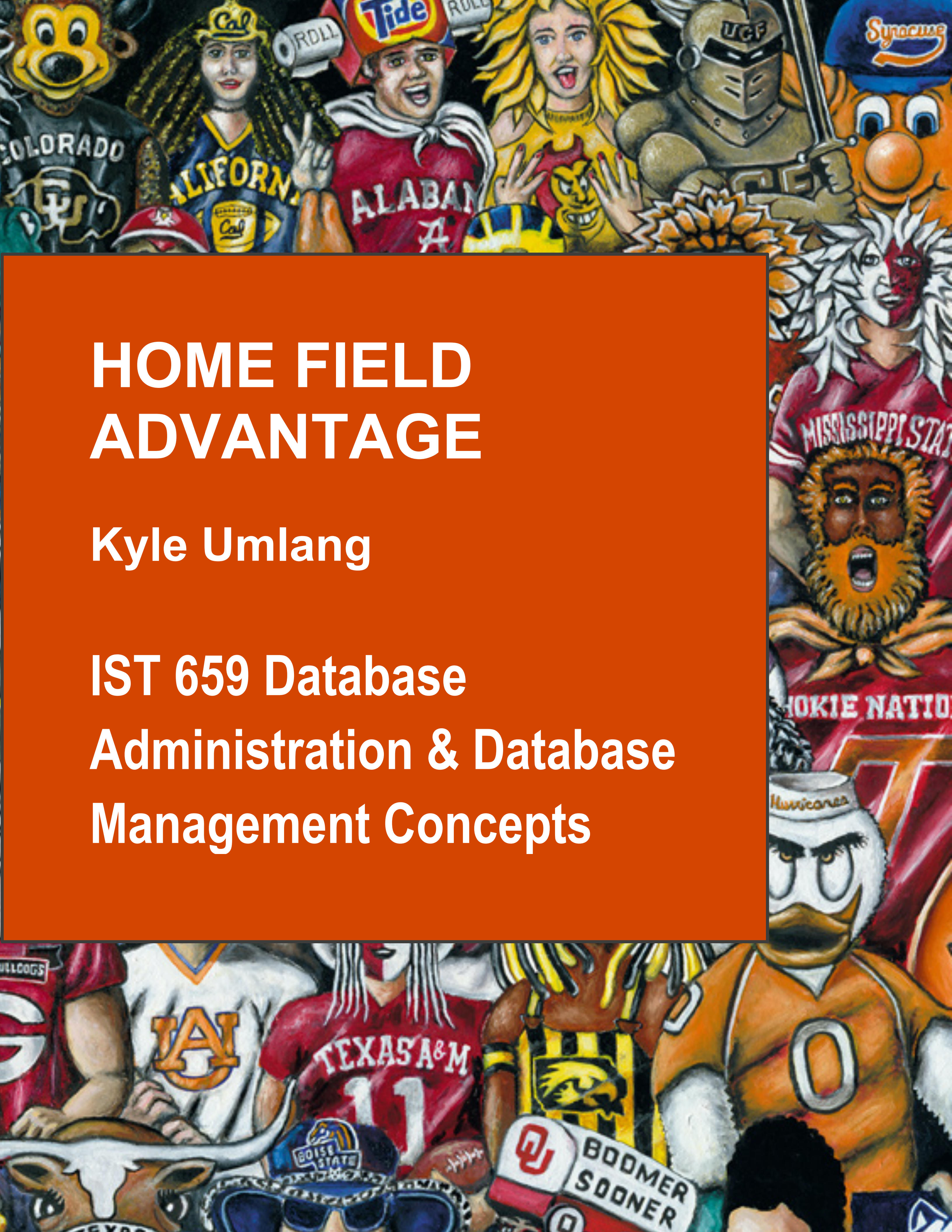


HOME FIELD ADVANTAGE

Kyle Umlang

IST 659 Database Administration & Database Management Concepts

Summary

In the wild world of sports, home field advantage is a well-known, yet almost immeasurable, benefit any player/fan would like to have on their side when entering a game. Does crowd size play a factor in this? How about average temperature? This database will provide an answer. When looking at a team's win percentage for home games vs. away games, it's hard to judge the full impact of one's home field advantage because in most sports, especially college football, anything can happen in the last possession. Whether it's a last second "hail mary", a blocked field goal or one of those wild trick plays that will live in replay eternity, a loss, no matter how it happens, is still a loss. This database digs deeper than wins and losses. It houses fifteen years of home and away yards and points per game for all 129 NCAA Division 1 Football Bowl Subdivision teams' offenses and defenses. This database will finally give sports analysts, college football enthusiasts, and loyal sports fans the answer they have unknowingly been questioning their entire fandom: is home field advantage real and more importantly, which teams stand to gain the most from theirs? Using SQL as the DBMS, and building all the tables listed later in the Normalized Logical Model, this database is able to quickly answer the following questions about Home Field Advantage:

Questions this database answers:

1. In the last 15 years, which team has had the best Home Field Advantage in a single season?

2. Which team has had the best overall Home Field Advantage over the past 5 years combined?

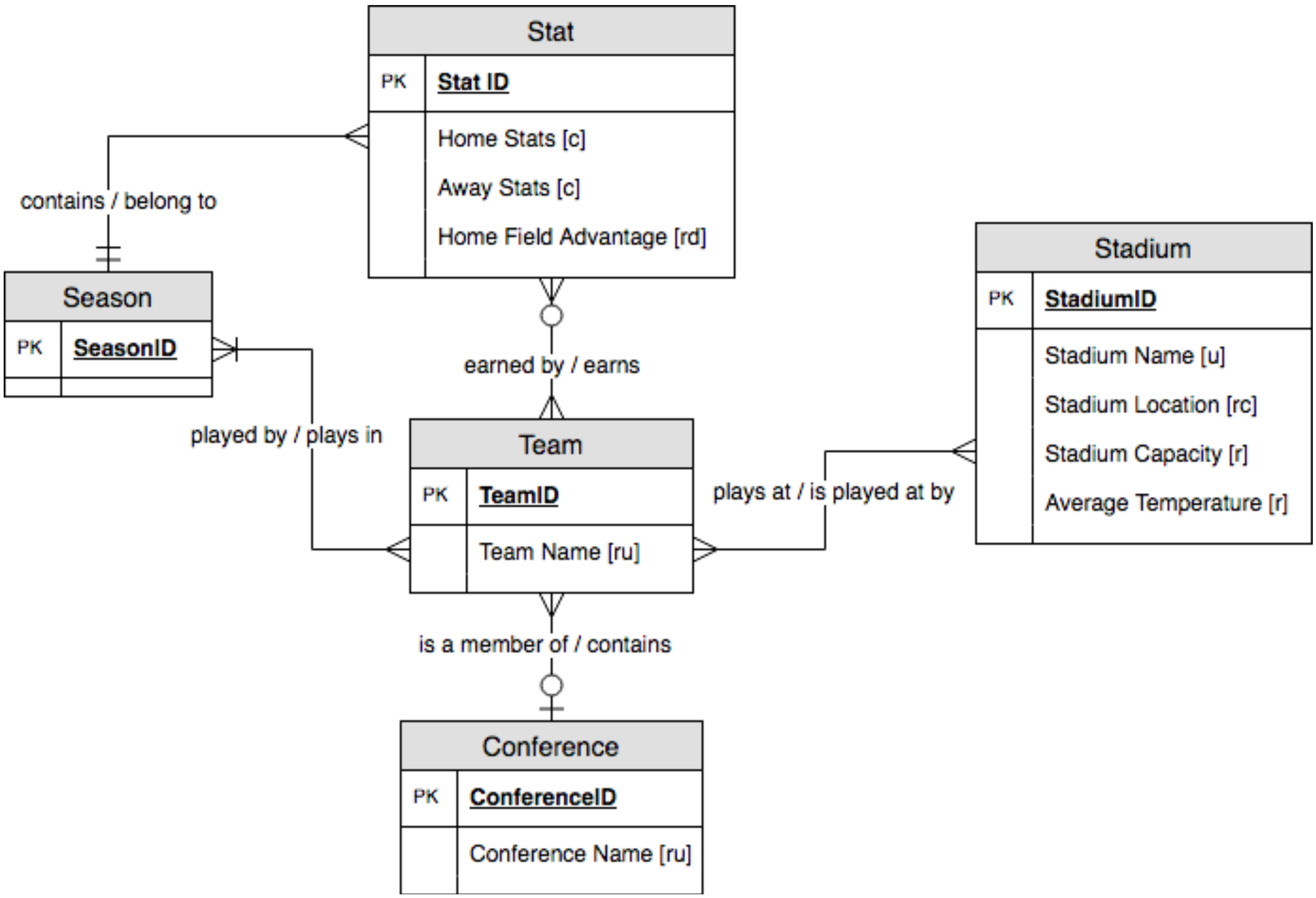
3. Is there a correlation between the size of a team's stadium and their team's Home Field Advantage?

4. Is there a correlation between the average temperature of a stadium's location and their team's Home Field Advantage?

5. Based on Home Field Advantage, can this DB accurately predict the winner of each Top 25 Week 1 Matchups held September 1, 2018?

Conceptual Model

Stakeholders for this database are sports analysts, college football enthusiasts, coaches, gamblers and general sports fans. As of 2017, there are 129 Division 1 Football Bowl Subdivision teams. 125 are broken out into 10 conferences based on geographical location and 4 teams do not belong to a conference (Independents). A conference contains many teams ranging from 10-14 members. A team plays 12 games in a season and can play in one or more seasons as a FBS team. Around 10 of these games are against other FBS teams. All data in this database is from games involving two FBS teams and is presented as “per game” averages to make up for those teams that play more or less than the 10 FBS game average. Each season, teams play roughly half of their games at their own stadium, known as home games. The stats from these home games, known as home stats, (*home yards per game, home yards allowed per game, home points per game, home points allowed per game*) will be compared against the stats from the games at opponents stadiums, or away stats (*away yards per game, away yards allowed per game, away points per game, away points allowed per game*) to determine the difference in performance, or home field advantage (*HFA*). Each season, teams can have a new set of home and away stats. Teams with positive season HFAs will be those whose yards per point difference between offense and defense is greater at home. An all-time HFA can be calculated using 15 year averages of all the stats which can then be compared to stadium location and capacity to determine if there’s any correlations.



Glossary

Below is a glossary and definitions describing the entities, attributes, and relationships in the above Conceptual ERD model.

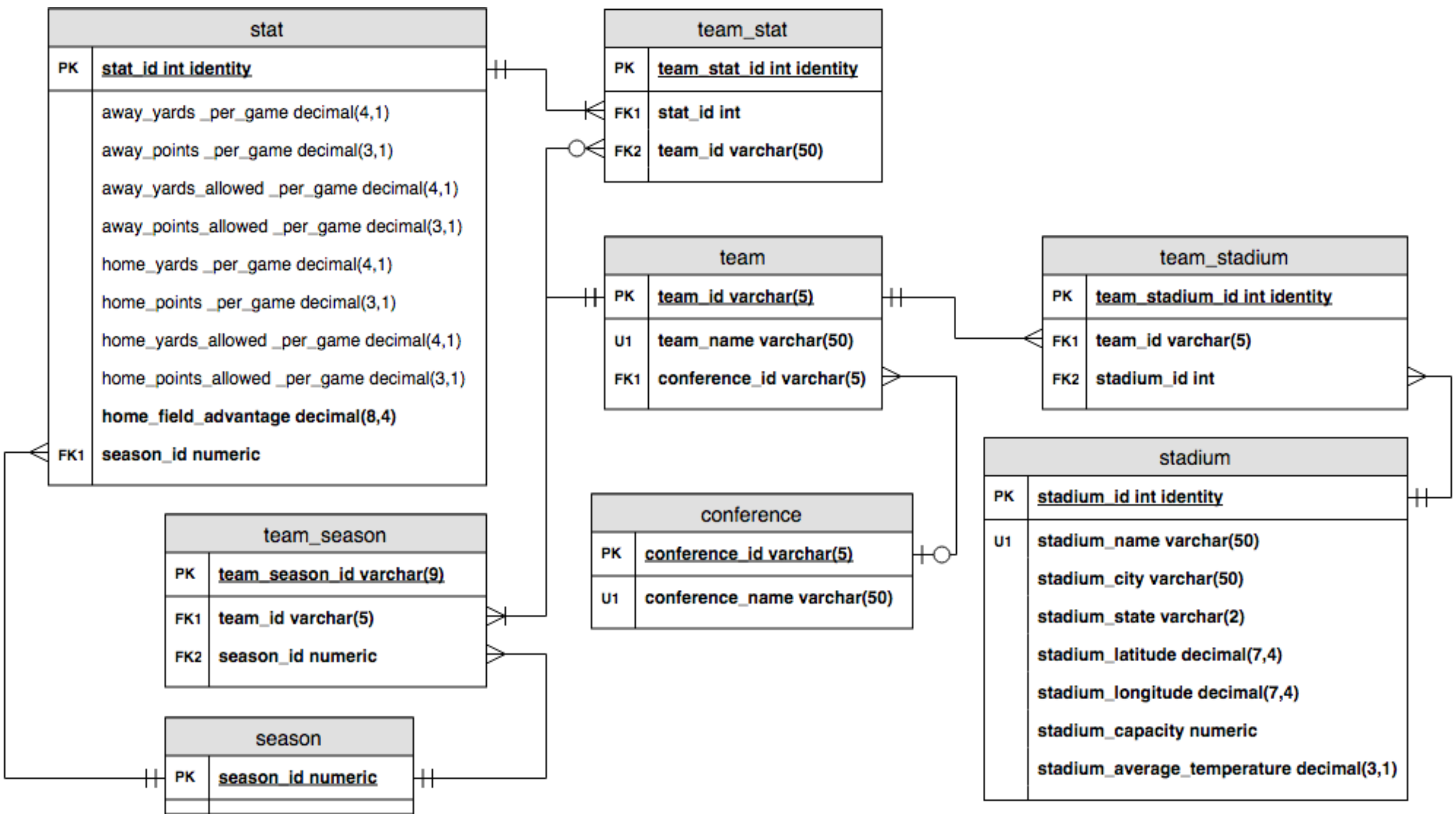
Entity	**Attribute**
Team	Team Name [ru]
Stat	Home Stats [c] Away Stats [c] Home Field Advantage [rd]
Conference	Conference Name [ru]
Stadium	Stadium Name [u] Stadium Location [rc] Stadium Capacity [r] Stadium Average Temperature [r]
Season	
Relationships	
Each team plays at many stadiums, each stadium is played at by many teams A team can be a member of one and only one conference and a conference can contain many teams as members Each team plays in one or more seasons, while each season is played by many teams A team can earn 0 or more stats, while a stat can be earned by many teams Each season contains many stats, while each stat belongs to one and only one season	

