_id INT,  
User_id INT,  
User_rating INT,  
PRIMARY KEY (G_id, User_id),  
FOREIGN KEY (G_id) REFERENCES  
Game(G_id),  
FOREIGN KEY (User_id) REFERENCES  
Users(User_id));
```

```
CREATE TABLE Published_by(  
G_id INT NOT NULL,  
C_id INT,  
PRIMARY KEY (G_id,C_id),  
FOREIGN KEY (G_id) REFERENCES  
Game(G_id),  
FOREIGN KEY (C_id) REFERENCES  
Game_company(C_id));
```

```
CREATE TABLE Average_to_star(  
Average_rating INT,  
Star_rating INT,  
PRIMARY KEY (Average_rating));
```

```
CREATE TABLE Reviewed_Total_Rating(  
G_id INT,  
Game_rating_id INT,  
Average_rating INT,  
PRIMARY KEY (G_id,Game_rating_id),  
FOREIGN KEY (G_id)  
REFERENCES Game(G_id)  
ON DELETE CASCADE,  
FOREIGN KEY (Average_rating)  
REFERENCES  
average_to_star(average_rating));
```

```
CREATE TABLE Record_achievement(  
Record_time TIME,  
Achievement INT,  
Badge CHAR(25),  
PRIMARY KEY (Record_time));
```



```
CREATE TABLE Game_history(  
User_id INT,  
Game_id INT,  
Record_time TIME,  
PRIMARY KEY (User_id,Game_id),  
FOREIGN KEY (User_id) REFERENCES  
Users(User_id) ON DELETE CASCADE,  
FOREIGN KEY (Record_time) REFERENCES  
Record_achievement(Record_time));
```

```
CREATE TABLE Prefer_Preference(  
User_id INT,  
Game_tag_1 INT,  
Game_tag_2 INT,  
Game_tag_3 INT,  
PRIMARY KEY (User_id),  
FOREIGN KEY (User_id) REFERENCES  
Users(User_id),  
FOREIGN KEY (Game_tag_1) REFERENCES  
Genre(Genre_id),  
FOREIGN KEY (Game_tag_2) REFERENCES  
Genre(Genre_id),  
FOREIGN KEY (Game_tag_3) REFERENCES  
Genre(Genre_id));
```

Populate Table

SELECT * FROM `user_age`				SELECT * FROM `users`				SELECT * FROM `platform`					
<input type="checkbox"/> Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] []				<input type="checkbox"/> Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]				<input type="checkbox"/> Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] []					
<input type="checkbox"/> Show all Number of rows: 25		Filter rows: Search this		<input type="checkbox"/> Show all Number of rows: 25		Filter rows: Search this		<input type="checkbox"/> Show all Number of rows: 25		Filter rows: Search this			
Extra options				Extra options				Extra options					
← T →		User_id	name	← T →		User_id	name	← T →		User_id	name		
		Year_joined	Account_age			Year_joined	Model_id			Year_joined	Model_id		
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2022	0	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2022	1	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2022	1		
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2023	1	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2022	Gaven Great	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2022	2		
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2024	2	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2022	Rick N Mory	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2022	3		
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2025	3	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2022	Best Player	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2022	4		
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2026	4	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2022	3-Year-Old Kid	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2022	5		
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete				<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2022	Your Prof	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2022	6		
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete				<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2022	Kraken	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2022	7		
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete				<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2022	Mad Dog	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2022	8		
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete				<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2022	BroCode	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2022	9		
SELECT * FROM `viewer`				SELECT * FROM `pc`				SELECT * FROM `mobile`					
<input type="checkbox"/> Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] []				<input type="checkbox"/> Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]				<input type="checkbox"/> Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] []					
<input type="checkbox"/> Show all Number of rows: 25		Filter rows: Search this		<input type="checkbox"/> Show all Number of rows: 25		Filter rows: Search this table		<input type="checkbox"/> Show all Number of rows: 25		Filter rows: Search this			
Extra options				Extra options				Extra options					
← T →		User_id	num_game_rated	← T →		Model_id	CPU	GPU	RAM	← T →		Model_id	OS
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		1000	2	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		3	def CPU	def GPU	def RAM	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2	default mobile
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		1001	1	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		6	Intel	RTX 3060	16	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		5	Android
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		1002	1	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		9	Intel i7	RTX 3080	12	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		8	iOS
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		1003	0	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		12	AMD Ryzen	RTX 3090	12	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		11	Linux
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		1004	0	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		15	AMD Ryzen 9	RX 6600	16	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		14	Windows
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		1005	0	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete						<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete			
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		1006	0	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete						<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete			
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		1007	1	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete						<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete			
SELECT * FROM `player`				SELECT * FROM `console`				SELECT * FROM `mobile`					
<input type="checkbox"/> Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] []				<input type="checkbox"/> Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]				<input type="checkbox"/> Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] []					
<input type="checkbox"/> Show all Number of rows: 25		Filter rows: Search this		<input type="checkbox"/> Show all Number of rows: 25		Filter rows: Search this		<input type="checkbox"/> Show all Number of rows: 25		Filter rows: Search this			
Extra options				Extra options				Extra options					
← T →		User_id	Num_game_played	← T →		Model_id	Generation	← T →		Model_id	OS		
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		1000	5	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		1	default console	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		2	default mobile		
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		1006	5	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		4	Nintendo Switch	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		5	Android		
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		1011	5	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		7	PlayStation 4	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		8	iOS		
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		1004	10	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		10	PlayStation 5	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		11	Linux		
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		1002	20	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		13	Xbox One S	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		12	Windows		
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		1001	50	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		16	Nintendo DS	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		13			
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		1010	100	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		19	LONSUN Classic	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		14			
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete				<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		22	HAndPE Retro Classic	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		15			
<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete				<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		25	Wii	<input type="checkbox"/> Edit Copy <input type="checkbox"/> Delete		16			

Populate Table

