

# **CS4D2A: INFORMATION MANAGEMENT SYSTEMS II**

## **PROJECT REPORT**



**Submitted To: Vincent Wade, Seamus Lawless**

**Submitted By:  
Kavin Gupta  
17317505**

## DESCRIPTION:

For This Project, I decided to make a football database. Being a fan of football, I thought this would keep me interested throughout the whole project. I chose this theme to accommodate the league play of Football.

This League play has two distinct regions: North America and Europe.

In these of these regions, teams play with every other team and each game is played in a patch. Regarding the game themselves, there are no account for draws/ties, only wins and losses are taken into account or if there is a tie at the scoreboard.

The aim of the database is therefore to simulate the league play in the two regions and to manage all the data associated with it, which allows searching for specific statistical data, such as which player has better kills, death and assists in a given Champion and which team has more victories in the competition.

For this database, I have several teams, and each team will have several players and coaches. There are 14 entities in my database.

**Person:** Each person has a person id which is the primary key, game\_id for what they play, name and their nationality. This is the general table.

**Team:** Each team has an id, a tag, a name, a date of foundation, a region to which it belongs, five or more players, zero or more coaches / analysts, a total number of wins and losses in the competition, a list of sponsoring brands, such as a list of the games in which it participates.

**Player:** Each player has an associated id, a game id, a name, a nationality, an age (date of birth), a position, a team to which they belong and a list of individual statistics in matches participates (playing champion, kills, deaths, assists, etc ...).

**Coach:** Each coach has an associated id, a game id, a name, a nationality, a role (coach / analyst) and a team to which he belongs.

**Game:** Each game has an identification code, a date (year, month, day, hour and minutes), a duration, two participating teams (one on the blue side of the map, one on the red side), a winner, and one list of statistics associated with each participating player. It also has an associated day and week and the region to which it belongs.

**Champion:** Each champion has associated an id, a name and a position, a list of statistics for the games in which it is chosen, such as a list of patches where it is disabled (due to bugs).

**Day:** Each day has associated one game per team and one week to which it belongs.

**Week:** Each week has associated two games per team, an MVP player, a patch in which the games are played and the region to which the competition belongs.

**Patch:** Each patch has a version associated with it, one or more weeks it is played and a list of champions that are disabled in the competition.

**Region:** It corresponds to the region where the tournament takes place, it can be in the European or North American region, being that they are different championships.

**Brand:** Each brand has associated an id, a name and a list of the teams it sponsors.

**Sponsors:** This is a table which shows which brand is associated with which team. It can be done as a view also but I preferred to keep it as a table. This is just for normalization reasons and easy to grab.

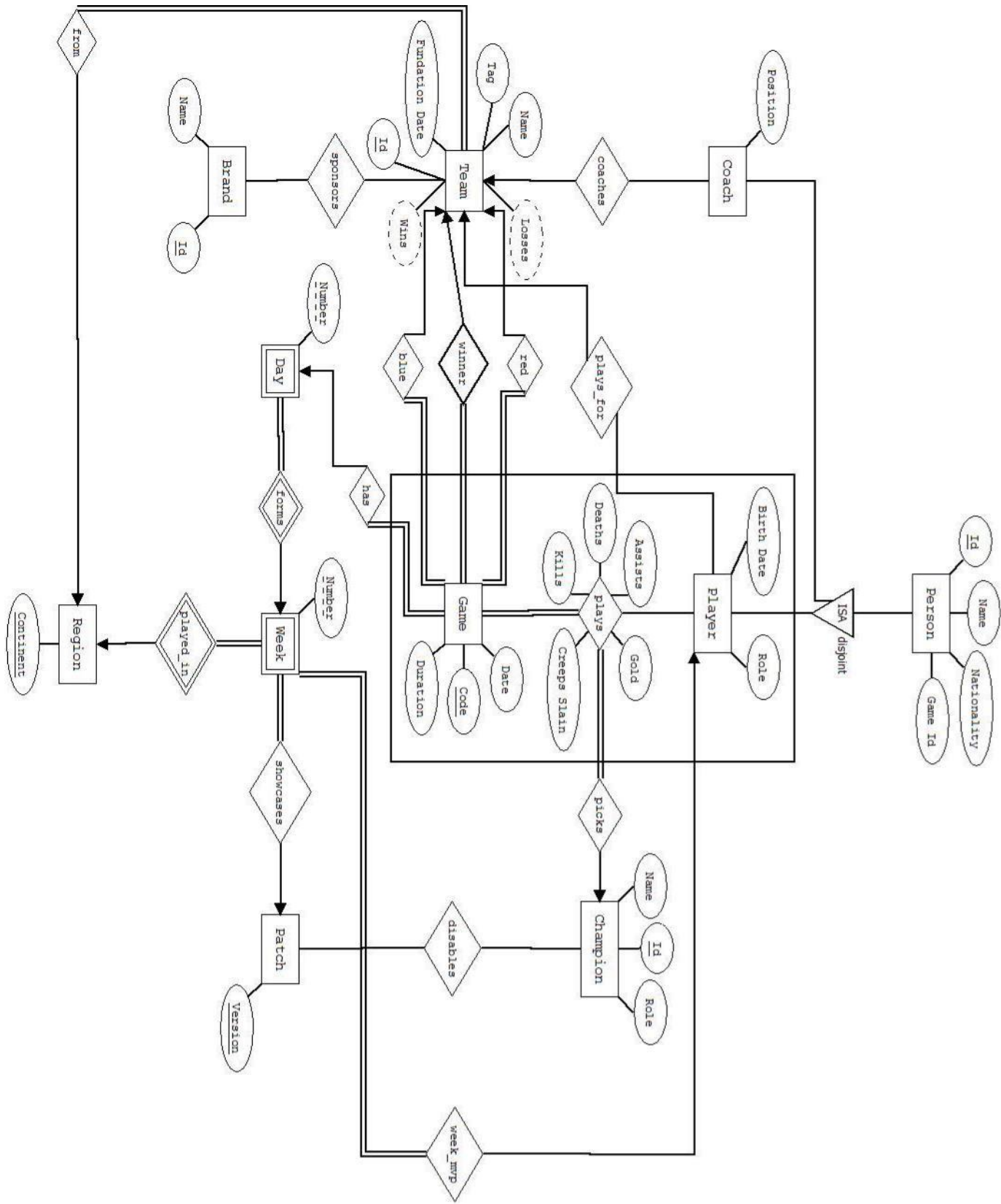
**Disables:** This is the same as the Sponsors table for Patch and Champions containing details about the version and the champion ID.

