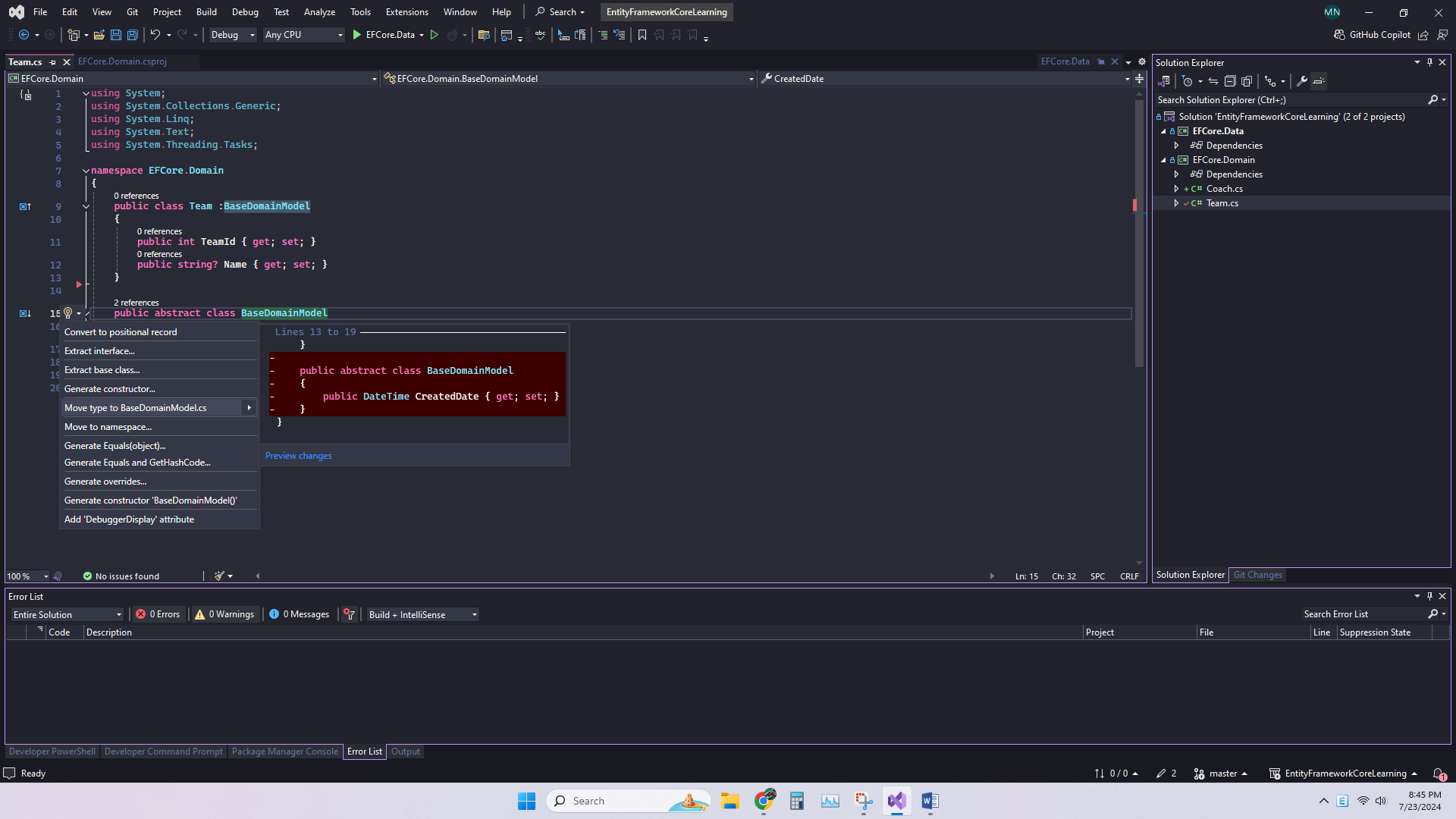
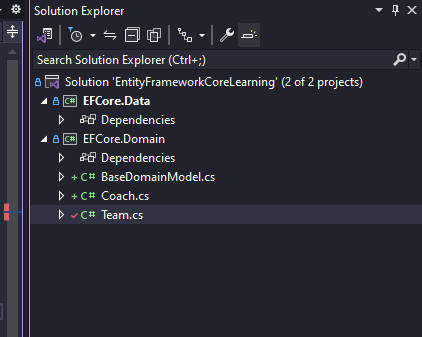
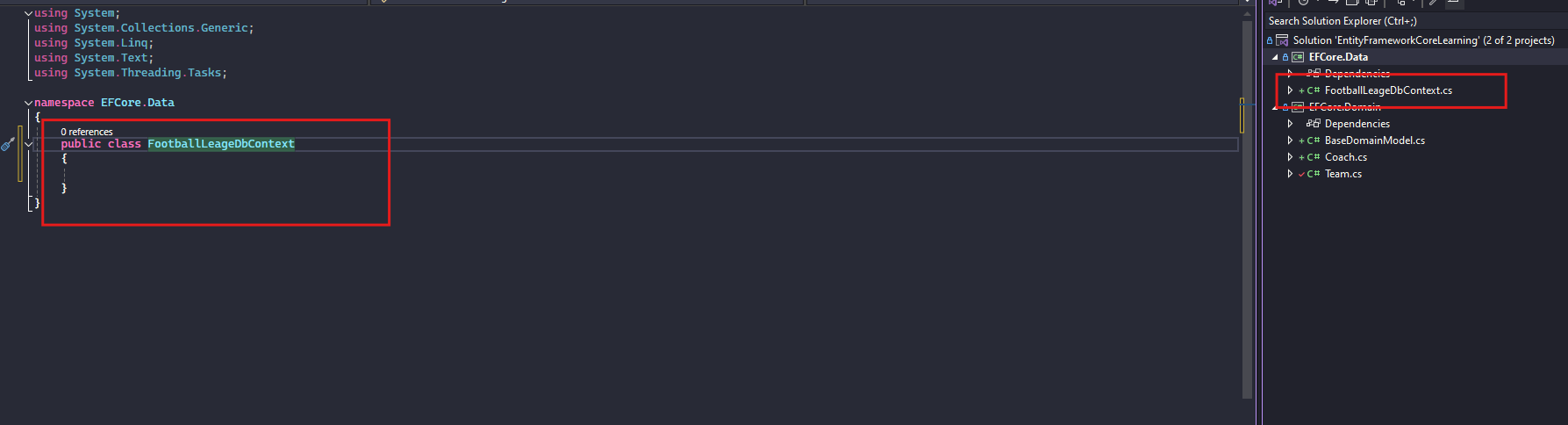
# Create Data Model