TEAM NAME	Name of a school's football team
HOME STATS (HS)	Stats earned while playing at team's stadium (HOME YARDS ALLOWED PER GAME / HOME POINTS ALLOWED PER GAME) - (HOME YARDS PER GAME / HOME POINTS PER GAME)
AWAY STATS (AS)	Stats earned while playing at team's stadium (AWAY YARDS ALLOWED PER GAME / AWAY POINTS ALLOWED PER GAME) - (AWAY YARDS PER GAME / AWAY POINTS PER GAME)
HOME FIELD ADVANTAGE	(HS – AS) / AS
CONFERENCE NAME	Name of the conference in which a school's football team belongs
STADIUM NAME	Name of a school's football team's stadium
STADIUM LOCATION	City, state, latitude and longitude of a stadium
STADIUM CAPACITY	Number of people a stadium can hold
STADIUM AVERAGE TEMPERATURE	Average temperature of where the stadium is located

Normalized Logical Model

After reviewing the data and normalizing the conceptual model, the below diagram shows the final logical model that will be implemented in the database. The primary keys for Team ID and Conference ID are set to varchar(5) because this abbreviation nomenclature is synonymous in the sports world with actual names of teams (example – Kentucky = UK, Oklahoma = OU, South Eastern Conference = SEC).

- Stadium capacity and season ID are strictly numbers so their data type is set to NUMERIC
- Stadium name, stadium city, conference name and team name are all varchar(50) while stadium state is varchar(2)
- Home Field Advantage is DECIMAL (8,4) as this data type will be presented as a percentage (Ex – 0.3765 = 37.65%) with possibilities of being over 100%
- All yard averages are DECIMAL(4,1) while all point averages are DECIMAL(3,1)
- Stadium Latitude and Longitude are DECIMAL(7,4) as that is the standard for coordinates
- Stadium Average Temperature is DECIMAL(3,1)



Physical Database Design

SQL DDL Commands

Below is the SQL statement I used to create all the tables in my database and is executable from top to bottom. For convenience, I have included the necessary drop table statements at the beginning of the script.

```
-- Check for and drop Team_Stat Table
IF exists (SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE table_name = 'Team_Stat')
BEGIN
	DROP TABLE Team_Stat
END
GO
-- Check for and drop Team Season Table
IF exists (SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE table_name = 'Team_Season')
BEGIN
	DROP TABLE Team_Season
END
GO
-- Check for and drop Stat Table
IF exists (SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE table_name = 'Stat')
BEGIN
	DROP TABLE Stat
END
GO
-- Check for and drop Season Table
IF exists (SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE table_name = 'Season')
BEGIN
	DROP TABLE Season
END
GO
-- Check for and drop Team Stadium Table
IF exists (SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE table_name = 'Team_Stadium')
BEGIN
	DROP TABLE Team_Stadium
END
GO
-- Check for and drop Team Table
IF exists (SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE table_name = 'Team')
BEGIN
	DROP TABLE Team
END
GO
-- Check for and drop Stadium Table
IF exists (SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE table_name = 'Stadium')
BEGIN
	DROP TABLE Stadium
END
GO
-- Check for and drop Conference Table
IF exists (SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE table_name = 'Conference')
BEGIN
	DROP TABLE Conference
END
GO
```

```sql
-- Create Table Conference
CREATE TABLE Conference (
       conference_id varchar(5),
       conference_name varchar(50) not null
       CONSTRAINT PK_Conference PRIMARY KEY (conference_id),
       CONSTRAINT U1_Conference UNIQUE (conference_name))

-- Create Table Stadium
CREATE TABLE Stadium (
       stadium_id int identity,
       stadium_name varchar(50) not null,
       stadium_city varchar(50) not null,
       stadium_state varchar(2) not null,
       stadium_latitude decimal(7,4) not null,
       stadium_longitude decimal(7,4) not null,
       stadium_capacity numeric not null,
       stadium_average_temperature decimal(3,1) not null
       CONSTRAINT PK_Stadium PRIMARY KEY (stadium_id),
       CONSTRAINT U1_Stadium UNIQUE (stadium_name))

-- Create Table Team
CREATE TABLE Team (
       team_id varchar(5),
       team_name varchar(50) not null,
       conference_id varchar(5) not null
       CONSTRAINT PK_Team PRIMARY KEY (team_id),
       CONSTRAINT U1_Team UNIQUE (team_name),
       CONSTRAINT FK1_Team FOREIGN KEY (conference_id) REFERENCES Conference(conference_id))

-- Create Table Team Stadium
CREATE TABLE Team_Stadium (
       team_stadium_id int identity,
       team_id varchar(5) not null,
       stadium_id int not null
       CONSTRAINT PK_Team_Stadium PRIMARY KEY (team_stadium_id),
       CONSTRAINT FK1_Team_Stadium FOREIGN KEY (team_id) REFERENCES Team(team_id),
       CONSTRAINT FK2_Team_Stadium FOREIGN KEY (stadium_id) REFERENCES Stadium(stadium_id))

-- Create Table Season
CREATE TABLE Season (
       season_id numeric
       CONSTRAINT PK_Season PRIMARY KEY (season_id))

-- Create Table Stat
CREATE TABLE Stat (
       stat_id int identity,
       away_yards_per_game decimal(4,1),
       away_points_per_game decimal(3,1),
       away_yards_allowed_per_game decimal(4,1),
       away_points_allowed_per_game decimal(3,1),
       home_yards_per_game decimal(4,1),
       home_points_per_game decimal(3,1),
       home_yards_allowed_per_game decimal(4,1),
       home_points_allowed_per_game decimal(3,1),
       home_field_advantage decimal(8,4) not null,
       season_id numeric not null
       CONSTRAINT PK_Stat PRIMARY KEY (stat_id),
       CONSTRAINT FK1_Stat FOREIGN KEY (season_id) REFERENCES Season(season_id))
```

```sql
-- Create Table Team Season
CREATE TABLE Team_Season (
       team_season_id varchar(9),
       team_id varchar(5) not null,
       season_id numeric not null
       CONSTRAINT PK_Team_Season PRIMARY KEY (team_season_id),
       CONSTRAINT FK1_Team_Season FOREIGN KEY (team_id) REFERENCES Team(team_id),
       CONSTRAINT FK2_Team_Season FOREIGN KEY (season_id) REFERENCES Season(season_id))

-- Create Table Team Stat
CREATE TABLE Team_Stat (
       team_stat_id int identity,
       stat_id int not null,
       team_id varchar(5) not null
       CONSTRAINT PK_Team_Stat PRIMARY KEY (team_stat_id),
       CONSTRAINT FK1_Team_Stat FOREIGN KEY (stat_id) REFERENCES Stat(stat_id),
       CONSTRAINT FK2_Team_Stat FOREIGN KEY (team_id) REFERENCES Team(team_id))
```

