University of British Columbia, Vancouver
Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: 2

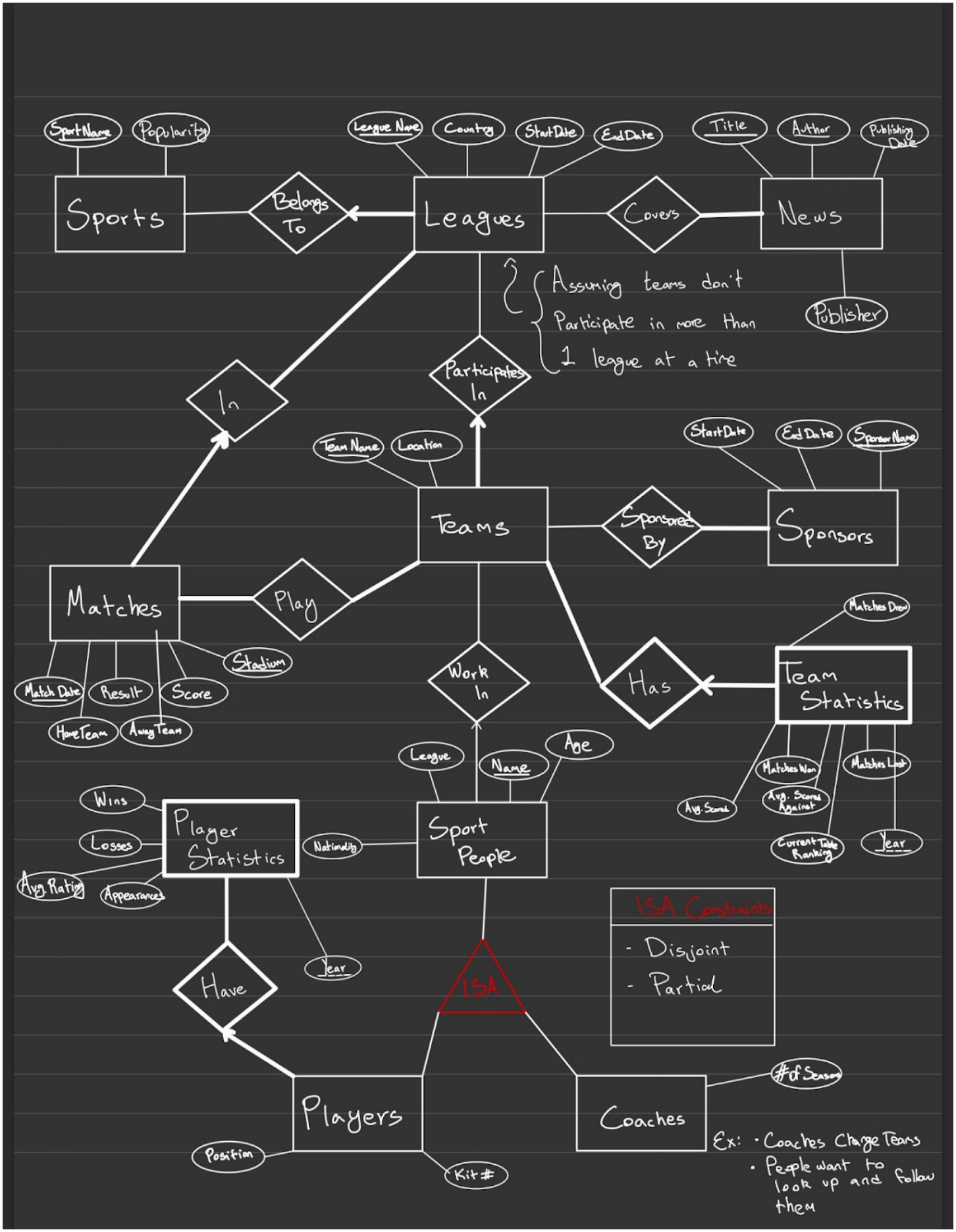
Date: Oct. 20th, 2023

Group Number: 152

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Shariq Hassan	48238018	e1d2g	shariqihassan@gmail.com
Nima Motieifard	66492497	b8m2i	nima.motieifard@gmail.com
Farouk Sharkia	53860227	n7d3b	farouks@student.ubc.ca

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)
In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