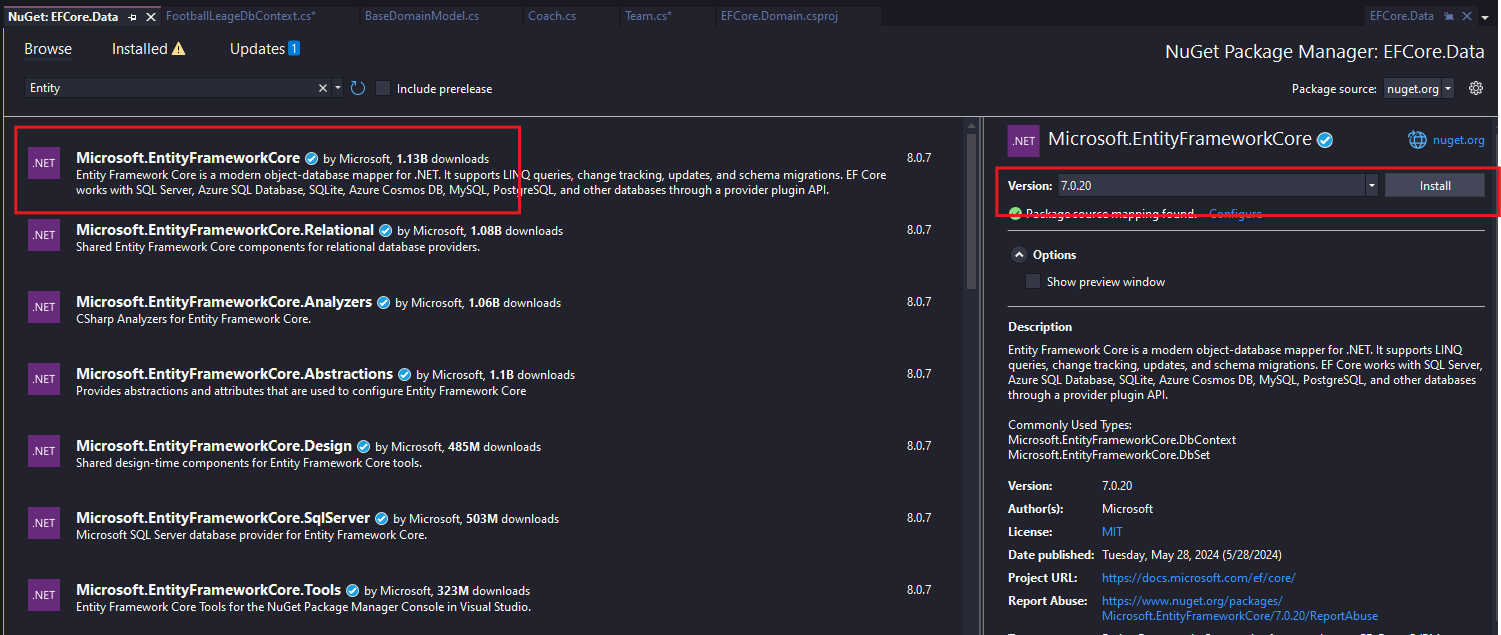
Create Model



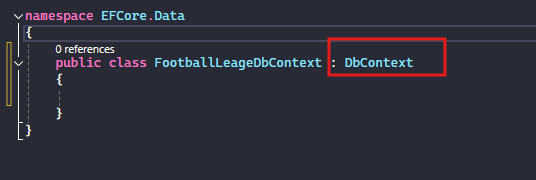
# Create Db Context



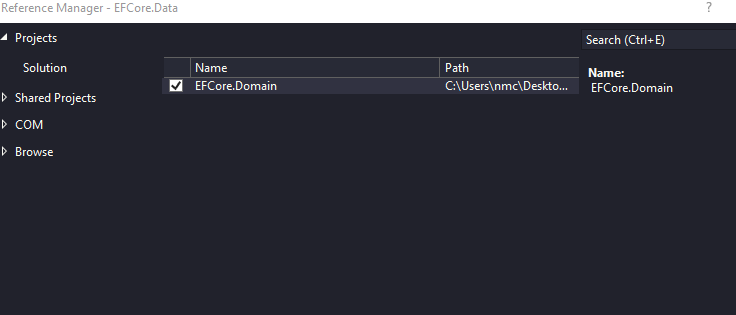
Setting package EFCore on NET 7



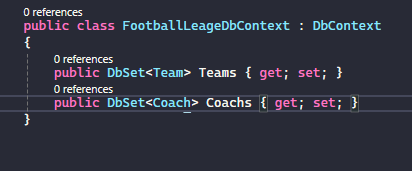
Add db context



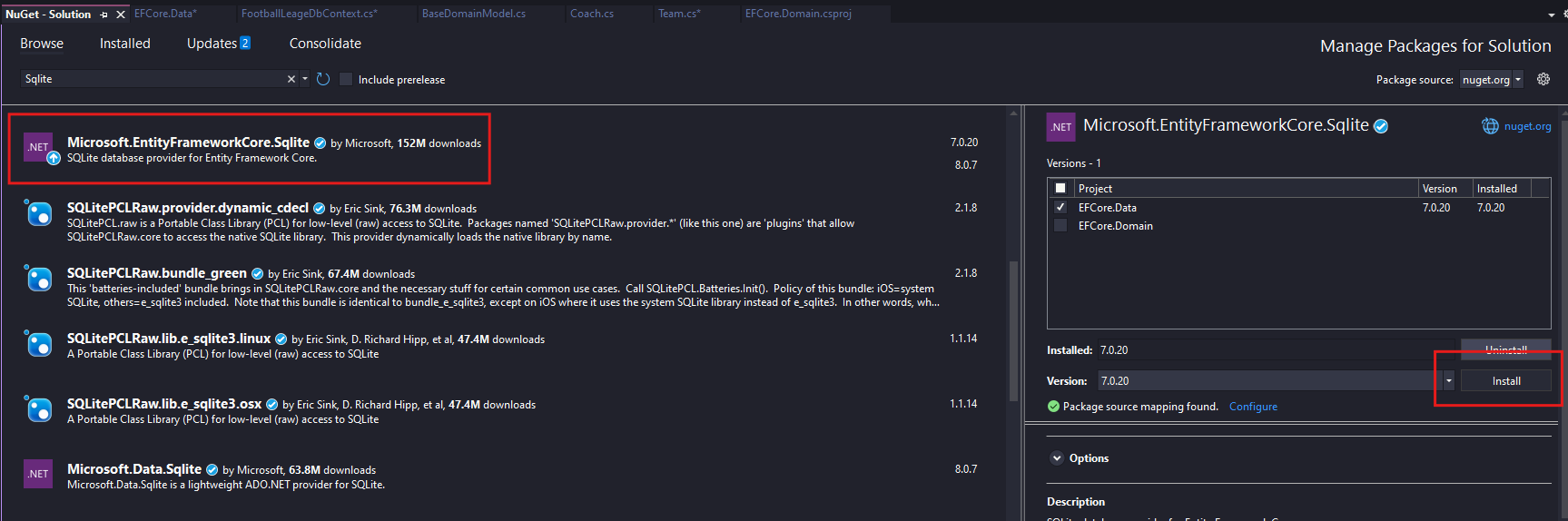
Thêm project ref để kết hợp với project domain



