

Příkazy pro RDBS

Průměr záznamů na celou databázi

```
SELECT (  
    (SELECT COUNT(*) FROM players) + (SELECT COUNT(*) FROM clubs) + (SELECT COUNT(*)  
    FROM leagues) + (SELECT COUNT(*) FROM stadiums) + (SELECT COUNT(*) FROM cities) +  
    (SELECT COUNT(*) FROM nations) + (SELECT COUNT(*) FROM positions)  
    ) / 7 AS avg_records_per_table;
```

Počet záznamů na tabulku

```
SELECT 'players' AS table_name, COUNT(*) AS row_count FROM players  
  
UNION ALL  
  
SELECT 'clubs', COUNT(*) FROM clubs  
  
UNION ALL  
  
SELECT 'leagues', COUNT(*) FROM leagues  
  
UNION ALL  
  
SELECT 'stadiums', COUNT(*) FROM stadiums  
  
UNION ALL  
  
SELECT 'cities', COUNT(*) FROM cities  
  
UNION ALL  
  
SELECT 'nations', COUNT(*) FROM nations  
  
UNION ALL  
  
SELECT 'positions', COUNT(*) FROM positions;
```

Select s vnořeným dotazem

```
SELECT first_name, last_name, club_id  
  
FROM players  
  
WHERE club_id = (  
    SELECT club_id  
    FROM clubs  
    WHERE club_id = (  
        SELECT club_id  
        FROM leagues  
        WHERE league_id = (  
            SELECT league_id  
            FROM stadiums  
            WHERE stadium_id = (  
                SELECT stadium_id  
                FROM cities  
                WHERE city_id = (  
                    SELECT city_id  
                    FROM nations  
                    WHERE nation_id = (  
                        SELECT nation_id  
                        FROM positions  
                        WHERE position_id = 1  
                    )  
                )  
            )  
        )  
    )  
)
```

```
SELECT club_id FROM clubs WHERE name = 'Real Madrid'  
);
```

Select s agregační funkcí a group by

```
SELECT leagues.name AS league_name, COUNT(clubs.club_id) AS total_clubs  
FROM leagues  
JOIN clubs ON leagues.league_id = clubs.league_id  
GROUP BY leagues.name;
```

Select řešící hierarchii nebo rekurzi

```
SELECT p1.first_name AS player, p1.last_name, p2.first_name AS teammate,  
p2.last_name AS teammate_last  
FROM players p1  
JOIN players p2 ON p1.club_id = p2.club_id AND p1.player_id <> p2.player_id  
ORDER BY p1.club_id;
```

VIEW

```
CREATE OR REPLACE VIEW player_info AS  
SELECT p.first_name, p.last_name, c.name AS club_name, l.name AS league_name,  
n.name AS nationality  
FROM players p  
JOIN clubs c ON p.club_id = c.club_id  
JOIN leagues l ON c.league_id = l.league_id  
JOIN nations n ON p.nation_id = n.nation_id;
```

Index na nějaké slupci

```
CREATE INDEX idx_club_name ON clubs(name);
```

```
INSERT INTO clubs VALUES (NULL, 'Liverpool', 1, 1, 2025-01-02);
```

Funkce(volá se takto: SELECT get_avg_stadium_capacity('La Liga');)

```
DELIMITER // CREATE FUNCTION get_avg_stadium_capacity(league_name
VARCHAR(100))
```

```
RETURNS INT DETERMINISTIC
```

```
BEGIN
```

```
DECLARE avg_capacity INT;
```

```
SELECT AVG(s.capacity) INTO avg_capacity
```

```
FROM stadiums s
```

```
JOIN clubs c ON s.stadium_id = c.stadium_id
```

```
JOIN leagues l ON c.league_id = l.league_id
```

```
WHERE l.name = league_name;
```

```
RETURN avg_capacity;
```

```
END //
```

```
DELIMITER ;
```

Procedura(CALL update_shirt_numbers())

```
DELIMITER // CREATE PROCEDURE update_shirt_numbers() BEGIN DECLARE done INT
DEFAULT 0; DECLARE p_id INT; DECLARE new_number INT; DECLARE cur CURSOR FOR
SELECT player_id FROM players; DECLARE CONTINUE HANDLER FOR NOT FOUND SET
done = 1; OPEN cur; read_loop: LOOP FETCH cur INTO p_id; IF done THEN LEAVE
read_loop; END IF; SET new_number = FLOOR(1 + (RAND() * 99)); UPDATE players SET
shirt_number = new_number WHERE player_id = p_id; END LOOP; CLOSE cur; END //
```

```
DELIMITER ;
```

Trigger

```
DELIMITER // CREATE TRIGGER after_player_update AFTER UPDATE ON players FOR EACH
ROW BEGIN INSERT INTO player_updates (player_id, changed_at) VALUES (NEW.player_id,
NOW()); END //
```

Transakce

```
START TRANSACTION;
```

```
-- Vytvoření nového klubu
```

```
INSERT INTO clubs (name, league_id, stadium_id, foundation_date) VALUES ('New FC', 1,  
1, '2025-02-05');
```

```
--Uložení ID nového klubu
```

```
SET @new_club_id = LAST_INSERT_ID();
```

```
-- Přidání hráče do nového klubu
```

```
INSERT INTO players (first_name, last_name, birth_date, position_id, nation_id, club_id,  
shirt_number, right_footed) VALUES ('John', 'Doe', '2000-01-01', 1, 1, @new_club_id, 10, 1);
```

```
COMMIT;
```

Správa uživatelů

```
CREATE USER 'football_admin'@'localhost' IDENTIFIED BY 'pass'; GRANT SELECT, INSERT,  
UPDATE ON football_top_20.* TO 'football_admin'@'localhost';
```

LOCK

```
LOCK TABLE players WRITE;
```

```
UPDATE players SET right_footed = 1 WHERE position_id = 2;
```

```
UNLOCK TABLES;
```