

DBMS - Mini Project

ONLINE GAME  
MANAGEMENT

Submitted By:

Name : Shreyas S

SRN : PES1UG20CS408

V Semester Section :G

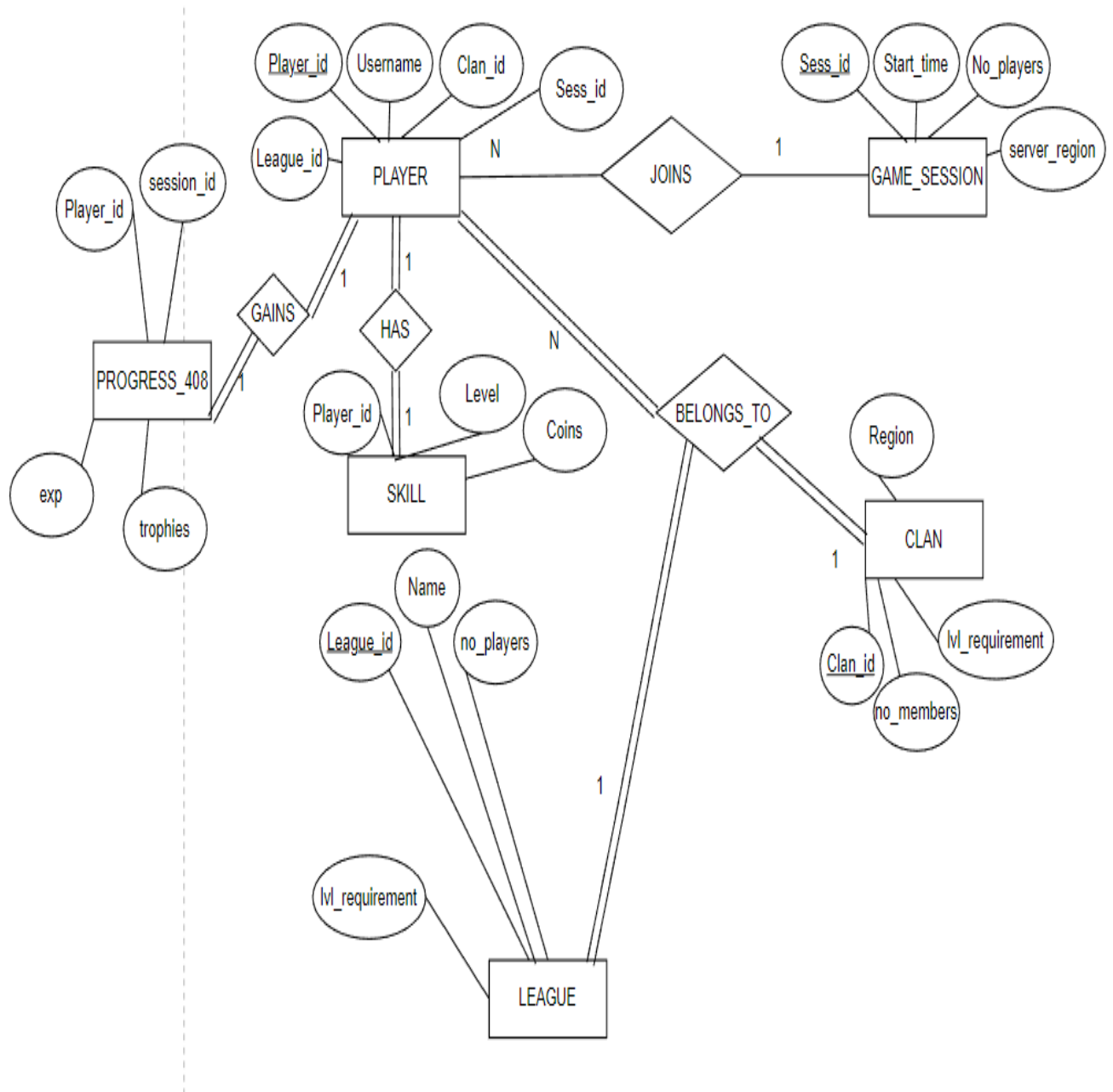
## **Short Description and Scope of the Project**

Every game has a database to keep track of all the players, interactions and leaderboards. This is a sample representation of how similar game applications organize and access data of the players. The project was developed in MySQL using MariaDB. The Frontend was developed using streamlit. It is equipped with triggers, procedures and functions along with the application of cursors for better analysis, and processing of the data.