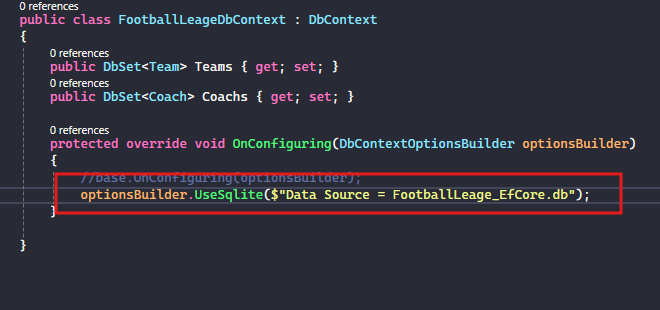
Thêm DbSet



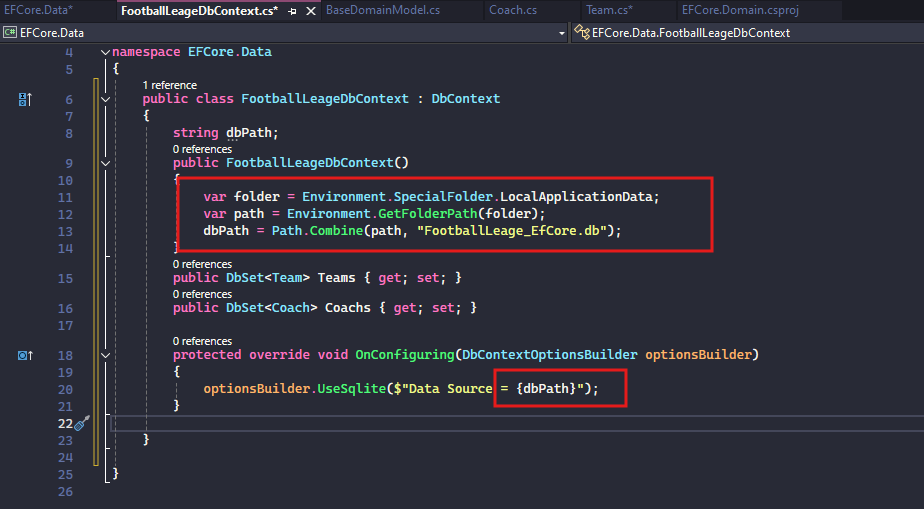
# Tải thư viện cho Efcore hỗ trợ Sqlite



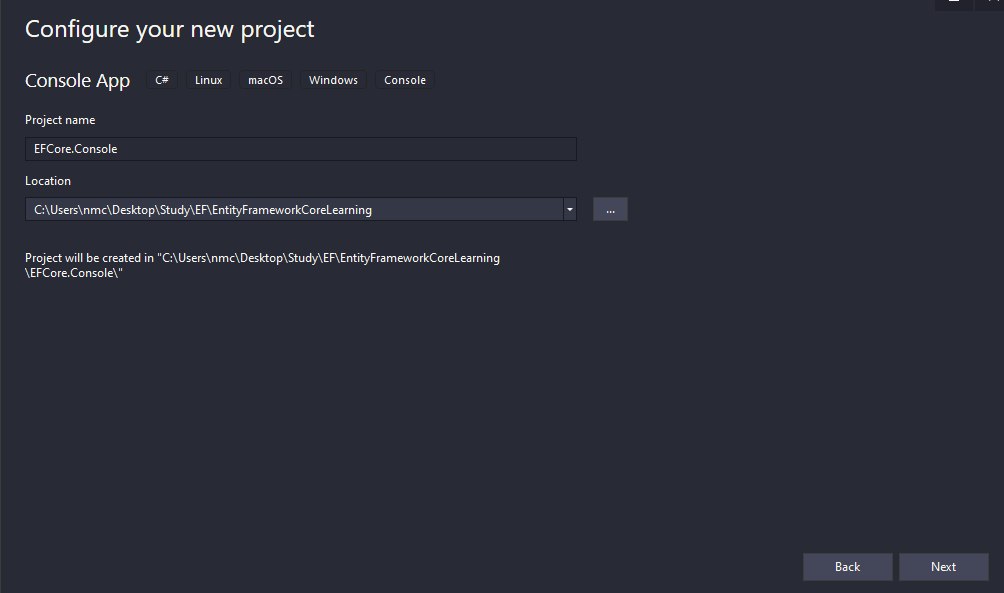
Khai báo dữ liệu cho việc tạo database với sqlite



