NHL DATABASE SYSTEM

Table of Contents

INTRODUCTION

The National Hockey League is a professional ice hockey league in North America, currently comprising 31 teams: 24 in the United States and 7 in Canada. NHL is considered to be the premier professional ice hockey league in the world and one of the major professional sports leagues in the United States and Canada.

Each regulation game is 60 minutes long. The game is composed of three 20-minute periods with an intermission between periods. At the end of regulation time, the team with the most goals wins the game. If a game is tied after regulation time, overtime ensues. During the regular season, overtime is a five-minute, three-on-three sudden-death period, in which whoever scores a goal first will win the game.

This project deals with building an automated database which includes all the details of players, teams and matches. Subsequently insights about the NHL database are reported in Tableau.

Following are the analytical questions the report tries to answer:

- How many hits are recorded each season?
- Who was the hits leader for each season?
- What is the distribution of yearly hit totals per player? (Selected Top Players)
- What is the distribution of yearly hit totals per team?
- Who is taking the most hits?
- How many goals are recorded each season?
- Who scored the most goals each season?
- What is the distribution of yearly goal totals per player? (Selected Top Players)
- What is the distribution of yearly goal totals per team?

DATA DESCRPTION

The data represents all the official metrics measured for each game in the NHL from 2012-2013 to 2018-2019 season (6 seasons). This dataset not only includes the results and player stats of NHL games but also includes details on individual plays such as shots, goals and stoppages including date & time. The dataset is incomplete as there are some games where no plays information is available. The dataset comprises of 9 tables that shows multiple information about the type of game played, venue, date and time, the statistical information of each player, team, and the result of each game. The dataset tables are as follows:

1. **team_info:** Contains the static information on teams such as names. It can be merged into other tables by the key field: team_id. While team_id is unique, franchise_id is not.
    - team_id: Key field as assigned by the NHL
    - franchiseId
    - shortName
    - teamName
    - abbreviation

- link

2. **player_info:** Contains the static information on players such as names. It can be merged into other tables by the key field: player_id.
   - player_id: Key field as assigned by the NHL
   - firstName
   - lastName
   - nationality
   - birthCity
   - primaryPosition
   - birthdate
   - link
3. **game:** Consists of a record of each game in the dataset. It contains 16 columns describing the venue, date and time the game was played, the number of goals scored by each team and ID assigned to different team and type of game played.
   - game_id: Game key field as assigned by the NHL
   - season
   - type: R=Regular Season, P=Playoff
   - date_time
   - date_time_GMT
   - away_team_id
   - home_team_id
   - away_goals
   - home_goals
   - outcome
   - home_rink_side_start: Indicates the direction of play relative to the Time/Score keepers. XY coordinates are relative to this
   - venue
   - venue_link
   - venue_time_zone_id
   - venue_time_zone_offset
   - venue_time_zone_tz
4. **game_plays:** Explains the individual plays that make up a game. A key field play_id allows the players involved in a play to be found in the game_plays_players table.
   - play_id
   - game_id
   - play_num
   - team_id_for
   - team_id_against
   - event
   - secondaryType
   - x: x-coordinate recorded by NHL official
   - y: y-coordinate recorded by NHL official
   - period
   - periodType
   - periodTime
   - periodTimeRemaining

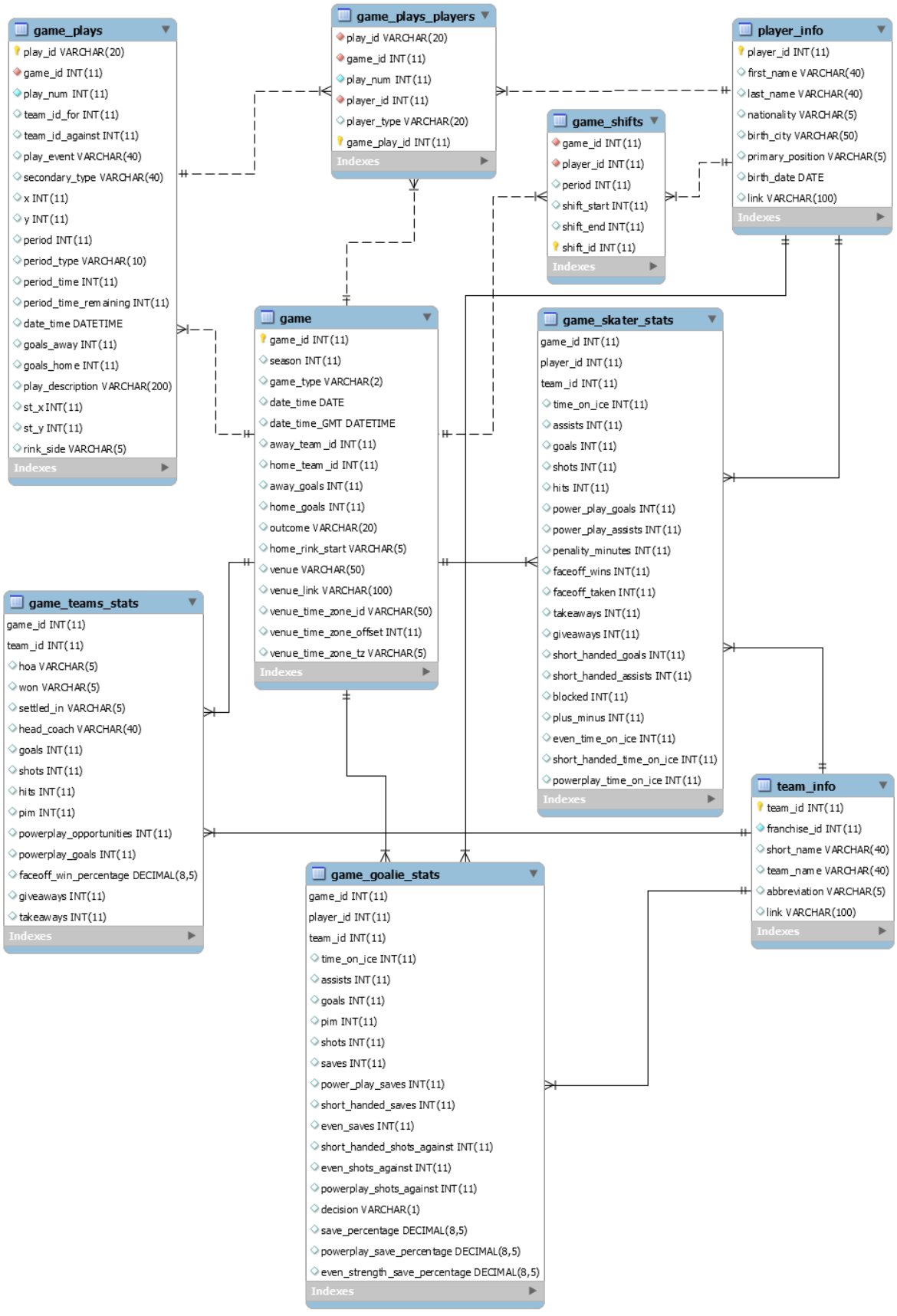
- dateTime
- goals_away
- goals_home
- description
- st_x: x coordinate adjusted to be for attacking left to right
- st_y: y coordinate adjusted to be for attacking left to right
- rink_side