More SQL DDL

Below are 5 programming objects consisting of views, functions and stored procedures.

```sql
-- (#1) View to retrieve all Stats for all teams for all seasons
CREATE VIEW AllTeamStats AS
       SELECT t.team_id, st.season_id, st.home_yards_allowed_per_game,
st.home_points_allowed_per_game, st.away_yards_allowed_per_game,
       st.away_points_allowed_per_game, home_field_advantage
       FROM stat st, team_stat tst, team t
       WHERE tst.stat_id = st.stat_id and tst.team_id = t.team_id
GO

SELECT * FROM AllTeamStats


-- (#2) View to retrieve list of all team data
CREATE VIEW AllTeamInfo AS
       SELECT t.team_name, c.conference_name, s.stadium_name, s.stadium_city, s.stadium_state,
s.stadium_capacity, s.stadium_average_temperature
       FROM team t, team_stadium ts, stadium s,  conference c
       WHERE t.team_id = ts.team_id and s.stadium_id = ts.stadium_id and t.conference_id =
c.conference_id
GO

SELECT * FROM AllTeamInfo


-- (#3) View to retrieve list of teams by Conference
CREATE VIEW AllConferences AS
       SELECT c.conference_name, t.team_name
       FROM conference c, team t
       WHERE t.conference_id = c.conference_id
       GROUP by c.conference_name, t.team_name
GO

SELECT * FROM AllConferences
```

```sql
-- (#4) Procedure to update a stadium’s capacity
-- The first parameter is the stadium_name for the user to change
-- The second is the new capacity

CREATE PROCEDURE UpdateStadiumCapacity(@stadium_name varchar(50), @newCapacity numeric)
AS
BEGIN
       UPDATE stadium SET stadium_capacity = @newCapacity
       WHERE stadium_name = @stadium_name
END
GO


EXEC UpdateStadiumCapacity 'Beaver Stadium', '110000'



SELECT * FROM Stadium WHERE stadium_name = 'Beaver Stadium'



-- (#5) Function to retrieve a specific Team's Home Field Advantage
-- Gets the stat_id of the stat record that matches the parameter
-- and assigns that value to @returnValue.

CREATE FUNCTION dbo.HomeFieldAdvantageLookup(@statid int)
RETURNS decimal(8,4) AS -- Home Field Advantage is a decimal
BEGIN

       DECLARE @returnValue decimal(8,4) -- Matches the function's return type

       SELECT @returnValue = home_field_advantage FROM stat
       WHERE stat_id = @statid

       RETURN @returnValue
END
GO


SELECT dbo.HomeFieldAdvantageLookup('1234')
```

Data Creation

SQL DML INSERT Statements

Below are 10 SQL INSERT Statements for creating data throughout the various tables in my database. NOTE – The data in the below insert statements does not represent all of my data.

```
-- (#1) Adding data to the Conference table
INSERT INTO Conference
              (conference_id, conference_name)
VALUES ('B12','Big 12 Conference'),
       ('AAC','American Athletic Conference'),
       ('MAC','Mid-American Conference')
```

```
-- (#2) Adding data to the Stadium table
INSERT INTO Stadium
              (stadium_name, stadium_city, stadium_state, stadium_latitude, stadium_longitude,
stadium_capacity,stadium_average_temperature)
VALUES ('Alamodome','San Antonio','TX','29.4722','-98.5247','65000','80.4'),
       ('Albertsons Stadium','Boise','ID','43.6135','-116.2035','36387','64'),
       ('Allen E. Paulson Stadium','Statesboro','GA','32.4376','-81.775','25000','79.4')
```

```
-- (#3) Adding data to the Team table
INSERT INTO Team
              (team_id, team_name,conference_id)
VALUES ('A&M','Texas A&M','SEC'),
       ('AF','Air Force','MWC'),
       ('AKRON','University of Akron','MAC'
```

```
-- (#4) Adding data to the Team_Stadium table
INSERT INTO Team_Stadium
              (team_id, stadium_id)
VALUES ('A&M','70'),
('AF','41'),
('AKRON','57')
```

```
-- (#5) Adding data to the Season table
INSERT INTO Season
              (season_id)
VALUES ('2003'),
('2004'),
('2005')
```

```sql
-- (#6) Adding data to the Stat table
INSERT INTO Stat
              (away_yards_per_game, away_points_per_game, away_yards_allowed_per_game,
away_points_allowed_per_game, home_yards_per_game, home_points_per_game,
home_yards_allowed_per_game, home_points_allowed_per_game, Home_Field_Advantage, season_id)
VALUES('427','31.2','405.7','26.7','447.2','28.5','343','19.7','0.139893991115429','2010'),
      ('398.7','23.9','372.7','22.1','432','33.8','325.4','19.6','19.966145538591','2010'),
      ('250.7','14.5','421.3','35.8','269.2','12.6','448.2','33.6','-0.312030811147658','2010')



-- (#7) Adding data to the Team_Stat table
INSERT INTO Team_Stat
              (stat_id, team_id)
VALUES
('965','FIU'),
('966','FL'),
('967','FLST')



-- (#8) Adding data to the Team_Season table
INSERT INTO Team_Season
              (team_season_id, team_id, season_id)
VALUES  ('2011LTECH','LTECH','2011'),
        ('2011LOU','LOU','2011'),
        ('2011LSU','LSU','2011')



-- (#9) Adding MORE data to the Conference table
INSERT INTO Conference
              (conference_id, conference_name)
VALUES ('B1G','Big Ten Conference'),
       ('SUN','Sun Belt Conference'),
       ('PAC','Pac-12 Conference')



-- (#10) Adding MORE data to the Stadium table
INSERT INTO Stadium
              (stadium_name, stadium_city, stadium_state, stadium_latitude, stadium_longitude,
stadium_capacity,stadium_average_temperature)
VALUES ('Cajun Field','Lafayette','LA','30.2097','-92.0314','41426','79.8'),
       ('California Memorial Stadium','Berkeley','CA','37.8723','-122.276','62717','69.2'),
       ('Camp Randall Stadium','Madison','WI','43.0809','-89.3921','80321','56.8')
```