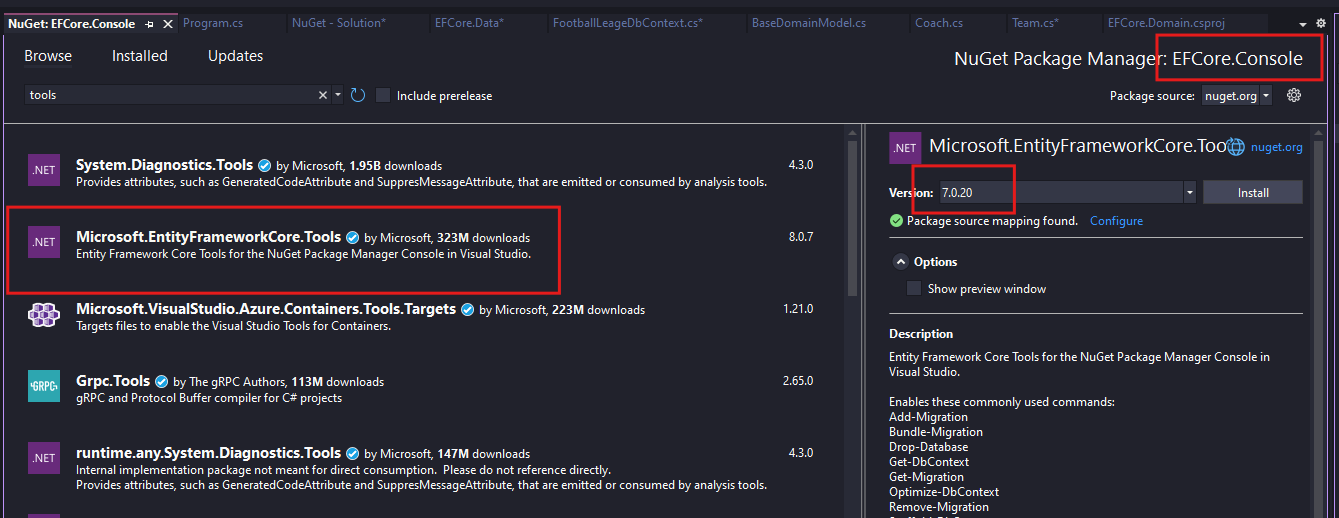
Sửa lại đường dẫn database



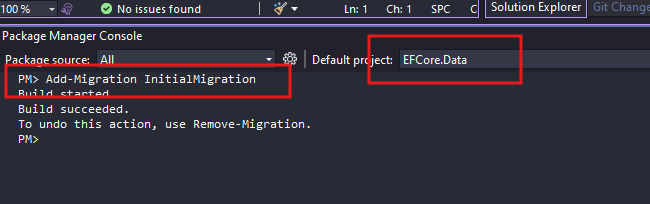
# Thêm Console Project



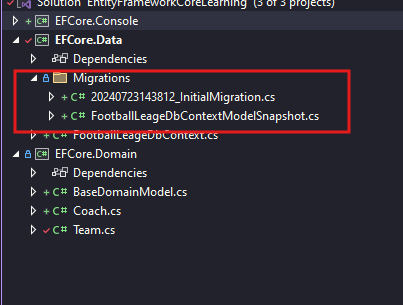
Cài đặt tools

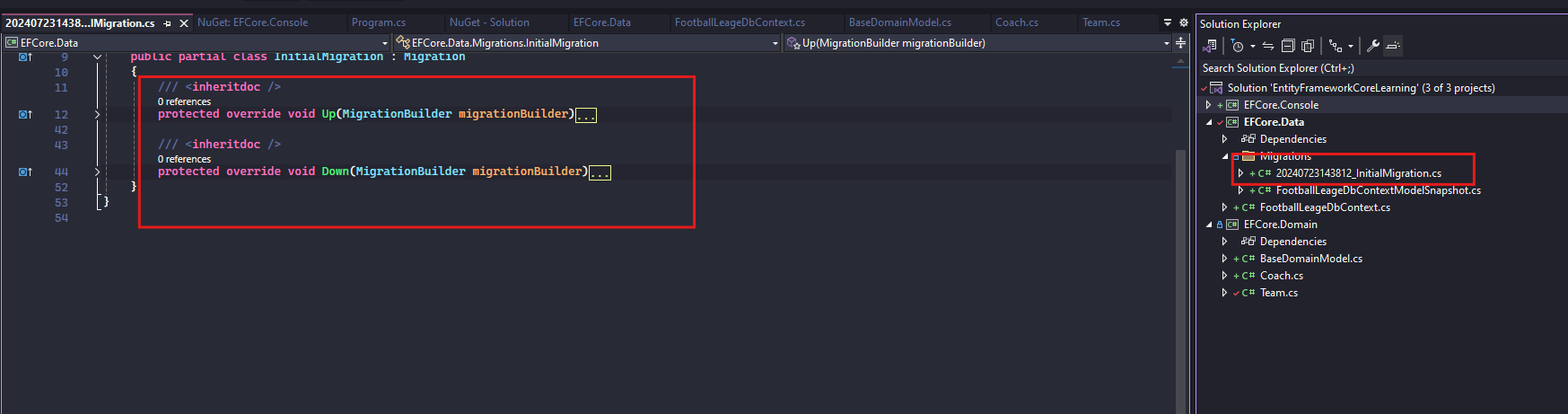


# Add Migration , đây là cơ chế tạo ra database từ code



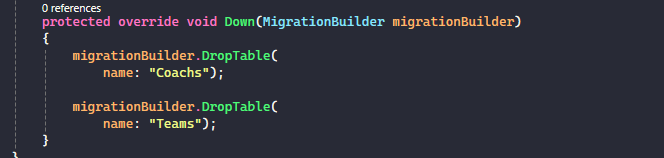
Xuất hiện thư mục Migration





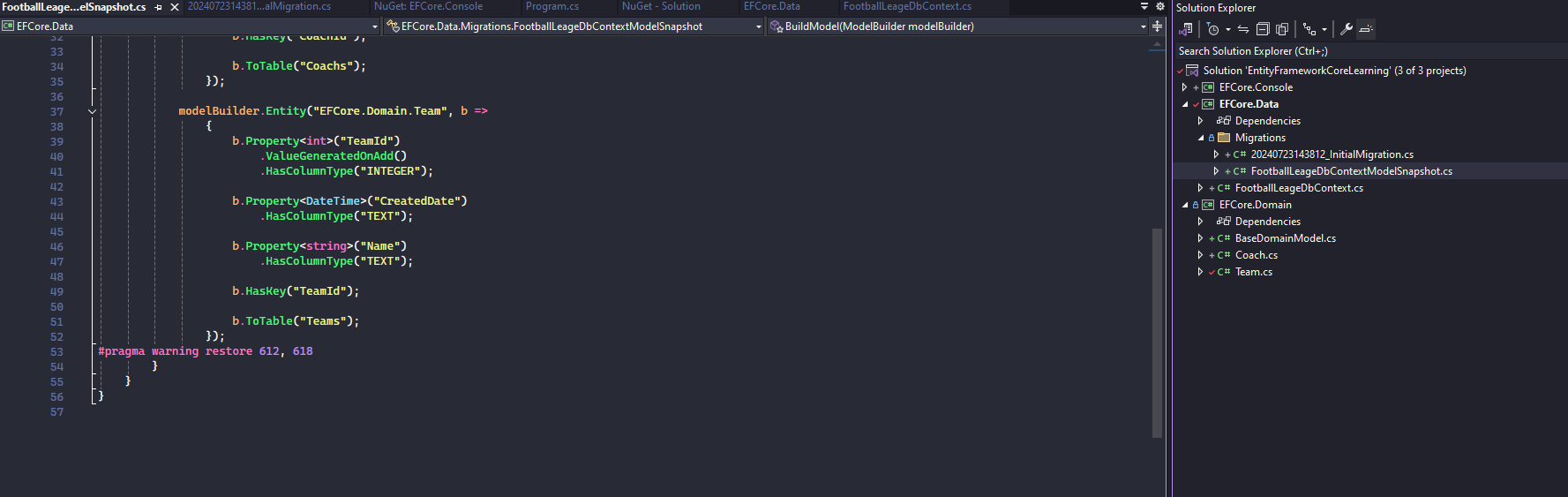
Code sinh ra tự động ở file Migration có hàm Up và Down