5. **game_plays_players:** Contains the details regarding players involved in a play. It can be joined with the game_plays table to get information on the play, and with the player_info table to get information on the players.
   - play_id
   - game_id
   - play_num
   - player_id
   - playerType
6. **game_shifts:** Contains the information of which players were on the ice at which times for each game.
   - game_id
   - player_id
   - period
   - shift_start
   - shift_end
7. **game_teams_stats:** Records the basic stats for each team per game. Each game will have two entries in this table, one for each team.
   - game_id
   - team_id
   - HoA: Whether the team is Home or Away
   - Won: Did the team win. There are no draws in Hockey
   - settled_in: Whether the team won/lost in normal regulation time, over time, or in a shootout
   - head_coach
   - goals
   - shots
   - hits
   - pim
   - powerPlayOpportunities
   - powerPlayGoals
   - faceOffWinPercentage
   - giveaways
   - takeaways
8. **game_skater_stats:** Records the basic stats for each skater that played in each game. There will often be 36 entries per game, one for each skater on each team. Sometimes there might be less entries.
   - game_id
   - player_id
   - team_id

- timeOnIce
- assists
- goals
- shots
- hits
- powerPlayGoals
- powerPlayAssists
- penaltyMinutes
- faceOffWins
- faceoffTaken
- takeaways
- giveaways
- shortHandedGoals
- shortHandedAssists
- blocked
- plusMinus
- evenTimeOnIce
- shortHandedTimeOnIce
- powerPlayTimeOnIce

9. **game_goalie_stats:** Records the basic stats for each goalie that had some ice time in each game. If the team's "backup" goalie is not used, they are not recorded in this table.
    - game_id
    - player_id
    - team_id
    - timeOnIce
    - assists
    - goals
    - pim
    - shots
    - saves
    - powerPlaySaves
    - shortHandedSaves
    - evenSaves
    - shortHandedShotsAgainst
    - evenShotsAgainst
    - powerPlayShotsAgainst
    - decision
    - savePercentage
    - powerPlaySavePercentage
    - evenStrengthSavePercentage

ENTITY RELATIONSHIP DIAGRAM (ERD)



CREATING DATABASE

```
DROP DATABASE IF EXISTS nhl;
CREATE DATABASE nhl;

SHOW DATABASES;
```

```
+--------------------+
| Database           |
+--------------------+
| ap                 |
| ex                 |
| information_schema |
| mysql              |
| nhl                |
| om                 |
| performance_schema |
| sys                |
+--------------------+
```

CREATING TABLES

9 tables are created for NHL dataset which contains the same number of tables and retains the integrity of the columns. The tables created are namely,