**Plays:** This contains the data per match given with the game code. It include kills, assists and deaths.

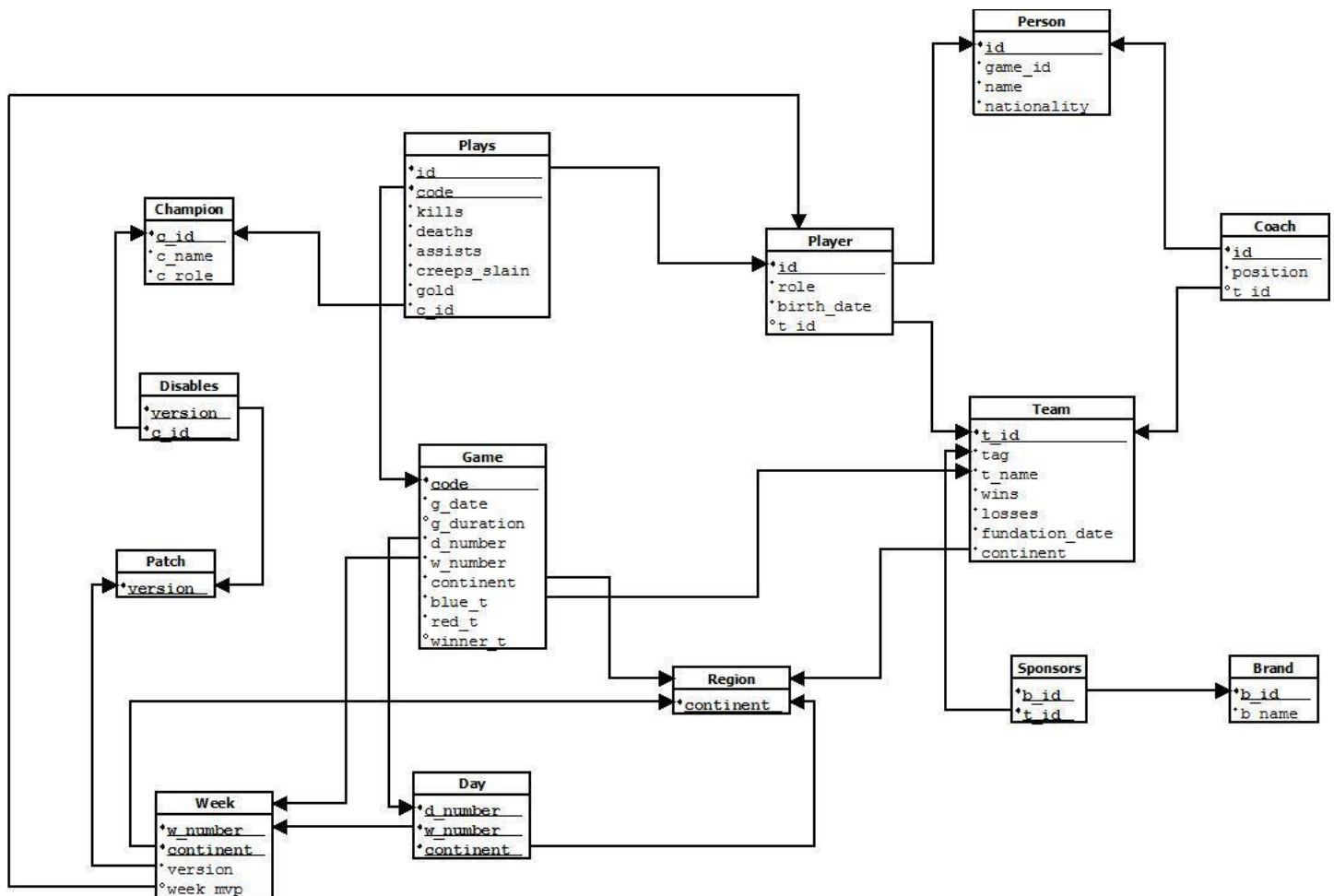
There were some security features added with the aim of giving certain functions to specific people. For example, only coaches should be allowed to add players in a team.