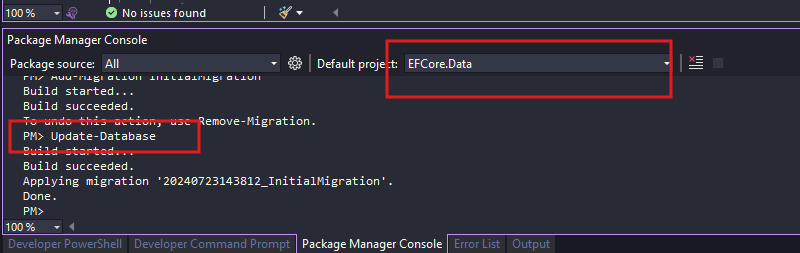


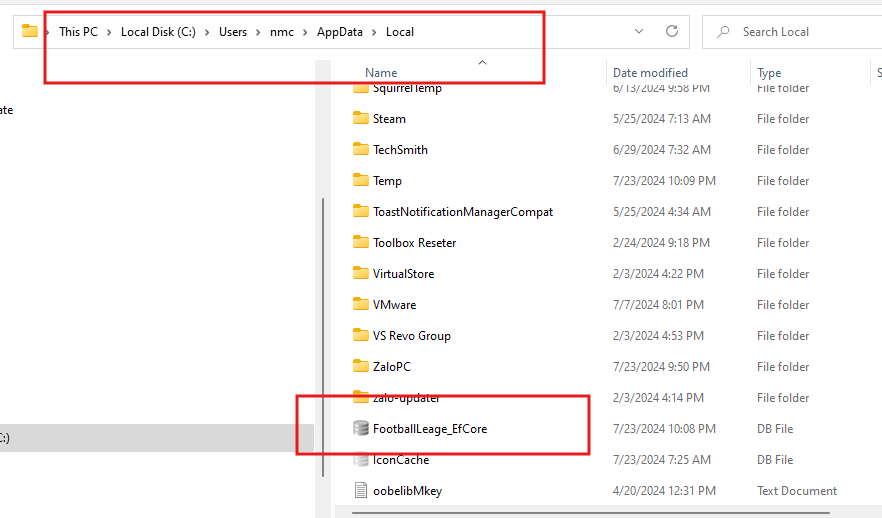


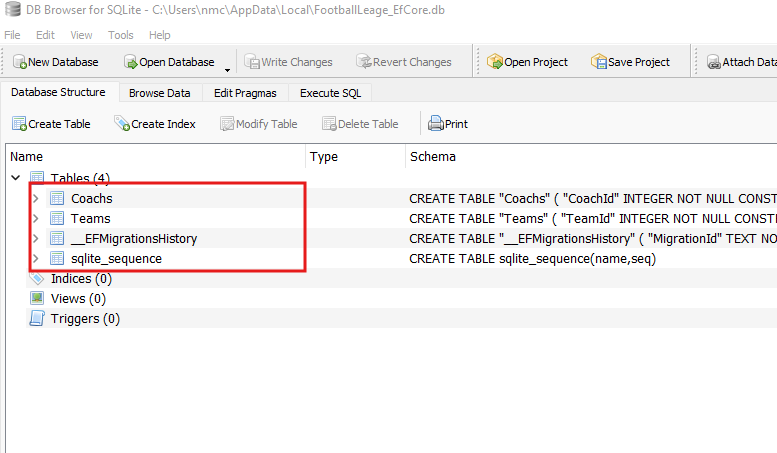
File snapshoot

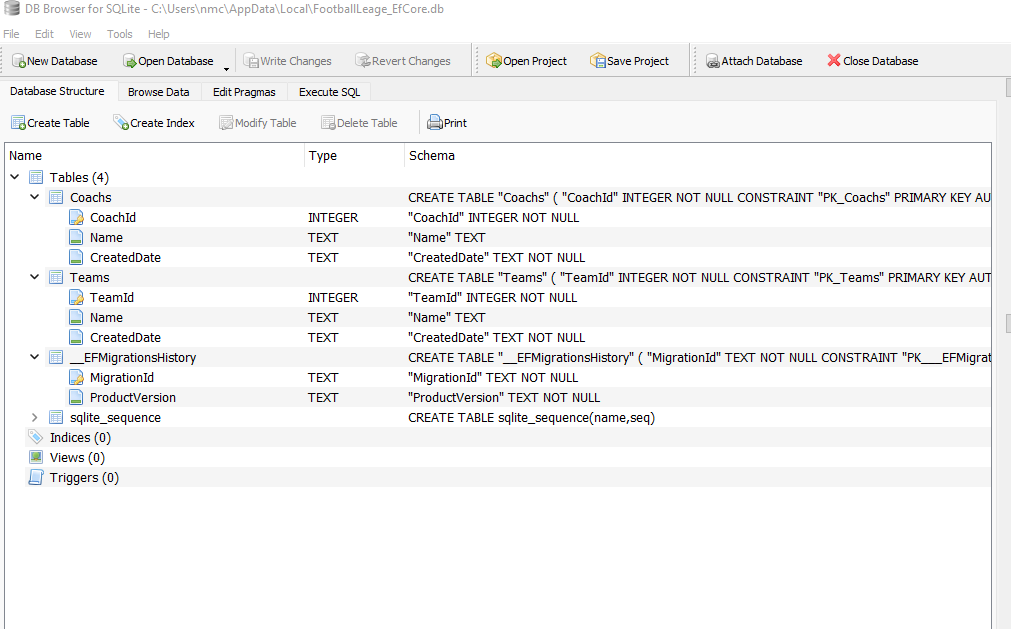


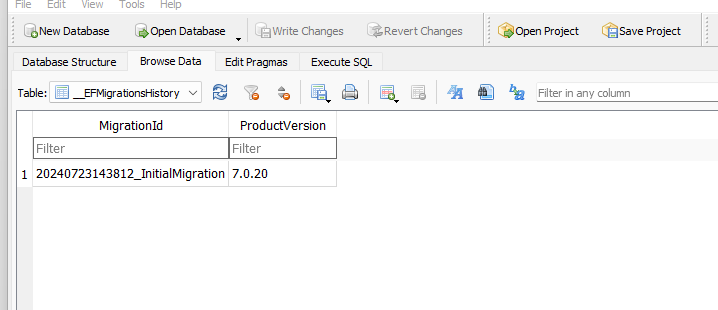




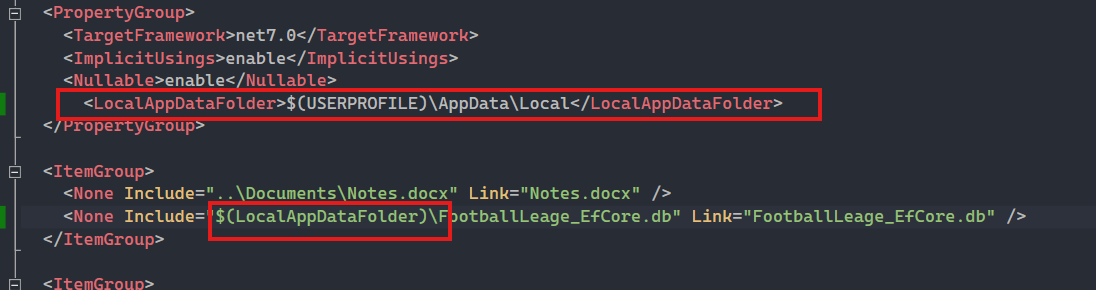




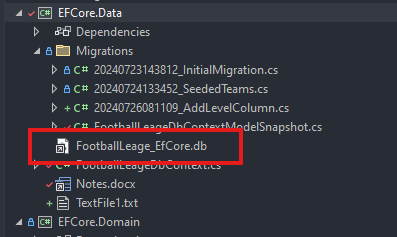




Thêm link trên project để mở database xem khi cần

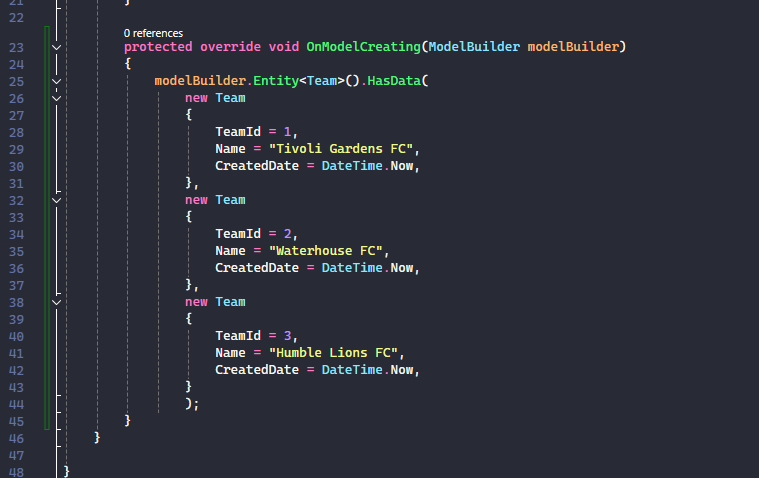


Chúng ta sẽ Add newitem với as link

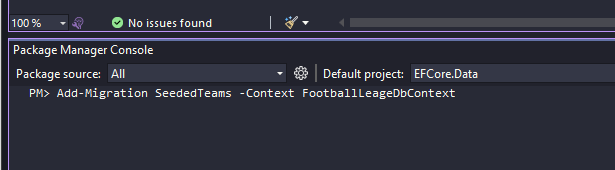


# Thêm dữ liệu hàng loạt

Sử dụng phương thức OnModelCreateing để tạo dữ liệu và thêm vào database



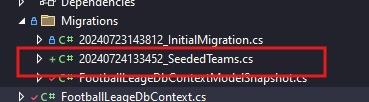
Sau đó Add Migration mới



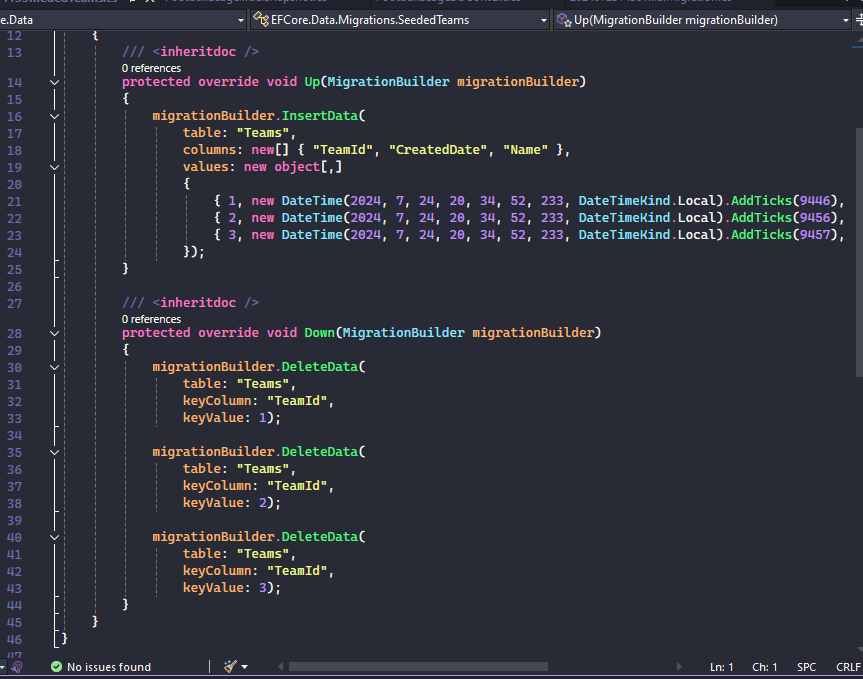
Cú pháp trên có : Add-Migration Mô Tả -Context Tên DbContext cần thêm.

