CSU33041 INFORMATION MANAGEMENT II

14333615

SiKai

Database SQL Project.

Description of the application and Database constraint

The purpose of this application is to fulfil the need of the storing professional CSGO matches statistics throughout the lifespan before it's forgotten. The purposes of holding professional CSGO competitions are earning money and determining which team is the best CSGO team at that period. A CSGO player must join a team before playing professionally in the scene. Sponsors sponsor teams. A CSGO team can have many players, more than required to play on the stage. A game between 2 teams is term a match. A CSGO team often plays best of three against each other and this process is term a competition. A tournament must consist of many competitions between the attending teams. A tournament must choose one map pool which contains 7 different maps out of the list.

Initially, this application only has 8 maps include team, player, sponsor, tournament, competition, match, map and map pool. After applying conceptual design, the number of tables is settled onto 10. Sponsor table is added onto the list since a team can have multiple sponsors and sponsors can sponsor many teams and the team attending table also suits the idea. A tournament is joined by many teams, and a team can join multiple tournaments.

The team table contains 8 attributes, team id, team name, location, region, start date, disband, running time and total winning. Team ID is a unique attribute that identifies each team. Start date declares the date when the team was found. Disband declares the date the team dissolves. Since not all the team had a financial crisis, this field can be a NULL. The running time declares the total amount of days between and start date and disband date and calculated automatically if the disband value is updated by the database manager using a trigger mechanism. All the attribute in this team don't contain composite value and each attribute is solely dependent on the primary key. A check statements that prevent if the date of disbanding is earlier than the date of founding the team.

The Sponsor team contains 3 attributes, sponsor id, sponsor name and sponsor web link. Sponsor id is a unique attribute that identifies itself. All the attribute in this team don't contain composite value and each attribute is solely dependent on the primary key.

The sponsor list table represents the M: N relationship exists between team and sponsor. The team id and sponsor id are foreign keys that reference team id and sponsor id in both team table and sponsor table respectively.

The player table contains 11 attributes, player id, team id, player role, player nickname, player forename, player surname, age, nationality, join team, quit team and onboard time. The player id is a unique attribute that identifies each tuple. The team id is a foreign key references the team id in the team table. Since some players stay in the team the whole time, the quit team attribute can be a NULL for those players. The value of onboard of each tuple is automatically calculated if the corresponding quit team time is updated. A check statement is needed to prevent that the date of quit team is earlier than the date of

join team. The relationship between team and player is 1:N, a team can have multiple players, but multiple players can only join one team, hence the team id attribute can be added into the 1 side, the player side. A check statement, CONSTRAINT CHECK (PLAYER_ROLE IN ('IGL','Entry Fragger','Lurker', 'Supporter','AWPER')), ensures that the play role attribute assigned to each tuple is one of the given one. All the attribute in this team don't contain composite value and each attribute is solely dependent on the primary key.

The tournament table contains 9 attributes, tournaments id, tournament name, start date, end date, prize pool, location, champion, MVP player and map pool. Tournament id is a unique attribute that identifies each tournament. A check statement is needed to ensure that the start date is earlier than the end date. The champion, MVP player and map pool are foreign keys that reference team id, player id and map pool id from team, player and map pool table respectively. Since a tournament has the chance of ongoing and the champion and MVP player of the tournament haven't been decided, the value of those two attributes can be NULL. All the attribute in this team don't contain composite value and each attribute is solely dependent on the primary key.

The team attending table represents the M: N relationship exists between teams and tournaments. Teams can join multiple tournaments and tournaments are joined by multiple teams. The team id and tournaments id are foreign keys that reference the team id and tournament id in the team table and tournament id respectively.

The map table contains 6 attributes, map id, map name, terrorist, counter-terrorist, location and theme. The map id is a unique attribute that identifies each map. All the attribute in this team don't contain composite value and each attribute is solely dependent on the primary key.

The map pool contains 8 attributes, the map pool id and 7 different map id. The relationship between the map and the map pool is tricky. Intuitively, it seems like an M: N relationship, since a map pool contains multiple maps and maps can be included into different map pools. However, CSGO declares that a tournament can only have 7 maps to play. This regulates the range of choices and reduces the relationship from M: N to 7 to N which can be realized in a single table. 20 checks statements are needed to ensure that duplicate maps don't add into a single map pool. All the attribute in this team don't contain composite value and each attribute is solely dependent on the primary key.

The competition table contains 6 attributes, the competition id, team A, team B and the competition winner, competition score and the tournament id. The competition id is a unique attribute that identifies itself. Since the competition is not yet finished, the competition score and competition winner can contain NULL values. Team A and Team B are foreign keys that reference the team id in the team table. Since the relationship between competition and tournament is N:1, a tournament contains multiple competition and multiple competitions belong to a tournament, thus the tournament id can be added into the competition table. A semantic constraint cannot be addressed is that both team A and team B have to be teams that attend this tournament. All the attribute in this team don't contain composite value and each attribute is solely dependent on the primary key.

The match table contains 13 attributes, match id, competition id, tournament id, team a, team b, score, map played, most kills, most damaged, most assistant, most AWP kills, most first kill and Best rating player. Most kills, most damaged, most assistant, most AWP

kills, most first kill and Best rating player are foreign keys that reference player id in the player table. Since the relationship between match and tournament & competition is 1:N relationship, a tournament and competition can contain multiple matches and a match only belongs to a competition and a tournament, thus the competition id and tournament id can be added into the match table. Some semantic constraints cannot be addressed are that both team A and team B have to be teams that attend this competition, players who get all those titles have to be members of teams join the competition and the map they play have to be one map out of the map pool chosen by the tournaments. All the attribute in this team don't contain composite value and each attribute is solely dependent on the primary key.