- **team_info**
- **player_info**
- **game**
- **game_plays**
- **game_plays_players**
- **game_shifts**
- **game_teams_stats**
- **game_skater_stats**
- **game_goalie_stats**

```
USE nhl;

DROP TABLE IF EXISTS team_info;
CREATE TABLE team_info (
    team_id INT NOT NULL,
    franchise_id INT NOT NULL,
    short_name VARCHAR(40) NULL,
    team_name VARCHAR(40) NULL,
    abbreviation VARCHAR(5) NULL,
    link VARCHAR(100) NULL
) COMMENT 'Table contains basic team information';

DROP TABLE IF EXISTS player_info;
CREATE TABLE player_info (
    player_id INT NOT NULL,
    first_name VARCHAR(40) NULL,
    last_name VARCHAR(40) NULL,
    nationality VARCHAR(5) NULL,
    birth_city VARCHAR(50) NULL,
    primary_position VARCHAR(5) NULL,
    birth_date DATE NULL,
    link VARCHAR(100) NULL
) COMMENT 'Table contains basic player information';
```

```sql
DROP TABLE IF EXISTS game;
CREATE TABLE game (
    game_id INT NOT NULL,
    season INT NULL,
    game_type VARCHAR(2) NULL,
    date_time DATE NULL,
    date_time_GMT DATETIME NULL,
    away_team_id INT NULL,
    home_team_id INT NULL,
    away_goals INT NULL,
    home_goals INT NULL,
    outcome VARCHAR(20) NULL,
    home_rink_start VARCHAR(5) NULL,
    venue VARCHAR(50) NULL,
    venue_link VARCHAR(100) NULL,
    venue_time_zone_id VARCHAR(50) NULL,
    venue_time_zone_offset INT NULL,
    venue_time_zone_tz VARCHAR(5) NULL
) COMMENT 'Table contains records of every game';


DROP TABLE IF EXISTS game_plays;
CREATE TABLE game_plays (
    play_id VARCHAR(20) NOT NULL,
    game_id INT NOT NULL,
    play_num INT NOT NULL,
    team_id_for INT NULL,
    team_id_against INT NULL,
    play_event VARCHAR(40) NULL,
    secondary_type VARCHAR(40) NULL,
    x INT NULL,
    y INT NULL,
    period INT NULL,
    period_type VARCHAR(10) NULL,
    period_time INT NULL,
    period_time_remaining INT NULL,
    date_time DATETIME NULL,
    goals_away INT NULL,
    goals_home INT NULL,
    play_description VARCHAR(200) NULL,
    st_x INT NULL,
    st_y INT NULL,
    rink_side VARCHAR(5) NULL
) COMMENT 'Table contains information of individual plays that make up a game';

DROP TABLE IF EXISTS game_plays_players;
CREATE TABLE game_plays_players (
    play_id VARCHAR(20) NOT NULL,
    game_id INT NOT NULL,
    play_num INT NOT NULL,
    player_id INT NOT NULL,
    player_type VARCHAR(20) NULL
) COMMENT 'Table contains information about players involved in a play';


DROP TABLE IF EXISTS game_shifts;
CREATE TABLE game_shifts (
    game_id INT NOT NULL,
    player_id INT NOT NULL,
    period INT NULL,
    shift_start INT NULL,
    shift_end INT NULL
) COMMENT 'Table contains details of which players were on the ice and at which times for each game';

DROP TABLE IF EXISTS game_teams_stats;
CREATE TABLE game_teams_stats (
    game_id INT NOT NULL,
    team_id INT NOT NULL,
    hoa VARCHAR(5) NULL,
    won VARCHAR(5) NULL,
    settled_in VARCHAR(5) NULL,
    head_coach VARCHAR(40) NULL,
    goals INT NULL,
    shots INT NULL,
    hits INT NULL,
    pim INT NULL,
    powerplay_opportunities INT NULL,
    powerplay_goals INT NULL,
    faceoff_win_percentage DECIMAL(8,5) NULL,
    giveaways INT NULL,
    takeaways INT NULL
) COMMENT 'Table contains basic stats for each team per game';
```

```
DROP TABLE IF EXISTS game_skater_stats;
CREATE TABLE game_skater_stats (
    game_id INT NOT NULL,
    player_id INT NOT NULL,
    team_id INT NOT NULL,
    time_on_ice INT NULL,
    assists INT NULL,
    goals INT NULL,
    shots INT NULL,
    hits INT NULL,
    power_play_goals INT NULL,
    power_play_assists INT NULL,
    penality_minutes INT NULL,
    faceoff_wins INT NULL,
    faceoff_taken INT NULL,
    takeaways INT NULL,
    giveaways INT NULL,
    short_handed_goals INT NULL,
    short_handed_assists INT NULL,
    blocked INT NULL,
    plus_minus INT NULL,
    even_time_on_ice INT NULL,
    short_handed_time_on_ice INT NULL,
    powerplay_time_on_ice INT NULL
) COMMENT 'Table contains basic stats for each skater that played in a game';
```

```
DROP TABLE IF EXISTS game_goalie_stats;
CREATE TABLE game_goalie_stats (
    game_id INT NOT NULL,
    player_id INT NOT NULL,
    team_id INT NOT NULL,
    time_on_ice INT NULL,
    assists INT NULL,
    goals INT NULL,
    pim INT NULL,
    shots INT NULL,
    saves INT NULL,
    power_play_saves INT NULL,
    short_handed_saves INT NULL,
    even_saves INT NULL,
    short_handed_shots_against INT NULL,
    even_shots_against INT NULL,
    powerplay_shots_against INT NULL,
    decision VARCHAR(1) NULL,
    save_percentage DECIMAL(8,5) NULL,
    powerplay_save_percentage DECIMAL(8,5) NULL,
    even_strength_save_percentage DECIMAL(8,5) NULL
) COMMENT 'Table contains basic stats for each goalie that played in a game';
```

LOADING DATA

The csv files are read and loaded into the created empty tables.

```
SET GLOBAL local_infile = ON;

USE nhl;

LOAD DATA LOCAL INFILE
'nhl-game-data\\team_info.csv'
INTO TABLE team_info
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY  '\r\n'
IGNORE 1 LINES;

LOAD DATA LOCAL INFILE
'nhl-game-data\\player_info.csv'
INTO TABLE player_info
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY  '\r\n'
IGNORE 1 LINES;
```

```
LOAD DATA LOCAL INFILE
'nhl-game-data\\game.csv'
INTO TABLE game
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY  '\r\n'
IGNORE 1 LINES;

LOAD DATA LOCAL INFILE
'nhl-game-data\\game_plays.csv'
INTO TABLE game_plays
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY  '\r\n'
IGNORE 1 LINES;

LOAD DATA LOCAL INFILE
'nhl-game-data\\game_plays_players.csv'
INTO TABLE game_plays_players
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY  '\r\n'
IGNORE 1 LINES;

LOAD DATA LOCAL INFILE
'nhl-game-data\\game_shifts.csv'
INTO TABLE game_shifts
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY  '\r\n'
IGNORE 1 LINES;


LOAD DATA LOCAL INFILE
'nhl-game-data\\game_teams_stats.csv'
INTO TABLE game_teams_stats
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY  '\r\n'
IGNORE 1 LINES;

LOAD DATA LOCAL INFILE
'nhl-game-data\\game_skater_stats.csv'
INTO TABLE game_skater_stats
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY  '\r\n'
IGNORE 1 LINES;

LOAD DATA LOCAL INFILE
'nhl-game-data\\game_goalie_stats.csv'
INTO TABLE game_goalie_stats
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY  '\r\n'
IGNORE 1 LINES;

SELECT TABLE_NAME,
    TABLE_ROWS
FROM INFORMATION_SCHEMA.TABLES
WHERE TABLE_SCHEMA ='nhl';
```