Việc xác định –Context là giúp phân biệt các Dbcontext nếu trong Project của bạn chứa nhiều DbContext

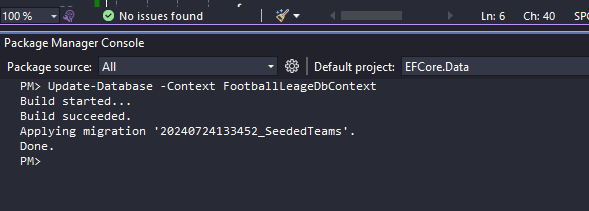
Sau khi Add-Migration xong sẽ xuất hiện file Migration mới



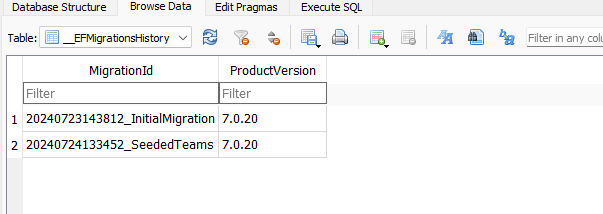
Và code mới tạo ra cũng có hai phương thức Up và Down



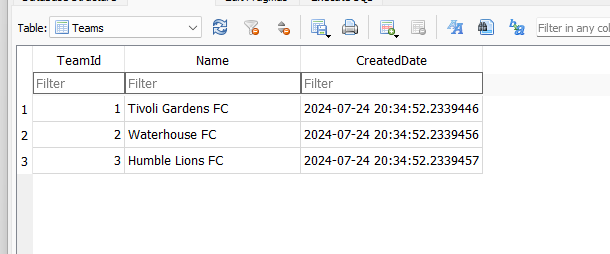
Sau khi Add Migration xong chúng ta sẽ update Database bằng cú pháp bên dưới, cũng xác định DbContext cụ thể



Khi mở cơ sở dữ liệu chúng ta sẽ thấy một mục mới trong bảng Migration

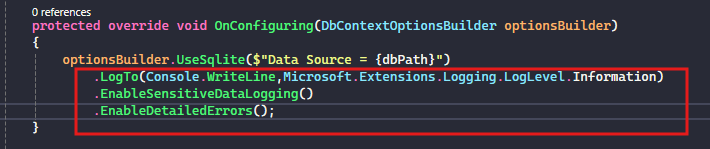


Đồng thời các dữ liệu trong bảng Team cũng được thêm vào



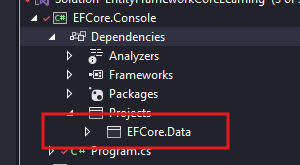
# Thêm tính năng ghi Log vào EF core dưới dạng Console

Ở phần phương thức OnConfiguring chúng ta sẽ thêm các tính năng ghi log các dữ liệu nhạy cảm ra màn hình console, nó đại diện cho các câu lệnh SQL khi chúng ta dùng Linq



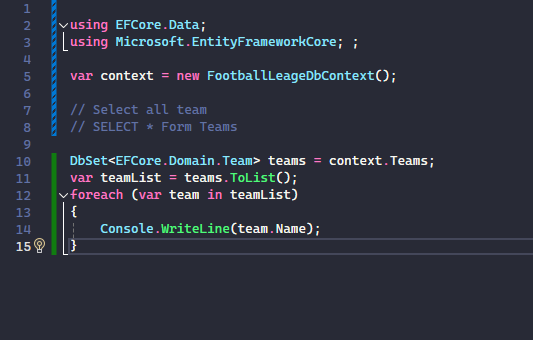
# Truy vấn cơ bản

Ở project Console chúng ta sẽ thêm Reference đến project EFCore.Data để có thể làm việc với DbContext

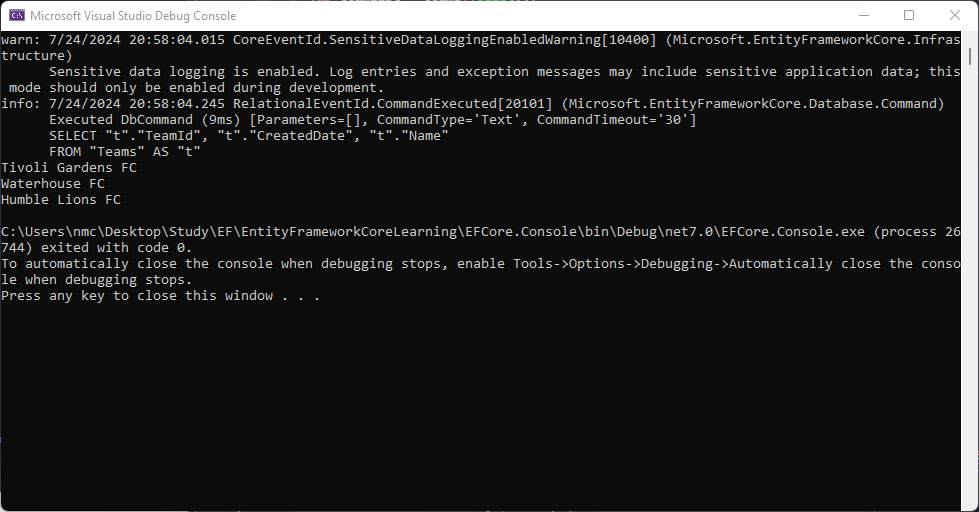


Trong file Program.cs chúng ta thêm code sau

Đoạn code này giúp tạo ra dbcontext , sau đó gọi dbset ra và chuyển các dữ liệu sang dạng list