Security Policy

The role-based access control is implemented in this application. The purpose of this application fulfils the public need of viewing professional CSGO matches details. This simple requirement breaks the user of this application into two types, content uploader(Supplier) and people who read the data(Consumer).In this application, there is a supplier manager who is in charge of hiring new suppliers, granting proper privileges to those and revoking from them. The responsibility of the supplier is limited in the field of updating columns, inserting and delete tuple of the existing tables.

This supplier manager has all the control over the database, in order to do so, execute this statement by the database administrator:

```
CREATE ROLE SUPPLIER_MANAGER;
GRANT ALL ON csgostats.* TO SUPPLIER_MANAGER WITH GRANT OPTION;
```

From now then, this supplier manager can grant privileges to suppliers who work for him.

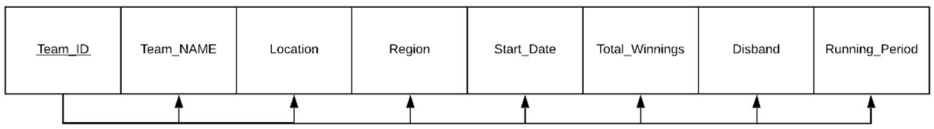
```
CREATE ROLE SUPPLIER;
GRANT INSERT,DELETE,UPDATE TO SUPPLIER.
GRANT SELECT TO SUPPLIER.
```

Since all the information stored in the database are public, there is no need to create view to limit the amount of data can be seen by the consumer. The following privileges suffice for consumers.

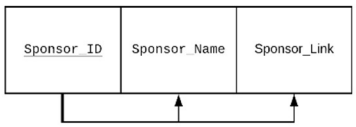
```
CREATE ROLE CONSUMER;
GRANT SELECT TO CONSUMER;
```