TABLE_NAME	TABLE_ROWS
game	11418
game_goalie_stats	24645
game_plays	3508212
game_plays_players	5321029
game_shifts	8715163
game_skater_stats	408756
game_teams_stats	22834
player_info	2409
team_info	33

UPDATING PRIMARY AND FOREIGN KEYS

After loading the data, primary keys and foreign keys are updated to main data consistency.

```
USE nhl;

ALTER TABLE player_info ADD PRIMARY KEY (player_id);
ALTER TABLE team_info ADD PRIMARY KEY (team_id);
ALTER TABLE game ADD PRIMARY KEY (game_id);
ALTER TABLE game_plays ADD PRIMARY KEY (play_id);
ALTER TABLE game_plays_players ADD game_play_id INT NOT NULL AUTO_INCREMENT PRIMARY KEY;
ALTER TABLE game_shifts ADD shift_id INT NOT NULL AUTO_INCREMENT PRIMARY KEY;
ALTER TABLE game_teams_stats ADD PRIMARY KEY (game_id, team_id);
ALTER TABLE game_skater_stats ADD PRIMARY KEY (game_id, team_id, player_id);
ALTER TABLE game_goalie_stats ADD PRIMARY KEY (game_id, team_id, player_id);

ALTER TABLE game_plays ADD CONSTRAINT fk_game_id_gp_g FOREIGN KEY (game_id)
REFERENCES game(game_id);
ALTER TABLE game_plays_players ADD CONSTRAINT fk_game_id_gpp_g FOREIGN KEY (game_id)
REFERENCES game(game_id);
ALTER TABLE game_shifts ADD CONSTRAINT fk_game_id_gs_g FOREIGN KEY (game_id)
REFERENCES game(game_id);
ALTER TABLE game_teams_stats ADD CONSTRAINT fk_game_id_gts_g FOREIGN KEY (game_id)
REFERENCES game(game_id);
ALTER TABLE game_skater_stats ADD CONSTRAINT fk_game_id_gss_g FOREIGN KEY (game_id)
REFERENCES game(game_id);
ALTER TABLE game_goalie_stats ADD CONSTRAINT fk_game_id_ggs_g FOREIGN KEY (game_id)
REFERENCES game(game_id);

ALTER TABLE game_plays_players ADD CONSTRAINT fk_player_id_gpp_pi FOREIGN KEY (player_id)
REFERENCES player_info(player_id);
ALTER TABLE game_shifts ADD CONSTRAINT fk_player_id_gs_pi FOREIGN KEY (player_id)
REFERENCES player_info(player_id);
ALTER TABLE game_skater_stats ADD CONSTRAINT fk_player_id_gss_pi FOREIGN KEY (player_id)
REFERENCES player_info(player_id);
ALTER TABLE game_goalie_stats ADD CONSTRAINT fk_player_id_ggs_pi FOREIGN KEY (player_id)
REFERENCES player_info(player_id);


ALTER TABLE game_teams_stats ADD CONSTRAINT fk_team_id_gts_ti FOREIGN KEY (team_id)
REFERENCES team_info(team_id);
ALTER TABLE game_skater_stats ADD CONSTRAINT fk_team_id_gss_ti FOREIGN KEY (team_id)
REFERENCES team_info(team_id);
ALTER TABLE game_goalie_stats ADD CONSTRAINT fk_team_id_ggs_ti FOREIGN KEY (team_id)
REFERENCES team_info(team_id);

ALTER TABLE game_plays_players ADD CONSTRAINT fk_play_id_gpp_gp FOREIGN KEY (play_id)
REFERENCES game_plays(play_id);
```

CREATING VIEWS

Five views are created for simplifying analytical queries which require complex query design.

- **hits**

```
CREATE OR REPLACE VIEW hits AS
SELECT gp.*,
    gpp.player_id,
    pi.first_name,
    pi.last_name,
    pi.primary_position,
    pi.birth_date
FROM game_plays AS gp
LEFT JOIN game_plays_players AS gpp
    USING (play_id)
LEFT JOIN player_info AS pi
    USING (player_id)
WHERE gp.play_event = "Hit"
    AND gpp.player_type = "Hitter";
```

- **player_hits**

```
CREATE OR REPLACE VIEW player_hits AS
SELECT SUBSTRING(game_id, 1, 4) AS season,
    player_id,
    COUNT(*) AS hit_count
FROM hits
GROUP BY season,
    player_id;
```

- **beats**

```
CREATE OR REPLACE VIEW beats AS
SELECT gp.*,
    gpp.player_id,
    pi.first_name,
    pi.last_name,
    pi.primary_position,
    pi.birth_date
FROM game_plays AS gp
LEFT JOIN game_plays_players AS gpp
    USING (play_id)
LEFT JOIN player_info AS pi
    USING (player_id)
WHERE gp.play_event = "Hit"
    AND gpp.player_type = "Hittee";
```