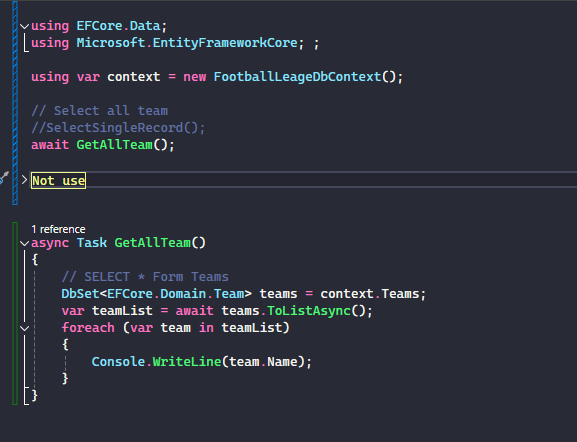


Cửa sổ console sẽ có dòng thông báo , dòng lệnh sql cụ thể và dòng kết quả

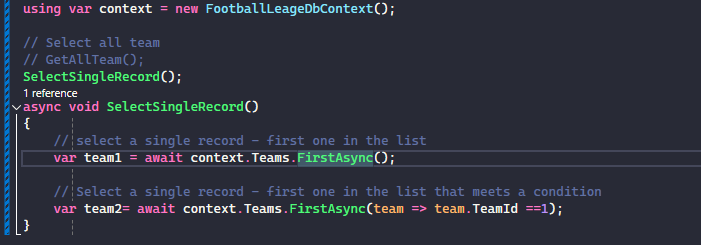


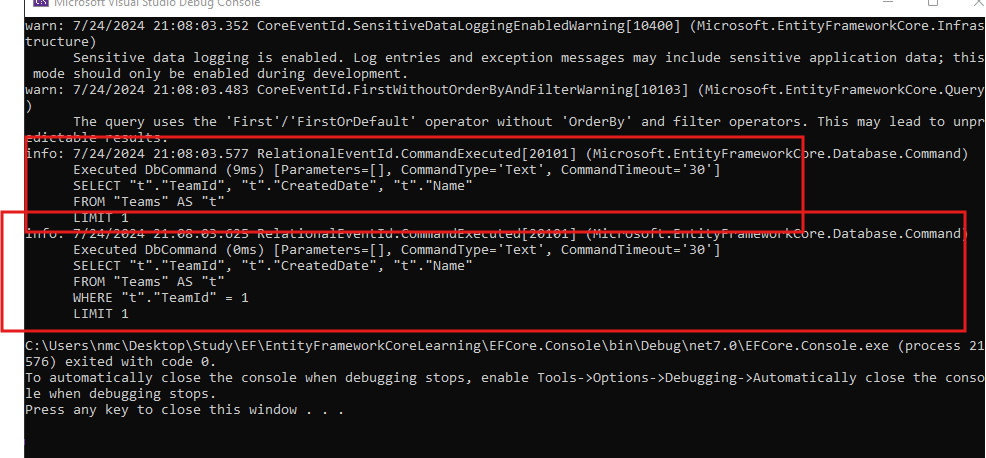
# Dùng Async cho truy vấn

Sửa lại một ít truy vấn cũ sang async



Truy vấn với một recorrd



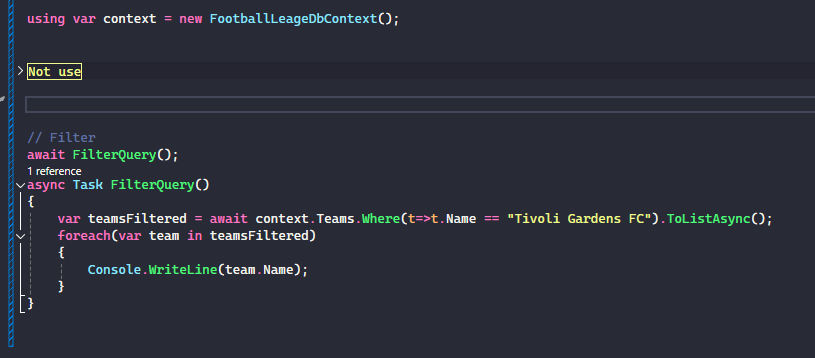


Ngoài ra chúng ta có thể thử FirstOrDefault để tránh gặp lỗi nếu null

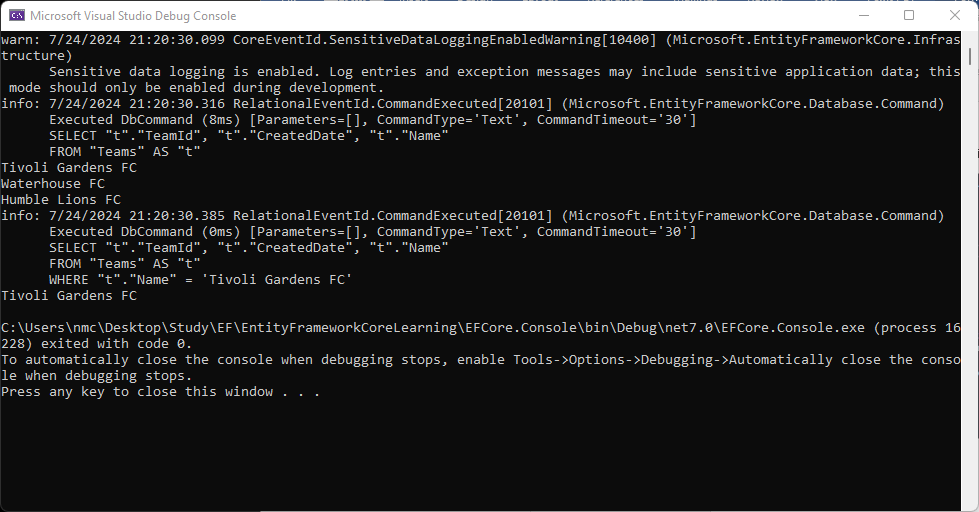
Các lệnh nên thử như single , find dùng để truy vấn rất thuận tiện

# Bộ lọc dữ liệu

Chúng ta sẽ dùng where để tim ra tên đội bóng và in ra



Kêt quả



# Cách cập nhật lại DB khi kéo source từ Git về

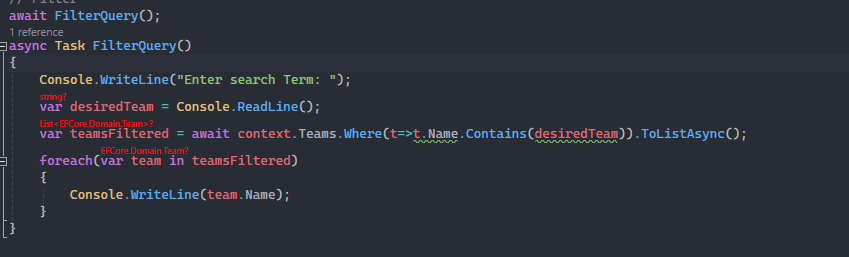
Sử dụng command : Update-Database

# Các bộ lọc dữ liệu bổ sung Contains, Like

Tìm kiếm theo thuật ngữ

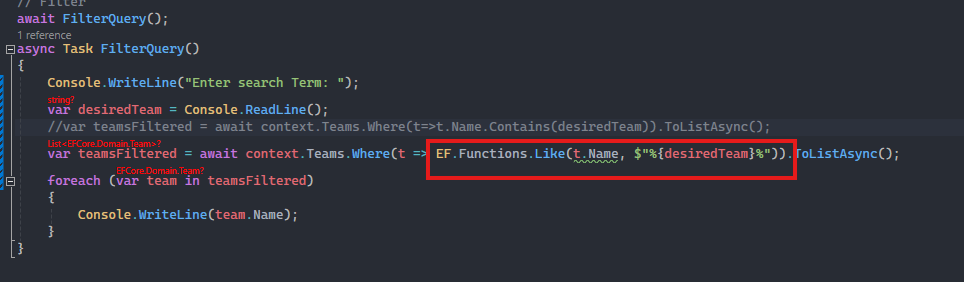
Sử dụng Contains để kiếm một cụm từ không cần nguyên văn

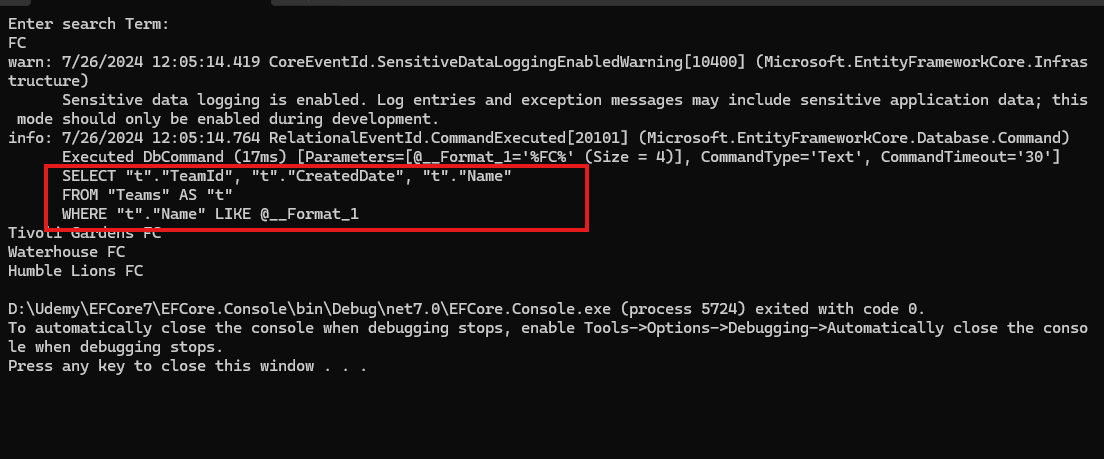
Ví dụ khi search từ “ FC” thì sẽ show ra tất cả các kết quả tên có chứa cụm “FC”