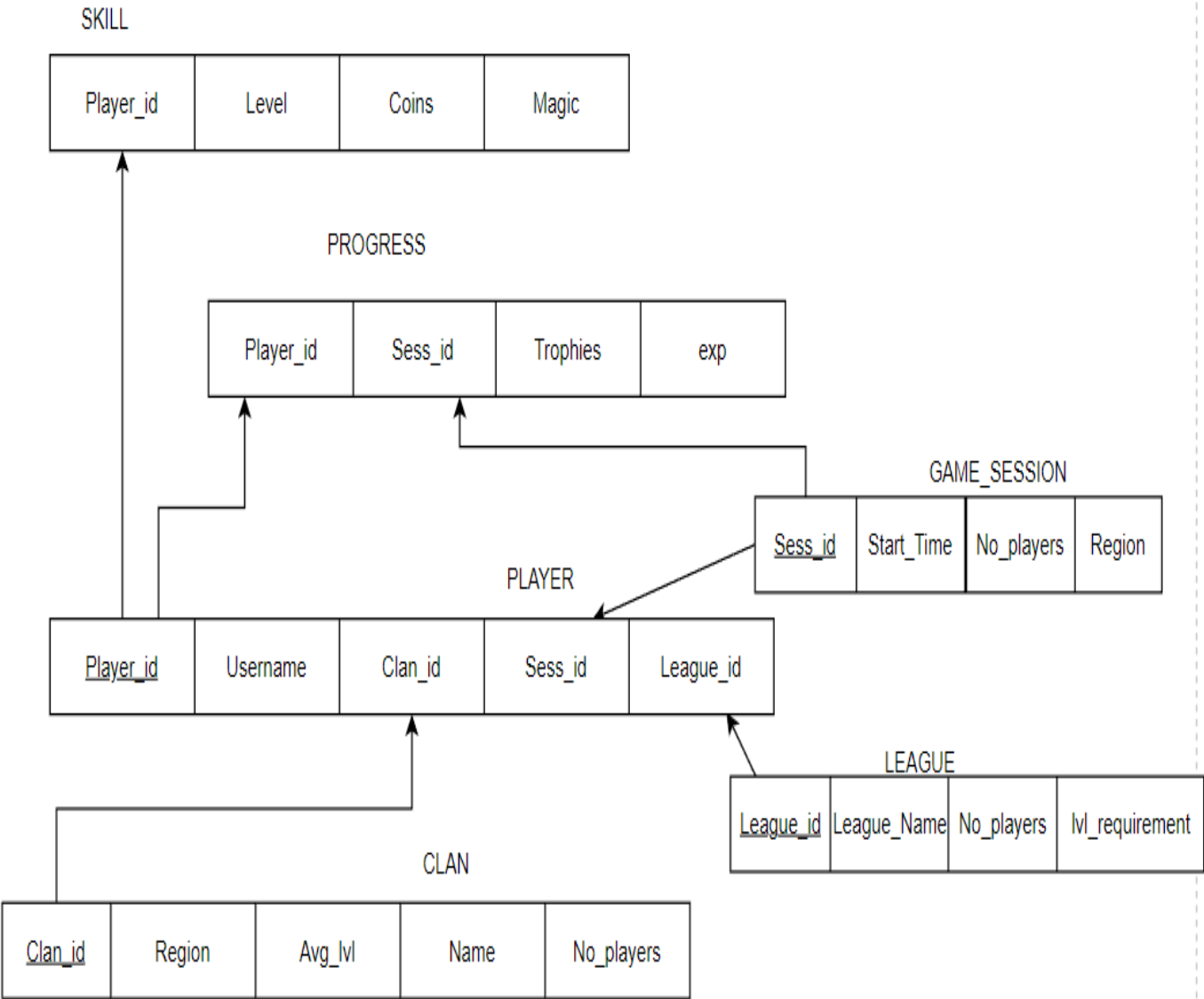
### **SCOPE:**

The database tries to emulate the database of an online RPG ( role playing game) wherein the main purpose of the database is to handle the players along with their respective properties. The players can belong to various clans of their choice and the purpose behind this is to group similar players based on their stats. The stats of a player is used to build comparisons , select matchups etc. Going into the details of these stats, the player belongs to a league, which wrt a RDBMS, is its own table with its own primary key. The assignment of a player's league to a player is done on the basis of the amount of trophies the player has along with the player's level. The level of the player is also used to level-gate entries into clans and leagues. Tables like game-session are used to keep track of players currently online which also assigns them game\_session ids based on the region of the player. Moreover a table called progress calculates a player's progress in a particular session.

## ER Diagram



Relational Schema :



## DDL statements - Building the database

-- Creating the database:

```
CREATE database mini_proj_408;
```

-- Creating the tables:

```
CREATE TABLE LEAGUE_408(  
league_id INT NOT NULL,  
league_name VARCHAR(20) NOT NULL,  
--league_level INT NOT NULL,  
No_players INT NOT NULL,  
lvl_req INT NOT NULL,  
PRIMARY KEY(league_id)  
);  
CREATE TABLE GAME_SESSION_408(  
session_id INT NOT NULL,  
start_time TIME NOT NULL,  
No_players INT NOT NULL,  
server_region VARCHAR(20) NOT NULL,  
PRIMARY KEY(session_id)  
);
```

```
CREATE TABLE CLAN_408(  
clan_id INT NOT NULL,  
no_members INT NOT NULL,  
lvl_req INT NOT NULL,  
PRIMARY KEY(clan_id)  
);  
--PLAYER ENTITY  
CREATE TABLE PLAYER_408(  
player_id INT NOT NULL,  
username VARCHAR(20) NOT NULL,  
clan_id INT NOT NULL,  
league_id INT NOT NULL,  
session_id INT NOT NULL,  
PRIMARY KEY(player_id),  
FOREIGN KEY(clan_id) REFERENCES CLAN_408(clan_id),  
FOREIGN KEY(league_id) REFERENCES
```

```
LEAGUE_408(league_id),  
FOREIGN KEY(session_id) REFERENCES  
GAME_SESSION_408(session_id)  
);
```

--SKILLS ENTITY

```
CREATE TABLE SKILLS_408(  
    player_id INT NOT NULL,  
    level INT NOT NULL,  
    coins INT NOT NULL,  
    FOREIGN KEY(player_id) REFERENCES  
PLAYER_408(player_id)  
);
```

-- Progress Entity

```
CREATE TABLE PROGRESS_408(  
    player_id INT NOT NULL,  
    --session_id INT NOT NULL,  
    trophies INT NOT NULL,  
    exp INT ,  
    FOREIGN KEY(player_id) REFERENCES  
PLAYER_408(player_id)  
);
```