- **goals**

```
CREATE OR REPLACE VIEW goals AS
SELECT gp.*,
    gpp.player_id,
    pi.first_name,
    pi.last_name,
    pi.primary_position,
    pi.birth_date
FROM game_plays AS gp
LEFT JOIN game_plays_players AS gpp
    USING (play_id)
LEFT JOIN player_info AS pi
    USING (player_id)
WHERE gp.play_event = "Goal"
    AND gpp.player_type = "Scorer";
```

- **player_goals**

```
CREATE OR REPLACE VIEW player_goals AS
SELECT SUBSTRING(game_id, 1, 4) AS season,
    player_id,
    COUNT(*) AS goal_count
FROM goals
GROUP BY season,
    player_id;
```

ANALYTICAL TASK 1

How many hits are recorded each season?

Code:

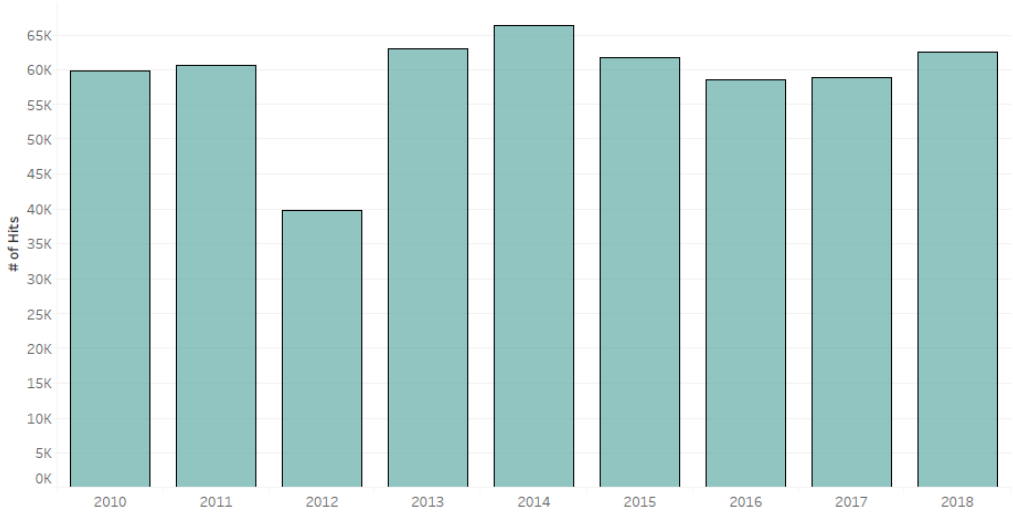
```
SELECT SUBSTRING(game_id, 1, 4) AS season,
    COUNT(*) AS hit_count
FROM hits
GROUP BY season;
```

SQL Result:

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

season	hit_count
2011	60519
2012	39794
2013	62973
2014	66342
2015	61615
2016	58635
2017	58962
2018	62576

Tableau Plot:



ANALYTICAL TASK 2

Who was the hits leader for each season?

Code:

```
SELECT *,
    RANK() OVER (
        PARTITION BY season
        ORDER BY hit_count DESC
    ) AS season_rank
FROM (
    SELECT SUBSTRING(h.game_id, 1, 4) AS season,
        h.team_id_for,
        ti.short_name,
        ti.team_name,
        COUNT(*) AS hit_count
    FROM hits AS h
    LEFT JOIN team_info AS ti
        ON h.team_id_for = ti.team_id
    GROUP BY season,
        h.team_id_for
) AS team_hits;
```

SQL Result:

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

season	player_id	first_name	last_name	primary_position	hit_count
2010	8470606	Dustin	Brown	RW	330
2011	8470606	Dustin	Brown	RW	383
2012	8474709	Matt	Martin	LW	275
2013	8470606	Dustin	Brown	RW	367
2014	8474709	Matt	Martin	LW	432
2015	8474709	Matt	Martin	LW	424
2016	8474697	Mark	Borowiecki	D	370
2017	8476880	Tom	Wilson	RW	344
2018	8471817	Ryan	Reaves	RW	343

Tableau Plot:

ANALYTICAL TASK 3

What is the distribution of yearly hit totals per player? (Selected Top Players)

Code:

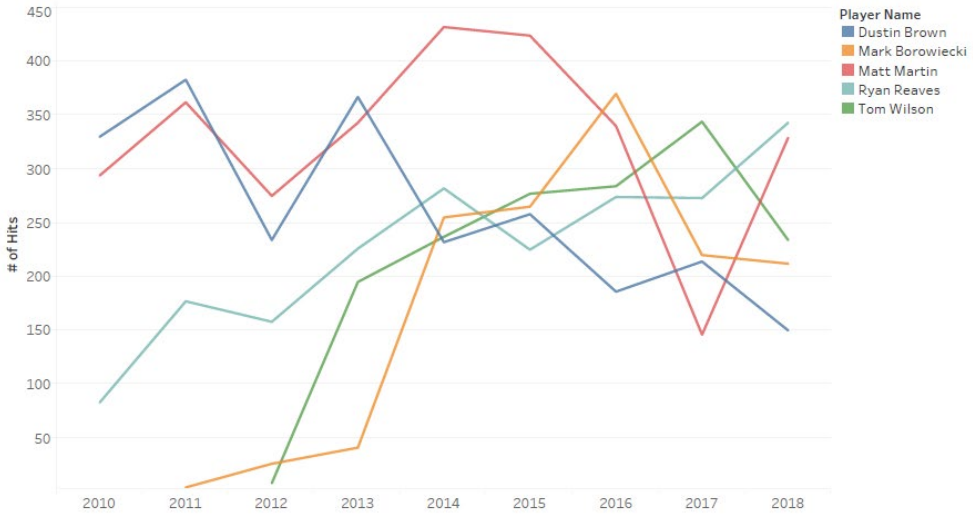
```
SELECT SUBSTRING(game_id, 1, 4) AS season,
    player_id,
    first_name,
    last_name,
    primary_position,
    COUNT(*) AS hit_count
FROM hits
GROUP BY season,
    player_id;
```