There were also some triggers implemented in order to ensure that the integrity and accuracy of the database is upheld, for example, when a player gets retired, its player id should be deleted from all tables with foreign keys.

ENTITY RELATIONSHIP DIAGRAM:



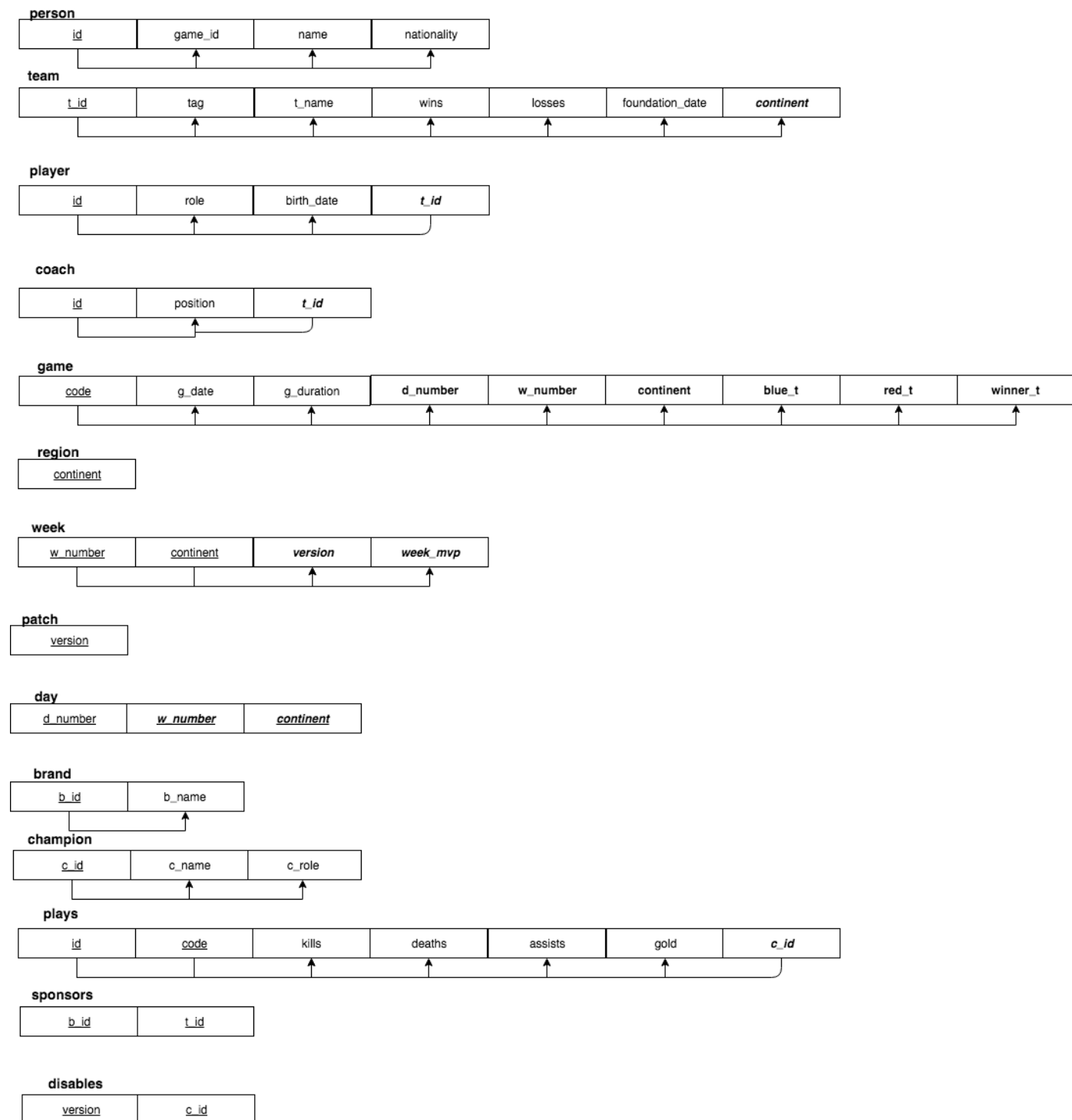
## RELATIONAL SCHEMA:



### Note:

If a value is underlined in the above diagram, it is Primary Key.