-- test:

```
CREATE TABLE attack_408(  
    player_id INT NOT NULL,  
    session_id INT NOT NULL,  
    player2_id INT NOT NULL,  
    XP INT NOT NULL,  
    FOREIGN KEY(player_id) REFERENCES  
PLAYER_408(player_id),  
  
    FOREIGN KEY(session_id) REFERENCES  
GAME_SESSION_408(session_id),  
    FOREIGN KEY(player2_id) REFERENCES  
PLAYER_408(player_id)  
);
```

## Populating the Database

```
-- Creating the database:
CREATE database mini_proj_408;

-- Creating the tables:
CREATE TABLE LEAGUE_408(
league_id INT NOT NULL,
league_name VARCHAR(20) NOT NULL,
--league_level INT NOT NULL,
No_players INT NOT NULL,
lvl_req INT NOT NULL,
PRIMARY KEY(league_id)
);

CREATE TABLE GAME_SESSION_408(
session_id INT NOT NULL,
start_time TIME NOT NULL,
No_players INT NOT NULL,
server_region VARCHAR(20) NOT NULL,
PRIMARY KEY(session_id)
);

CREATE TABLE CLAN_408(
clan_id INT NOT NULL,
no_members INT NOT NULL,
lvl_req INT NOT NULL,
PRIMARY KEY(clan_id)
);

--PLAYER ENTITY
CREATE TABLE PLAYER_408(
player_id INT NOT NULL,
username VARCHAR(20) NOT NULL,
clan_id INT NOT NULL,
league_id INT NOT NULL,
session_id INT NOT NULL,
PRIMARY KEY(player_id),
FOREIGN KEY(clan_id) REFERENCES
```

```

CLAN_408(clan_id),
FOREIGN KEY(league_id) REFERENCES
LEAGUE_408(league_id),
FOREIGN KEY(session_id) REFERENCES
GAME_SESSION_408(session_id)
);

--SKILLS ENTITY
CREATE TABLE SKILLS_408(
    player_id INT NOT NULL,
    level INT NOT NULL,
    coins INT NOT NULL,
    FOREIGN KEY(player_id) REFERENCES
PLAYER_408(player_id)
);

-- Progress Entity
CREATE TABLE PROGRESS_408(
    player_id INT NOT NULL,
    --session_id INT NOT NULL,
    trophies INT NOT NULL,
    exp INT ,
    FOREIGN KEY(player_id) REFERENCES
PLAYER_408(player_id)
);

-- test:
CREATE TABLE attack_408(
    player_id INT NOT NULL,
    session_id INT NOT NULL,
    player2_id INT NOT NULL,
    XP INT NOT NULL,
    FOREIGN KEY(player_id) REFERENCES
PLAYER_408(player_id),

    FOREIGN KEY(session_id) REFERENCES

```



```
GAME_SESSION_408(session_id),  
    FOREIGN KEY(player2_id) REFERENCES  
PLAYER_408(player_id)  
  
);
```

## Join Queries

Showcase at least 4 join queries

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

### 1) QUERY:

Get the details from a join of the player\_408 and table progress\_408 to display a progress\_table join

#### SQL:

```
CREATE VIEW PLAYER_PROGRESS_408 AS
SELECT PLAYER_408.player_id, PLAYER_408.username,
PROGRESS_408.trophies, PROGRESS_408.exp
FROM PLAYER_408
INNER JOIN PROGRESS_408
ON PLAYER_408.player_id = PROGRESS_408.player_id;
```

#### OUTPUT:

```
MariaDB [mini_proj_408]> CREATE VIEW PLAYER_PROGRESS_408 AS
-> SELECT PLAYER_408.player_id, PLAYER_408.username, PROGRESS_408.trophies, PROGRESS_408.exp
-> FROM PLAYER_408
-> INNER JOIN PROGRESS_408
-> ON PLAYER_408.player_id = PROGRESS_408.player_id;
Query OK, 0 rows affected (0.008 sec)

MariaDB [mini_proj_408]> select * from player_progress_408;
+-----+-----+-----+-----+
| player_id | username | trophies | exp |
+-----+-----+-----+-----+
| 69 | noob | 16 | 40 |
| 70 | maniac | 599 | 79 |
| 71 | jesse | 400 | 45 |
| 72 | hog_rider | 345 | 55 |
| 73 | waltuh | 660 | 21 |
| 74 | pekka | 2 | 0 |
| 75 | kkk | 800 | 0 |
+-----+-----+-----+-----+
7 rows in set (0.001 sec)

MariaDB [mini_proj_408]> |
```

**2) QUERY:** Get the list of player\_id ,username, level, coins by joining 3 tables ie player\_408, skills\_408 and progress\_408

### SQL:

```
CREATE VIEW PLAYER_SKILLS_PROGRESS_OUTER_408 AS
SELECT PLAYER_408.player_id, PLAYER_408.username,
SKILLS_408.level, SKILLS_408.coins, PROGRESS_408.trophies,
PROGRESS_408.exp
FROM PLAYER_408
LEFT OUTER JOIN SKILLS_408
ON PLAYER_408.player_id = SKILLS_408.player_id
LEFT OUTER JOIN PROGRESS_408
ON PLAYER_408.player_id = PROGRESS_408.player_id;
```