Data Manipulation

SQL DML UPDATE Statements

Below are 5 SQL INSERT Statements for updating/deleting data throughout the various tables in my database.

```
-- (#1) UPDATING Missouri's Conference to the SEC
-- since they moved from the Big Twelve Conference in 2012

UPDATE Team SET conference_id = 'SEC'
WHERE team_name = 'Missouri'


-- (#2) DELETING the 2005 USC Season because of Reggie Bush's violations

DELETE Team_Season
WHERE team_id = 'USC' and season_id = '2005'


-- (#3) UPDATING Kent State's Team Name to the correct name.
-- It was incorrectly labeled as 'Kentucky State'

UPDATE Team SET team_name = 'Kent State University'
WHERE team_id = 'KENST'


-- (#4) DELETING the Big East Conference as it is no longer active

DELETE Conference
WHERE conference_id = 'BEC'


-- (#5) UPDATING Wake Forest's Stadium's State

UPDATE Stadium SET stadium_state = 'NC'
WHERE stadium_name = 'BB&T Field'
```

Answering Data Questions

SQL DML SELECT Statements

Below are the sql statements used to answer the 5 questions posed at the beginning of this document:

1. In the last 15 years, which team has had the best Home Field Advantage in a single season?

   Answer: The 2006 Boston College team had the best Home Field Advantage of all time

```
SELECT t.team_id, st.season_id, st.home_field_advantage
FROM stat st, team_stat tst, team t
WHERE tst.stat_id = st.stat_id and tst.team_id = t.team_id
ORDER by st.home_field_advantage DESC
```

	team_id	season_id	home_field_advantage
1	BC	2006	6947.4024
2	FL	2006	908.6678
3	LAFAY	2011	660.2035
4	BAMA	2007	612.4875
5	APPST	2014	611.5674
6	MSHL	2007	551.5795
7	UCONN	2007	525.9635
8	BYU	2010	364.0047
9	UCLA	2012	165.1928
10	CINCY	2003	135.1333

2. Which team has had the best overall Home Field Advantage over the past 5 years combined?

Answer: Appalachian State has had the best Home Field Advantage in the last 5 years combined.

```
SELECT t.team_id, avg(st.home_field_advantage)
FROM stat st, team_stat tst, team t, season s
WHERE tst.stat_id = st.stat_id and tst.team_id = t.team_id and st.season_id in ('2013',
'2014', '2015', '2016', '2017')
GROUP BY t.team_id
ORDER by avg(st.home_field_advantage) DESC
```

Results | Messages

	team_id	(No column name)
1	APPST	153.762325
2	ECU	25.753580
3	CCAR	16.159900
4	TOL	15.842360
5	WIS	12.001120
6	BAMA	11.979620
7	MSHL	11.360640
8	COL	10.881320
9	PUR	10.419540
10	WMICH	9.136900

3. Is there a correlation between the size of a team's stadium and their team's Home Field Advantage?

Answer: No there's no correlation. (***NOTE – I looked at the past 5 years only as the last 5 years fully represents a team's present day status. I don't want a team's successes or failures 15 years ago to affect what their Home Field Advantage actually is today)*** Looking at the top 20 teams by stadium capacity you can see only 3 teams (Alabama, Florida and Wisconsin) hold one of the top 20 spots (green colored cells) regarding Home Field Advantage. Outliers like Ohio State rank 3rd in capacity but 101st in HFA whereas Coastal Carolina (not pictured) ranks 127th in capacity but 3rd in HFA. Below the table is the SQL select statement that gives you the data. I added a simple pivot table and conditional formatting to the data in order to get the colorful table to the right.

Results | Messages

	team_id	stadium_capacity	home_field_advantage
1	A&M	102733	-1.239800
2	A&M	102733	-0.559100
3	A&M	102733	1.123200
4	A&M	102733	6.262200
5	A&M	102733	17.426000
6	AF	46692	-0.244400
7	AF	46692	-0.209900
8	AF	46692	1.054900
9	AF	46692	3.421300
10	AF	46692	4.056700
11	AKRON	30000	-3.497200
12	AKRON	30000	0.248400
13	AKRON	30000	2.222100
14	AKRON	30000	2.668900
15	AKRON	30000	4.982600
16	APPST	30000	0.148800
17	APPST	30000	1.625400
18	APPST	30000	1.707700
19	APPST	30000	611.567400
20	ARK	72000	0.292100
21	ARK	72000	2.324200
22	ARK	72000	4.288000

team_id	stadium_capacity	home_field_advantage
MICH	107601	2.23778129
PENN	106572	1.71393705
OHST	104944	0.67081153
A&M	102733	4.60249701
TENN	102455	2.22170805
LSU	102321	3.36844389
BAMA	101821	11.979612
TEX	100119	-0.04240357
USC	93607	5.63846911
UGA	92746	0.41622437
UCLA	90888	0.27049601
FLOR	88548	7.70571656
AU	87451	1.72228873
OU	86112	2.05641611
NEB	85458	0.3430943
CLEM	81500	3.47706175
WIS	80321	12.0011208
SCAR	80250	1.30135715
FLST	79560	3.86164945

```
SELECT ts.team_id, s.stadium_capacity,
AVG(st.home_field_advantage) as home_field_advantage
FROM stat st, stadium s, team_stadium ts, team_stat tstat, season sea
WHERE st.stat_id = tstat.stat_id and ts.team_id = tstat.team_id
and s.stadium_id = ts.stadium_id and sea.season_id = st.season_id
and sea.season_id in ('2013', '2014', '2015', '2016', '2017')
GROUP BY ts.team_id, s.stadium_capacity, st.home_field_advantage
```

