

-- This document lists the queries during ETL of raw data

-- Importing data from csv to sqlite3

```
.mode csv raw_team_stats
.import data/team_stats.csv raw_team_stats
.mode csv raw_team_basic_profile
.import data/team_basic_profile_info.csv raw_team_basic_profile
.mode csv raw_player_basic_profile
.import data/player_basic_profile.csv raw_player_basic_profile
.mode csv raw_player_salary
.import data/player_salary.csv raw_player_salary
.mode csv raw_team_basic_profile
.import data/player_basic_profile_info.csv raw_team_basic_profile
.mode csv raw_player_stats
.import data/player_stats.csv raw_player_stats
```

-- Cleaning data in raw tables

-- remove '\$' and ',' in salary

```
UPDATE raw_player_salary SET salary =
replace( replace( salary, '$', '' ) , ',', '' )
```

-- remove unnecessary empty rows in raw_player_stats table

```
DELETE FROM raw_player_stats WHERE Season = '' AND Lg = ''
```

-- remove 'Did not play' records from raw_player_stats table

```
DELETE FROM raw_player_stats WHERE Tm ISNULL AND Lg ISNULL AND Pos
ISNULL
```

-- update heights of players to inches instead of feet-inch

```
UPDATE raw_player_basic_profile SET ht=cast(substr(trim(ht),0,2)
AS INT)*12+ cast(substr(ht,3) AS INT)
```

-- Clean False as 0 and True as 1 for active flag in
raw_player_basic_profile

```
UPDATE raw_player_basic_profile SET active=1 WHERE active =
"True"
UPDATE raw_player_basic_profile SET active=0 WHERE active =
"False"
```

-- Remove '*' from season names in raw_player_stats

```
UPDATE raw_player_stats SET Season =  
trim(replace(Season, '★', ''), ' ') WHERE Season like '%★'
```

```
UPDATE raw_player_basic_profile SET  
name=replace(name, '★', '')
```