### OUTPUT:

```
MariaDB [mini_proj_408]> CREATE VIEW PLAYER_SKILLS_PROGRESS_OUTER_408 AS
-> SELECT PLAYER_408.player_id, PLAYER_408.username, SKILLS_408.level, SKILLS_408.coins, PROGRESS_408.trophies, PROGRESS_408.exp
-> FROM PLAYER_408
-> LEFT OUTER JOIN SKILLS_408
-> ON PLAYER_408.player_id = SKILLS_408.player_id
-> LEFT OUTER JOIN PROGRESS_408
-> ON PLAYER_408.player_id = PROGRESS_408.player_id;
Query OK, 0 rows affected (0.006 sec)

MariaDB [mini_proj_408]> SELECT * FROM PLAYER_SKILLS_PROGRESS_OUTER_408;
+-----+-----+-----+-----+-----+-----+
| player_id | username | level | coins | trophies | exp |
+-----+-----+-----+-----+-----+-----+
| 69 | noob | 3 | 12000 | 16 | 40 |
| 70 | maniac | 14 | 1000000 | 599 | 79 |
| 71 | jesse | 12 | 500000 | 400 | 45 |
| 72 | hog_rider | 10 | 320000 | 345 | 55 |
| 73 | waltuh | 14 | 4000000 | 660 | 21 |
| 74 | pekka | 3 | 9000 | 2 | 0 |
| 75 | kkk | 16 | 10000000 | 800 | 0 |
+-----+-----+-----+-----+-----+-----+
7 rows in set (0.001 sec)

MariaDB [mini_proj_408]>
```

**3) QUERY:** To get a view of all players belonging to a particular game session.

### SQL:

```
CREATE VIEW GAME_SESSION_PLAYER_408 AS
SELECT GAME_SESSION_408.session_id,
```

```

GAME_SESSION_408.server_region, PLAYER_408.player_id,
PLAYER_408.username
FROM GAME_SESSION_408
LEFT OUTER JOIN PLAYER_408
ON GAME_SESSION_408.session_id = PLAYER_408.session_id;

```

## OUTPUT:

```

MariaDB [mini_proj_408]> CREATE VIEW GAME_SESSION_PLAYER_408 AS
  -> SELECT GAME_SESSION_408.session_id, GAME_SESSION_408.server_region, PLAYER_408.player_id, PLAYER_408.username
  -> FROM GAME_SESSION_408
  -> LEFT OUTER JOIN PLAYER_408
  -> ON GAME_SESSION_408.session_id = PLAYER_408.session_id;
Query OK, 0 rows affected (0.002 sec)

MariaDB [mini_proj_408]> SELECT * FROM GAME_SESSION_PLAYER_408;
+-----+-----+-----+-----+
| session_id | server_region | player_id | username |
+-----+-----+-----+-----+
| 10 | India | 70 | maniac |
| 10 | India | 71 | jesse |
| 11 | Europe | 73 | waltuh |
| 11 | Europe | 75 | kkk |
| 12 | Singapore | 69 | noob |
| 12 | Singapore | 74 | pekka |
| 13 | India | 72 | hog_rider |
| 14 | Europe | NULL | NULL |
| 15 | Singapore | NULL | NULL |
+-----+-----+-----+-----+
9 rows in set (0.001 sec)

MariaDB [mini_proj_408]>

```

4)

**QUERY:** To get the details of players along with clan details if they belonging to a particular game session

**SQL:**

```

CREATE VIEW GAME_SESSION_PLAYER_CLAN_408 AS
SELECT GAME_SESSION_408.session_id,
GAME_SESSION_408.server_region, PLAYER_408.player_id,
PLAYER_408.username, CLAN_408.clan_id, CLAN_408.no_members,
CLAN_408.lv_req
FROM GAME_SESSION_408

```

RIGHT OUTER JOIN PLAYER\_408  
ON GAME\_SESSION\_408.session\_id = PLAYER\_408.session\_id  
RIGHT OUTER JOIN CLAN\_408  
ON PLAYER\_408.clan\_id = CLAN\_408.clan\_id;

## OUTPUT:

```
MariaDB [mini_proj_408]> CREATE VIEW GAME_SESSION_PLAYER_CLAN_408 AS
-> SELECT GAME_SESSION_408.session_id, GAME_SESSION_408.server_region, PLAYER_408.player_id, PLAYER_408.username, CL
AN_408.clan_id, CLAN_408.no_members, CLAN_408.lvl_req
-> FROM GAME_SESSION_408
-> RIGHT OUTER JOIN PLAYER_408
-> ON GAME_SESSION_408.session_id = PLAYER_408.session_id
-> RIGHT OUTER JOIN CLAN_408
-> ON PLAYER_408.clan_id = CLAN_408.clan_id;
Query OK, 0 rows affected (0.006 sec)

MariaDB [mini_proj_408]> select * from game_session_player_clan_408;
+-----+-----+-----+-----+-----+-----+-----+
| session_id | server_region | player_id | username | clan_id | no_members | lvl_req |
+-----+-----+-----+-----+-----+-----+-----+
| 12 | Singapore | 69 | noob | 23450 | 24 | 3 |
| NULL | NULL | NULL | NULL | 23451 | 16 | 15 |
| NULL | NULL | NULL | NULL | 23452 | 34 | 8 |
| 12 | Singapore | 74 | pekka | 23453 | 23 | 3 |
| 10 | India | 71 | jesse | 23454 | 9 | 12 |
| NULL | NULL | NULL | NULL | 23455 | 9 | 6 |
| 10 | India | 70 | maniac | 23456 | 15 | 14 |
| 11 | Europe | 73 | waltuh | 23456 | 15 | 14 |
| 11 | Europe | 75 | kkk | 23457 | 2 | 16 |
| 13 | India | 72 | hog_rider | 23458 | 8 | 10 |
+-----+-----+-----+-----+-----+-----+-----+
10 rows in set (0.001 sec)

MariaDB [mini_proj_408]>
```

## Aggregate Functions

Showcase at least 4 Aggregate function queries

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

### 1) QUERY:

Get the average value of number of trophies for all players in a particular clan.