4. Is there a correlation between the average temperature of a stadium's location and their team's Home Field Advantage?

Answer: Yes there is a small correlation. (***NOTE – I looked at the past 5 years only as the last 5 years fully represents a team's present day status. I don't want a team's successes or failures 15 years ago to affect what their Home Field Advantage actually is today)*** Looking at the top 20 teams by stadium average temperature, you can see that 4 teams (Central Florida, Miami, South Florida and Florida) hold one of the top 20 spots (green colored cells) regarding Home Field Advantage. These 4 teams all take advantage of these weather conditions when playing opponents at home. However there are some outliers like Florida Atlantic and Florida International. They are also in Florida and rank 2nd and 3rd in average temperature but 114th and 109th in HFA, so the outliers don't make it a strong correlation. Below the table is the SQL select statement that gives you the data. I added a simple pivot table and conditional formatting in order to get the colorful table to the right.

Results | Messages

	team_id	stadium_average_temperature	home_field_advantage
1	A&M	80.2	-1.239800
2	A&M	80.2	-0.559100
3	A&M	80.2	1.123200
4	A&M	80.2	6.262200
5	A&M	80.2	17.426000
6	AF	54.8	-0.244400
7	AF	54.8	-0.209900
8	AF	54.8	1.054900
9	AF	54.8	3.421300
10	AF	54.8	4.056700
11	AKRON	61.4	-3.497200
12	AKRON	61.4	0.248400
13	AKRON	61.4	2.222100
14	AKRON	61.4	2.668900
15	AKRON	61.4	4.982600
16	APPST	62.2	0.148800
17	APPST	62.2	1.625400
18	APPST	62.2	1.707700
19	APPST	62.2	611.567400
20	ARK	60.6	0.292100
21	ARK	60.6	2.324200
22	ARK	60.6	4.288000

team_id	stadium_average_temperature	home_field_advantage
HAWA	86.0	2.24235785
FIU	85.2	0.24996219
FAU	84.8	0.45758234
ARZ	83.8	5.43587582
UTEP	83.8	1.61964289
MIA	83.0	7.42819011
UCF	83.0	7.29387139
USF	82.6	7.97156859
UNLV	82.2	-0.39452108
TEX	80.8	-0.04240357
UCLA	80.6	0.27049601
TXST	80.6	0.68499305
UTSA	80.4	0.55100391
A&M	80.2	4.60249701
FLOR	80.0	7.70571656
FLST	79.8	3.86164945
LALAF	79.8	4.01760456
GASTH	79.4	1.27338436
RICE	79.2	2.19938159
UH	79.2	2.07814984

```
SELECT ts.team_id, s.stadium_average_temperature,
AVG(st.home_field_advantage) as home_field_advantage
FROM stat st, stadium s, team_stadium ts, team_stat tstat, season sea
WHERE st.stat_id = tstat.stat_id and ts.team_id = tstat.team_id
and s.stadium_id = ts.stadium_id and sea.season_id = st.season_id
and sea.season_id in ('2013', '2014', '2015', '2016', '2017')
GROUP BY ts.team_id, s.stadium_average_temperature, st.home_field_advantage
```

5. Based on Home Field Advantage, can this DB accurately predict the winner of each Top 25 Week 1 Matchups held September 1, 2018?

Answer: Yes, it accurately predicted the winners of 6/8 games (75%) involving a Top 25 ranked team. (***NOTE – I looked at the past 5 years only as the last 5 years fully represents a team's present day status. I don't want a team's successes or failures 15 years ago to affect what their Home Field Advantage actually is today)*** Most week 1 games involving a ranked team are usually against "cupcake teams" and the majority of Top 25 vs. Top 25 games are played at neutral stadiums, so I tried my best to fill the remaining games with similarly skilled opponents who also played at an actual home stadiums. As an extra little test, I tried to see if my DB could also correctly predict the score of any of these games and it did get very close to a few of the scores. Here are the results of those predictions:

My Database		Actual Scores	
#14 Michigan	24	#14 Michigan	17
#12 Notre Dame	**34**	**#12 Notre Dame**	**24**
#20 Virginia Tech	25	**#20 Virginia Tech**	**24**
#19 Florida State	**44**	#19 Florida State	3
Oregon State	17	Oregon State	31
#5 Ohio State	**92**	**#5 Ohio State**	**77**
North Carolina	23	North Carolina	17
California	**37**	**California**	**24**
BYU	27	**BYU**	**28**
Arizona	**32**	Arizona	23
FAU	16	FAU	14
#7 Oklahoma	**62**	**#7 Oklahoma**	**63**
San Diego State	22	San Diego State	10
#13 Stanford	**50**	**#13 Stanford**	**31**
Western Kentucky	20	Western Kentucky	3
#4 Wisconsin	**61**	**#4 Wisconsin**	**34**

Below is the SQL select statement that gives you the above score prediction data:

```sql
SELECT t.team_id AS Team, st.away_yards_per_game AS YardsPG,
st.away_yards_allowed_per_game AS YardsAllowedPG,
st.away_points_per_game AS PointsPG, st.away_points_allowed_per_game AS PointsAllowedPG,
((AVG(st.away_yards_allowed_per_game)/AVG(st.away_points_allowed_per_game))-
(AVG(st.away_yards_per_game)/AVG(st.away_points_per_game))) AS YPP
 FROM stat st, team_stat tstat, season sea, team t, team_season ts
WHERE st.stat_id = tstat.stat_id and ts.team_id = tstat.team_id and t.team_id =
ts.team_id and sea.season_id = st.season_id
and sea.season_id in ('2013', '2014', '2015', '2016', '2017')
and t.team_name in ('Michigan')
GROUP BY t.team_id, st.away_yards_per_game, st.away_yards_allowed_per_game,
st.away_points_per_game, st.away_points_allowed_per_game
UNION
SELECT t.team_id AS home_team, st.home_yards_per_game, st.home_yards_allowed_per_game,
st.home_points_per_game, st.home_points_allowed_per_game,
((AVG(st.home_yards_allowed_per_game)/AVG(st.home_points_allowed_per_game))-
(AVG(st.home_yards_per_game)/AVG(st.home_points_per_game))) AS HOMEYPP
 FROM stat st, team_stat tstat, season sea, team t, team_season ts
WHERE st.stat_id = tstat.stat_id and ts.team_id = tstat.team_id and t.team_id =
ts.team_id and sea.season_id = st.season_id
and sea.season_id in ('2013', '2014', '2015', '2016', '2017')
and t.team_name in ('Notre Dame')
GROUP BY t.team_id, st.home_yards_per_game, st.home_yards_allowed_per_game,
st.home_points_per_game, st.home_points_allowed_per_game
```

Once I have the below results, I paste them into Excel where I have formulas to calculate the averages for the data and then calculate projected yards and points and yards per point using a mix of the away team and home team allowed averages and finally taking that projected YPP and divide by the average number of yards for both teams to calculate a projected score. If the number is a weird football score (ie, 22 and 33) I round it to nearest normal football score (ie, 24 and 34).