ER Diagram


Sport Name
Popularity
Sports
Belongs To
League Name
Country
Start Date
End Date
Leagues
Covers
Title
Author
Publishing Date
News
Publisher
Assuming teams don't Participate in more than 1 league at a time
In
Participates In
Team Name
Location
Teams
Start Date
End Date
Sponsor Name
Sponsored By
Sponsors
Matches
Play
Stadium
Match Date
Result
Score
Home Team
Away Team
Work In
Has
Matches Drew
Team Statistics
Matches Won
Matches Lost
Avg. Scored
Avg. Scored Against
Current Table Ranking
Year
League
Name
Age
Wins
Losses
Avg. Rating
Appearances
Player Statistics
Nationality
Sport People
Have
Year
ISA
ISA Constraints
- Disjoint
- Partial
Players
Coaches
# of Seasons
Position
Kit #
Ex: • Coaches Change Teams
• People want to look up and follow them

Project Summary

The domain of this application is primarily in sports entertainment. It is for fans of various sports to be able to look up information on specific teams, players, leagues and so on. It is designed to give people an easy and convenient method for accessing and staying up to date on the ever changing information in the world of sports entertainment.
The database centers around providing data about a few leagues in a certain team sport. The scope is limited to information relevant to the teams participating in those leagues and does not go into too much detail about subsidiary facts or data. The target audience of such a service is mainly fans of these sports as well as sports analysts who wish to keep track of various teams or player achievement in specific leagues as well comparing and evaluating team and player performances.

ER Diagram Changes:

- The TA pointed out redundant attributes present in the Teams Entity and SportPeople Entity. Therefore we removed the attribute sportName(from Teams & SportPeople) and teamName(from SportPeople) as we could now understand that it was indeed redundant and unnecessary as they are passed down accordingly from the relationships.
- As we did not quite understand what the TA meant by “If you want to use it as a list, there should be lines connecting from every use of SportName”, we talked to Professor Pottinger about this query and said she herself did not understand this comment. She recommended adding a meaningful attribute to the Sports Entity and keeping it. The TA also commented on our “# of leagues” not being a meaningful attribute which we agree with and aimed to follow the Professor’s recommendation of adding one. Hence we replaced “# of leagues” with popularity which we feel is an attribute sport fans care about.
- We also removed Kit# as a Key from the sub entity Players as they do not have PK’s which we mistakenly overlooked.
- Another thing we overlooked was the entity Sponsors having only one attribute, therefore we deemed it better to move the attributes (StartDate, EndDate) from the relation SponsoredBy to the entity Sponsors so that we could satisfy this requirement.
- We also removed the attribute referees from Matches as they are a SportPeople and in our opinion it is not vital to know the referee of a match for a user.
- We were given a note about our cardinality for our weak entities (namely Teams & TeamStatistics, Players & PlayerStatistics) being one-to-one relationships which is not allowed. Therefore we chose to make it a one-to-many relationship by signifying year as an attribute and partial key indicating that one team can have many statistics (different years = different statistics). We did the same with Players.

- The TA also suggested hand drawn diagrams to be harder to follow however we think we have done a good job to keep it legible and tried to improve it again. If there are any major concerns we would be happy to change it and use an application.

Assumptions:

1) News:

- An author works only for one publisher.
- An author publishes at most one news article a day.
- No 2 authors have the exact same name (first name, middle name, last name) work for the same publisher.

2) Matches:

- Teams don't play more than one game a day
- No more than 1 match in league is played in a stadium per day
- Result: winning team name, in case of a tie: tie

2) Sports:

- Popularity: Amount of fans (i.e: Soccer: 3.5 Billion, Basketball: 800 Million, etc.)

Relational Model:

Sports(SportName: char, Popularity: int)
Primary Key: SportName
Candidate Key: SportName
Foreign Key: SportName NOT NULL, UNIQUE
FDs:
SportName –> { # of Leagues, Popularity}

Note:
- Popularity: Amount of fans (i.e: Soccer: 3.5 Billion, Basketball: 800 Million, etc.)

Leagues(LeagueName: char, Country: char, StartDate: char, EndDate: char, **SportName: Char**)
Primary Key: LeagueName
Candidate Key: LeagueName
Foreign Key: SportName NOT NULL, UNIQUE
FDs:
LeagueName → {LeagueName, Country, StartDate, EndDate, SportName}

News(Title: char, Author: char, PublishingDate: char, Publisher: char)
Primary Key: Title, Author
Candidate Key: Title, Author
Foreign Key: N/A
FDs:
Title, Author → {PublicationDate}
PublicationDate, Title → {Author}
Author → {Publisher}

Assumptions:
- An author works only for one publisher.
- An author publishes at most one news article a day.
- No 2 authors have the exact same name (first name, middle name, last name) work for the same publisher.