### SQL:

select avg(trophies) from PROGRESS\_408 where player\_id in (select player\_id from PLAYER\_408 where clan\_id = 23456);

### OUTPUT:

```
MariaDB [mini_proj_408]> select avg(trophies) from PROGRESS_408 where player_id in (select player_id from PLAYER_408 where clan_id = 23456);
+-----+
| avg(trophies) |
+-----+
|      629.5000 |
+-----+
1 row in set (0.000 sec)
```

### 2)

### QUERY:

Get the count of players belonging to a particular clan

**SQL:**

select count(player\_id) from PLAYER\_408 where clan\_id = 23458;

**OUTPUT:**

```
MariaDB [mini_proj_408]> select count(player_id) from PLAYER_408 where clan_id = 23458;
+-----+
| count(player_id) |
+-----+
|                1 |
+-----+
1 row in set (0.000 sec)

MariaDB [mini_proj_408]> |
```

3)

**QUERY:**

Get the total number of trophies for a particular clan

**SQL:**

select sum(trophies) from PROGRESS\_408 where player\_id in (select player\_id from PLAYER\_408 where clan\_id = 23458);

**OUTPUT:**

```
MariaDB [mini_proj_408]> select sum(trophies) from PROGRESS_408 where player_id in (select player_id from PLAYER_408 where clan_id = 23458);
+-----+
| sum(trophies) |
+-----+
|            345 |
+-----+
1 row in set (0.072 sec)
```

4)

**QUERY:**

Get the max value of coins a player holds from all clans where the

level requirement for the clan is 10.

### SQL:

```
select max(coins) from SKILLS_408 where player_id in (select player_id from  
PLAYER_408 where clan_id in (select clan_id from CLAN_408 where lvl_req  
= 10));
```

### OUTPUT:

```
MariaDB [mini_proj_408]> select max(coins) from SKILLS_408 where player_id in (select player_id from PLAYER_408 where cl  
an_id in (select clan_id from CLAN_408 where lvl_req = 10));  
+-----+  
| max(coins) |  
+-----+  
|      320000 |  
+-----+  
1 row in set (0.016 sec)
```



## Set Operations

Showcase at least 4 Set Operations queries

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

### 1) QUERY: Union of people based on Europe and India

#### SQL:

```
(Select player_408.username,game_session_408.server_region
FROM player_408
INNER JOIN game_session_408
ON player_408.session_id = game_session_408.session_id
WHERE game_session_408.server_region = 'India')
UNION
(Select player_408.username,game_session_408.server_region
FROM player_408
INNER JOIN game_session_408
ON player_408.session_id = game_session_408.session_id
WHERE game_session_408.server_region = 'Europe');
```

#### OUTPUT:

```

MariaDB [mini_proj_408]> (Select player_408.username,game_session_408.server_region
-> FROM player_408
-> INNER JOIN game_session_408
-> ON player_408.session_id = game_session_408.session_id
-> WHERE game_session_408.server_region = 'India')
-> UNION
-> (Select player_408.username,game_session_408.server_region
-> FROM player_408
-> INNER JOIN game_session_408
-> ON player_408.session_id = game_session_408.session_id
-> WHERE game_session_408.server_region = 'Europe');
+-----+-----+
| username | server_region |
+-----+-----+
| maniac   | India         |
| jesse    | India         |
| hog_rider| India         |
| waltuh   | Europe        |
| kkk      | Europe        |
+-----+-----+
5 rows in set (0.001 sec)

MariaDB [mini_proj_408]> |

```

2)

**QUERY:** Get a list of players not belonging to a particular clan

**SQL:**

```

Select player_408.username, clan_408.clan_id FROM
player_408,clan_408
WHERE player_408.clan_id = clan_408.clan_id
EXCEPT
Select player_408.username, clan_408.clan_id FROM
player_408,clan_408
WHERE player_408.clan_id = clan_408.clan_id AND clan_408.clan_id
= 242747;

```

**OUTPUT:**

```

MariaDB [mini_proj_408]> Select player_408.username, clan_408.clan_id FROM player_408,clan_408
-> WHERE player_408.clan_id = clan_408.clan_id
-> EXCEPT
-> Select player_408.username, clan_408.clan_id FROM player_408,clan_408
-> WHERE player_408.clan_id = clan_408.clan_id AND clan_408.clan_id = 242747;
+-----+-----+
| username | clan_id |
+-----+-----+
| noob     | 23450   |
| maniac   | 23456   |
| jesse    | 23454   |
| hog_rider| 23458   |
| waltuh   | 23456   |
| pekka    | 23453   |
| kkk      | 23457   |
+-----+-----+
7 rows in set (0.007 sec)

```

3)

**QUERY:** Get a list of all player details( coins, level) having their player level to be 10

**SQL:**

```

SELECT SKILLS_408.coins , skills_408.level, player_408.username
FROM skills_408
INNER JOIN player_408 ON skills_408.player_id =
player_408.player_id
intersect
SELECT SKILLS_408.coins , skills_408.level, player_408.username
FROM skills_408
INNER JOIN player_408 ON skills_408.player_id =
player_408.player_id
WHERE skills_408.level = 10;

```

**OUTPUT:**

```

MariaDB [mini_proj_408]> SELECT SKILLS_408.coins , skills_408.level, player_408.username FROM skills_408
-> INNER JOIN player_408 ON skills_408.player_id = player_408.player_id
-> intersect
-> SELECT SKILLS_408.coins , skills_408.level, player_408.username FROM skills_408
-> INNER JOIN player_408 ON skills_408.player_id = player_408.player_id
-> WHERE skills_408.level = 10;
+-----+-----+-----+
| coins | level | username |
+-----+-----+-----+
| 320000 | 10 | hog_rider |
+-----+-----+-----+
1 row in set (0.001 sec)

```

#### 4) QUERY:

Get a list of all players along with their username and player\_id using the union all operator.