Functional Dependency

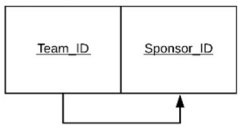
Team



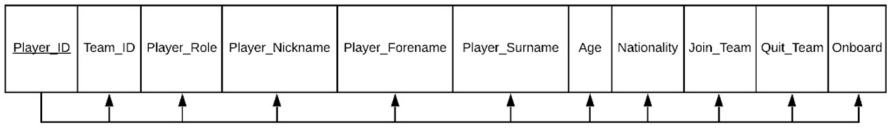
Sponsor



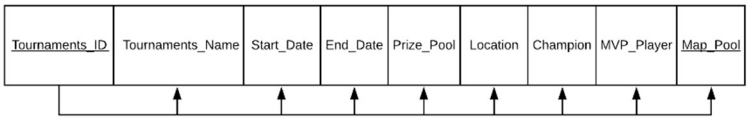
Sponsor_Table



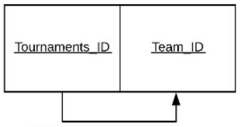
Player



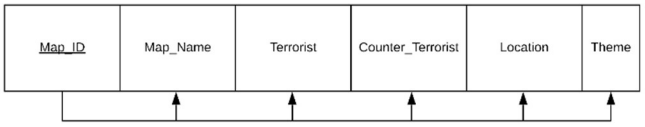
Tournaments



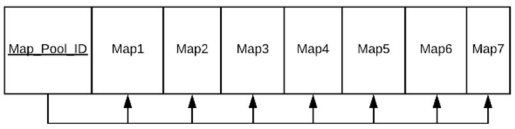
Team_Attending



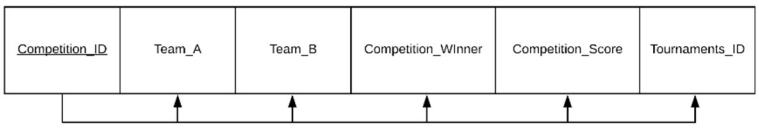
Map



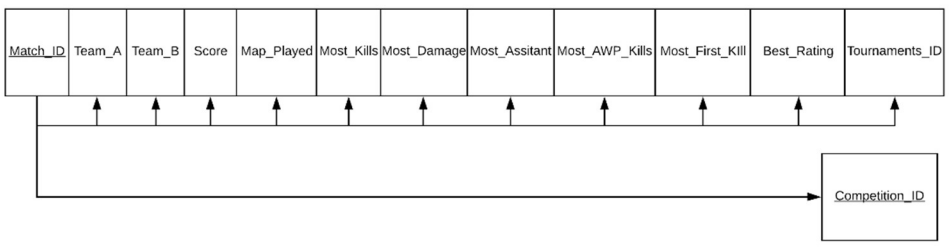
Map_Pool

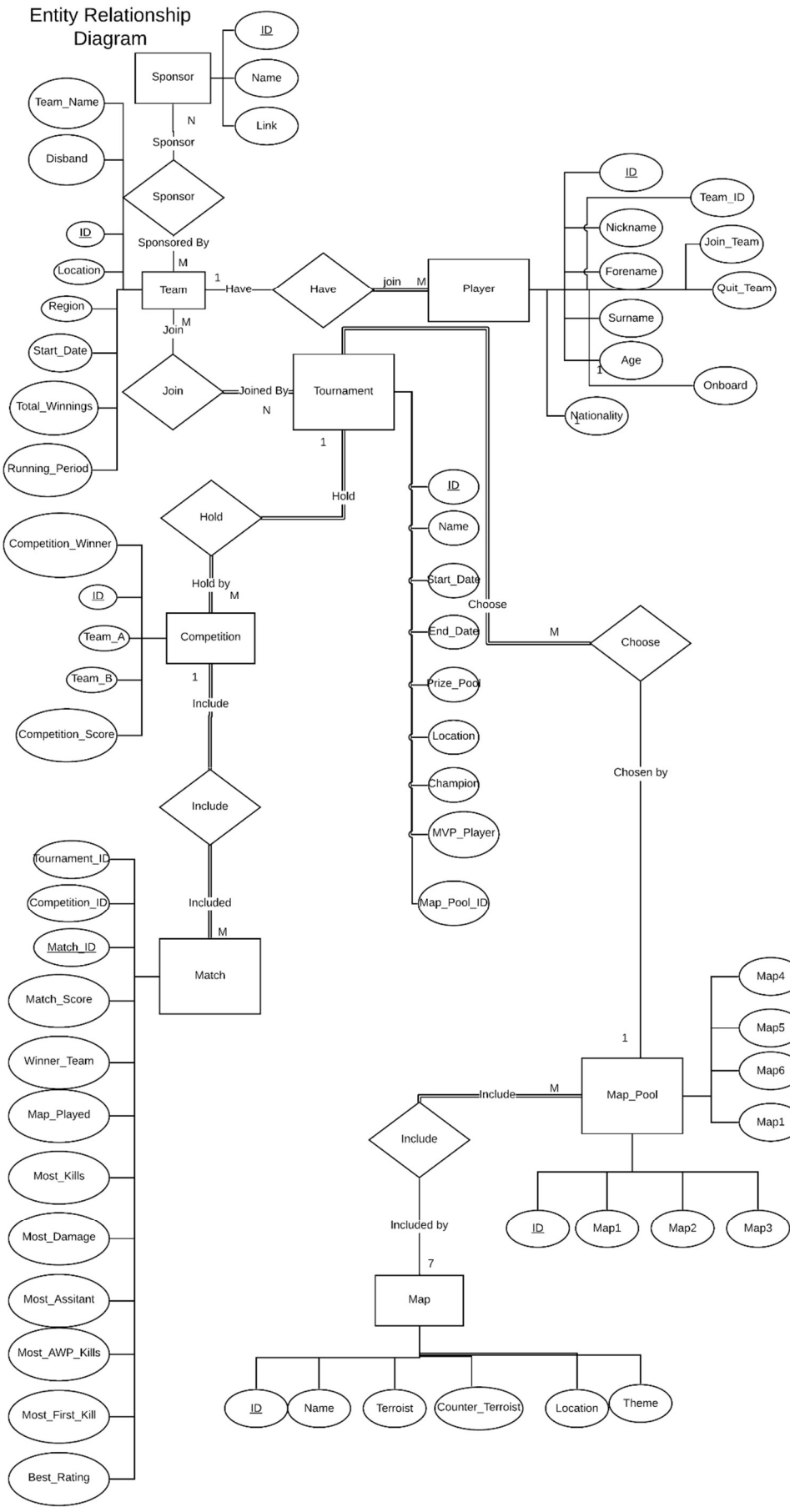


Competition

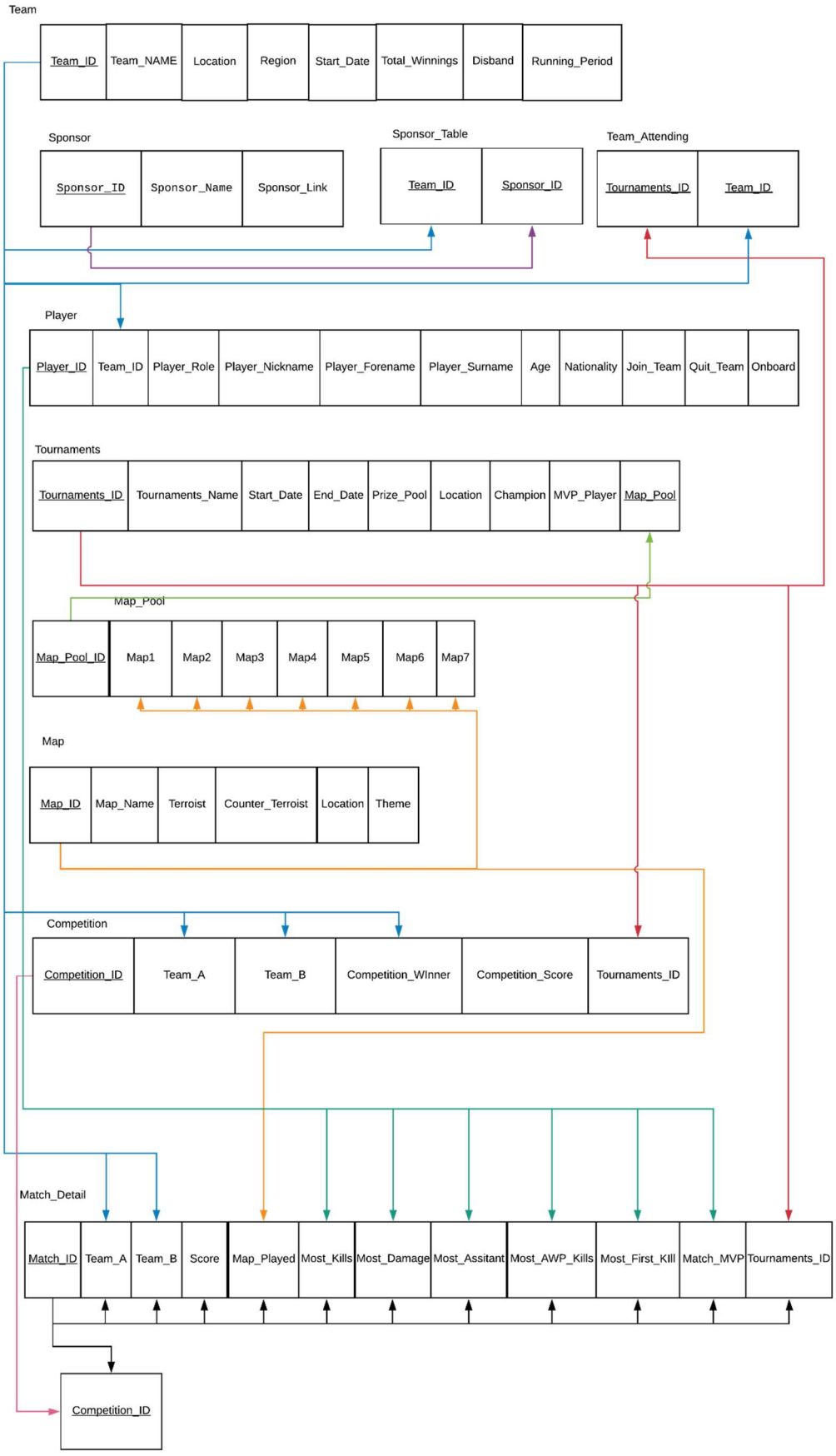


Match_Detail




Entity Relationship Diagram
Sponsor
ID
Name
Link
N
Sponsor
Team_Name
Disband
Sponsor
Sponsored By
M
ID
Location
Region
Start_Date
Total_Winnings
Running_Period
Team
1
Have
Have
join
M
Player
ID
Team_ID
Nickname
Join_Team
Forename
Quit_Team
Surname
Age
Onboard
Nationality
1
Join
M
Join
Joined By
N
Tournament
1
Hold
Hold
ID
Name
Start_Date
End_Date
Prize_Pool
Location
Champion
MVP_Player
Map_Pool_ID
Competition_Winner
ID
Team_A
Team_B
Competition_Score
Hold by
M
Competition
1
Include
Include
Choose
M
Choose