Teams(TeamName: char, Location: char, **LeagueName: Char**)
Primary Key: TeamName
Candidate Key: TeamName
Foreign Key: LeaguName NOT NULL, UNIQUE

FDs:

TeamName → {Location}
(e.g. Los Angeles Lakers, Manchester United FC, FC Barcelona Handbol)
LeagueName, TeamName → {Location}

Matches(MatchDate: char, Result: char, Score: char, Stadium: char, HomeTeam: char, AwayTeam: char, **LeagueName: char**)
Primary Key: MatchDate, Stadium
Candidate Key: MatchDate, Stadium
Foreign Key: LeagueName NOT NULL, UNIQUE

FDs:
MatchDate, Stadium → {Score, Result, HomeTeam, AwayTeam, LeagueName}
HomeTeam → {Stadium}
LeagueName, MatchDate → {Result, HomeTeam, AwayTeam}
MatchDate, HomeTeam, AwayTeam → {Score, Result, LeagueName, stadium}
Stadium → {HomeTeam}

Assumption:
- Teams don't play more than one game a day
- No more than 1 match in league is played in a stadium per day
- Result: winning team name, in case of a tie: tie

Sponsors(SponsorName: char, startDate: char, endDate: char)
Primary Key: SponsorName
Candidate Key: SponsorName
Foreign Key: N/A

FDs:
SponsorName -> {startDate, endDate}

SponsoredBy(**TeamName: char**, **SponsorName: char**)
Primary Key: TeamName, SponsorName
Candidate Key: TeamName, SponsorName
Foreign Key: TeamName, SponsorName NOT NULL, UNIQUE

FDs:
TeamName → {SponsorName}

TeamStatistics(MatchDrew: int, AverageScored: double, MatchesWon: int, MatchesLost: int, AverageScoredAgainst: double, CurrentTableRanking: int, year: int, **TeamName: char**)

Primary Key: Year, TeamName
Candidate Key: Year, TeamName, CurrentTableRanking
Foreign Key: TeamName NOT NULL, UNIQUE

FDs:

Year, teamName → {MatchDrew, AverageScored, MatchesWon, MatchesLost, AvarageScoredAgainst, CurrentTableRanking, Year, teamName, sportName}

SportPeople(Nationality: char, League: char, Name: char, Age: int, **TeamName: char**)
Primary Key: Name
Candidate Key: Name
Foreign Key:TeamName NOT NULL, UNIQUE

FDs:

Name, teamName → {Name, League, Age, Nationality}

Coaches(NumberOfSeasons: int, **Name: char**, **TeamName: char**)
Primary Key: Name, TeamName
Candidate Key: Name, TeamName
Foreign Key: Name, TeamName

FDs:
Name, SportName, TeamName → { Name, SportName, TeamName, #ofSeasons}

Players(position: char, kitNum: int, **Name: char**, **SportName: char**)
Primary Key: Name, SportName
Candidate Key: Name, SportName, KitNum
Foreign Key: Name, SportName

FDs:
Name, sportName, TeamName → {Name, sportName, kitNum, position}

PlayerStatistics(Wins: int, year: int, Losses: int, AverageRating: double, Appearances: double, **Name: char**, **SportName: char**)
Primary Key: Year, Name, SportName
Candidate Key: Year, Name, SportName
Foreign Key: Name, SportName NOT NULL, UNIQUE

FDs:

Year, SportName, Name -> {Wins, Year, Losses, AverageRating, Appearances, SportName, Name}

Covers(**<u>LeagueName: char, NewsTitle: char, NewsAuthor: char</u>**)

Primary Key: LeagueName, NewsTitle, NewsAuthor
Candidate Key: LeagueName, NewsTitle, NewsAuthor
Foreign Key: LeagueName, NewsTitle, NewsAuthor NOT NULL, UNIQUE

FDs:

LeagueName, NewsTitle → NewsAuthor

Play(**<u>MatchDate: char</u>**, **<u>Stadium: char</u>**, **<u>TeamName: char</u>**)

Primary Key: MatchDate, Stadium, TeamName
Candidate Key: MatchDate, Stadium, TeamName
Foreign Key: MatchDate, Stadium, TeamName NOT NULL, UNIQUE

FDs:

MatchDate, TeamName → {Stadium}

Normalization:

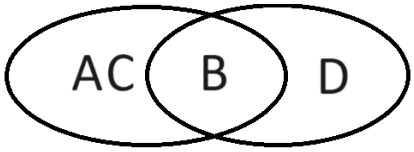
Some of The functional dependencies in News and Matches have to be normalized since they are not in BCNF.