#### SQL:

```

SELECT player_408.username, player_408.player_id FROM player_408
UNION ALL
SELECT player_408.username, player_408.player_id FROM player_408
WHERE player_408.player_id = 70;

```

#### OUTPUT:

```

MariaDB [mini_proj_408]> SELECT player_408.username, player_408.player_id FROM player_408
-> UNION ALL
-> SELECT player_408.username, player_408.player_id FROM player_408
-> WHERE player_408.player_id = 70;
+-----+-----+
| username | player_id |
+-----+-----+
| noob     | 69        |
| maniac   | 70        |
| jesse    | 71        |
| hog_rider | 72        |
| waltuh   | 73        |
| pekka    | 74        |
| kkk      | 75        |
| maniac   | 70        |
+-----+-----+
8 rows in set (0.001 sec)

```

## Functions and Procedures

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

### PROCEDURE:

**OBJ:** To give a list of players having trophies lesser than a given amount

### SQL:

```
DROP PROCEDURE IF EXISTS players_below_trophy;

DELIMITER $$

CREATE PROCEDURE players_below_trophy(IN trophymax INT)
BEGIN
SELECT player_408.player_id,player_408.username,
progress_408.trophies FROM player_408 JOIN
progress_408 ON player_408.player_id=progress_408.player_id WHERE
progress_408.trophies<=400
ORDER BY progress_408.trophies DESC;
END $$

DELIMITER ;
```

### OUTPUT:

```
MariaDB [mini_proj_408]> DROP PROCEDURE IF EXISTS players_below_trophy;
Query OK, 0 rows affected (0.008 sec)

MariaDB [mini_proj_408]> DELIMITER $$
MariaDB [mini_proj_408]>
MariaDB [mini_proj_408]> CREATE PROCEDURE players_below_trophy(IN trophymax INT)
-> BEGIN
-> SELECT player_408.player_id,player_408.username, progress_408.trophies FROM player_408 JOIN
-> progress_408 ON player_408.player_id=progress_408.player_id WHERE progress_408.trophies<=400
-> ORDER BY progress_408.trophies DESC;
-> END $$
Query OK, 0 rows affected (0.003 sec)

MariaDB [mini_proj_408]> DELIMITER ;
MariaDB [mini_proj_408]> call players_below_trophy(400);
+-----+-----+-----+
| player_id | username | trophies |
+-----+-----+-----+
| 71 | jesse | 400 |
| 72 | hog_rider | 345 |
| 69 | noob | 16 |
| 74 | pekka | 2 |
+-----+-----+-----+
4 rows in set (0.002 sec)

Query OK, 0 rows affected (0.007 sec)

MariaDB [mini_proj_408]> |
```

## FUNCTION:

**OBJ:** To give the sum of all the coins belonging to players belonging to a particular clan.

## QUERY:

```
DROP FUNCTION IF EXISTS PES1UG20CS408_FUNC;

DELIMITER $$

CREATE FUNCTION PES1UG20CS408_FUNC(clan_id INT)
RETURNS INT
BEGIN
DECLARE total_coins INT;

SELECT SUM(coins) INTO total_coins FROM skills_408 WHERE
skills_408.player_id IN (SELECT player_408.player_id FROM player_408
WHERE player_408.clan_id = clan_id);

RETURN total_coins;

END$$

DELIMITER ;
```

## OUTPUT:

```
MariaDB [mini_proj_408]> DELIMITER $$
MariaDB [mini_proj_408]> CREATE FUNCTION PES1UG20CS408_FUNC(clan_id INT)
-> RETURNS INT
-> BEGIN
-> DECLARE total_coins INT;
-> SELECT SUM(coins) INTO total_coins FROM skills_408 WHERE skills_408.player_id IN (SELECT player_408.player_id FRO
M player_408 WHERE player_408.clan_id = clan_id);
-> RETURN total_coins;
-> END$$
Query OK, 0 rows affected (0.003 sec)

MariaDB [mini_proj_408]> DELIMITER ;
MariaDB [mini_proj_408]> SELECT PES1UG20CS408_FUNC(23454);
+-----+
| PES1UG20CS408_FUNC(23454) |
+-----+
|                500000 |
+-----+
1 row in set (0.001 sec)

MariaDB [mini_proj_408]> |
```

## Triggers and Cursors

Create a Trigger and a Cursor. State the objective. Run and display the results.