SQL Result:

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

season	player_id	first_name	last_name	primary_position	hit_count
2010	8474125	Carl	Gunnarsson	D	63
2010	8467400	Francois	Beauchemin	D	127
2010	8469474	Colby	Armstrong	RW	88
2010	8469521	Tomas	Plekanec	C	61
2010	8471310	Dustin	Boyd	C	12
2010	8471346	Kris	Versteeg	RW	54
2010	8471371	Mike	Brown	RW	105
2010	8468635	Travis	Moen	LW	139
2010	8470654	Maxim	Lapierre	C	272

Tableau Plot:



ANALYTICAL TASK 4

What is the distribution of yearly hit totals per team?

Code:

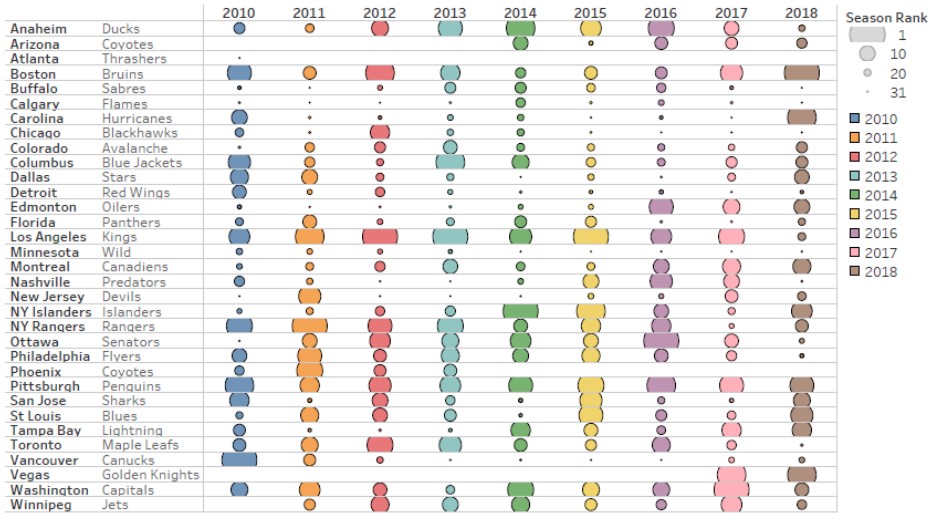
```
SELECT *,
    RANK() OVER (
        PARTITION BY season
        ORDER BY hit_count DESC
    ) AS season_rank
FROM (
    SELECT SUBSTRING(h.game_id, 1, 4) AS season,
        h.team_id_for,
        ti.short_name,
        ti.team_name,
        COUNT(*) AS hit_count
    FROM hits AS h
    LEFT JOIN team_info AS ti
        ON h.team_id_for = ti.team_id
    GROUP BY season,
        h.team_id_for
) AS team_hits;
```

SQL Result:

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

season	team_id_for	short_name	team_name	hit_count
2010	10	Toronto	Maple Leafs	2048
2010	8	Montreal	Canadiens	1795
2010	4	Philadelphia	Flyers	2096
2010	5	Pittsburgh	Penguins	2480
2010	12	Carolina	Hurricanes	2128
2010	30	Minnesota	Wild	1821
2010	16	Chicago	Blackhawks	1936
2010	21	Colorado	Avalanche	1657
2010	22	Edmonton	Oilers	1753

Tableau Plot:



ANALYTICAL TASK 5

Who is taking the most hits?

Code:

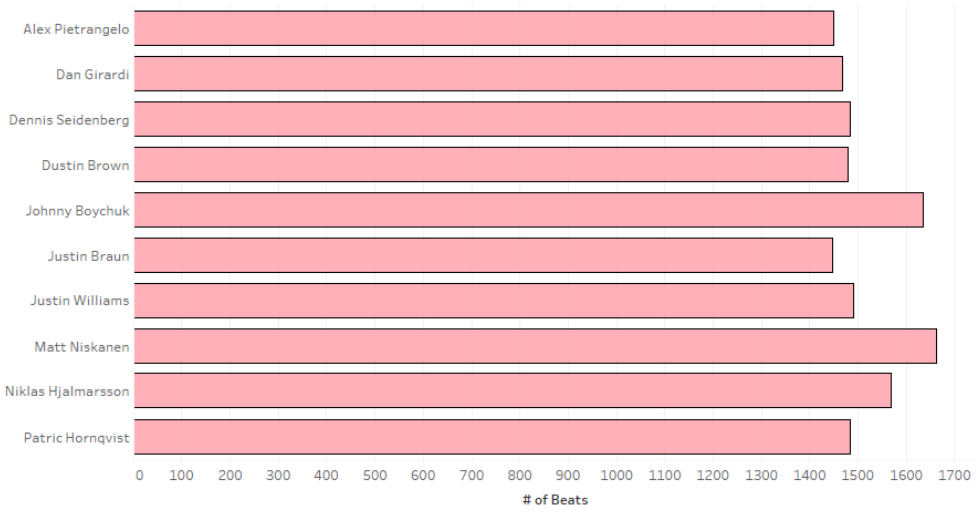
```
SELECT b.player_id,
    pi.first_name,
    pi.last_name,
    pi.primary_position,
    COUNT(*) AS beat_count
FROM beats AS b
LEFT JOIN player_info AS pi
    USING(player_id)
GROUP BY player_id
ORDER BY beat_count DESC
LIMIT 10;
```