Chosen by
Tournament_ID
Competition_ID
Match_ID
Match_Score
Winner_Team
Map_Played
Most_Kills
Most_Damage
Most_Assitant
Most_AWP_Kills
Most_First_Kill
Best_Rating
Included
M
Match
1
Include
M
Map_Pool
Map4
Map5
Map6
Map1
Include
Included by
7
Map
ID
Map1
Map2
Map3
ID
Name
Terroist
Counter_Terroist
Location
Theme

Functional Dependency



Trigger Implementation

```
CREATE TRIGGER CALCULATE_RUNNING_PERIOD
BEFORE UPDATE ON TEAM
FOR EACH ROW
 SET NEW.RUNNING_PERIOD = TIMESTAMPDIFF(DAY,OLD.START_DATE,NEW.DISBAND);

CREATE TRIGGER CALCULATE_PLAYER_ONBOARD
BEFORE UPDATE ON PLAYER
FOR EACH ROW
SET NEW.ONBOARD = TIMESTAMPDIFF(DAY,OLD.JOIN_TEAM,NEW.QUIT_TEAM);
```

The usage of the first triggers is to automatically calculate the amount time of a team join the CSGO professional scene actively before it disbands and store the value into the running period attribute slot of the corresponding tuple.

The second one calculates the onboard time of a player in a team and automatically store the value into the onboard attribute slot of the player entry in the player table.