News(Title: char, Author: char, PublishingDate: char, Publisher: char)

FD's:

Title, Author → {PublicationDate}
PublicationDate, Title → {Author}
Author → {Publisher}

Normalization:
A: Title
B: Author
C: PublishingDate
D: Publisher
Which gives us
AB → C
B → D
AC → B
Closures:
AB+ = {A,B,C,D}
B+ = {B,D}
AC+ = {A,B,C,D}



This gives us:
R1(ABC), R2(BD)
Final answer:

News_1(title: char, Author: char, PublishingDate: char),
News_2(Author: char, Publisher: char)

Matches(MatchDate: char, Result: char, Score: char, Stadium: char, HomeTeam: char, AwayTeam: char, **LeagueName: char**)

FD's:

MatchDate, Stadium → {Score, Result, HomeTeam, AwayTeam, LeagueName}

HomeTeam → {Stadium}
LeagueName, MatchDate → {Result, HomeTeam, AwayTeam}
MatchDate, HomeTeam, AwayTeam → {Score, Result, AwayTeam, LeagueName, stadium}
Stadium → {HomeTeam}

Normalization:
MatchDate: A
Result: B
Score: C
Stadium: D
HomeTeam: E
AwayTeam: F
LeagueName: G

Matches(A, B, C, D, E, F, G)
AD → BCEFG
E → D (Violates BCNF)
AG → BEF
AEF → BCDG
D → E (Violates BCNF)

Closures:
AD += {A, D,B, C, E, F, G}
E+= {E, D}
AG+= {A, G, B, E, F}
AEF+= {A, E, F, B, C, D, G}
D+= {D, E}

R(AEG), R(ABCEFG), R(DE)
AE → G
A G → E B C F
D → E
E → D
MatchesR1(MatchDate: char, HomeTeam: char, LeagueName: Char)
MatchesR2(MatchDate: char, Result: char, score: char, HomeTeam: char, AwayTeam: char, LeagueName: char)
MatchesR3(HomeTeam: char, Stadium: char)

SQL/DDL:

Sports(SportName: char, popularity: int)

```
CREATE TABLE Sports (
        SportName           CHAR(20) PRIMARY KEY,
        popularity          INTEGER );

CREATE TABLE Leagues(
        LeagueName          CHAR(30) PRIMARY KEY,
        Country             CHAR(20),
        StartDate           CHAR(10),
        EndDate             CHAR(10) );

CREATE TABLE News_1(
        Title               CHAR(300),
        Author              CHAR(20),
        PublishingDate      CHAR(10),
        FOREIGN KEY PRIMARY KEY(Title, Author) );

CREATE TABLE News_2(
        Author              CHAR(20) PRIMARY KEY,
        Publisher           CHAR(20) );

CREATE TABLE Teams(
        TeamName            CHAR(30) PRIMARY KEY,
        Location            CHAR(20)
        LeagueName                  CHAR(30)
        FOREIGN KEY(LeagueName)
        REFERENCES Leagues
        ON DELETE CASCADE
        ON UPDATE CASCADE);

CREATE TABLE Matches_1(
        MatchDate           CHAR(10),
        HomeTown            CHAR(20),
        LeagueName          CHAR(20),
        PRIMARY KEY(MatchDate, Stadium),
        FOREIGN KEY(LeagueName)
        REFERENCES Matches_2
        ON DELETE CASCADE
        ON UPDATE CASCADE);
```

```
CREATE TABLE Matches_2(
        MatchDate           CHAR(10),
        Stadium             CHAR(20),
        Result              CHAR(20),
        Score               CHAR(20),
        HomeTeam            CHAR(20),
        AwayTeam            CHAR(20),
        LeagueName          CHAR(20)
        PRIMARY KEY(LeagueName));

CREATE TABLE Matches_3(
        MatchDate           CHAR(10),
        HomeTeam            CHAR(20),
        PRIMARY KEY(MatchDate, Stadium),
        PRIMARY KEY(MatchDate, Stadium)
        FOREIGN KEY(LeagueName)
        REFERENCES Matches_2
        ON DELETE CASCADE
        ON UPDATE CASCADE));


CREATE TABLE Sponsors(
        SponsorName         CHAR(20) PRIMARY KEY,
        StartDate           CHAR(10),
        EndDate             CHAR(10) );

CREATE TABLE SponsoredBy(
        SponsorName                 CHAR(20),
        TeamName            CHAR(30),
        PRIMARY KEY(SponsorName, TeamName),
        FOREIGN KEY(SponsorName)
        REFERENCES Sponsors
        ON DELETE SET DEFAULT
        ON UPDATE SET DEFAULT,
        FOREIGN KEY(TeamName)
        REFERENCES Teams
        ON DELETE SET DEFAULT
        ON UPDATE SET DEFAULT);
```

```
CREATE TABLE TeamStatistics(
        MatchesDrew             INTEGER,
        MatchesWon              INTEGER,
        MatchesLost             INTEGER,
        AvarageScored           DOUBLE,
        AvarageScoredAgainst    DOUBLE,
        CurrentTableRanking     INTEGER,
        Year                    INTEGER,
        TeamName                CHAR(20),
        PRIMARY KEY(Year, TeamName)
        FOREIGN KEY(TeamName)
        REFERENCES Teams
        ON DELETE CASCADE
        ON UPDATE CASCADE);

CREATE TABLE SportPeople(
        Nationality             CHAR(20),
        Name                    CHAR(30),
        Age                     INTEGER,
        TeamName                CHAR(30),
        PRIMARY KEY (Nationality, Name, Age),
        FOREIGN KEY(TeamName)
        REFERENCES Teams
        ON DELETE SET DEFAULT
        ON UPDATE SET DEFAULT);

CREATE TABLE Coaches(
        NumOfSeasons            INTEGER,
        Name                    CHAR(20),
        TeamName                CHAR(20),
        PRIMARY KEY(Name, TeamName),
        FOREIGN KEY(Name)
        REFERENCES SportPeople
        ON DELETE CASCADE
        ON UPDATE CASCADE,
        FOREIGN KEY(TeamName)
        REFERENCES SportPeople
        ON DELETE CASCADE
        ON UPDATE CASCADE);

CREATE TABLE Players(
        Position                CHAR(20),
        KitNum                  INTEGER,
        Name                    CHAR(20),
```

```
    TeamName            CHAR(20),
    PRIMARY KEY (Name, SportName)
    FOREIGN KEY(Name)
    REFERENCES Players
    ON DELETE CASCADE
    ON UPDATE CASCADE);



CREATE TABLE PlayersStatistics(
    Wins                INTEGER,
    Losses                  INTEGER,
    Year                INTEGER,
    AvarageRating       DOUBLE,
    Appearances         DOUBLE,
    Name                CHAR(20),
    PRIMARY KEY(Year),
    FOREIGN KEY(Name)
    REFERENCES Players
    ON DELETE CASCADE
    ON UPDATE CASCADE);

CREATE TABLE Covers (
    LeagueName              CHAR(30),
    NewsTitle               CHAR(300),
    NewsAuthor              CHAR(20),
    PRIMARY KEY(LeagueName, NewsTitle, NewsAuthor),
    FOREIGN KEY(LeagueName)
    REFERENCES Leagues
    ON DELETE CASCADE
    ON UPDATE CASCADE,
    FOREIGN KEY(NewsTitle)
    REFERENCES News
    ON DELETE CASCADE
    ON UPDATE CASCADE,
    FOREIGN KEY(NewsAuthor)
    REFERENCES News
    ON DELETE CASCADE
    ON UPDATE CASCADE);

CREATE TABLE SponsoredBy(
    TeamName            CHAR(20),
    startDate           CHAR(10),
```

endDate CHAR(10),
SponsorName CHAR(20),
PRIMARY KEY(TeamName, SponsorName),
FOREIGN KEY(TeamName)
REFERENCES Team
ON DELETE CASCADE
ON UPDATE CASCADE,
FOREIGN KEY(SponsorName)
REFERENCES Sponsors
ON DELETE CASCADE
ON UPDATE CASCADE);