SQL Result:

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

player_id	first_name	last_name	primary_position	beat_count
8471702	Matt	Niskanen	D	1663
8470187	Johnny	Boychuk	D	1635
8471769	Niklas	Hjalmarsson	D	1569
8468508	Justin	Williams	RW	1489
8469619	Dennis	Seidenberg	D	1484
8471887	Patric	Hornqvist	RW	1483
8470606	Dustin	Brown	RW	1479
8471958	Dan	Girardi	D	1468
8474565	Alex	Pietrangelo	D	1449

Tableau Plot:



ANALYTICAL TASK 6

How many goals are recorded each season?

Code:

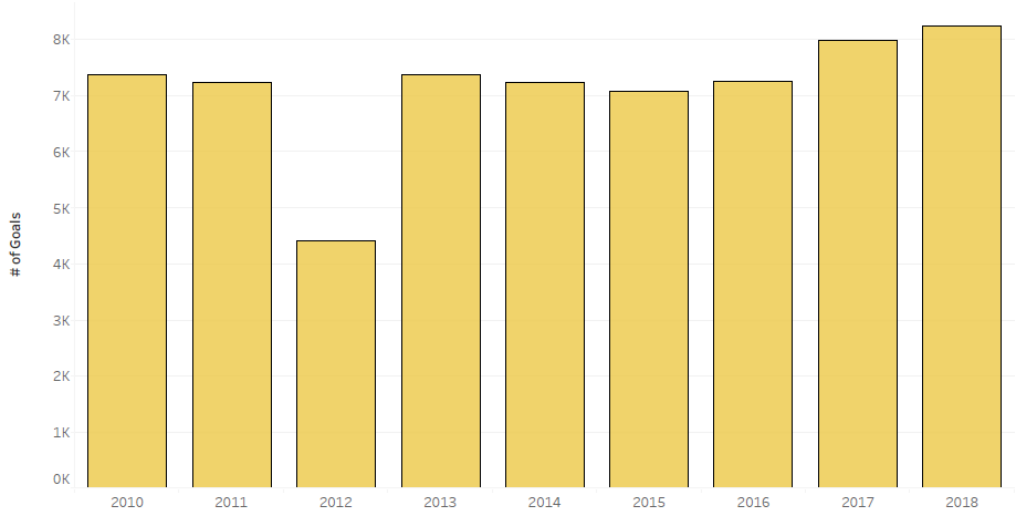
```
SELECT SUBSTRING(game_id, 1, 4) AS season,
    COUNT(*) AS goal_count
FROM goals
GROUP BY season;
```

SQL Result:

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

season	goal_count
2010	7376
2011	7233
2012	4411
2013	7377
2014	7243
2015	7082
2016	7251
2017	7992
2018	8245

Tableau Plot:



ANALYTICAL TASK 7

Who scored the most goals each season?

Code:

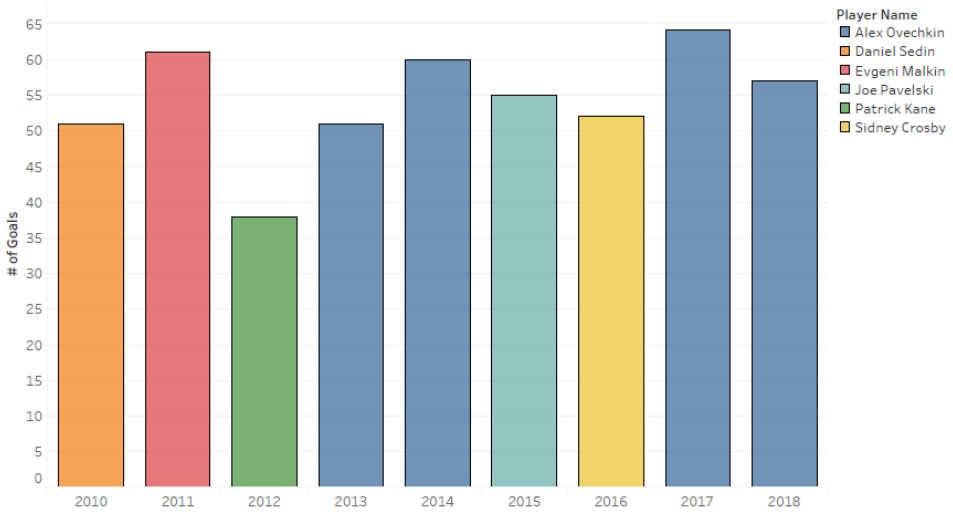
```
SELECT pg2.season,
    pg2.player_id,
    pi.first_name,
    pi.last_name,
    pi.primary_position,
    pg2.goal_count
FROM (
    SELECT season,
        MAX(goal_count) AS goal_count
    FROM player_goals
    GROUP BY season
) AS pg1
JOIN player_goals AS pg2
    USING (season,
        goal_count)
LEFT JOIN player_info AS pi
    USING (player_id);
```

SQL Result:

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

season	team_id_for	short_name	team_name	goal_count
2010	10	Toronto	Maple Leafs	225
2010	8	Montreal	Canadiens	230
2010	4	Philadelphia	Flyers	281
2010	5	Pittsburgh	Penguins	257
2010	12	Carolina	Hurricanes	239
2010	30	Minnesota	Wild	210
2010	21	Colorado	Avalanche	219
2010	16	Chicago	Blackhawks	270
2010	22	Edmonton	Oilers	194