### TRIGGER:

**OBJECTIVE:** The objective of the trigger is to make sure that, when a player's details from the player\_408 table is updated, the new details match the level requirements of the tables **league\_408** & **clan\_408**

### SQL:

```
drop trigger if exists player_408_lvl_req_check_update;
delimiter //

create trigger player_408_lvl_req_check_update
before update on PLAYER_408
for each row
begin
    if ( (select lvl_req from CLAN_408 where clan_id = new.clan_id) > (select
level from SKILLS_408 where player_id = new.player_id) ) then
        signal sqlstate '45000' set message_text = 'Player level is not high
enough to join clan';
    end if;

    if ( (select lvl_req from LEAGUE_408 where league_id = new.league_id) >
(select level from SKILLS_408 where player_id = new.player_id) ) then
        signal sqlstate '45000' set message_text = 'Player level is not high
enough to join league';
    end if;
end//
delimiter ;
```

### OUTPUT:

```
MariaDB [mini_proj_408]> UPDATE player_408 SET league_id = '12307' WHERE player_408.player_id = 69;
ERROR 1644 (45000): Player level is not high enough to join league
MariaDB [mini_proj_408]> |
```

```

MariaDB [mini_proj_408]> UPDATE player_408 SET league_id = '12307' WHERE player_408.player_id = 69;
ERROR 1644 (45000): Player level is not high enough to join league
MariaDB [mini_proj_408]> UPDATE 'player_408' SET 'clan_id' = '23457' WHERE 'player_408`.`player_id` = 69;
ERROR 1644 (45000): Player level is not high enough to join clan
MariaDB [mini_proj_408]> |

```

## CURSOR:

**OBJECTIVE:** Use the cursor to locate entries where username is a given argument for a trigger. This is then used to update concurrent values of the row ie no of trophies

## SQL:

```

--
DROP PROCEDURE IF EXISTS update_trophies;
DELIMITER $$
CREATE PROCEDURE update_trophies(IN uname
VARCHAR(20), IN new_trophies INT)
BEGIN
    DECLARE cont INT default 0;
    DECLARE pid INT;
    DECLARE C1 CURSOR FOR SELECT
player_408.player_id FROM player_408 WHERE
player_408.username=uname;

    DECLARE CONTINUE HANDLER FOR NOT FOUND SET
cont = 1;

    -- SELECT player_408.player_id FROM
player_408;

    OPEN C1;
    11: LOOP
        FETCH C1 INTO pid;
        if cont=1 THEN

```



```

        leave 11;
    end if;
    -- select C1;
    select pid;

        UPDATE    progress_408    SET
trophies=new_trophies WHERE player_id = pid;
    END LOOP 11;
    CLOSE C1;
END $$
DELIMITER ;

call update_trophies('jesse', 999);

```

## OUTPUT:

```

MariaDB [mini_proj_408]> UPDATE player_408 SET league_id = '12307' WHERE player_408.player_id = 69;
ERROR 1644 (45000): Player level is not high enough to join league
MariaDB [mini_proj_408]> UPDATE `player_408` SET `clan_id` = '23457' WHERE `player_408`.`player_id` = 69;
ERROR 1644 (45000): Player level is not high enough to join clan
MariaDB [mini_proj_408]> DROP PROCEDURE IF EXISTS update_trophies;
Query OK, 0 rows affected, 1 warning (0.000 sec)

```

```

MariaDB [mini_proj_408]> DELIMITER $$
MariaDB [mini_proj_408]> CREATE PROCEDURE update_trophies(IN username VARCHAR(20), IN trophies INT)
-> BEGIN
-> DECLARE player_id INT;
-> DECLARE C1 CURSOR FOR SELECT player_id FROM player_408 WHERE username = username;
-> DECLARE CONTINUE HANDLER FOR NOT FOUND SET player_id = 0;
-> OPEN C1;
-> FETCH C1 INTO player_id;
-> IF player_id = 70 THEN
-> INSERT INTO player_408(username) VALUES(username);
-> SET player_id = LAST_INSERT_ID();
-> END IF;
-> UPDATE progress_408 SET trophies = trophies WHERE player_id = player_id;
-> CLOSE C1;
-> END $$
Query OK, 0 rows affected (0.008 sec)

```

```

MariaDB [mini_proj_408]> delIMITER ;
MariaDB [mini_proj_408]> CALL UPDATE_TROPHIES('PEKKA',788);
Query OK, 0 rows affected (0.007 sec)

```

```

MariaDB [mini_proj_408]> call update_trophies('jesse',788);

```

```

+-----+
| pid |
+-----+
| 71 |
+-----+

```

1 row in set (0.095 sec)