CREATE TABLE Play(
MatchDate CHAR(10),
Stadium CHAR(20),
TeamName CHAR(20),
PRIMARY KEY(MatchDate, Stadium, TeamName, SportName),
FOREIGN KEY(MatchDate)
REFERENCES Matches
ON DELETE CASCADE
ON UPDATE CASCADE,
FOREIGN KEY(Stadium**)**
REFERENCES Matches
ON DELETE CASCADE
ON UPDATE CASCADE,
FOREIGN KEY(TeamName**)**
REFERENCES Teams
ON DELETE CASCADE
ON UPDATE CASCADE);

Inserts:

Date char format: DD/MM/YY

LEAGUES:

// Leagues Inserts

INSERT
INTO Leagues (LeagueName, Country*, StartDate, EndDate)
VALUES
(“English Premier League”, “England”, “11/08/23”, “19/05/24”),
(“EHF European League”, “europe”, “27/08/22”, “28/05/23”),
(“NBA”, “USA”, “18/10/22”, “12/06/23”),
(“MLB”, “USA”, “07/04/22”, “05/11/22”),

(“NHL”, “Canada”, “07/10/22”, “13/06/23”)

LeagueName	Country/Location	StartDate	EndDate
English Premier League	England	11/08/23	19/05/24
EHF European League	Europe	27/08/22	28/05/23
NBA	USA	18/10/22	12/06/23
MLB	USA	07/04/22	05/11/22
NHL	Canada	07/10/22	13/06/23

*maybe we can change this to location

NEWS:

INSERT
INTO News_1(title, Author, PublishingDate)
VALUES (“Missed opportunities finally catch up with Phillies in Game 3 loss to Diamondbacks”, “Mike Santa Barbara”, “19/10/23”),
(“Jerry Jeudy responds to viral interaction with Steve Smith Sr. after NFL great went on a rant about Broncos WR”, “ Garrett Podell”, 19/10/23),
(“ 'Then the alligators got him': Inside Ja Morant's 18-month downfall”, “Baxter Holmes”, “18/10/23”),
(“Man City could begin new era of European dominance”, “James Robson”, “09/06/23”),
(“How the growth of wearable technology is transforming football”, “Mark Carey”, “18/10/23”),
(“Fede Valverde and Eduardo Camavinga agree new Real Madrid contracts”, “Mario Cortegana”, “12/10/23”);

INSERT
INTO News_2(Author, Publisher)
VALUES (“Mike Santa Barbara”, “yardbarker.com”),
(“Garrett Podell”, “cbssports.com”),
(“Baxter Holmes”, “ESPN”),
(“James Robson”, “TSN”),
(“Mark Carey”, “The Athletic”),
(“Mario Cortegana”, “The Athletic”);

TEAMS:

// Teams Insert

INSERT
INTO Teams (TeamName, Location)
VALUES
("Los Angeles Lakers", "Los Angeles, CA, USA"),
("Vancouver Canucks", "Vancouver, BC, Canada"),
("FC Barcelona", "Barcelona, Spain"),
("New York Yankees", "New York, NY, USA")
("Baltimore Ravens", "Baltimore, MD")

TeamName	Location
Los Angeles Lakers	Los Angeles, CA, USA
Vancouver Canucks	Vancouver, BC, Canada
FC Barcelona	Barcelona, Spain
New York Yankees	New York, NY, USA
Baltimore Ravens	Baltimore, MD

MATCHES:

INSERT
INTO Matches_1(MatchDaTe, HomeTeam, LeagueName)
VALUES
("30/05/19", "Toronto Raptors" ,"NBA"),
("23/08/20", "Paris Saint-Germain FC", "UEFA Champions League"),
("13/2/22", "Los Angeles Rams", "NFL"),
("20/09/23", "draw" , "Galatasaray S.K.", "UEFA Champions League"),
("15/06/22", "Colorado Avalanche","NHL");

MatchesR2(MatchDate: char, Result: char, score: char, HomeTeam: char, AwayTeam: char, LeagueName: char)

INSERT
INTO Matches_2
VALUES
("30/05/19", "Toronto Raptors" , "109-118" ,"Toronto Raptors", "Golden State Warriors", "NBA"),
("23/08/20", "FC Bayern Munich", "0-1",, "Paris Saint-Germain FC", "FC Bayern Munich", "UEFA Champions League"),

(“13/2/22”, “Los Angeles Rams”, “23-20”, “Los Angeles Rams”, “Cincinnati Bengals”),
(“20/09/23”, “draw” , “2-2”, “Galatasaray S.K.”, “FC Copenhagen”, “UEFA Champions League”),
(“15/06/22”, “Colorado Avalanche”, “3-4”, “Colorado Avalanche”, “Tampa Bay Lightning”, “NHL”);