## Functional Dependency Diagram.



**Note\*:** The primary keys are underlined while the foreign keys are emboldened.

## List of Normalized Tables

**person** (id, game\_id, name, nationality)

**team** ( t\_id, tag, t\_name, wins, losses, foundation\_date, *continent*)

**player** (id, role, birth\_date, *t\_id*)

**coach** (id, position, *t\_id*)

**region** (continent)

**patch** (version)

**brand** (b\_id, b\_name)

**champion** (c\_id, c\_name, c\_role)

**game** (code, g\_date, g\_duration, *d\_number*, *w\_number*, *continent*, *blue\_t*, *red\_t*, *winner\_t*)

**week** (w\_number, continent, *version*, *week\_mvp*) (can be converted into two tables as its fails second normal form).

**plays** (id, code, kills, deaths, assists, gold, *c\_id*) (can be converted into two tables as its fails second normal form).

The other tables are just composite primary keys. They were made to validate brand and champions tables and one brand can sponsor many teams and similarly there can be multiple champions.

**Note:** The primary keys are underlined while the foreign keys are in italic.

## Semantic Constraints:

Semantic constraints are used to help ensure that a database accurately reflects the real world in structure and content. These are restrictions on the permitted values in database tables. They can be scheme based or explicit constraints. They also can be static or dynamic.

In my database, I have used several semantic constraints. The CHECK constraint and NOT NULL.

The CHECK Constraint specifies conditions on data inserted into a column. Each row inserted into a table must meet these conditions.

The NOT NULL which accounts for columns not having any empty cell.

The columns which are identified as Primary key has NOT NULL and UNIQUE constraint applied by default.

In my database, the Semantic Constraints are:

- CHECK constraint in Region table to check if the continent is Europe or North America otherwise give error.
- CHECK constraint in Team Table to check if the number of wins and losses are greater than equal to 0 and less than equal to matches played.
- NOT NULL constraint in Team table and UNIQUE constraint to make the team tag and name unique.
- CHECK constraint in Player table to limit the roles performed by a player and same in the Coach table.
- CHECK constraint in Week table to account for some negative integers and higher bound. Same in the Day table.
- CHECK constraint in Play table to contain only non-negative integers.
- NOT NULL constraint almost in all columns to make it more real-world experience.



## SECURITY:

Security in a database is an important consideration when designing and implementing a database, since an unsecure database can pose many risks such as data privacy violations. With reference to the database implemented, security is imperative because if there were no restrictions applied to the database, any person can edit other member's personal details, etc

There are some privileges that only have access to some people. It is being applied in this project to give authentication to some people.

### 1) Giving privileges to coach.

Create ROLE coach;

GRANT CREATE, DELETE ON player TO coach;

GRANT UPDATE, SELECT ON team TO coach WITH GRANT OPTION;

### 2) Giving Privileges To User.

CREATE ROLE user IDENTIFIED BY qwertyuiop ; (IDENTIFIED BY PASSWORD)

GRANT CREATE, UPDATE, SELECT ON game TO user;

GRANT UPDATE, DELETE ON plays TO user;

GRANT SELECT ON week, day TO user;

With giving some privileges to certain people, certain restrictions should also be implemented.

In this project, I restricted coach to delete data from person table.

To remove privileges from staff members, a command similar to the following can be used:

REVOKE DELETE ON person FROM coach;

## VIEW CREATION:

- 1) create or replace view person\_players as select person.id, game\_id, name, nationality, role, birth\_date as age, birth\_date, t\_id from person inner join player;





























id	game_id	name	nationality	role	age	birth_date	t_id
101	Balls	An Le	USA	Top	1994-06-22	1994-06-22	11
102	Balls	An Le	USA	Jungler	1993-01-01	1993-01-01	11
103	Balls	An Le	USA	Mid	1993-02-02	1993-02-02	11
104	Balls	An Le	USA	AD Carry	1994-03-19	1994-03-19	11
105	Balls	An Le	USA	Support	1989-06-15	1989-06-15	11
106	Balls	An Le	USA	Top	1994-11-12	1994-11-12	12
107	Balls	An Le	USA	Jungler	1991-05-10	1991-05-10	12
108	Balls	An Le	USA	Mid	1995-10-14	1995-10-14	12
109	Balls	An Le	USA	AD Carry	1993-07-19	1993-07-19	12
110	Balls	An Le	USA	Support	1992-09-08	1992-09-08	12
111	Balls	An Le	USA	Top	1993-03-03	1993-03-03	13
112	Balls	An Le	USA	Jungler	1993-04-04	1993-04-04	13
113	Balls	An Le	USA	Mid	1995-07-05	1995-07-05	13
114	Balls	An Le	USA	AD Carry	1993-05-05	1993-05-05	13
115	Balls	An Le	USA	Support	1993-06-06	1993-06-06	13
116	Balls	An Le	USA	Top	1995-05-20	1995-05-20	14
117	Balls	An Le	USA	Jungler	1992-05-23	1992-05-23	14
118	Balls	An Le	USA	Mid	1993-07-07	1993-07-07	14
119	Balls	An Le	USA	AD Carry	1997-05-08	1997-05-08	14
120	Balls	An Le	USA	Support	1993-08-08	1993-08-08	14
101	Meteos	William Hartman	USA	Top	1994-06-22	1994-06-22	11
102	Meteos	William Hartman	USA	Jungler	1993-01-01	1993-01-01	11
103	Meteos	William Hartman	USA	Mid	1993-02-02	1993-02-02	11
104	Meteos	William Hartman	USA	AD Carry	1994-03-19	1994-03-19	11
105	Meteos	William Hartman	USA	Support	1989-06-15	1989-06-15	11