```
SELECT * FROM `prefer_preference`
```

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

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

Extra options

	User_id	Game_tag_1	Game_tag_2	Game_tag_3
<input type="checkbox"/>	1000	2	4	5
<input type="checkbox"/>	1002	3	8	9
<input type="checkbox"/>	1003	9	10	6
<input type="checkbox"/>	1004	7	1	8
<input type="checkbox"/>	1008	6	3	1

```
SELECT * FROM `record_achievement`
```

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

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

Extra options

	Record_time	Achievement	Badge
<input type="checkbox"/>	05:00:00	1 Starter	
<input type="checkbox"/>	15:00:00	2 Novice	
<input type="checkbox"/>	20:00:00	3 Apprentice	
<input type="checkbox"/>	50:00:00	4 Adept	
<input type="checkbox"/>	100:00:00	5 Master	
<input type="checkbox"/>	200:00:00	6 Exalted	
<input type="checkbox"/>	500:00:00	7 Legendary	

```
SELECT * FROM `reviewed_total_rating`
```

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

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

Extra options

	G_id	Game_rating_id	Average_rating
<input type="checkbox"/>	2	1	0
<input type="checkbox"/>	8	1	6
<input type="checkbox"/>	5	1	7
<input type="checkbox"/>	7	1	7
<input type="checkbox"/>	4	1	8
<input type="checkbox"/>	1	1	9
<input type="checkbox"/>	6	1	9
<input type="checkbox"/>	3	1	10

```
SELECT * FROM `rate`
```

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

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

Extra options

	G_id	User_id	User_rating
<input type="checkbox"/>	1	1000	9
<input type="checkbox"/>	3	1007	10
<input type="checkbox"/>	4	1002	8
<input type="checkbox"/>	5	1001	7
<input type="checkbox"/>	7	1000	7
<input type="checkbox"/>	8	1010	6

```
SELECT * FROM `average_to_star`
```

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

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

Extra options

	Average_rating	Star_rating
<input type="checkbox"/>	0	0
<input type="checkbox"/>	1	0
<input type="checkbox"/>	2	1
<input type="checkbox"/>	3	1
<input type="checkbox"/>	4	2
<input type="checkbox"/>	5	2
<input type="checkbox"/>	6	3
<input type="checkbox"/>	7	3