Tableau Plot:



ANALYTICAL TASK 8

What is the distribution of yearly goal totals per player? (Selected Top Players)

Code:

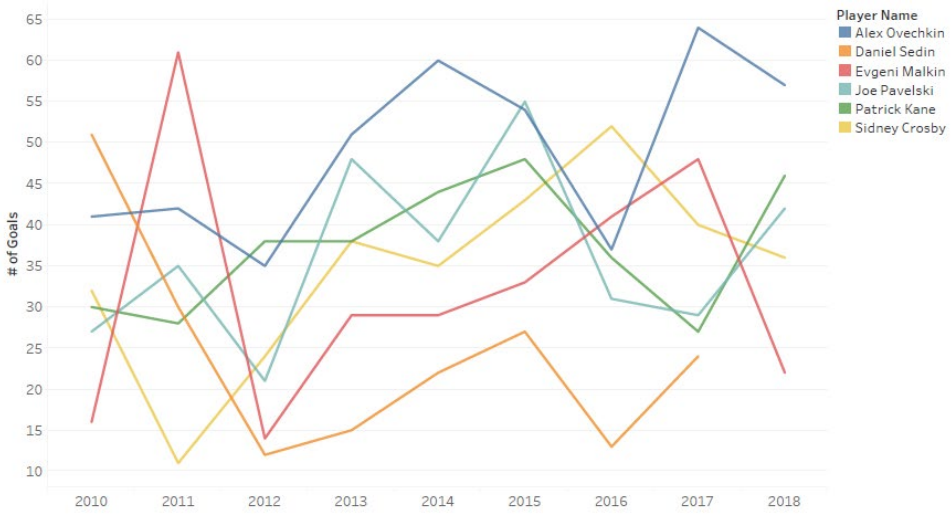
```
SELECT SUBSTRING(game_id, 1, 4) AS season,
    player_id,
    first_name,
    last_name,
    primary_position,
    COUNT(*) AS goal_count
FROM goals
GROUP BY season,
    player_id;
```

SQL Result:

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

season	player_id	first_name	last_name	primary_position	goal_count
2010	8470667	Clarke	MacArthur	LW	21
2010	8467831	Jeff	Halpern	C	11
2010	8470283	Tim	Brent	C	8
2010	8473548	Phil	Kessel	RW	33
2010	8471310	Dustin	Boyd	C	1
2010	8464975	Daniel	Briere	C	42
2010	8467361	Blair	Betts	C	5
2010	8471311	Tyler	Kennedy	C	23
2010	8473512	Claude	Giroux	C	27

Tableau Plot:



ANALYTICAL TASK 9

What is the distribution of yearly goal totals per team?

Code:

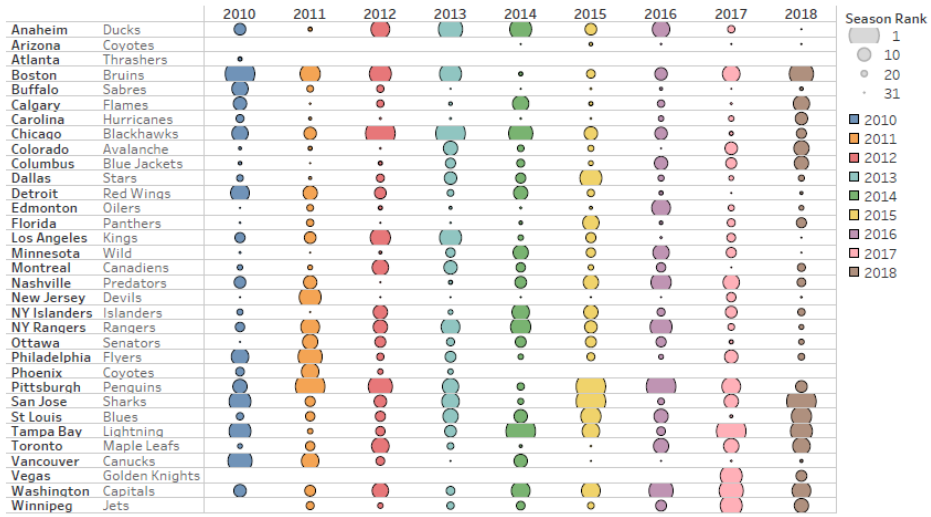
```
SELECT *,
    RANK() OVER (
        PARTITION BY season
        ORDER BY goal_count DESC
    ) season_rank
FROM (
    SELECT SUBSTRING(g.game_id, 1, 4) AS season,
        g.team_id_for,
        ti.short_name,
        ti.team_name,
        COUNT(*) AS goal_count
    FROM goals AS g
    LEFT JOIN team_info AS ti
        ON g.team_id_for = ti.team_id
    GROUP BY season,
        g.team_id_for
) AS team_goals;
```

SQL Result:

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

season	team_id_for	short_name	team_name	goal_count
2010	10	Toronto	Maple Leafs	225
2010	8	Montreal	Canadiens	230
2010	4	Philadelphia	Flyers	281
2010	5	Pittsburgh	Penguins	257
2010	12	Carolina	Hurricanes	239
2010	30	Minnesota	Wild	210
2010	21	Colorado	Avalanche	219
2010	16	Chicago	Blackhawks	270
2010	22	Edmonton	Oilers	194

Tableau Plot:



REFERENCES

- https://en.wikipedia.org/wiki/National_Hockey_League
- https://www.kaggle.com/martinellis/nhl-game-data
- https://www.kaggle.com/murphydan/checking-in-on-checks-and-other-nhl-data