# Query Scenarios and SQL Syntax

This document outlines the query scenarios for an MMA database and includes the SQL syntax used to answer each scenario. The queries have been designed to extract meaningful insights from the database, tailored for a business managing MMA fighters, coaches, gyms, sponsors, and contracts.

## 1. Retrieve fighters' records for those weighing between 155 to 200 lbs.

This query retrieves the names, records, and weights of fighters who fall within the 155 to 200-pound weight range. This information can be useful for identifying competitors in specific weight classes for matchmaking or promotions.

SQL Syntax:

SELECT F.Name AS FighterName, FR.Wins, FR.Losses, FR.Draws, F.Weight  
FROM Fighter F  
JOIN FightRecord FR ON F.FighterID = FR.FighterID  
WHERE F.Weight BETWEEN 155 AND 200;

## 2. Get a combined list of fighters, their coaches, gyms, and sponsors.

This query combines data from multiple tables to provide a comprehensive overview of each fighter, including their coach, gym affiliation, and sponsor. This helps in analyzing relationships between fighters and their support systems.

SQL Syntax:

SELECT F.Name AS FighterName, C.Name AS CoachName, G.Name AS GymName, S.Name AS SponsorName  
FROM Fighter F  
JOIN Coach C ON F.FighterID = C.CoachID  
JOIN Gym G ON C.GymID = G.GymID  
JOIN FighterSponsor FS ON F.FighterID = FS.FighterID  
JOIN Sponsor S ON FS.SponsorID = S.SponsorID;

## 3. Retrieve fighters' contracts sorted by payment amount from highest to lowest.

This query ranks fighters by the payment amounts in their contracts, from highest to lowest. This can help the organization identify top earners and assess their value to the business.

SQL Syntax:

SELECT F.Name AS FighterName, FC.Payment  
FROM Fighter F  
JOIN FightContract FC ON F.FighterID = FC.FighterID  
ORDER BY FC.Payment DESC;

## 4. Get a list of fighters who have no recent injuries.

This query identifies fighters who have no recent injuries, indicating they are fit for competition. This is critical for scheduling fights and maintaining health standards.

SQL Syntax:

SELECT F.Name AS FighterName, MR.RecentInjury  
FROM Fighter F  
JOIN MedicalRecord MR ON F.FighterID = MR.FighterID  
WHERE MR.RecentInjury = 'No Recent Injury';

## 5. Calculate the total number of wins by fighters grouped by their country of origin.

This query calculates the total wins for fighters from each country. It can be used to analyze geographic trends and evaluate the performance of fighters based on their nationality.

SQL Syntax:

SELECT F.CountryOfOrigin, SUM(FR.Wins) AS TotalWins  
FROM Fighter F  
JOIN FightRecord FR ON F.FighterID = FR.FighterID  
GROUP BY F.CountryOfOrigin  
ORDER BY TotalWins DESC;

## 6. List fighters and the gyms they are associated with.

This query links fighters to the gyms they are training at. It provides insight into training affiliations and can be used to highlight the prominence of certain gyms.

SQL Syntax:

SELECT F.Name AS FighterName, G.Name AS GymName  
FROM Fighter F  
JOIN Gym G ON F.FighterID = G.GymID;

## 7. Get a list of fighters sponsored by a specific company, e.g., 'Nike'.

This query retrieves fighters sponsored by a specific company, such as 'Nike'. This can help track sponsorship relationships and evaluate brand partnerships.

SQL Syntax:

SELECT F.Name AS FighterName, S.Name AS SponsorName  
FROM Fighter F  
JOIN FighterSponsor FS ON F.FighterID = FS.FighterID  
JOIN Sponsor S ON FS.SponsorID = S.SponsorID  
WHERE S.Name = 'Nike';

## 8. Retrieve all coaches and their specialties.

This query lists all coaches along with their specialties. This information helps in understanding the expertise available within the organization.

SQL Syntax:

SELECT C.Name AS CoachName, C.Specialty  
FROM Coach C;

## 9. Find fighters with contracts that are expiring in the next 6 months.

This query identifies fighters whose contracts will expire within the next 6 months. It helps the organization plan contract renewals and negotiations proactively.

SQL Syntax:

SELECT F.Name AS FighterName, FC.ContractEnd  
FROM Fighter F  
JOIN FightContract FC ON F.FighterID = FC.FighterID  
WHERE FC.ContractEnd BETWEEN CURDATE() AND DATE\_ADD(CURDATE(), INTERVAL 6 MONTH);

## 10. Calculate the total contract payments for fighters grouped by their weight class.

This query calculates the total payments made to fighters in each weight class. It can be used to analyze how financial resources are allocated across different divisions.

SQL Syntax:

SELECT F.WeightClass, SUM(FC.Payment) AS TotalPayment  
FROM Fighter F  
JOIN FightContract FC ON F.FighterID = FC.FighterID  
GROUP BY F.WeightClass  
ORDER BY TotalPayment DESC;

## 11. Retrieve all fighters, their weight class, gym, and coach information using a view.

This query uses a view to combine fighter, gym, and coach information in a single result. It simplifies queries for reporting purposes and ensures consistency in data retrieval.

SQL Syntax:

-- Create a view for fighters with their gyms and coaches  
CREATE VIEW FighterGymCoach AS  
SELECT   
 F.FighterID,  
 F.Name AS FighterName,  
 F.WeightClass,  
 G.Name AS GymName,  
 C.Name AS CoachName  
FROM Fighter F  
JOIN Coach C ON F.FighterID = C.CoachID  
JOIN Gym G ON C.GymID = G.GymID;  
  
-- Query the view  
SELECT FighterName, WeightClass, GymName, CoachName  
FROM FighterGymCoach  
WHERE WeightClass = 'Lightweight';