Query OK, 1 row affected (0.111 sec)

```

MariaDB [mini_proj_408]> |

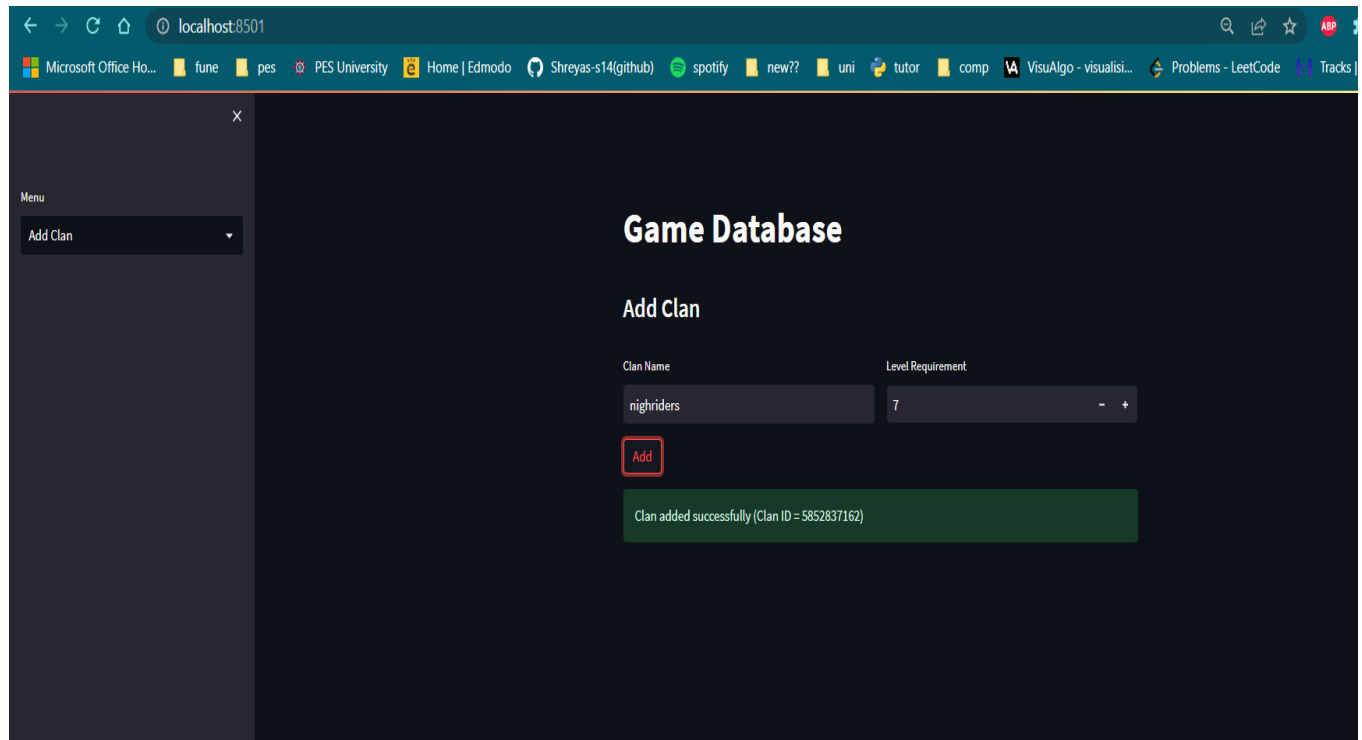
```

## **Developing a Frontend**

The frontend should support

1. Addition, Modification and Deletion of records from any chosen table
2. There should be an window to accept and run any SQL statement and display the result

**CREATE A NEW ENTRY IN CLAN\_408 TABLE:**



**ADD A NEW ENTRY TO THE PLAYER\_408 TABLE:**

X

Menu

Add Player

# Game Database

## Add Player

Username

Level

Clan Name

Trophies

Region

Xp

Coins

Add

Player added successfully

zalinsky

8

nightriders

699

India

69

1022

READ VALUES FROM THREE TABLES ie PLAYER\_408, CLAN\_408 AND LEAGUE-408

# Game Database

Read

View all players

|   | Username  | League   | Level | Trophies | Experience | Coins    | Server Region |
|---|-----------|----------|-------|----------|------------|----------|---------------|
| 0 | noob      | Bronze   | 3     | 16       | 40         | 12000    | Singapore     |
| 1 | maniac    | Master   | 14    | 599      | 79         | 1000000  | India         |
| 2 | jesse     | Crystal  | 12    | 400      | 45         | 500000   | India         |
| 3 | hog_rider | Champion | 10    | 345      | 55         | 320000   | India         |
| 4 | waltuh    | Master   | 14    | 660      | 21         | 4000000  | Europe        |
| 5 | pekka     | Bronze   | 3     | 2        | 0          | 9000     | Singapore     |
| 6 | kkk       | Titan    | 16    | 800      | 0          | 10000000 | Europe        |
| 7 | zalinsky  | Platinum | 8     | 699      | 69         | 1022     | India         |

Menu

Read

Clan Information

|   | Clan ID    | Number of members | Level Requirement | Average Trophies | Average Experience | Average... |
|---|------------|-------------------|-------------------|------------------|--------------------|------------|
| 0 | 23450      | 24                | 3                 | 160000           | 400000             | 300000     |
| 1 | 23453      | 23                | 3                 | 20000            | 0                  | 30000      |
| 2 | 23454      | 9                 | 12                | 4000000          | 450000             | 120000     |
| 3 | 23456      | 15                | 14                | 6295000          | 500000             | 140000     |
| 4 | 23457      | 2                 | 16                | 8000000          | 0                  | 160000     |
| 5 | 23458      | 8                 | 10                | 3450000          | 550000             | 100000     |
| 6 | 2147483647 | 1                 | 7                 | 6990000          | 690000             | 800000     |

League Information

|   | League   | Number of Players | Level Requirement |
|---|----------|-------------------|-------------------|
| 0 | Bronze   | 20                | 2                 |
| 1 | Silver   | 25                | 4                 |
| 2 | Gold     | 24                | 6                 |
| 3 | Platinum | 30                | 8                 |
| 4 | Champion | 21                | 10                |
| 5 | Crystal  | 19                | 12                |
| 6 | Master   | 16                | 14                |

DELETE VALUE FROM PLAYER\_408 TABLE:

Menu

Delete Player

Add Player

Add Clan

Read

Update Player

Update Clan

Delete Player

Delete Clan

Game Database

Username

zalinsky

Delete

Player removed

ENTER CUSTOM COMMAND INTO THE FRONTEND:

# Game Database

Query Input

```
SELECT PLAYER_408.player_id, PLAYER_408.username, PROGRESS_408.trophies, PROGRESS_408.exp F
```

Execute

|   | player_id | username  | trophies | exp |
|---|-----------|-----------|----------|-----|
| 0 | 69        | noob      | 16       | 40  |
| 1 | 70        | maniac    | 599      | 79  |
| 2 | 71        | jesse     | 400      | 45  |
| 3 | 72        | hog_rider | 345      | 55  |
| 4 | 73        | waltuh    | 660      | 21  |
| 5 | 74        | pekka     | 2        | 0   |
| 6 | 75        | kkk       | 800      | 0   |