- 2) create or replace view person\_coaches as select person.id, game\_id, name, nationality, position, t\_id from person inner join coach

id	game_id	name	nationality	position	t_id
101	Balls	An Le	USA	Coach	11
101	Balls	An Le	USA	Analyst	11
101	Balls	An Le	USA	Coach	13
101	Balls	An Le	USA	Coach	14
101	Balls	An Le	USA	Analyst	15
101	Balls	An Le	USA	Analyst	16
101	Balls	An Le	USA	Coach	12
101	Balls	An Le	USA	Coach	16
101	Balls	An Le	USA	Coach	15
102	Meteos	William Hartman	USA	Coach	11
102	Meteos	William Hartman	USA	Analyst	11
102	Meteos	William Hartman	USA	Coach	13
102	Meteos	William Hartman	USA	Coach	14
102	Meteos	William Hartman	USA	Analyst	15
102	Meteos	William Hartman	USA	Analyst	16
102	Meteos	William Hartman	USA	Coach	12
102	Meteos	William Hartman	USA	Coach	16
102	Meteos	William Hartman	USA	Coach	15
103	Incarnati0n	Nicolaj Jensen	Denmark	Coach	11
103	Incarnati0n	Nicolaj Jensen	Denmark	Analyst	11
103	Incarnati0n	Nicolaj Jensen	Denmark	Coach	13
103	Incarnati0n	Nicolaj Jensen	Denmark	Coach	14
103	Incarnati0n	Nicolaj Jensen	Denmark	Analyst	15
103	Incarnati0n	Nicolaj Jensen	Denmark	Analyst	16
103	Incarnati0n	Nicolaj Jensen	Denmark	Coach	12

## JOINS

- 1) `SELECT champion.c_id, c_name FROM champion LEFT JOIN plays ON champion.c_id = plays.c_id ORDER BY plays.c_id;`

						c_id	c_name	
<input type="checkbox"/>		Edit		Copy		Delete	102	Akali
<input type="checkbox"/>		Edit		Copy		Delete	103	Alistar
<input type="checkbox"/>		Edit		Copy		Delete	106	Draven
<input type="checkbox"/>		Edit		Copy		Delete	101	Ahri
<input type="checkbox"/>		<b>You can also edit most values by double-clicking directly on them.</b>					104	Annie
<input type="checkbox"/>		Edit		Copy		Delete	105	Caitlyn
<input type="checkbox"/>		Edit		Copy		Delete	107	Elise
<input type="checkbox"/>		Edit		Copy		Delete	108	Evelynn
<input type="checkbox"/>		Edit		Copy		Delete	109	Ezreal
<input type="checkbox"/>		Edit		Copy		Delete	110	Graves

- 2) `SELECT game.g_date,game.g_duration FROM game INNER JOIN plays ON game.code=plays.code;`

				g_date	g_duration
<input type="checkbox"/>		Edit	Copy  Delete	2004-01-20 15:17:00	35
<input type="checkbox"/>		Edit	Copy  Delete	2004-01-20 15:17:00	35
<input type="checkbox"/>		Edit	Copy  Delete	2004-01-20 15:19:30	28
<input type="checkbox"/>		Edit	Copy  Delete	2001-01-20 15:13:00	45
<input type="checkbox"/>		Edit	Copy  Delete	2001-01-20 15:14:30	38
<input type="checkbox"/>		Edit	Copy  Delete	2001-01-20 15:16:00	31

3)

`SELECT person.name AS name, player.role AS role, coach.position AS position FROM person INNER JOIN player ON person.id = player.id INNER JOIN coach ON person.id = coach.id;`

## UPDATE STATEMENT

1)

UPDATE person SET name = 'Alfred Schmidt', nationality= 'Dutch' WHERE id = 107;

id	game_id	name	nationality
101	Balls	An Le	USA
102	Meteos	William Hartman	USA
103	Incarnati0n	Nicolaj Jensen	Denmark
104	Sneaky	Zachary Scuderi	USA
105	LemonNation	Daerek Hart	USA
106	ZionSpartan	Darshan Upadhyaya	Canada
107	Xmithie	Alfred Schmidt	Dutch
108	Pobelter	Eugene Park	USA
109	Doublelift	Yiliang Peng	USA



2) UPDATE week SET version = '8'WHERE continent = 'Europe';

w_number	continent	version	week_mvp
1	Europe	8	114
1	<u>North America</u>	5	101
2	Europe	8	115
2	North America	5	103
3	Europe	8	119
3	North America	6	106
4	North America	6	109
5	North America	7	110
6	North America	8	112



## TRIGGERS

```
1) DELIMITER //
CREATE TRIGGER `DeletePlayer` AFTER DELETE ON `player`
FOR EACH ROW begin

delete from plays

where id = old.id;

delete from player

where id = old.id;

delete from person

where id = old.id;

end
//
DELIMITER;
```

This trigger will delete the player from all tables where the player id is referenced as foreign key.

```
2)
DELIMITER //
CREATE TRIGGER `boston_vehicle_check` BEFORE INSERT ON `game`
FOR EACH ROW BEGIN
  IF (New.g_duration < 0) THEN
    SIGNAL SQLSTATE '02002' SET MESSAGE_TEXT = 'Game duration less than 0';
  END IF;
END
//
DELIMITER ;
```

This trigger gives an exception when game duration entered is less than zero.

