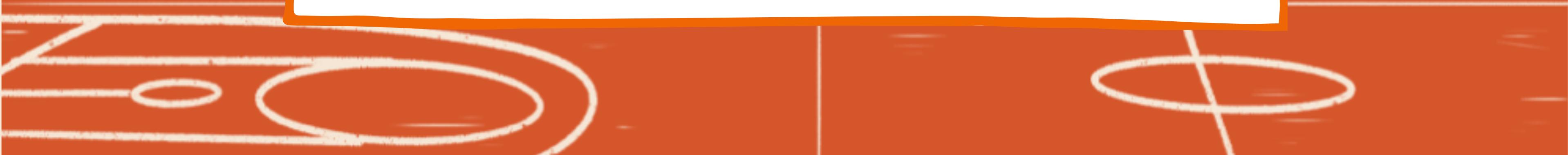


# **NBA 2024-2025 Statistics Database**

**AZIZ UMARBAEV**







DESIGN AND  
IMPLEMENT A  
COMPLETE DATABASE  
SYSTEM



SHOWCASE  
ADVANCED SQL  
CAPABILITIES



IMPLEMENT REAL-  
WORLD SPORTS  
ANALYTICS

## Core Objectives



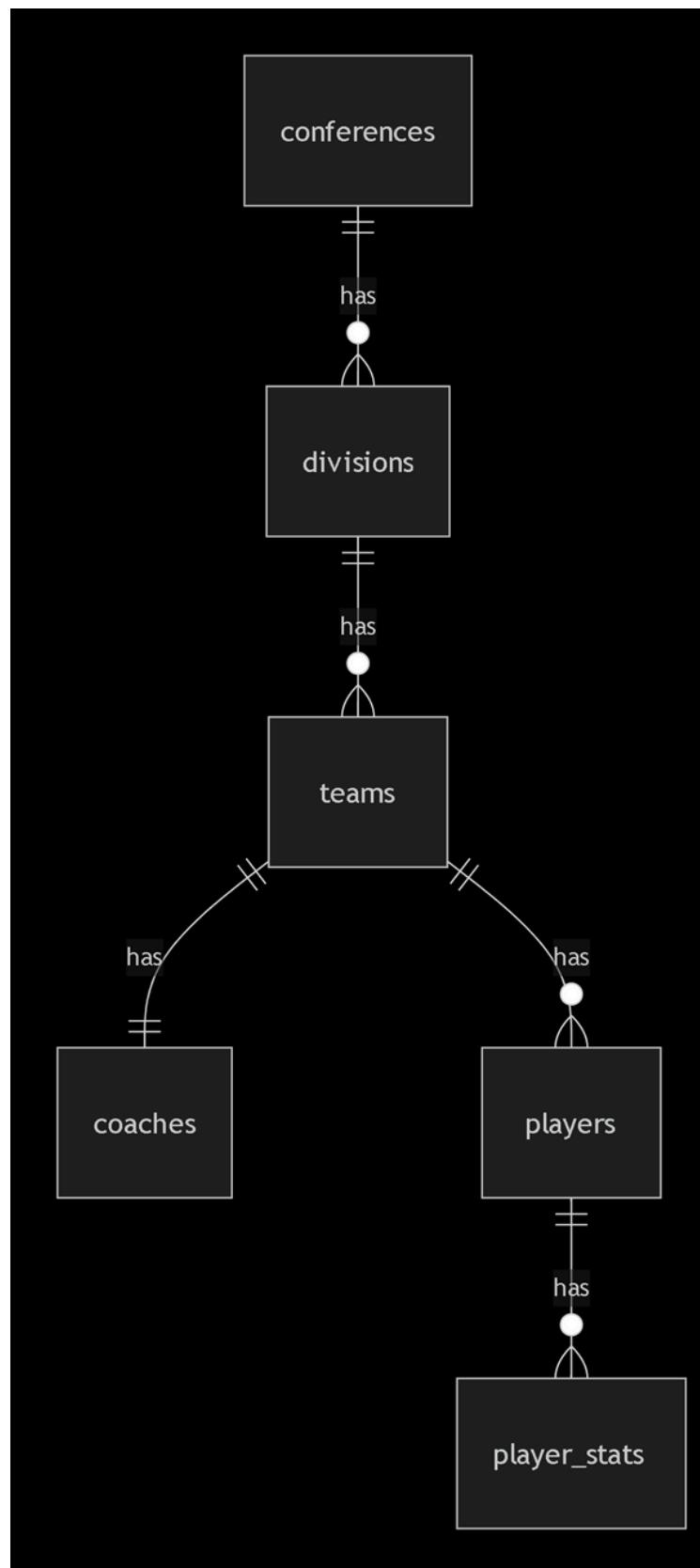
A COMPREHENSIVE  
POSTGRESQL DATABASE  
TRACKING THE COMPLETE  
2024-2025 NBA SEASON