```
SELECT * FROM `game_played_award`
```

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

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

Extra options

	Num_game_played	Award
<input type="checkbox"/>	5	Newbie
<input type="checkbox"/>	10	Starter
<input type="checkbox"/>	15	Try-Hard
<input type="checkbox"/>	20	Typical Gamer
<input type="checkbox"/>	50	Expert
<input type="checkbox"/>	100	Game Master
<input type="checkbox"/>	200	Ace

Populate Table

<code>SELECT * FROM `game`</code>
<input type="checkbox"/> Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
<input type="checkbox"/> Show all Number of rows: 25 <input type="button" value="▼"/> Filter rows: Search this table

Extra options

	G_id	name	Published_Year
<input type="checkbox"/> Edit Copy Delete	1	Minecraft	2011
<input type="checkbox"/> Edit Copy Delete	2	Animal Crossing: New Horizons	2021
<input type="checkbox"/> Edit Copy Delete	3	League Of Legends	2009
<input type="checkbox"/> Edit Copy Delete	4	Valorant	2020
<input type="checkbox"/> Edit Copy Delete	5	Genshin Impact	2020
<input type="checkbox"/> Edit Copy Delete	6	PUBG	2017
<input type="checkbox"/> Edit Copy Delete	7	Clash of Clan	2012
<input type="checkbox"/> Edit Copy Delete	8	Dead by Daylight	2016

`SELECT * FROM `game_company``

<input type="checkbox"/> Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code]
<input type="checkbox"/> Show all Number of rows: 25 <input type="button" value="▼"/>
Filter rows: Search this table

Extra options

	C_id	name
<input type="checkbox"/> Edit Copy Delete	1	Mojang Studios
<input type="checkbox"/> Edit Copy Delete	2	Sony
<input type="checkbox"/> Edit Copy Delete	3	Xbox
<input type="checkbox"/> Edit Copy Delete	4	Nintendo
<input type="checkbox"/> Edit Copy Delete	5	Riot Games
<input type="checkbox"/> Edit Copy Delete	6	miHoYo
<input type="checkbox"/> Edit Copy Delete	7	Krafton
<input type="checkbox"/> Edit Copy Delete	8	Tencent Games
<input type="checkbox"/> Edit Copy Delete	9	Supercell
<input type="checkbox"/> Edit Copy Delete	10	Behaviour Interactive

`SELECT * FROM `published_by``

<input type="checkbox"/> Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code]
<input type="checkbox"/> Show all Number of rows: 25 <input type="button" value="▼"/>
Filter rows: Search this table

Extra options

	G_id	C_id
<input type="checkbox"/> Edit Copy Delete	1	1
<input type="checkbox"/> Edit Copy Delete	2	4
<input type="checkbox"/> Edit Copy Delete	3	5
<input type="checkbox"/> Edit Copy Delete	4	5
<input type="checkbox"/> Edit Copy Delete	5	6
<input type="checkbox"/> Edit Copy Delete	6	7
<input type="checkbox"/> Edit Copy Delete	7	9
<input type="checkbox"/> Edit Copy Delete	8	10

<code>SELECT * FROM `genre`</code>
<input type="checkbox"/> Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
<input type="checkbox"/> Show all Number of rows: 25 <input type="button" value="▼"/> Filter rows: Search this table

Extra options

	Genre_id	Genre_Type
<input type="checkbox"/> Edit Copy Delete	1	Sandbox
<input type="checkbox"/> Edit Copy Delete	2	Real-time strategy
<input type="checkbox"/> Edit Copy Delete	3	Shooters(FPS and TPS)
<input type="checkbox"/> Edit Copy Delete	4	Multiplayer online battle
<input type="checkbox"/> Edit Copy Delete	5	Role-playing