## ADDITIONAL FEATURES:

**Variables:** It is an object that can hold a single data value of a specific type. Variables are used to hold a value in data to be tested or to save data to be returned by a stored procedure.

They have local scope and are only visible within the batch or procedure where they are defined.

### 1) Variable stored as an ID to show data.

```
DECLARE @Myvariable int
SET @Myvariable = 101
Select role, birth_date, t_id, name
FROM player, person
WHERE id>=@Myvariable;
```

## Functions:

This is a user defined function that accepts parameters, perform actions and returns a result in a value.

### 1) Function to return player kills.

```
create or replace function avg_kills(player in integer)
return integer is
    pkills integer;
begin
    select avg(kills) into pkills from plays p where player = p.id;

    return pkills;
end;
```

## APPENDIX 1:

Code used to create tables:

**1) Person:**

```
CREATE TABLE person (id INTEGER (3) NOT NULL,  
  
    game_id VARCHAR (35) not null unique,  
  
    name varchar(35) not null,  
  
    nationality varchar(35) not null,  
  
    primary key (id));
```

**2) Region:**

```
create table region (continent varchar(13) not null check (continent in ('Europe',  
'North America')), primary key(continent));
```

**3) Team:**

```
create table team (t_id integer(3) not null,  
    tag varchar (6) not null unique,  
    t_name varchar (24) not null unique,  
    wins integer (2) not null check (wins >= 0 and wins < 19),  
    losses integer (2) not null check(losses >= 0 and losses < 19),  
    foundation_date date not null,  
    continent varchar (13) not null,  
    primary key(t_id),  
    foreign key(continent) references region (continent));
```

**4) Player:**

```
create table player(id integer(3) not null,  
    role varchar(8) not null check(role in ('Top', 'Jungler', 'Mid', 'AD Carry', 'Support')),  
    birth_date date not null, t_id integer(3),  
    primary key(id),  
    foreign key(id) references person(id),  
    foreign key(t_id) references team(t_id));
```

**5) Coach:**

```
create table coach(id integer(3) not null,  
    position varchar(7) not null check(position in ('Coach', 'Analyst')),  
    t_id integer(3),  
    primary key (id),  
    foreign key (id) references person(id),  
    foreign key(t_id) references team(t_id));
```

**6) Brand:**

```
create table brand(b_id integer(3) not null,  
b_name varchar(35) not null unique,  
primary key(b_id));
```

**7) Champion:** create table champion(c\_id integer(3) not null,

```
c_name varchar(35) not null unique,  
c_role varchar(8) not null check(c_role in ('Assassin', 'Fighter', 'Mage', 'Marksman',  
'Support', 'Tank')),  
primary key (c_id));
```

**8) Patch:** create table patch(version integer(2) not null check(version >= 5 and version < 9), primary key(version));

**9) Week:**create table week(w\_number integer(2) not null check(w\_number > 0 and w\_number< 10),

```
continent varchar(13) not null,  
version integer(2) not null,  
week_mvp integer(3),  
primary key(w_integer, continent),  
foreign key(continent) references region(continent),  
foreign key(version) references patch(version),  
foreign key(week_mvp) references player(id));
```

**10) Day:** create table day(d\_number integer(1) not null check(d\_number > 0 and d\_number < 3),  
w\_number integer(2) not null,  
continent varchar(13) not null,  
primary key(d\_number, w\_number, continent), foreign key(w\_number, continent) references week(w\_number, continent));

**11) Game:** create table game(code integer(3) not null,

```
g_date timestamp not null unique,  
g_duration integer(3) check(g_duration > 0),  
d_number integer(1) not null,  
w_number integer(2) not null,
```



```
continent varchar(13) not null,  
blue_t integer(3) not null,  
red_t integer(3) not null,  
winner_t integer(3),  
primary key(code),  
foreign key(d_number, w_number, continent) references day(d_number, w_number,  
continent),  
foreign key(blue_t) references team(t_id), foreign key(red_t) references team(t_id),  
foreign key(winner_t) references team(t_id));
```

**12) Sponsors:** create table sponsors(b\_id integer(3) not null,  
t\_id integer(3) not null,  
primary key(b\_id, t\_id),  
foreign key(b\_id) references brand(b\_id),  
foreign key(t\_id) references team(t\_id));

**13) Disables:** create table disables(version integer(2) not null, c\_id integer(3) not null,  
primary key(version, c\_id),  
foreign key(version) references patch(version),  
foreign key(c\_id) references champion(c\_id));