# DATABASE ARCHITECTURE

CONFERENCES (1) → DIVISIONS (many) → TEAMS (many) → PLAYERS (many) → PLAYER\_STATS (many)

→ COACHES (1:1 with teams)





# DATABASE ARCHITECTURE

ONE CONFERENCE HAS MANY DIVISIONS  
ONE DIVISION HAS MANY TEAMS  
ONE TEAM HAS MANY PLAYERS  
ONE TEAM HAS ONE CURRENT COACH  
ONE PLAYER HAS MANY SEASON STATS

## NORMALIZED SCHEMA

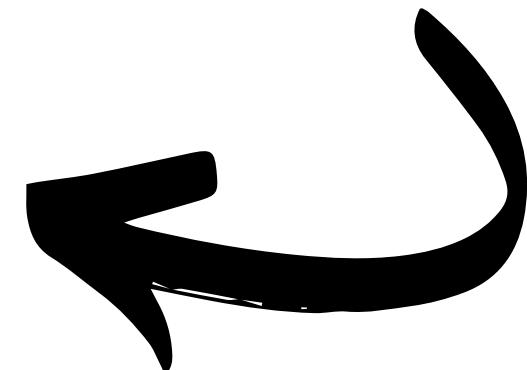
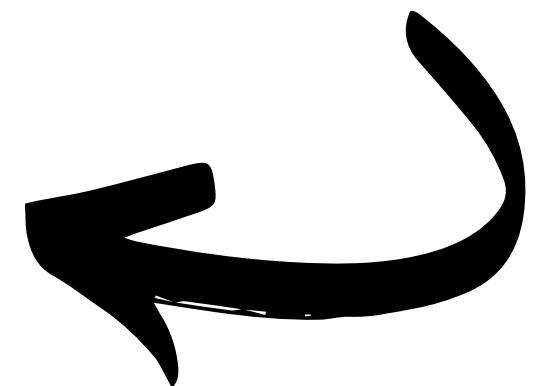


Table	Purpose	Records
conferences	Eastern/Western conferences	2
divisions	6 NBA divisions	6
teams	30 NBA teams with stats	30
players	Player biographical data	150+
player_stats	Season averages per player	150+
coaches	Head coaches	30

# Project Structure & Organization

```
NBA-DATABASE/  
    ├── DATABASE/  
    │   ├── SCHEMA/  
    │   ├── DATA/  
    │   └── FUNCTIONS/  
    ├── QUERIES/  
    ├── DOCS/  
    └── README.MD
```