DATABASE IMPLEMENTATION

```
DROP DATABASE csgoStats;
CREATE DATABASE csgoStats;
use csgoStats;
CREATE TABLE TEAM(
	TEAM_ID Integer NOT NULL UNIQUE,
	TEAM_NAME VARCHAR(255) NOT NULL,
	LOCATION VARCHAR(255) NOT NULL,
	REGION VARCHAR(255) NOT NULL,
	START_DATE DATE NOT NULL,
	DISBAND DATE,
	RUNNING_PERIOD INT,
	TOTAL_WINNING Integer NOT NULL,
	PRIMARY KEY (TEAM_ID)

);
CREATE TABLE SPONSOR(
	SPONSOR_ID INTEGER NOT NULL UNIQUE,
	SPONSOR_NAME VARCHAR(255) NOT NULL,
	SPONSOR_LINK VARCHAR(255) NOT NULL,
	PRIMARY KEY (SPONSOR_ID)
```

```
);
CREATE TABLE SPONSOR_TABLE(
    TEAM_ID INTEGER NOT NULL,
    SPONSOR_ID INTEGER NOT NULL,
    FOREIGN KEY (TEAM_ID) REFERENCES TEAM(TEAM_ID),
    FOREIGN KEY (SPONSOR_ID) REFERENCES SPONSOR(SPONSOR_ID),
    PRIMARY KEY(TEAM_ID,SPONSOR_ID)
);
CREATE TABLE PLAYER(
    PLAYER_ID INTEGER NOT NULL UNIQUE,
    TEAM_ID INTEGER NOT NULL,
    PLAYER_ROLE VARCHAR(255),
    NICKNAME VARCHAR(255) NOT NULL,
    FORENAME VARCHAR(255) NOT NULL,
    SURNAME VARCHAR(255) NOT NULL,
    AGE INTEGER NOT NULL,
    NATIONALITY VARCHAR(255) NOT NULL,
    JOIN_TEAM DATE NOT NULL,
    QUIT_TEAM DATE,
    ONBOARD INTEGER,
    PRIMARY KEY (PLAYER_ID),
    FOREIGN KEY (TEAM_ID) REFERENCES TEAM(TEAM_ID),
    CONSTRAINT CHECK (PLAYER_ROLE IN ('IGL','Entry
Fragger','Lurker','Supporter','AWPER'))
);
CREATE TABLE MAP(
    MAP_ID INTEGER NOT NULL UNIQUE,
    MAP_NAME VARCHAR(255) NOT NULL,
    TERROIST VARCHAR(255) NOT NULL,
    COUNTER_TERROIST VARCHAR(255) NOT NULL,
    LOCATION VARCHAR(255) NOT NULL,
    THEME VARCHAR(255) NOT NULL,
    PRIMARY KEY(MAP_ID)
);
CREATE TABLE MAP_POOL(
    MAP_POOL_ID INTEGER NOT NULL,
    MAP1 INTEGER NOT NULL,
    MAP2 INTEGER NOT NULL,
    MAP3 INTEGER NOT NULL,
    MAP4 INTEGER NOT NULL,
    MAP5 INTEGER NOT NULL,
    MAP6 INTEGER NOT NULL,
    MAP7 INTEGER NOT NULL,
    PRIMARY KEY(MAP_POOL_ID),
```

```
    FOREIGN KEY (MAP1) REFERENCES MAP(MAP_ID),
    FOREIGN KEY (MAP2) REFERENCES MAP(MAP_ID),
    FOREIGN KEY (MAP3) REFERENCES MAP(MAP_ID),
    FOREIGN KEY (MAP4) REFERENCES MAP(MAP_ID),
    FOREIGN KEY (MAP5) REFERENCES MAP(MAP_ID),
    FOREIGN KEY (MAP6) REFERENCES MAP(MAP_ID),
    FOREIGN KEY (MAP7) REFERENCES MAP(MAP_ID),
    CHECK (MAP1 != MAP2),CHECK (MAP1 != MAP3),CHECK (MAP1 != MAP4),CHECK
(MAP1 != MAP5),CHECK (MAP1 != MAP6),CHECK (MAP1 != MAP7),
    CHECK (MAP2 != MAP3),CHECK (MAP2 != MAP4),CHECK (MAP2 != MAP5),CHECK
(MAP2 != MAP6),CHECK (MAP2 != MAP7),
    CHECK (MAP3 != MAP4),CHECK (MAP3 != MAP5),CHECK (MAP3 != MAP6),CHECK
(MAP3 != MAP7),
    CHECK (MAP4 != MAP5),CHECK (MAP4 != MAP6),CHECK (MAP4 != MAP7),
    CHECK (MAP5 != MAP6),CHECK (MAP5 != MAP6),
    CHECK (MAP6 != MAP7)
);
CREATE TABLE TOURNAMENT(
    TOURNAMENT_ID INTEGER NOT NULL UNIQUE,
    TOURNAMENT_NAME VARCHAR(255) NOT NULL,
    START_DATE DATE NOT NULL,
    END_DATE DATE NOT NULL,
    PRIZE_POOL INTEGER NOT NULL,
    LOCATION VARCHAR(255) NOT NULL,
    CHAMPION INTEGER,
    MVP_PLAYER INTEGER,
    MAP_POOL INTEGER NOT NULL,
    CHECK (START_DATE <= END_DATE),
    PRIMARY KEY (TOURNAMENT_ID),
    FOREIGN KEY (CHAMPION) REFERENCES TEAM(TEAM_ID),
    FOREIGN KEY (MVP_PLAYER) REFERENCES PLAYER (PLAYER_ID),
    FOREIGN KEY (MAP_POOL) REFERENCES MAP_POOL (MAP_POOL_ID)
);
CREATE TABLE TEAM_ATTENDING(
    TOURNAMENT_ID INTEGER NOT NULL,
    TEAM_ID INTEGER NOT NULL,
    FOREIGN KEY (TOURNAMENT_ID) REFERENCES TOURNAMENT(TOURNAMENT_ID),
    FOREIGN KEY (TEAM_ID) REFERENCES TEAM(TEAM_ID)
);
CREATE TABLE COMPETITION(
    TOURNAMENT_ID INTEGER NOT NULL,
    COMPETITION_ID INTEGER NOT NULL UNIQUE,
    TEAM_A INTEGER NOT NULL,
    TEAM_B INTEGER NOT NULL,
```

```
    COMPETITION_WINNER INTEGER,
    COMPETITION_SCORE VARCHAR(255),
    FOREIGN KEY (TEAM_A) REFERENCES TEAM(TEAM_ID),
    FOREIGN KEY (TEAM_B) REFERENCES TEAM(TEAM_ID),
    FOREIGN KEY (COMPETITION_WINNER) REFERENCES TEAM(TEAM_ID),
    PRIMARY KEY (COMPETITION_ID)
);
CREATE TABLE MATCH_DETAIL(
    TOURNAMENT_ID INTEGER NOT NULL,
    COMPETITION_ID INTEGER NOT NULL,
    MATCH_ID INTEGER NOT NULL UNIQUE,
    MATCH_WINNER INTEGER NOT NULL,
    MATCH_SCORE VARCHAR(255) NOT NULL,
    MAP_PLAYED INTEGER NOT NULL NOT NULL,
    MOST_KILLS INTEGER NOT NULL,
    MOST_DAMAGE INTEGER NOT NULL,
    MOST_ASSITANT INTEGER NOT NULL,
    MOST_AWP_KILLS INTEGER NOT NULL,
    MOST_FIRST_KILLS INTEGER NOT NULL,
    BEST_RATING INTEGER NOT NULL,
    FOREIGN KEY (MATCH_WINNER) REFERENCES TEAM(TEAM_ID),
    FOREIGN KEY (MAP_PLAYED) REFERENCES MAP(MAP_ID),
    FOREIGN KEY (MOST_KILLS) REFERENCES PLAYER(PLAYER_ID),
    FOREIGN KEY (MOST_DAMAGE) REFERENCES PLAYER(PLAYER_ID),
    FOREIGN KEY (MOST_ASSITANT) REFERENCES PLAYER(PLAYER_ID),
    FOREIGN KEY (MOST_AWP_KILLS) REFERENCES PLAYER(PLAYER_ID),
    FOREIGN KEY (MOST_FIRST_KILLS) REFERENCES PLAYER(PLAYER_ID),
    FOREIGN KEY (BEST_RATING) REFERENCES PLAYER(PLAYER_ID),
    PRIMARY KEY (MATCH_ID)
);
```

