

LINQ and EF Core Tasks for Code_Academy Project

Task 1: Query Data Using LINQ

- 1. Retrieve All Users:** Write a LINQ query to fetch and display all users from your database.
- 2. Filter Users by Age:** Create a LINQ query to get all users older than 18.
- 3. Search Users by Name:** Implement a LINQ query that searches for users whose names contain a specific string.
- 4. Sorting:** Sort the users by their registration date in descending order.

Task 2: Perform Aggregation Operations

- 1. Calculate the total number of users in the database.**
- 2. Find the average age of the users.**
- 3. Determine the maximum and minimum age among the users.**

Task 3: CRUD Operations with EF Core

- 1. Create:** Add a new user to the database with mock data.
- 2. Read:** Fetch and display a user by their ID.
- 3. Update:** Update the email or name of a specific user by ID.
- 4. Delete:** Remove a user by their ID.

Task 4: Advanced LINQ Tasks

- 1. Grouping:** Group users by their age and display the number of users in each group.
- 2. Join Operation:** Assuming you have another table like Courses, write a LINQ query to join users with their enrolled courses.
- 3. Projection:** Select and display only the user names and their email addresses as a new list.

Task 5: Implementing Relationships in EF Core

- 1. Add Relationships:** If you have entities like User and Course, implement a many-to-many relationship between them.
- 2. Eager Loading:** Write a query to include related data (e.g., fetch users and their enrolled courses together).

3. Lazy Loading: Configure lazy loading for a related entity and access it to verify its

Task 6: Write LINQ Queries with Complex Conditions

- 1. Fetch users whose names start with 'A' and are older than 25.**
- 2. Retrieve the first 5 users sorted by their age.**
- 3. Check if all users have a non-null email address.**

Task 7: Performance Optimization

- 1. Use AsNoTracking in EF Core to fetch data where updates are not required.**
- 2. Compare the execution time of LINQ queries with and without .ToList() in a loop.**

Deliverables

- 1. Push your code for each task to a separate branch in your repository (e.g., task1, task2, task3, task4, task5, task6, task7).**
- 2. Add comments in your code to explain the logic.**
- 3. Share the link to the updated repository for review and feedback.**