-- Clean old team names from raw_player_salary and
raw_player_stats table

```
UPDATE raw_player_salary SET team_url='/teams/NJN/' WHERE  
team_url='/teams/BRK/';  
UPDATE raw_player_salary SET team_url='/teams/CHA/' WHERE  
team_url IN ('/teams/CHH/', '/teams/CHO/');  
UPDATE raw_player_salary SET team_url='/teams/NOH/' WHERE  
team_url IN ('/teams/NOK/', '/teams/NOP/');  
UPDATE raw_player_salary SET team_url='/teams/OKC/' WHERE  
team_url='/teams/SEA/';  
UPDATE raw_player_salary SET team_url='/teams/MEM/' WHERE  
team_url='/teams/VAN/';  
UPDATE raw_player_salary SET team_url='/teams/WAS/' WHERE  
team_url='/teams/WSB/';  
UPDATE raw_player_stats SET Tm_url='/teams/WAS/' WHERE Tm_url ='/  
teams/WSB/';  
UPDATE raw_player_stats SET Tm_url='/teams/UTS/' WHERE Tm_url='/  
teams/ANA/';  
UPDATE raw_player_stats SET Tm_url='/teams/WAS/' WHERE Tm_url='/  
teams/BAL/';  
UPDATE raw_player_stats SET Tm_url='/teams/LAC/' WHERE Tm_url='/  
teams/BUF/';  
UPDATE raw_player_stats SET Tm_url='/teams/WAS/' WHERE Tm_url='/  
teams/CAP/';  
UPDATE raw_player_stats SET Tm_url='/teams/SSL/' WHERE Tm_url='/  
teams/CAR/';  
UPDATE raw_player_stats SET Tm_url='/teams/WAS/' WHERE Tm_url='/  
teams/CHP/';  
UPDATE raw_player_stats SET Tm_url='/teams/WAS/' WHERE Tm_url='/  
teams/CHZ/';  
UPDATE raw_player_stats SET Tm_url='/teams/SAC/' WHERE Tm_url='/  
teams/CIN/';  
UPDATE raw_player_stats SET Tm_url='/teams/SAS/' WHERE Tm_url='/  
teams/DLC/';  
UPDATE raw_player_stats SET Tm_url='/teams/DEN/' WHERE Tm_url='/  
teams/DNA/';  
UPDATE raw_player_stats SET Tm_url='/teams/DEN/' WHERE Tm_url='/  
teams/DNR/';  
UPDATE raw_player_stats SET Tm_url='/teams/DET/' WHERE Tm_url='/  
teams/FTW/';  
UPDATE raw_player_stats SET Tm_url='/teams/SSL/' WHERE Tm_url='/
```

```

teams/HSM/';
UPDATE raw_player_stats SET Tm_url='/teams/IND/' WHERE Tm_url='/
teams/INA/';
UPDATE raw_player_stats SET Tm_url='/teams/SAC/' WHERE Tm_url='/
teams/KCK/';
UPDATE raw_player_stats SET Tm_url='/teams/SAC/' WHERE Tm_url='/
teams/KCO/';
UPDATE raw_player_stats SET Tm_url='/teams/UTS/' WHERE Tm_url='/
teams/LAS/';
UPDATE raw_player_stats SET Tm_url='/teams/ATL/' WHERE Tm_url='/
teams/MLH/';
UPDATE raw_player_stats SET Tm_url='/teams/FL0/' WHERE Tm_url='/
teams/MMF/';
UPDATE raw_player_stats SET Tm_url='/teams/MMS/' WHERE Tm_url='/
teams/MMP/';
UPDATE raw_player_stats SET Tm_url='/teams/MMS/' WHERE Tm_url='/
teams/MMT/';
UPDATE raw_player_stats SET Tm_url='/teams/LAL/' WHERE Tm_url='/
teams/MNL/';
UPDATE raw_player_stats SET Tm_url='/teams/FL0/' WHERE Tm_url='/
teams/MNM/';
UPDATE raw_player_stats SET Tm_url='/teams/PTC/' WHERE Tm_url='/
teams/MNP/';
UPDATE raw_player_stats SET Tm_url='/teams/NJN/' WHERE Tm_url='/
teams/NJA/';
UPDATE raw_player_stats SET Tm_url='/teams/MMS/' WHERE Tm_url='/
teams/NOB/';
UPDATE raw_player_stats SET Tm_url='/teams/UTA/' WHERE Tm_url='/
teams/NOJ/';
UPDATE raw_player_stats SET Tm_url='/teams/NJN/' WHERE Tm_url='/
teams/NYA/';
UPDATE raw_player_stats SET Tm_url='/teams/NJN/' WHERE Tm_url='/
teams/NYN/';
UPDATE raw_player_stats SET Tm_url='/teams/VIR/' WHERE Tm_url='/
teams/OAK/';
UPDATE raw_player_stats SET Tm_url='/teams/GSW/' WHERE Tm_url='/
teams/PHW/';
UPDATE raw_player_stats SET Tm_url='/teams/PTC/' WHERE Tm_url='/
teams/PTP/';
UPDATE raw_player_stats SET Tm_url='/teams/SAC/' WHERE Tm_url='/
teams/ROC/';
UPDATE raw_player_stats SET Tm_url='/teams/SAS/' WHERE Tm_url='/
teams/SAA/';
UPDATE raw_player_stats SET Tm_url='/teams/SDS/' WHERE Tm_url='/
teams/SDA/';
UPDATE raw_player_stats SET Tm_url='/teams/LAC/' WHERE Tm_url='/
teams/SDC/';
UPDATE raw_player_stats SET Tm_url='/teams/HOU/' WHERE Tm_url='/
teams/SDR/';
UPDATE raw_player_stats SET Tm_url='/teams/GSW/' WHERE Tm_url='/
teams/SFW/';

```

```

UPDATE raw_player_stats SET Tm_url='/teams/ATL/' WHERE Tm_url='/
teams/STL/';
UPDATE raw_player_stats SET Tm_url='/teams/PHI/' WHERE Tm_url='/
teams/SYR/';
UPDATE raw_player_stats SET Tm_url='/teams/SAS/' WHERE Tm_url='/
teams/TEX/';
UPDATE raw_player_stats SET Tm_url='/teams/ATL/' WHERE Tm_url='/
teams/TRI/';
UPDATE raw_player_stats SET Tm_url='/teams/VIR/' WHERE Tm_url='/
teams/WSA/';

```

```

-- Create tables
-- Position table

```

```

CREATE TABLE position
(
    position_id INT PRIMARY KEY AUTOINCREMENT,
    position VARCHAR(10),
    description VARCHAR(40)
);

```

```

-- Player basic profile

```

```

CREATE TABLE player_basic
(
    player_id INTEGER PRIMARY KEY AUTOINCREMENT,
    name TEXT,
    from_year TEXT,
    to_year TEXT,
    pos INTEGER,
    ht INTEGER,
    wt INTEGER,
    dob TEXT,
    college TEXT,
    active BOOLEAN,
    url TEXT,
    FOREIGN KEY(pos) REFERENCES position(position_id)
);