DATA INSERTION

```
 use csgoStats;
# Insert TEAM Informatiom
INSERT INTO TEAM VALUE (1,'Astralis',"Denmark","Europe",'2016-01-
 18',NULL,NULL,7452634);
INSERT INTO TEAM VALUE (2,'AVANGAR',"Kazakhstan","CIS",'2017-07-
 17',NULL,NULL,727497);
INSERT INTO TEAM VALUE (3,'Renegades',"United States","North America",'2015-03-
 09',NULL,NULL,872098);
INSERT INTO TEAM VALUE (4,'NRG',"United States","North America",'2016-01-
 26',NULL,NULL,1088900);
INSERT INTO TEAM VALUE (5,'Natus Vincere',"Ukraine","CIS",'2012-11-
 04',NULL,NULL,3505092);
INSERT INTO TEAM VALUE (6,'ENCE',"Finland","Europe",'2013-04-13',NULL,NULL,878935);
```

```
INSERT INTO TEAM VALUE (7,'Vitality',"France","Europe",'2018-10-08',NULL,NULL,706805);
INSERT INTO TEAM VALUE (8,'Team Liquid',"United States","North America",'2015-01-
 13',NULL,NULL,4160812);
INSERT INTO TEAM VALUE (9,'mousesports',"Germany","Europe",'2012-08-
 16',NULL,NULL,2698417);
INSERT INTO TEAM VALUE (10,'CR4ZY',"Germany","Europe",'2017-10-
 20',NULL,NULL,275632);
INSERT INTO TEAM VALUE (11,'G2',"France","Europe",'2015-02-11',NULL,NULL,1787684);
INSERT INTO TEAM VALUE (12,'Faze',"Europe","Europe",'2016-01-20',NULL,NULL,3357347);
INSERT INTO TEAM VALUE (13,'North',"Denmark","Europe",'2017-01-
 13',NULL,NULL,1028250);
INSERT INTO TEAM VALUE (14,'MIBR',"Brazil","North America",'2018-06-
 23',NULL,NULL,922290);
INSERT INTO TEAM VALUE (15,'NiP',"Sweden","Europe",'2000-06-01',NULL,NULL,2830282);
INSERT INTO TEAM VALUE (16,'DreamEaster',"Russia","CIS",'2017-01-
 01',NULL,NULL,61908);

# Insert the sponsor information of the top 4 teams
#Astralis
INSERT INTO SPONSOR VALUE (1,'Unibet','https://www.unibet.com/');
INSERT INTO SPONSOR VALUE (2,'JACK & JONES','https://www.jackjones.com/ie/en/home');
INSERT INTO SPONSOR VALUE (3,'Turtle Beach','https://www.turtlebeach.com/');
INSERT INTO SPONSOR VALUE (4,'Logitech G','https://www.logitechg.com/en-roeu');
INSERT INTO SPONSOR VALUE (5,'OMEN by
 HP','https://www8.hp.com/us/en/gaming/omen.html');
INSERT INTO SPONSOR VALUE (6,'NOCCO','https://nocco.com/');
INSERT INTO SPONSOR VALUE (7,'Secretlab','https://secretlabchairs.co.uk/');
INSERT INTO SPONSOR_TABLE VALUE (1,1);
INSERT INTO SPONSOR_TABLE VALUE (1,2);
INSERT INTO SPONSOR_TABLE VALUE (1,3);
INSERT INTO SPONSOR_TABLE VALUE (1,4);
INSERT INTO SPONSOR_TABLE VALUE (1,5);
INSERT INTO SPONSOR_TABLE VALUE (1,6);
INSERT INTO SPONSOR_TABLE VALUE (1,7);
#AVANGAR
INSERT INTO SPONSOR VALUE (8,'GG.BET','https://gg.bet/en/betting');
INSERT INTO SPONSOR VALUE (9,'LEGION','https://legion.lenovo.com/');
INSERT INTO SPONSOR_TABLE VALUE (2,8);
INSERT INTO SPONSOR_TABLE VALUE (2,9);
#RENEDAGES
INSERT INTO SPONSOR VALUE (10,'NVIDIA','https://www.nvidia.com/en-gb/geforce/');
INSERT INTO SPONSOR VALUE (11,'HyperX','https://www.hyperxgaming.com/en');
INSERT INTO SPONSOR VALUE (12,'Alienware','https://www.dell.com/en-
 us/gaming/alienware');
```

```
INSERT INTO SPONSOR VALUE (13,'Twitch','https://www.twitch.tv/');
INSERT INTO SPONSOR VALUE
 (14,'Champion','https://www.championstore.com/en?utm_source=USReferral&utm_campai
 gn=akamaiUSredirect');
INSERT INTO SPONSOR VALUE (15,'RESPAWN','https://respawnproducts.com/');
INSERT INTO SPONSOR_TABLE VALUE (3,10);