# Data Population Strategy



**Foundational Data:**  
Conferences → Divisions → Teams



**Incremental Loading:**  
Team-by-team approach for testing

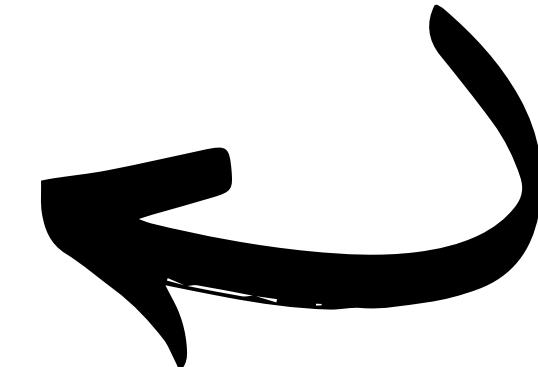
**Team Data:**  
Coaches → Players → Player Stats



```
SELECT
    p.first_name || ' ' || p.last_name AS player_name,
    t.team_name,
    c.conference_name AS conference,
    ps.position,
    ps.points_per_game AS ppg,
    ROUND(ps.field_goal_percentage * 100, 1) || '%' AS fg_pct,
    ROUND(ps.three_point_percentage * 100, 1) || '%' AS three_p_pct,
    ROUND(ps.effective_field_goal_percentage * 100, 1) || '%' AS efg_pct,
    ps.minutes_per_game AS mpg,
    ps.games_played
FROM player_stats ps
JOIN players p ON ps.player_id = p.player_id
JOIN teams t ON p.team_id = t.team_id
JOIN divisions d ON t.division_id = d.division_id
JOIN conferences c ON d.conference_id = c.conference_id
WHERE ps.season = '2024-25'
    AND ps.games_played > 20
ORDER BY ps.points_per_game DESC
LIMIT 15;
```



## Scoring Leaders with Advanced Shooting Stats



Top 15 scorers with detailed shooting breakdown

```
SELECT
    p.first_name || ' ' || p.last_name AS player_name,
    t.team_name,
    ps.position,
    ps.steals_per_game AS spg,
    ps.blocks_per_game AS bpg,
    ps.defensive_rebounds_per_game AS drpg,
    (ps.steals_per_game + ps.blocks_per_game) AS
    defensive_impact
FROM player_stats ps
JOIN players p ON ps.player_id = p.player_id
JOIN teams t ON p.team_id = t.team_id
WHERE ps.season = '2024-25'
    AND ps.games_played > 20
    AND (ps.steals_per_game > 1.5 OR ps.blocks_per_game > 1.0)
ORDER BY defensive_impact DESC;
```



## DEFENSIVE SPECIALISTS: Steals & Blocks Leaders



Find players  
who excel  
defensively

## Most Efficient Scorers



```
SELECT
    p.first_name || ' ' || p.last_name AS player_name,
    t.team_name,
    c.conference_name AS conference,
    ps.points_per_game AS ppg,
    ROUND(ps.field_goal_percentage * 100, 1) || '%' AS fg_pct,
    ROUND(ps.effective_field_goal_percentage * 100, 1) || '%' AS
    efg_pct,
    ps.field_goals_attempted_per_game AS fga
FROM player_stats ps
JOIN players p ON ps.player_id = p.player_id
JOIN teams t ON p.team_id = t.team_id
JOIN divisions d ON t.division_id = d.division_id
JOIN conferences c ON d.conference_id = c.conference_id
WHERE ps.season = '2024-25'
    AND ps.games_played > 20
    AND ps.points_per_game > 15
    AND ps.field_goal_percentage > 0.5
ORDER BY ps.points_per_game * ps.field_goal_percentage
DESC
LIMIT 10;
```



(High PPG + FG%)

```
SELECT
    p.first_name || ' ' || p.last_name AS player_name,
    t.team_name,
    ps.points_per_game AS ppg,
    ps.games_played,
    ps.games_started,
    ROUND(ps.games_started::DECIMAL / ps.games_played * 100, 1) || '%'
AS start_rate,
    ps.minutes_per_game AS mpg,
    ROUND(ps.points_per_game / NULLIF(ps.minutes_per_game, 0) * 36,
1) AS points_per_36
FROM player_stats ps
JOIN players p ON ps.player_id = p.player_id
JOIN teams t ON p.team_id = t.team_id
WHERE ps.season = '2024-25'
    AND ps.games_played > 40
    AND ps.games_started < ps.games_played * 0.5 -- Started less than
50% of games
    AND ps.points_per_game >= 12
    AND ps.minutes_per_game >= 20
ORDER BY ps.points_per_game DESC
LIMIT 10;
```



