

C# | .NET Bootcamp – Final Project

Part 1: Project Setup and LINQ Queries

Objective:

- You have been provided with a project template in C# along with SQLite database configuration.
- Write 10 queries using LINQ to retrieve specific information from the SQLite database.

Instructions:

1. LINQ Queries:

- Write 10 LINQ queries to retrieve the following information from the SQLite database:
 1. Retrieve the count, sum, and average of the total amounts from invoices using a single query.
 2. Fetch a random selection of the top 100 tracks.
 3. Retrieve the top 5 tracks that have the highest sales.
 4. Obtain all customers with invoices containing at least one line item priced less than \$2.
 5. Retrieve the top 5 loyal customers based on the total purchase amount across all their invoices.
 6. Retrieve a list of employees along with their respective managers.
 7. Calculate the average sales per month.
 8. Fetch the top 100 most popular genres for each country.
 9. Retrieve the top 100 most popular tracks based on customer preferences and purchases.
 10. Obtain a list of all customers and details of their last invoice, displaying null when a customer has not made any purchases.

Part 2: Dynamic Filter Expression

Objective:

Develop a dynamic filter generator that allows users to create and execute custom where clauses on the queries.

Instructions:

Implement a functionality that allows users to:

- Define query parameters (e.g., columns to select, filters, sorting).
- Generate and execute queries dynamically based on user inputs.

Part 3: Plugin Development for Loyalty Club Points Calculation**Objective:**

- Create a plugin system to calculate loyalty club points based on specific rules and conditions.

Instructions:**1. Plugin Development:**

- Design and implement a plugin architecture where loyalty club points can be calculated dynamically.
- Plugins should include:
 - Calculation of loyalty points based on the number of tracks purchased per month:
 - 1 track: 5 points
 - 2 tracks: 15 points
 - 3 or more tracks: 30 points
 - Seasonal bonus:
 - Customers earn an additional 50 points if they buy a track during spring (March, April, May).

2. Bonus Task: Parallel Loading:

- Implement parallel loading of plugins to enhance the performance and responsiveness of the application.
- Ensure that plugins can be loaded and executed concurrently without blocking the main application thread.

Submission Requirements:

- **Deadline:** 1403/04/31
- **Deliverables:**
 - Completed C# project with implemented LINQ queries.

- Dynamic query generator functionality.
- Plugin system for loyalty club points calculation.
- Bonus task: Documentation on parallel loading of plugins (if implemented).
- Ensure all code is well-commented and follows best practices.

Additional Notes:

- Properly handle exceptions and edge cases in your implementations.
- Feel free to extend the functionality beyond the given requirements if time permits.