```

```

INSERT INTO player_basic
(name, from_year, to_year, pos, ht, wt, dob, college, active)
SELECT r.name, r."from", r."to", p.position_id, r.ht, r.wt,
r.dob, r.college, r.active
FROM raw_player_basic_profile r, position p
WHERE r.pos = p.position

```

```

INSERT INTO player_basic (name,url)

```

```
SELECT name, url from raw_player_salary
WHERE url != '' AND
      url NOT IN (SELECT url FROM player_basic) GROUP BY name, url
```

```
INSERT INTO player_basic (name) SELECT name from
raw_player_salary WHERE url = '' GROUP BY name
```

— Create league table

```
CREATE TABLE league
(
    league_id INTEGER PRIMARY KEY AUTOINCREMENT,
    league VARCHAR(10),
    description VARCHAR(40)
);
```

-- Create season table

```
CREATE TABLE season
(
    season_id INTEGER PRIMARY KEY AUTOINCREMENT ,
    season VARCHAR(10)
);
```

```
INSERT INTO season (season) SELECT DISTINCT (Season) FROM
raw_player_stats WHERE Season NOT LIKE '%season%' AND
raw_player_stats.Season != 'Career' AND
raw_player_stats.Season != '' GROUP BY Season
```

-- Create team basic table

```
CREATE TABLE team_basic
(
    team_id INTEGER PRIMARY KEY AUTOINCREMENT,
    url TEXT,
    name TEXT,
    league_id INTEGER,
    "from" INTEGER,
    "to" INTEGER,
    years INTEGER,
    games INTEGER,
    wins INTEGER,
```

```
    losses INTEGER,  
    wlpercent REAL,  
    playoffs INTEGER,  
    div INTEGER,  
    conf INTEGER,  
    champ INTEGER,  
    FOREIGN KEY(league_id) REFERENCES league(league_id)  
);
```

```
INSERT INTO team_basic (url, name, league_id, "from",  
"to", years, games, wins, losses, wlpercent, playoffs,  
div, conf, champ) SELECT r.url, r.name, l.league_id,  
r."from", r."to", r.years, r.games, r.wins, r.losses,  
r.wlpercent, r.playoffs, r.div, r.conf, r.champ FROM  
raw_team_basic_profile r, league l WHERE r.lg=l.league;
```

-- Create season_salary table

```
CREATE TABLE player_season_salary  
(  
    player_id INTEGER,  
    team_id INTEGER,  
    season_id INTEGER,  
    rank INTEGER,  
    salary INTEGER,  
    PRIMARY KEY (player_id, team_id, season_id),  
    FOREIGN KEY(player_id) REFERENCES  
player_basic(player_id),  
    FOREIGN KEY(team_id) REFERENCES team_basic(team_id),  
    FOREIGN KEY(season_id) REFERENCES season(season_id)  
);
```

```
INSERT INTO player_season_salary (player_id, team_id, season_id,  
rank, salary) SELECT p.player_id,  
t.team_id,s.season_id,r.rank,r.salary FROM player_basic p,  
team_basic t, season s, raw_player_salary r WHERE ((r.url != '' AND  
r.url=p.url) OR (r.url = '' AND p.name = r.name)) AND t.url =  
r.team_url AND s.season = r.season GROUP BY p.player_id,  
t.team_id,s.season_id;
```

-- Create team stats table

```
CREATE TABLE team_stats
(
    team_id INTEGER,
    season_id INTEGER,
    lg_id INTEGER,
    W INTEGER,
    L INTEGER,
    "W/L%" REAL,
    Finish INTEGER,
    SRS REAL,
    Pace REAL,
    Rel_Pace REAL,
    ORtg REAL,
    Rel_ORtg REAL,
    DRtg REAL,
    Rel_DRtg REAL,
    top_ws INTEGER,
    PRIMARY KEY (team_id, season_id),
    FOREIGN KEY(team_id) REFERENCES team_basic(team_id),
    FOREIGN KEY(season_id) REFERENCES season(season_id),
    FOREIGN KEY(lg_id) REFERENCES league(league_id),
    FOREIGN KEY(top_ws) REFERENCES player_basic(player_id)
);
```

```
INSERT INTO team_stats (team_id, season_id, lg_id, W, L, "W/L%",
Finish, SRS, Pace, Rel_Pace, ORtg, Rel_ORtg, DRtg, Rel_DRtg,
top_ws) SELECT t.team_id,s.season_id,l.league_id, r.W, r.L, r."W/L%",
r.Finish, r.SRS, r.Pace, r.Rel_Pace, r.ORtg, r.Rel_ORtg
, r.DRtg, r.Rel_DRtg, p.player_id FROM team_basic t, season s,
league l, player_basic p, raw_team_stats r WHERE r.url = t.url AND
r.Season = s.season AND r.Lg = l.league AND r.top_ws_url = p.url;
```