**14) Plays:** create table plays(id integer(3) not null,  
code integer (3) not null,  
kills integer (2) not null check(kills >= 0),  
deaths integer (2) not null check(deaths >= 0),  
assists integer (2) not null check(assists >= 0),  
creeps\_slain integer(3) not null check(creeps\_slain >= 0), gold integer(5) not null check(gold  
>= 0), c\_id integer(3) not null,  
primary key(id, code),  
foreign key(id) references player(id),  
foreign key(code) references game(code),  
foreign key(c\_id) references champion(c\_id));

## APPENDIX 2:

Inserting data into Tables:

### 1) Inserting data to person table:

insert into person values (101, 'Balls', 'An Le', 'USA');

insert into person values (102, 'Meteos', 'William Hartman', 'USA');

insert into person values(103, 'IncarnatiOn', 'Nicolaj Jensen', 'Denmark');

insert into person values(104, 'Sneaky', 'Zachary Scuderi', 'USA');

insert into person values(105, 'LemonNation', 'Daerek Hart', 'USA');

insert into person values(106, 'ZionSpartan', 'Darshan Upadhyaya', 'Canada');

insert into erson values(107, 'Xmithie', 'Jake Puchero', 'Philippines');

insert into person values(108, 'Pobelter', 'Eugene Park', 'USA');

insert into person values(109, 'Doublelift', 'Yiliang Peng', 'USA');

insert into person values(110, 'Aphromoo', 'Zaqueri Black', 'USA');

insert into person values(111, 'Flaresz', 'Cuong Ta', 'USA');

insert into person values(112, 'Trashy', 'Jonas Andersen',  
'Denmark');

insert into person values(113, 'Innox', 'Tyson Kapler', 'Canada');

insert into person values(114, 'Otter', 'Brian Baniqued', 'USA');

insert into person values(115, 'Bodydrop', 'Adam Krauthaker', 'Canada');

insert into person values(116, 'Hauntzer', 'Kevin Yarnell', 'USA');

insert into person values(117, 'Move', 'Kang Min-su', 'South Korea');

insert into person values(118, 'Keane', 'Lae-Young Jang', 'South Korea');

insert into person values(119, 'Altec', 'Johnny Ru', 'Canada');

insert into person values(120, 'Bunny FuFuu', 'Michael Kurylo', 'USA');

insert into person values(201, 'Charlie', 'Charlie Lipsie', 'China');

insert into person values(202, 'Hai', 'Hai Lam', 'USA');

insert into person values(203, 'HuHi', 'Choi Jae-hyun', 'South Korea');

insert into person values(204, 'Lazydude', 'Brad Marx', 'USA');

insert into person values(205, 'LS', 'Nick De Cesare', 'USA');

insert into person values(206, 'Matthew Schmieder', 'Matthew Schmieder', 'USA');

insert into person values(207, 'Rico', 'Rico', 'USA');

insert into person values(208, 'chain', 'Kim Dong-woo', 'South Korea');

insert into person values(209, 'Fly', 'Sangchul Kim', 'South Korea');

insert into person values(210, 'Peter', 'Peter Zhang', 'China');

### 2) Inserting Data into Region Table:

insert into Region values('North America');

insert into Region values('Europe');

### 3) Inserting Data into Region Table:

```

insert into team values(11, 'C9', 'Cloud9', 0, 0, ('2012.12.01'), 'North America');
insert into Team values(12, 'CLG', 'Counter Logic Gaming', 0, 0, ('2010.04.01'),
'North America');
insert into team values(13, 'NME', 'Enemy Esports', 0, 0, ('2014.11.01'), 'North
America');
insert into team values(14, 'GV', 'Gravity', 0, 0, ('2015.01.01'), 'North America');
insert into team values(15, 'T8', 'Team 8', 0, 0, ('2013.12.01'), 'North America');
insert into team values(16, 'Dig', 'Team Dignitas', 0, 0, ('2011.09.01'), 'North
America');

```

#### **4) Inserting Data Into Player Table:**

```

insert into player values(101, 'Top', ('1994.06.22'), 11);
insert into player values(102, 'Jungler', ('1993.01.01'), 11);
insert into player values(103, 'Mid', ('1993.02.02'), 11);
insert into player values(104, 'AD Carry', ('1994.03.19'), 11);
insert into player values(105, 'Support', ('1989.06.15'), 11);
insert into player values(106, 'Top', ('1994.11.12'), 12);
insert into player values(107, 'Jungler', ('1991.05.10'), 12);
insert into player values(108, 'Mid', ('1995.10.14'), 12);
insert into player values(109, 'AD Carry', ('1993.07.19'), 12);
insert into player values(110, 'Support', ('1992.09.08'), 12);
insert into player values(111, 'Top', ('1993.03.03'), 13);
insert into player values(112, 'Jungler', ('1993.04.04'), 13);
insert into player values(113, 'Mid', ('1995.07.05'), 13);
insert into player values(114, 'AD Carry', ('1993.05.05'), 13);
insert into player values(115, 'Support', ('1993.06.06'), 13);
insert into player values(116, 'Top', ('1995.05.20'), 14);
insert into player values(117, 'Jungler', ('1992.05.23'), 14);
insert into player values(118, 'Mid', ('1993.07.07'), 14);
insert into player values(119, 'AD Carry', ('1997.05.08'), 14);
insert into player values(120, 'Support', ('1993.08.08'), 14);