Results | Messages

	Team	YardsPG	YardsAllowedPG	PointsPG	PointsAllowedPG	YPP
1	MICH	272.5	356.5	22.0	27.8	0.437378
2	MICH	287.8	376.4	14.6	28.6	-6.551489
3	MICH	324.3	304.3	23.6	21.3	0.544859
4	MICH	359.8	274.2	36.4	20.0	3.825385
5	MICH	409.3	318.3	31.8	19.0	3.881562
6	ND	393.0	371.7	23.5	18.5	3.368487
7	ND	424.3	386.7	28.8	27.0	-0.410416
8	ND	429.7	389.7	35.7	24.7	3.740913
9	ND	475.7	362.4	39.4	19.3	6.703598
10	ND	480.3	370.0	40.0	19.0	7.466184

Team	YardsPG	YardsAllowedPG	PointsPG	PointsAllowedPG	YPP
MICH	330.74	325.94	25.68	23.34	1.085583692
ND	440.6	376.1	33.48	21.7	4.171701656

Team	Projected Yards	Projected Points	Projected YPP	Projected Score
MICH	353.42	23.69	14.91853103	22.16974308
ND	383.27	28.41	13.4906723	32.65960289

Implementation

GUI Prototype

Below are 3 GUI screens for my database along with which tables each uses and screenshots of it in action. Please use the embedded hyperlinks to see the action for yourself.

#1 Home Field Advantage Tableau Dashboard

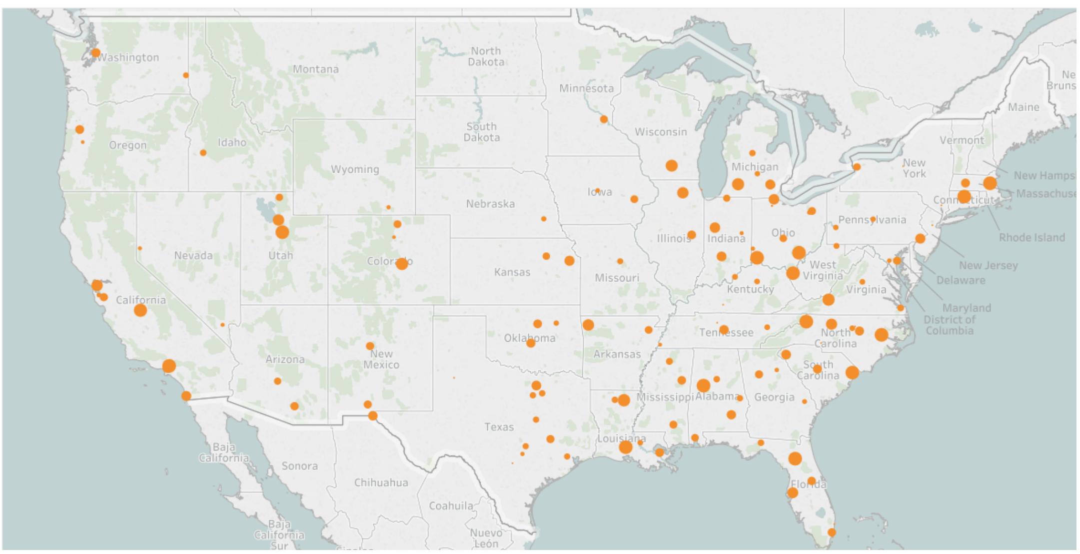
Tables used:

- Conference
- Stadium
- Team
- Team Stadium
- Stat
- Team_Stat

Dashboard in Action:

- 15 year HFA averages geo located for all 129 stadiums. Hover over a specific circle to get Home/Away yard and point stats as well as team and stadium info

Home Field Advantage



Team:	Alabama
Conference:	Southeastern Conference
Stadium Name:	Bryant-Denny Stadium
Capacity:	101,821
City/State:	Tuscaloosa, AL
Average Temp:	77°

Home		Away	
Yards PG	418	Yards PG	389
Yards Allowed PG	263	Yards Allowed PG	297
Points PG	32	Points PG	30
Points Allowed PG	13	Points Allowed PG	18

Home Field Advantage: **46.1**

Team:	Colorado
Conference:	Pac-12 Conference
Stadium Name:	Folsom Field
Capacity:	50,183
City/State:	Boulder, CO
Average Temp:	65°

Home		Away	
Yards PG	385	Yards PG	337
Yards Allowed PG	411	Yards Allowed PG	422
Points PG	26	Points PG	21
Points Allowed PG	29	Points Allowed PG	33