-- Create player_season_stats table

```
CREATE TABLE player_season_stats
(
    player_id INTEGER,
    season_id INTEGER,
    team_id INTEGER,
    Age INTEGER,
    Lg INTEGER,
    Pos INTEGER,
    G INTEGER,
    GS INTEGER,
    MP INTEGER,
    FG INTEGER,
    FGA INTEGER,
```

```

"FG%" REAL,
"3P" INTEGER,
"3PA" INTEGER,
"3P%" REAL,
"2P" INTEGER,
"2PA" INTEGER,
"2P%" REAL,
"eFG%" REAL,
FT INTEGER,
FTA INTEGER,
"FT%" REAL,
ORB INTEGER,
DRB INTEGER,
TRB INTEGER,
AST INTEGER,
STL INTEGER,
BLK INTEGER,
TOV INTEGER,
PF INTEGER,
PTS INTEGER,
PRIMARY KEY (player_id,season_id,team_id),
FOREIGN KEY(player_id) REFERENCES player_basic(player_id),
FOREIGN KEY(team_id) REFERENCES team_basic(team_id),
FOREIGN KEY(season_id) REFERENCES season(season_id),
FOREIGN KEY(Lg) REFERENCES league(league_id),
FOREIGN KEY(Pos) REFERENCES position(position_id)
);

```

```

INSERT INTO player_season_stats(player_id, season_id, team_id,
Age, Lg, Pos, G, GS, MP, FG, FGA, "FG%", "3P", "3PA", "3P%", "2P",
"2PA", "2P%", "eFG%", FT, FTA, "FT%", ORB, DRB, TRB, AST, STL,
BLK, TOV, PF, PTS) SELECT p.player_id,s.season_id, t.team_id,
r.Age, l.league_id,pos.position_id, r.G,r.GS,
r.MP,r.FG,r.FGA,r."FG%",r."3P",
r."3PA", r."3P%", r."2P", r."2PA", r."2P%", r."eFG%", r.FT,
r.FTA, r."FT%",r.ORB, r.DRB,r.TRB, r.AST,
r.STL, r.BLK, r.TOV, r.PF,r.PTS FROM player_basic p, season s,
team_basic t, league l, position pos, raw_player_stats r
WHERE p.url = r.player_url AND t.url = r.Tm_url AND r.Season =
s.season AND r.Lg = l.league AND pos.position =r.Pos;

```

-- Create career_stats table

```

CREATE TABLE player_career_stats
(
    player_id INTEGER,
    Lg INTEGER,
    Pos INTEGER,
    G INTEGER,

```



```

GS INTEGER,
MP INTEGER,
FG INTEGER,
FGA INTEGER,
"FG%" REAL,
"3P" INTEGER,
"3PA" INTEGER,
"3P%" REAL,
"2P" INTEGER,
"2PA" INTEGER,
"2P%" REAL,
"eFG%" REAL,
FT INTEGER,
FTA INTEGER,
"FT%" REAL,
ORB INTEGER,
DRB INTEGER,
TRB INTEGER,
AST INTEGER,
STL INTEGER,
BLK INTEGER,
TOV INTEGER,
PF INTEGER,
PTS INTEGER,
PRIMARY KEY (player_id,Lg),
FOREIGN KEY(player_id) REFERENCES player_basic(player_id),
FOREIGN KEY(Lg) REFERENCES league(league_id),
FOREIGN KEY(Pos) REFERENCES position(position_id)
);

```

```

INSERT INTO player_career_stats(player_id, Lg, Pos, G, GS, MP, FG,
FGA, "FG%", "3P", "3PA", "3P%", "2P", "2PA", "2P%", "eFG%", FT,
FTA, "FT%", ORB, DRB, TRB, AST, STL, BLK, TOV, PF, PTS)
SELECT p.player_id, l.league_id, pos.position_id, r.G, r.GS,
r.MP, r.FG, r.FGA, r."FG%", r."3P",
r."3PA", r."3P%", r."2P", r."2PA", r."2P%", r."eFG%", r.FT,
r.FTA, r."FT%", r.ORB, r.DRB, r.TRB, r.AST,
r.STL, r.BLK, r.TOV, r.PF, r.PTS FROM player_basic p, league l,
position pos, raw_player_stats r

```

— The queries for Part 2 questions can be found in the write up document ([GUIDEDPROJECT1_GROUP11.pdf](#))