<input type="checkbox"/> Edit Copy Delete	6	Simulation and sports
<input type="checkbox"/> Edit Copy Delete	7	Puzzlers and party games
<input type="checkbox"/> Edit Copy Delete	8	Action-adventure
<input type="checkbox"/> Edit Copy Delete	9	Survival and horror
<input type="checkbox"/> Edit Copy Delete	10	Platformer

<code>SELECT * FROM `collaborate`</code>
<input type="checkbox"/> Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code]
<input type="checkbox"/> Show all Number of rows: 25 <input type="button" value="▼"/> Filter rows: Search this table

Extra options

	A_C_id	B_C_id
<input type="checkbox"/> Edit Copy Delete	1	2
<input type="checkbox"/> Edit Copy Delete	1	3
<input type="checkbox"/> Edit Copy Delete	6	8
<input type="checkbox"/> Edit Copy Delete	7	2
<input type="checkbox"/> Edit Copy Delete	7	8

<code>SELECT * FROM `published_by`</code>
<input type="checkbox"/> Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code]
<input type="checkbox"/> Show all Number of rows: 25 <input type="button" value="▼"/> Filter rows: Search this table

Extra options

	Genre_id	G_id
<input type="checkbox"/> Edit Copy Delete	1	1
<input type="checkbox"/> Edit Copy Delete	2	7
<input type="checkbox"/> Edit Copy Delete	3	4
<input type="checkbox"/> Edit Copy Delete	3	6
<input type="checkbox"/> Edit Copy Delete	4	3
<input type="checkbox"/> Edit Copy Delete	4	6
<input type="checkbox"/> Edit Copy Delete	5	2
<input type="checkbox"/> Edit Copy Delete	6	2
<input type="checkbox"/> Edit Copy Delete	8	3
<input type="checkbox"/> Edit Copy Delete	9	5

<code>SELECT * FROM `game_history`</code>
<input type="checkbox"/> Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
<input type="checkbox"/> Show all Number of rows: 25 <input type="button" value="▼"/> Filter rows: Search this table

Extra options

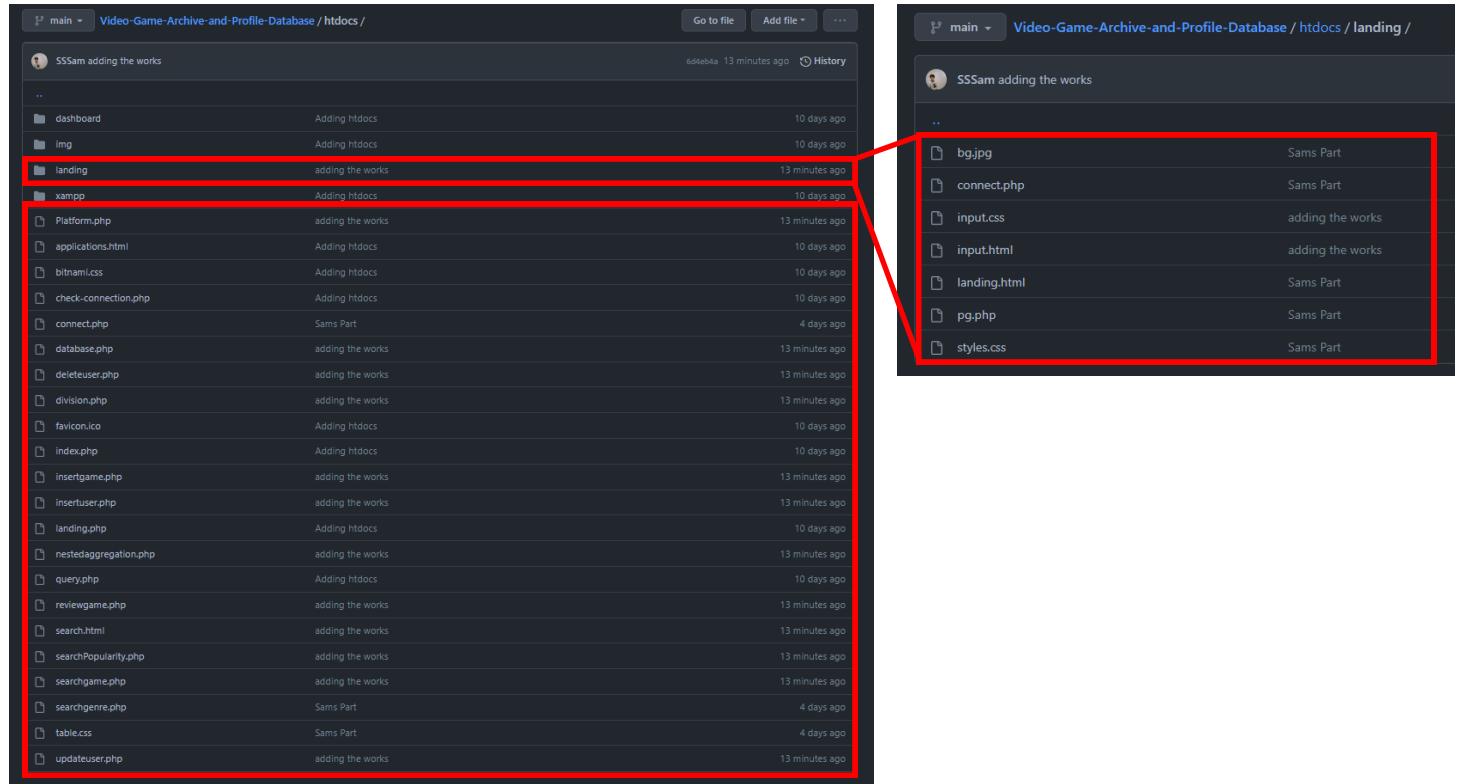
	User_id	Game_id	Record_time
<input type="checkbox"/> Edit Copy Delete	1009	8	15:00:00
<input type="checkbox"/> Edit Copy Delete	1005	6	20:00:00
<input type="checkbox"/> Edit Copy Delete	1000	1	50:00:00
<input type="checkbox"/> Edit Copy Delete	1000	7	100:00:00
<input type="checkbox"/> Edit Copy Delete	1001	3	100:00:00
<input type="checkbox"/> Edit Copy Delete	1010	2	100:00:00
<input type="checkbox"/> Edit Copy Delete	1001	5	200:00:00
<input type="checkbox"/> Edit Copy Delete	1002	4	200:00:00

Application Code:

Visit <https://github.com/SSSam/Video-Game-Archive-and-Profile-Database>

The repository consists of all the code that we wrote inside the htdocs folder.

There are php, css, html, and elements like jpg in it.



SQL Script:

Visit https://github.com/SSSam/Video-Game-Archive-and-Profile-Database/blob/main/video_game.sql

A single SQL script with over 900 lines used to create all the tables and data in our database.

The image shows a GitHub page for the 'video_game.sql' file. A large diagonal watermark 'PREVIEW' is across the page. The file content is a SQL dump with over 900 lines. Below the file content, there is a 'README.md' file which contains the following text:

Submitting our git commit hash:
Our latest commit
f0e7d85ffd713adbf2177b21018fec1467bf6198

At the bottom, there is a terminal session showing the latest commit details:

```
samuel@LAPTOP-L1OIE8MV MINGW64 ~/OneDrive/Desktop/Video-Game-Archive-and-Profile-Database (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.
nothing to commit, working tree clean
```

```
samuel@LAPTOP-L1OIE8MV MINGW64 ~/OneDrive/Desktop/Video-Game-Archive-and-Profile-Database (main)
$ git log
commit f0e7d85ffd713adbf2177b21018fec1467bf6198 (HEAD -> main, origin/main, origin/HEAD)
Date:   Fri Jul 29 17:41:04 2022 -0700
Author: SSSam <samuelfjen0509@gmail.com>
```

INTERFACE SNAPSHOT

Landing Page

Video Game Archive

This is a Video Game Database that allow you to rate games you have played. See, Update, Review yourself and other player's gaming record and more. Grind Hard, Play Hard to level up yourself in the ranking and unlock new titles!

[View Database](#)
[Input Your Data](#)
[Search Database](#)

General Video Game Database

Game Name	Published Year
Minecraft	2011
Animal Crossing	2021
League Of Legends	2009
Valorant	2020
Genshin Impact	2020
PUBG	2017
Clash of Clans	2012
Dead by Daylight	2016
Clash Royale	2016

Database Overview

Game Name	Published Year	Published By
Minecraft	2011	Mojang Studios
Animal Crossing	2021	Nintendo
League Of Legends	2009	Riot Games
Valorant	2020	Riot Games

Links in Database

Game Name	Rating	Number of Rating
Animal Crossing	8.0000	5
Clash of Clans	5.0000	4
Clash Royale	9.6667	3
Dead by Daylight	7.5000	4
Genshin Impact	6.7500	4
League Of Legends	5.8000	5
Minecraft	8.5000	4

User & Game Input

Adding Your Favourite Game

Game name:
Published Year:

Adding Player

Player Name:
Joined Year:

Update Player Name

Player ID:
Player New Name:

Delete My Name

Player ID:
Are you sure?

[Back to Main](#) [View Database](#) [Search](#)

SQL Query Search Interface

Searching in the Database

This Section returns games published after a certain year.

The attributes you can select from are:
• Game_name
• Game_type
• C_name (Game Company Name)

Enter the Attributes with Comma Separated

Type here:
Published After the Year of:

This Section returns game and its genre

The attributes you can select from are:
• Game_type
• Game_name
• Published_Year

The types of genre are:
• Sandbox
• Real-time strategy
• Multiplayer (PvP and PvE)
• Multiplayer battle
• Role-playing
• Survival and horror
• Sports and racing
• Platformer/Open World

Enter the Attributes with Comma Separated

Type here:
Specify genre types you are interested in:

SQL Query Search Result

Game	Popularity: Monthly Active Users
Minecraft	141000000
Valorant	120000000
League Of Legends	115000000
Clash of Clans	76000000
Genshin Impact	62820000
Animal Crossing	11000000
PUBG	527000
Dead by Daylight	40000

[Back to Main](#) [Update Info](#) [Search Database](#)

User Info Update

[Back to Main](#) [View Database](#) [Search](#)

Database Searching

This Section searches for Users

Enter the Username:
Type Here:

This Section searches for Games

Enter the Game Name:
Type Here:

This Section return Platform and games

Enter the Platform:
Type Here:

This Section searches for Users

Enter the Username:
Type Here:

List of all SQL Queries (1/2)

- Insertion Query - (**Insert data into Game company, game, and Rating**)
(insertgame.php, insertuser.php, reviewgame.php)

