Qt Database Exam: Football Edition

Database Concepts with Football Examples

Exam Structure

- 5 progressive exercises (30 minutes each)
- Focuses on SQLite with football/soccer examples
- Same difficulty level as previous exam
- Tests identical database concepts in sports context

1 Exercise 1: Football Database Fundamentals (30 minutes)

1. Design a database schema for tracking:
   - Teams (name, country, founded year)
   - Players (name, position, shirt number)
   - Matches (date, home team, away team, score)
2. Write Qt code to:
   (a) Create a SQLite database named ”football_manager.db”
   (b) Implement proper error handling for connection
   (c) Close connection when application exits
3. Explain how you would store a player’s position (GK, DEF, MID, FW) in the database

2 Exercise 2: Football Data Operations (30 minutes)

1. Write SQL queries to:
   - Insert Lionel Messi as a forward (FW) with shirt number 10
   - Update a player's position from MID to FW
   - Find all defenders (DEF) from Argentina
2. Create a QSqlQuery to count how many players have shirt numbers greater than 10
3. What's the advantage of using prepared statements when inserting match results?

3 Exercise 3: Football Match Viewer (30 minutes)

1. Create a Qt application with:
   - QTableView showing upcoming matches
   - QComboBox to filter by team
   - Refresh button
2. Implement the team filter using QSqlTableModel's setFilter()
3. How would you display match results (e.g., ”2-1”) in a single table column when the database stores home_goals and away_goals separately?

4 Exercise 4: Football Statistics (30 minutes)

1. Write SQL queries to:
   - Calculate average goals per match
   - Find the top 3 scoring teams
   - Count wins/losses/draws for a specific team

2. Create a Qt function that displays a player's career stats (goals, assists) using multiple database queries

3. How would you implement a "Team of the Week" feature based on match ratings?

5 Exercise 5: Football Database Best Practices (30 minutes)

1. How would you handle:

    - Player transfers between teams
    - Match postponements
    - Historical data archiving

2. Design a transaction for recording all match events (goals, cards, substitutions)

3. Propose a database schema versioning strategy for when new stats (like xG) need to be added

Key Database Concepts Covered

- Database Design: Tables, relationships, data types
- CRUD Operations: Creating, reading, updating football data
- Model-View: Displaying sports statistics
- Complex Queries: Football-specific analytics
- Transactions: Managing match events atomically