```

#### **5) Inserting Data into Coach table:**

```

insert into coach values(201, 'Coach', 11);

insert into coach values(202, 'Analyst', 11);

insert into coach values(204, 'Coach', 13);

insert into coach values(205, 'Coach', 14);

insert into coach values(206, 'Analyst', 15);

insert into coach values(207, 'Analyst', 16);

insert into coach values(208, 'Coach', 16);

insert into coach values(209, 'Coach', 12);

insert into coach values(210, 'Coach', 15);

```

## **6) Inserting Data into Brand Table:**

```
insert into brand values(21, 'KingstonHyperX');
insert into brand values(22, 'Logitech');
insert into brand values(23, 'Nissan');
insert into brand values(24, 'Nvidia');

insert into brand values(25, 'Ozone');

insert into brand values(26, 'Pringles');

insert into brand values(27, 'Razer');

insert into brand values(28, 'ROCCAT');

insert into brand values(29, 'SanDisk');

insert into brand values(30, 'SteelSeries');
```

## **7) Inserting Data into Champion Table:**

```
insert into champion values(101, 'Ahri', 'Mage');
insert into champion values(102, 'Akali', 'Assassin');
insert into champion values(103, 'Alistar', 'Tank');
insert into champion values(104, 'Annie', 'Mage');
insert into champion values(105, 'Caitlyn',
'Marksman');
insert into champion values(106, 'Draven',
'Marksman');
insert into champion values(107, 'Elise', 'Mage');
insert into champion values(108, 'Evelynn',
'Assassin');
insert into champion values(109, 'Ezreal',
'Marksman');
insert into champion values(110, 'Graves',
'Marksman');
```

## **8) Inserting Data into Patch Table:**

```
insert into patch values(5);
insert into patch values(6);
insert into patch values(7);
insert into patch values(8);
insert into patch values(9);
```

### **9) Inserting data into Week table:**

```
into week values(2, 'North America', 5, 103);
insert into week values(3, 'North America', 6, 106);
insert into week values(4, 'North America', 6, 109);
insert into week values(5, 'North America', 7, 110);
insert into week values(6, 'North America', 8, 112);
insert into week values(1, 'Europe', 5, 114);
insert into week values(2, 'Europe', 5, 115);
insert into week values(3, 'Europe', 9, 119);
```

### **10) Inserting Data into Day Table:**

```
insert into day values(1, 1, 'North America');
insert into day values(2, 1, 'North America');
insert into day values(1, 2, 'North America');
insert into day values(2, 2, 'North America');
insert into day values(1, 3, 'North America');
insert into day values(2, 3, 'North America');
insert into day values(1, 1, 'Europe');
insert into day values(2, 1, 'Europe');
insert into day values(1, 2, 'Europe');
insert into day values(2, 2, 'Europe');
insert into day values(1, 3, 'Europe');
insert into day values(2, 3, 'Europe'
);
```

### **11) Inserting Data into Game Table:**

```
insert into game values(151, ('01.01.2015.13.00'), 45, 1, 1, 'Europe', 11, 12, 12);
insert into game values(152, ('01.01.2015.14.30'), 38, 1, 1, 'Europe', 13, 14, 13);
insert into game values(153, ('01.01.2015.16.00'), 31, 1, 1, 'Europe', 15, 16, 16);
insert into game values(106, ('04.01.2015.17.00'), 35, 2, 1, 'North America', 12, 13, 12);
insert into game values(107, ('04.01.2015.18.30'), 28, 2, 1, 'North America', 11, 15, 15);
insert into game values(108, ('04.01.2015.19.30'), 28, 2, 1, 'North America', 14, 16, 16);
```

### **12) Inserting Data into Sponsor Table:**

```
insert into sponsors values(21, 11);
insert into sponsors values(21, 16);
insert into sponsors values(22, 12);
insert into sponsors values(22, 14);
insert into sponsors values(23, 16);
insert into sponsors values(24, 13);
```

```
insert into sponsors values(25, 11);  
insert into sponsors values(26, 12);
```

### **13) Inserting Data into Disable Table:**

```
insert into disables values(5, 103);  
insert into disables values(5, 108);  
insert into disables values(6, 105);  
insert into disables values(7, 110);  
insert into disables values(7, 109);  
insert into disables values(9, 101);
```

### **14) Inserting Data into Play Table:**

```
insert into plays values(101, 106, 3, 2, 11, 220, 16142, 110);  
insert into plays values(102, 106, 6, 0, 16, 174, 17126, 107);  
insert into plays values(103, 108, 4, 4, 8, 249, 16996, 108);  
insert into plays values(104, 151, 10, 1, 6, 282, 19328, 109);  
insert into plays values(105, 152, 1, 3, 19, 31, 13564, 104);  
insert into plays values(106, 153, 3, 4, 3, 207, 15762, 105);
```