## Sixth Man Candidates

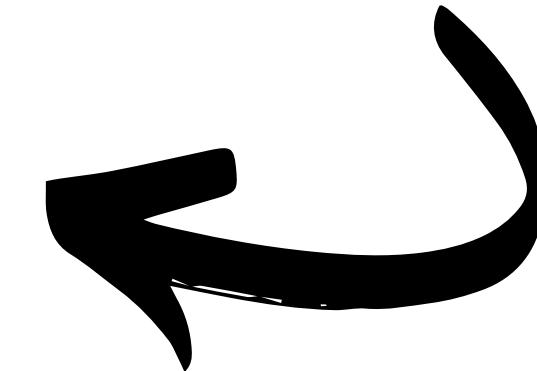


High-scoring players  
who don't start many  
games

```
SELECT
    t.team_name,
    c.conference_name AS conference,
    t.offensive_rating AS ortg,
    t.points_per_game AS ppg,
    t.pace,
    t.wins,
    t.losses
FROM teams t
JOIN divisions d ON t.division_id = d.division_id
JOIN conferences c ON d.conference_id = c.conference_id
ORDER BY t.offensive_rating DESC -- Higher is better for
offense
LIMIT 5;
```



## Top 5 Offensive Teams

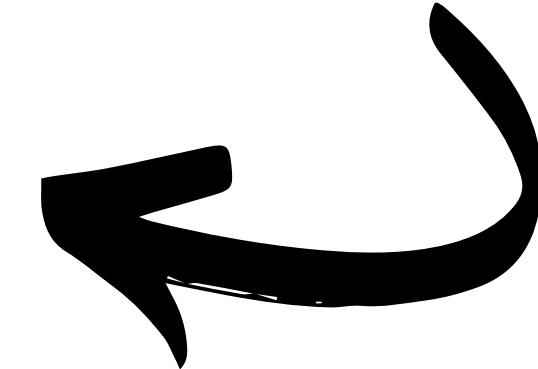


(Highest ORTG)

```
SELECT
    t.team_name,
    c.conference_name AS conference,
    t.defensive_rating AS drtg,
    t.points_allowed_per_game AS papg,
    t.wins,
    t.losses,
    ROUND(t.wins::DECIMAL / NULLIF(t.wins + t.losses, 0) * 100,
1) || '%' AS win_pct
FROM teams t
JOIN divisions d ON t.division_id = d.division_id
JOIN conferences c ON d.conference_id = c.conference_id
ORDER BY t.defensive_rating ASC -- Lower is better for
defense
LIMIT 5;
```

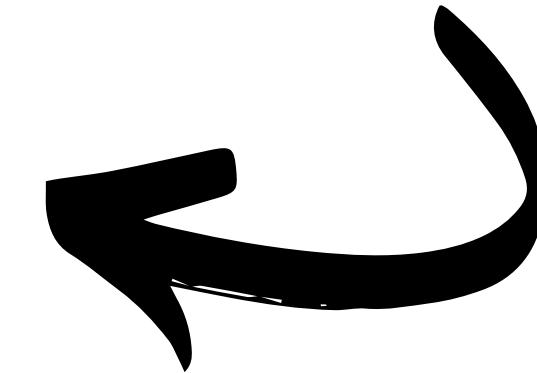


## Top 5 Defensive Teams



(Lowest DRTG)

## NET RATING



Top 5 Teams by Net Rating

```
SELECT
    t.team_name,
    c.conference_name AS conference,
    t.offensive_rating AS off_rtg,
    t.defensive_rating AS def_rtg,
    t.offensive_rating - t.defensive_rating AS net_rating,
    t.wins,
    t.losses,
    ROUND(t.wins::DECIMAL / NULLIF(t.wins + t.losses, 0) * 100,
1) || '%' AS win_pct
FROM teams t
JOIN divisions d ON t.division_id = d.division_id
JOIN conferences c ON d.conference_id = c.conference_id
ORDER BY net_rating DESC
LIMIT 5;
```





Thank  
you!