INSERT
INTO Matches_3(HomeTeam, Stadium)
VALUES (“Toronto Raptors”, “Scotiabank Arena”)
(“Estádio da Luz”, “Paris Saint-Germain FC”),
(“SoFi Stadium”, “Los Angeles Rams”),
(“Ali Sami Yen Spor Kompleksi”, “Galatasaray S.K.”),
(“Ball Arena”, “Colorado Avalanche”);

SPONSORS:

// Sponsors Inserts

```
INSERT
INTO  Sponsors (SponsorName, startDate, endDate)
 VALUES
("TeamViewer", "2021", "2026"),
("Zenni Optical", "2018", "2023"),
("Standard Chartered", "2010", "2027"),
("Spotify", "2022", "2027"),
("Jeep", "2012", "2024");
```

SponsorName	startDate	endDate
TeamViewer	2021	2026
Zenni Optical	2018	2023
Standard Chartered	2010	2027
Spotify	2022	2027
Jeep	2012	2024

SPONSORED_BY:

// SponsoredBy Inserts

```
INSERT
INTO  SponsoredBy (TeamName, SponsorName)
 VALUES
("Chelsea FC", "Infinite Athlete"),
("Toronto Raptors", "Sun Life Financial"),
("Vancouver Whitecaps FC", "Telus Communications Inc."),
("Chicago Bulls", "Motorola Mobility"),
("Golden State Warriors", "Rakuten")
```

TeamName	SponsorName
Chelsea FC	Infinite Athlete
Toronto Raptors	Sun Life Financial
Vancouver Whitecaps FC	Telus Communications Inc.
Chicago Bulls	Motorola Mobility
Golden State Warriors	Rakuten

TEAM_STATISTICS:

// TeamStatistics Inserts

```
INSERT
INTO  TeamStatistics (MatchesDrew, MatchesWon, MatchesLost,
AvarageScored,  AvarageScoredAgainst, CurrentTableRanking, Year,
TeamName)
 VALUES
(3, 9, 1, 1.77, 0.31, 1, 2020, “Chelsea FC”),
(0, 6, 2, 2.25, 1.0, 9, 2020,  “Juventus FC”),
(0, 65, 41, 111.8, 104.5, 2, 2021, “Boston Celtics”),
(0, 69, 35, 111.0, 105.5, 1, 2021, “Golden State Warriors”),
(3, 1, 2, 1.17, 1.67, 17, 2020, “FC Internazionale Milano”);
```

MatchesDrew	MatchesWon	MatchesLost	AvarageScored	AvarageScoredAgainst	CurrentTableRanking	Year	TeamName
3	9	1	1.77	0.31	1	2020	Chelsea FC
0	6	2	2.25	1.0	9	2020	Juventus FC
0	65	41	111.8	104.5	2	2021	Boston Celtics
0	69	35	111.0	105.5	1	2021	Golden State Warriors
3	1	2	1.17	1.67	17	2020	FC Internazionale Milano

SPORT_PEOPLE:

// SportPeople Inserts:

```
INSERT
INTO  SportPeople (Nationality, Name, Age)
 VALUES
(“Norwegian, “Erling Braut Haaland”, 23),
(“Canadian”, “Anthony Beauvillier”, 26),
(“American”, “Kevin Wayne Durant”, 35),
(“French”, “Karim Mostafa Benzema”, 35),
(“Argentine”, “Damián Emiliano Martínez”, 31);
```

Nationality	Name	Age
Norwegian	Erling Braut Haaland	23
Canadian	Anthony Beauvillier	26
American	Kevin Wayne Durant	35
French	Karim Mostafa Benzema	35
Argentine	Damián Emiliano Martínez	31

COACHES:

// Coaches Inserts:

```
INSERT
INTO Coaches (NumOfSeasons, Name, Age)
 VALUES
(10, “Darko Rajakovic”, 44),
(22, “Jürgen Klopp”, 56),
(17, “Stefano Pioli”, 57),
(20, “Frank Vogel”, 50),
(14, “Quin Snyder”, 56);

// numOfSeasons include seasons as head coach AND assistant coach
```

NumOfSeasons	Name	Age
10	Darko Rajakovic	44
22	Jürgen Klopp	56
17	Stefano Pioli	57

20	Frank Vogel	50
14	Quin Snyder	56