INSERT INTO SPONSOR_TABLE VALUE (3,11);
INSERT INTO SPONSOR_TABLE VALUE (3,12);
INSERT INTO SPONSOR_TABLE VALUE (3,13);
INSERT INTO SPONSOR_TABLE VALUE (3,14);
INSERT INTO SPONSOR_TABLE VALUE (3,15);
#NRG
INSERT INTO SPONSOR VALUE (16,'Events DC','http://www.eventsdc.com/');
INSERT INTO SPONSOR VALUE (17,'Sony Crackle','https://www.sonycrackle.com/out-of-
 region.html');
INSERT INTO SPONSOR VALUE (18,'OPSEAT','https://opseat.com/');
INSERT INTO SPONSOR VALUE (19,'Nighthawk Pro
 Gaming','https://www.netgear.com/gaming/');
INSERT INTO SPONSOR VALUE (20,'Qualcomm
 Snapdragon','https://www.qualcomm.com/snapdragon');
INSERT INTO SPONSOR VALUE (21,'Republic of Gamers','https://rog.asus.com/');
INSERT INTO SPONSOR VALUE (22,'Enjin Coin','https://enjin.io/');
INSERT INTO SPONSOR VALUE (23,'Cal Esports','https://recsports.berkeley.edu/esports/');
INSERT INTO SPONSOR VALUE (24,'ibuypower','https://www.ibuypower.com/');
INSERT INTO SPONSOR VALUE (25,'雷火电竞','http://www.e8552.com/');
INSERT INTO SPONSOR_TABLE VALUE (4,16);
INSERT INTO SPONSOR_TABLE VALUE (4,4);
INSERT INTO SPONSOR_TABLE VALUE (4,17);
INSERT INTO SPONSOR_TABLE VALUE (4,18);
INSERT INTO SPONSOR_TABLE VALUE (4,13);
INSERT INTO SPONSOR_TABLE VALUE (4,19);
INSERT INTO SPONSOR_TABLE VALUE (4,20);
INSERT INTO SPONSOR_TABLE VALUE (4,21);
INSERT INTO SPONSOR_TABLE VALUE (4,22);
INSERT INTO SPONSOR_TABLE VALUE (4,23);
INSERT INTO SPONSOR_TABLE VALUE (4,24);
INSERT INTO SPONSOR_TABLE VALUE (4,25);


# Insert Only the top 4 team players information
#Astralis
INSERT INTO PLAYER VALUE (1,1,'AWPER','dev1ce','Nicolai','Reedtz',24,'Denmark','2016-01-
 18',NULL,NULL);
INSERT INTO PLAYER VALUE (2,1,'Entry
```

```
  Fragger','dupreeh','Peter','Rasmussen',26,'Denmark','2016-01-18',NULL,NULL);
INSERT INTO PLAYER VALUE (3,1,'IGL','gla1ve','Lukas','Rossander',24,'Denmark','2016-10-
  24',NULL,NULL);
INSERT INTO PLAYER VALUE (4,1,'Lurker','Xyp9x','Andreas','Hojsleth',24,'Denmark','2016-01-
  18',NULL,NULL);
INSERT INTO PLAYER VALUE (5,1,'Entry Fragger','Magisk','Emil','Reif',21,'Denmark','2018-02-
  7',NULL,NULL);
#AVANGER
INSERT INTO PLAYER VALUE (6,2,'IGL','jame','Dzhami','Ali',21,'Russia','2017-08-
  09',NULL,NULL);
INSERT INTO PLAYER VALUE
  (7,2,'Supporter','AdreN','Dauren','Kystaubayev',29,'Kazakhstan','2019-06-25',NULL,NULL);
INSERT INTO PLAYER VALUE (8,2,'Lurker','buster','Timur','Tulepov',19,'Kazakhstan','2017-07-
  17',NULL,NULL);
INSERT INTO PLAYER VALUE (9,2,'Entry
  Fragger','SANJI','Sanjar','Kuliev',21,'Uzbekistan','2019-04-12',NULL,NULL);
INSERT INTO PLAYER VALUE (10,2,'Entry
  Fragger','qikert','Alexey','Golubev',20,'Kazakhstan','2017-07-17',NULL,NULL);
#Renedages #quit    at 2019-10-31
INSERT INTO PLAYER VALUE (11,3,'IGL','AZR','Aaron','Ward',27,'Australia','2015-06-
  19',NULL,NULL);
INSERT INTO PLAYER VALUE (12,3,'Supporter','Liazz','jay','Tregillgas',22,'Australia','2018-09-
  30',NULL,NULL);
INSERT INTO PLAYER VALUE (13,3,'Lurker','jkaem','joakim','Myrbostad',25,'Norway','2018-
  02-08',NULL,NULL);
INSERT INTO PLAYER VALUE (14,3,'Entry Fragger','jks','justin','Savage',23,'Australia','2015-
  06-19',NULL,NULL);
INSERT INTO PLAYER VALUE (15,3,'AWPER','Gratisfaction','Sean','Kaiwai',23,'New