```
$stmt = $conn->prepare(  
    "INSERT INTO game( game_name, Published_Year)  
    value(?, ?)");
```

```
$stmt = $conn->prepare(  
    "INSERT INTO users( userName, Year_joined)  
    value(?, ?)");
```

```
$stmt = $conn->prepare(  
    "INSERT INTO rate( G_id, User_id, User_rating )  
    values (?, ?, ?)");
```

- Deletion Query - (**Delete user data**) (deleteuser.php)

```
$updateName= 'DELETE FROM `users`  
WHERE User_id= ?';
```

- Update Query - (**Update user name**) (updateuser.php)

```
$updateName= 'UPDATE users  
SET userName= ? WHERE User_id= ?';  
$stmt= $conn->prepare($updateName);  
$stmt->bind_param('si', $newName, $ID);  
$stmt->execute();
```

- Selection Query - (**Return games published after a user inputted year**)
(searchgame.php)

```
att_cols = $_POST['game_name'];  
year_cols = $_POST['Published_Year'];  
conn = OpenCon();  
sql = "SELECT $att_cols  
FROM game, game_company, published_by  
WHERE Published_Year > $year_cols  
AND published_by.G_id= game.G_id  
AND published_by.C_id = game_company.C_id  
ORDER BY Published_Year ASC";
```

- Projection Query - (**Return games in a user inputted genre**)
(searchgenre.php)

```
$Genre =  
"SELECT genre.Genre_id, `Genre_Type`, `game_name`  
FROM genre, part_of, game  
WHERE part_of.Genre_id= genre.Genre_id  
AND part_of.G_id = game.G_id  
ORDER BY genre.genre_ID ASC";
```

- Join Query - (**Return games with a higher monthly user count than an inputted user value**) (searchpopularity.php) (database.php)