Home Field Advantage: **0.8**

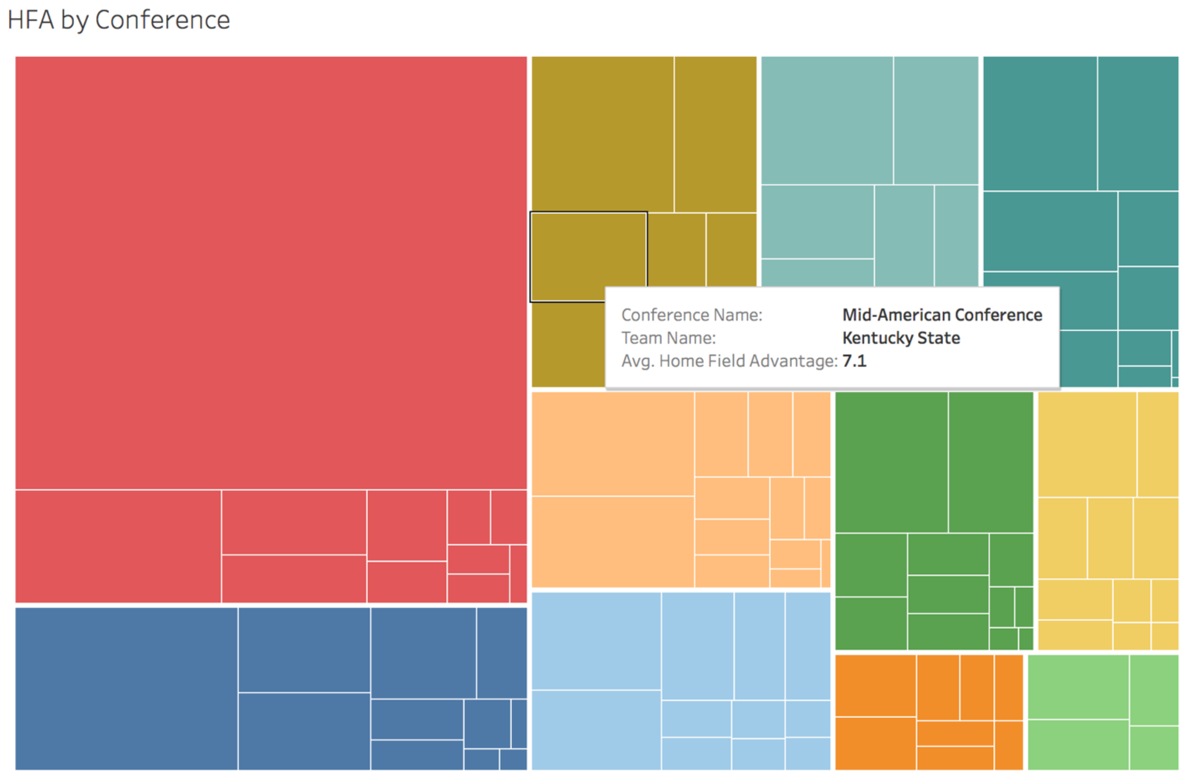
#2 HFA by Conference Tableau Dashboard

Tables used:

- Conference
- Team
- Team Stat
- Stat

Dashboard in Action:

- Home Field Advantage broken out by conference as well as team. Hover over a square to get the HFA for a specific team/conference



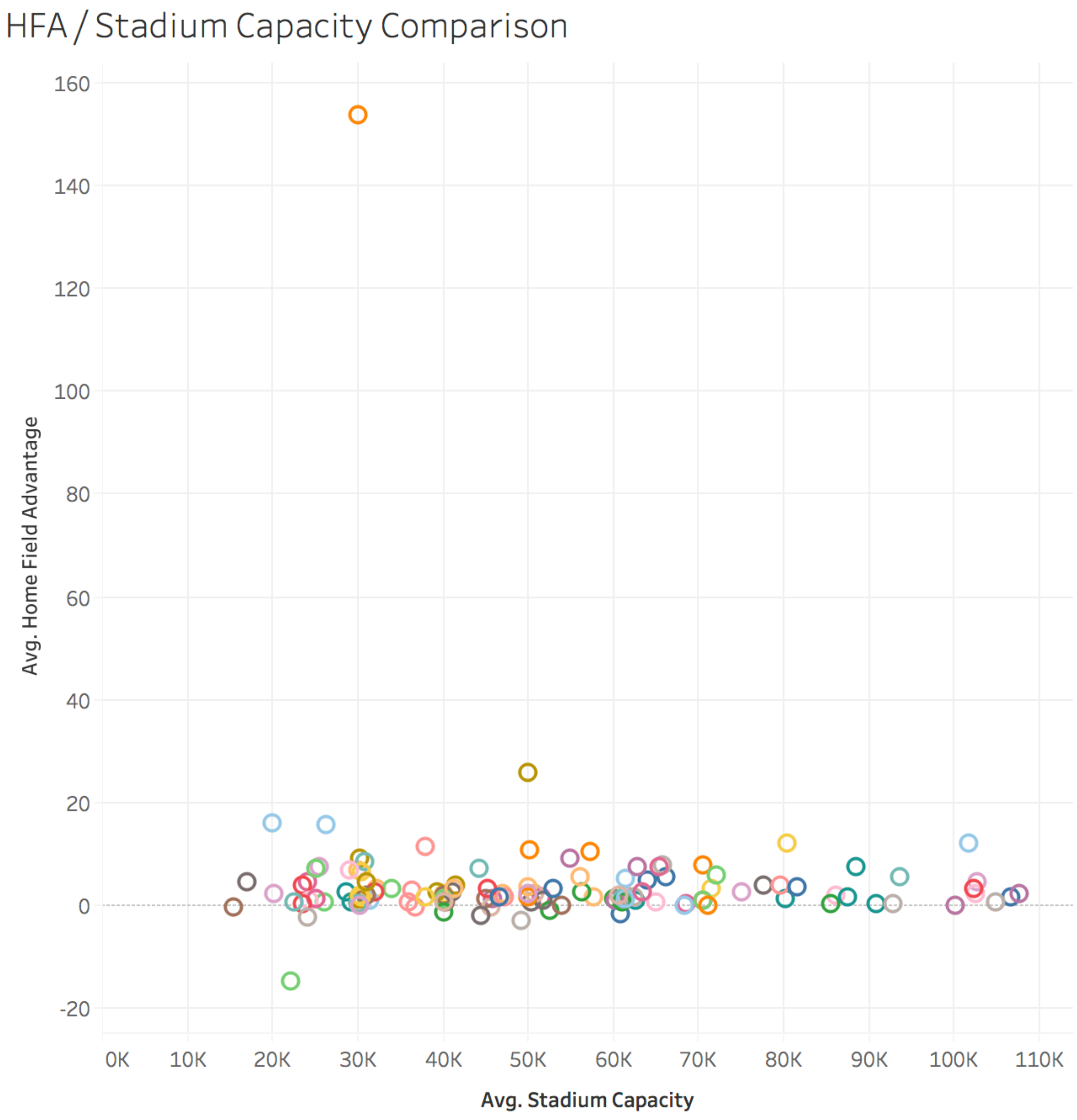
#3 HFA / Stadium Capacity Comparison Tableau Dashboard

Tables used:

- Team
- Team Stat
- Stat
- Stadium
- Team Stadium

Dashboard in Action:

- Scatter plot measuring all 129 team stadium's capacities vs Home Field Advantage



Reflection

After going through the entire process and completing my database, I have come to appreciate the SQL tables I currently work with in my profession. Knowing the time and energy that goes into a simple table, like in this project, I'm much more aware of the power a database has, as well as hard work and commitment it takes to provide that power. I assumed that all the data types I originally chose would work flawlessly in SQL, which made the implementation process take much longer than I anticipated. I also thought the SQL we used in labs would be the brunt of what I needed, but there were many instances where I encountered something not previously done before where it took some trial and error and Google searches to get it to work. The next database I build will go smoother as I will be much better at SQL and I'm confident it will take much less time to build my tables. I will also think about the normalized logical model more when first constructing my database. I understand so much more now about how tables are made, how to update them and keep them healthy / accurate and how I can find data better using SQL.