  Zealand','2018-09-30',NULL,NULL);
#NRG
INSERT INTO PLAYER VALUE (16,4,'IGL','stanislaw','Peter','Jarguz',25,'Canada','2019-06-
  11',NULL,NULL);
INSERT INTO PLAYER VALUE (17,4,'Entry Fragger','Ethan','Ethan','Arnold',19,'United
  States','2018-01-15',NULL,NULL);
INSERT INTO PLAYER VALUE (18,4,'Entry Fragger','tarik','tarik','Celik',23,'United States','2019-
  02-28',NULL,NULL);
INSERT INTO PLAYER VALUE (19,4,'Entry Fragger','Brehze','Vincent','Cayonte',21,'United
  States','2016-12-19',NULL,NULL);
INSERT INTO PLAYER VALUE (20,4,'AWPER','CeRq','Cvetelin','Dinitrov',19,'Bulgaria','2017-
  10-09',NULL,NULL);

#MAP
INSERT INTO MAP VALUE (1,'Dust II','Elite Crew','SEAL TEAM 6','Morocco','Desert');
INSERT INTO MAP VALUE (2,'Inferno','Separatist','SAS','Italty','Urban');
```

```
INSERT INTO MAP VALUE (3,'Mirage','Elite Crew','SAS','Morocco','Suburban');
INSERT INTO MAP VALUE (4,'Nuke','Phoenix','FBI','Northeastern United States','Industrial');
INSERT INTO MAP VALUE (5,'Overpass','Phoenix faction','GSG9','Berlin','Urban');
INSERT INTO MAP VALUE (6,'Train','Balkan','SEAL TEAM 6','Russia','Industrial');
INSERT INTO MAP VALUE (7,'Vertigo','Professionals','FBI','Skycraper Duilding','Urban');
INSERT INTO MAP VALUE (8,'Cobblestone','Phoenix fraction','GIGN','France','Castle');
INSERT INTO MAP VALUE (9,'Cache','Phoenix Connexion','GIGN','Ukraine','Industrial');
#MAP_POLL
INSERT INTO MAP_POOL VALUE (1,1,2,3,4,5,6,7);
INSERT INTO MAP_POOL VALUE (2,1,2,3,4,5,6,9);
#TOURNAMENT
INSERT INTO TOURNAMENT VALUE (1,'StarLadder Major Berlin 2019','2019-08-12','2019-
  09-23',1000000,'Berlin',1,1,1);
INSERT INTO TOURNAMENT VALUE (2,'IEM Katowice Major 2019','2019-02-13','2019-03-
  03',1000000,'Katowice',1,5,2);
#TEAM_ATTENDING
INSERT INTO TEAM_ATTENDING VALUE(1,1);
INSERT INTO TEAM_ATTENDING VALUE(1,2);
INSERT INTO TEAM_ATTENDING VALUE(1,3);
INSERT INTO TEAM_ATTENDING VALUE(1,4);
INSERT INTO TEAM_ATTENDING VALUE(1,5);
INSERT INTO TEAM_ATTENDING VALUE(1,6);
INSERT INTO TEAM_ATTENDING VALUE(1,7);
INSERT INTO TEAM_ATTENDING VALUE(1,8);
INSERT INTO TEAM_ATTENDING VALUE(1,9);
INSERT INTO TEAM_ATTENDING VALUE(1,10);
INSERT INTO TEAM_ATTENDING VALUE(1,11);
INSERT INTO TEAM_ATTENDING VALUE(1,12);
INSERT INTO TEAM_ATTENDING VALUE(1,13);
INSERT INTO TEAM_ATTENDING VALUE(1,14);
INSERT INTO TEAM_ATTENDING VALUE(1,15);
INSERT INTO TEAM_ATTENDING VALUE(1,16);
#COMPETITION     TOURNAMENT_ID,    COMPETITION_ID, TEAM_A, TEAM_B,
  COMPETITION_WINNER, COMPETITION_SCORE
INSERT INTO COMPETITION VALUE(1,1,1,2,1,'2:0');
INSERT INTO COMPETITION VALUE(1,2,1,4,1,'2:0');
INSERT INTO COMPETITION VALUE(1,3,2,3,2,'2:0');
#MATCHES
#TOURNAMENT_ID, COMPETITION_ID, MATCH_ID, MATCH_WINNER, MATCH_SCORE,
  MAP_PLAYED, MOST_KILLS, MOST_DAMAGE, MOST_ASSITANT, MOST_AWP_KILLS,
  MOST_FIRST_KILLS, BEST_RATING
INSERT INTO MATCH_DETAIL VALUE(1,1,1,1,'16:6',2,1,1,7,1,1,1);
INSERT INTO MATCH_DETAIL VALUE(1,1,2,1,'16:5',1,1,1,3,1,1,1);
INSERT INTO MATCH_DETAIL VALUE(1,2,3,1,'16:10',6,4,4,3,20,19,19);
```

```
INSERT INTO MATCH_DETAIL VALUE(1,2,4,1,'16:9',5,19,19,4,20,19,19);
INSERT INTO MATCH_DETAIL VALUE(1,3,5,2,'22:19',3,14,14,7,6,6,6);
INSERT INTO MATCH_DETAIL VALUE(1,3,6,2,'16:6',1,10,10,7,6,10,10);
```