```
$condition = $_POST['Rating_Constraint'];  
$conn = OpenCon();  
$sql = "SELECT game_name as Game,  
Popularity as 'Popularity: Monthly Active Users'  
FROM game, game_pop  
WHERE game.G_id= game_pop.G_id  
AND popularity > $condition  
GROUP BY game_name  
ORDER BY popularity DESC";
```

- Join Query - (**Return games with its publishing company**) (database.php)

```
$game_C =  
"SELECT `game_name`, `Published_Year`, `C_name`  
FROM game, game_company, published_by  
WHERE published_by.G_id= game.G_id  
AND published_by.C_id = game_company.C_id ";
```

- BONUS - Left join Group-by Query - (**Return all users and their preferences**)
(database.php)

```
$user_prefer =  
"SELECT `userName`, genre1.Genre_Type AS 'Tag1', genre2.Genre_Type AS 'Tag2', genre3.Genre_Type AS 'Tag3'  
FROM (users, prefer_preference, genre)  
LEFT JOIN genre genre1 ON genre1.Genre_id = prefer_preference.Game_tag_1  
LEFT JOIN genre genre2 ON genre2.Genre_id = prefer_preference.Game_tag_2  
LEFT JOIN genre genre3 ON genre3.Genre_id = prefer_preference.Game_tag_3  
WHERE users.User_id = prefer_preference.User_id  
GROUP BY users.userName";
```

List of all SQL Queires (2/2)

- Join Query - (*Return all games and which platform they are used on*) (Platform.php)

```
if($table == 'PC')
$result_sql =
"SELECT game.game_name as 'Game Name', `CPU`, `GPU`, `RAM`
FROM game, $table
JOIN runs on runs.Model_id = $table.Model_id
JOIN platform on platform.Model_id = $table.Model_id
Where game.G_id= runs.G_id";
```

```
else if ($table == "Console"){
$result_sql =
"SELECT game.game_name as 'Game Name', `Generation`
FROM game, $table
JOIN runs on runs.Model_id = $table.Model_id
JOIN platform on platform.Model_id = $table.Model_id
Where game.G_id= runs.G_id";
}
```

```
else if($table== "Mobile"){
$result_sql =
"SELECT game.game_name as 'Game Name', `OS`
FROM game, $table
JOIN runs on runs.Model_id = $table.Model_id
JOIN platform on platform.Model_id = $table.Model_id
Where game.G_id= runs.G_id";
}
```

- Aggregation Query - (*Return games with their calculated average rating and number of ratings for each game*) (database.php)

```
$game= "SELECT game.game_name as Game,
Avg(user_rating) as 'Average Rating'
FROM game, rate
WHERE game.G_id= rate.g_id
GROUP BY game.g_id";
```

```
$GameRate =
"SELECT game_name, AVG(User_rating) AS avgRating, count('*') AS 'Total Number of Rating'
FROM game, rate
WHERE game.G_id= rate.g_id
GROUP BY game_name";
```

- SQL Query - (*Returns user data, game data, and game company data*) (database.php)

```
$user =
"SELECT `User_id`, `userName`, `Year_joined`
FROM users";
```

```
$Collaborate =
"SELECT A.C_name AS C_name1, B.C_name AS C_name2
FROM game_company A, game_company B, collaborate
WHERE collaborate.A_C_id= A.C_id
AND collaborate.B_C_id= B.C_id";
```

```
$game_C =
"SELECT `g_id`, `game_name`, `Published_Year`
FROM game";
```

- Nested Aggregation with group-by Query - (*Return all games and games with the highest and lowest rating*) (nestedaggregation.php)

```
$bestgame= "SELECT
max(avggame.avgrate) as 'Best Game Rating',
min(avggame.avgrate) as 'Worst Game Rating'
FROM (
    SELECT Avg(user_rating) as avgrate
    FROM game, rate
    WHERE game.G_id= rate.g_id
    GROUP BY game.g_id) avggame";
```

- Division Query (*Return users that have rated ALL games*) (division.php)

```
$sqlFinal="SELECT `userName` as 'All the users who rated All the games in this database'
FROM users as s
WHERE NOT EXISTS (
    ( SELECT rated.g_id FROM rate as rated)
    EXCEPT
    (SELECT SP.g_id FROM rate SP WHERE SP.user_id= s.user_id))";
```