Tìm kiếm dùng hàm có sẵn trong EF, ví dụ như hàm Like





Đồng nghĩa với câu lệnh SQL: select \* from teams where name like ‘%FC%’

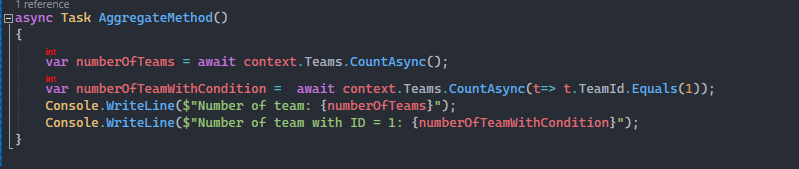
Phương pháp thay thế cú pháp Linq ( nói sau – không nên dùng vì ngược với SQL , gây rối)

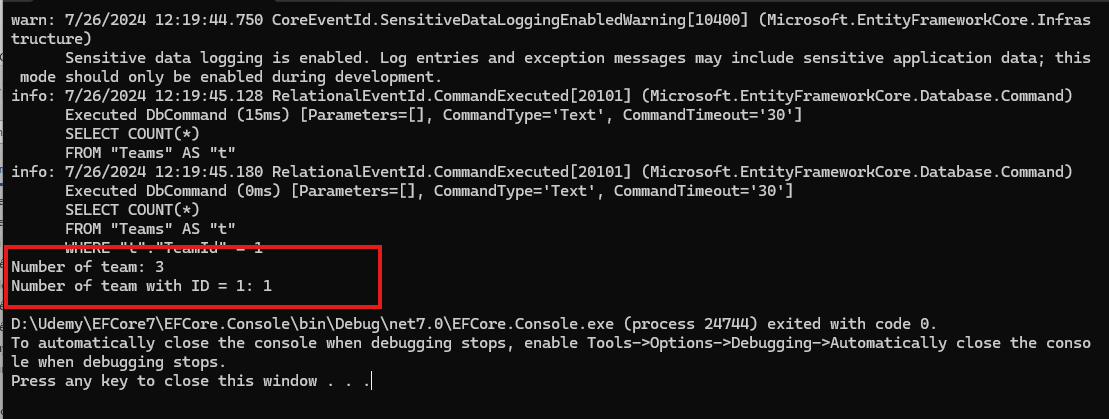
# Một số phương thức tổng hợp dữ liệu

Đếm với hàm CountAsync

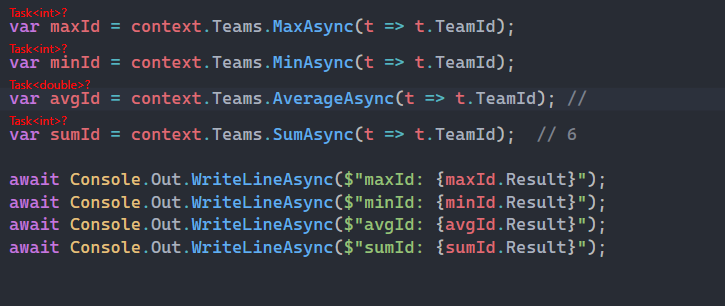
Hàm 1 : đếm tất các Teams

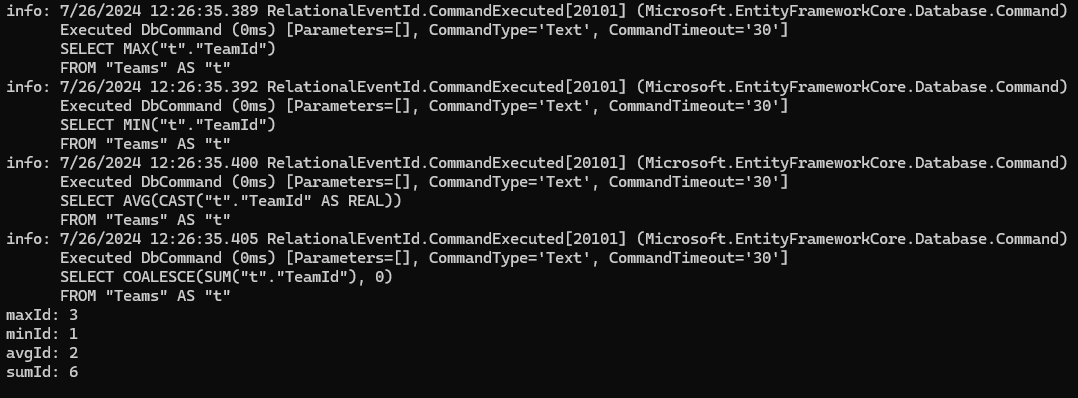
Hàm 2: đếm team có ID = 1





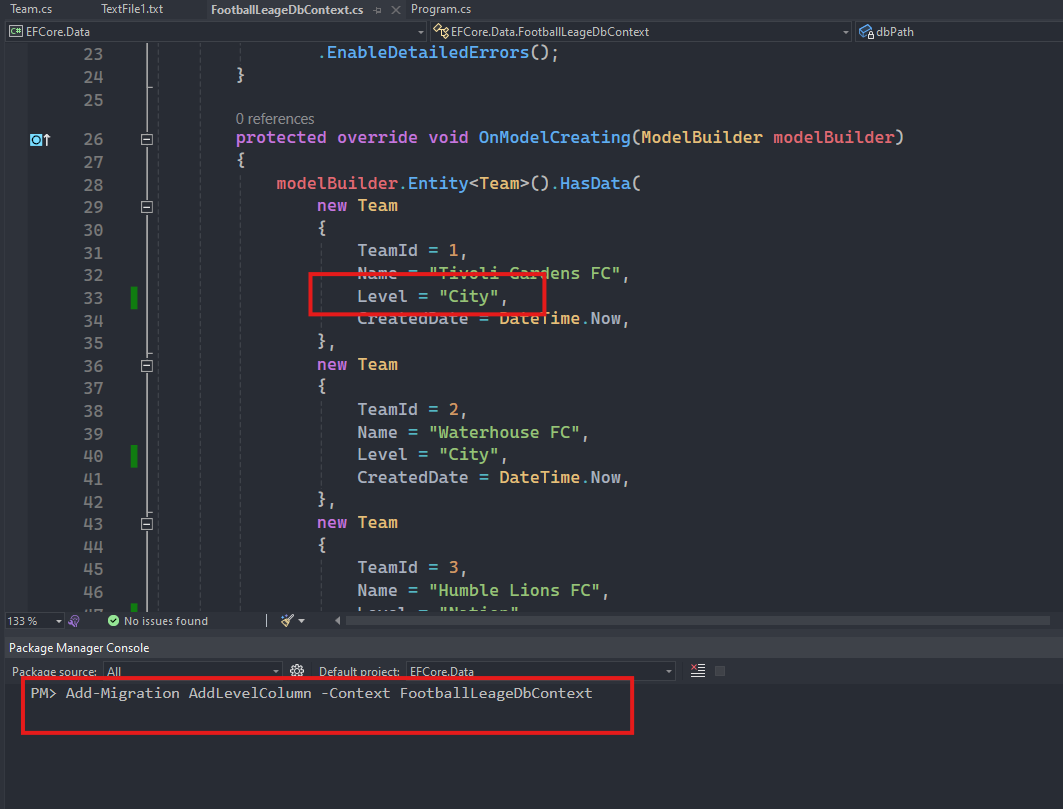
Hàm Max, Min Avg để tìm lớn nhất bé nhất và trung bình

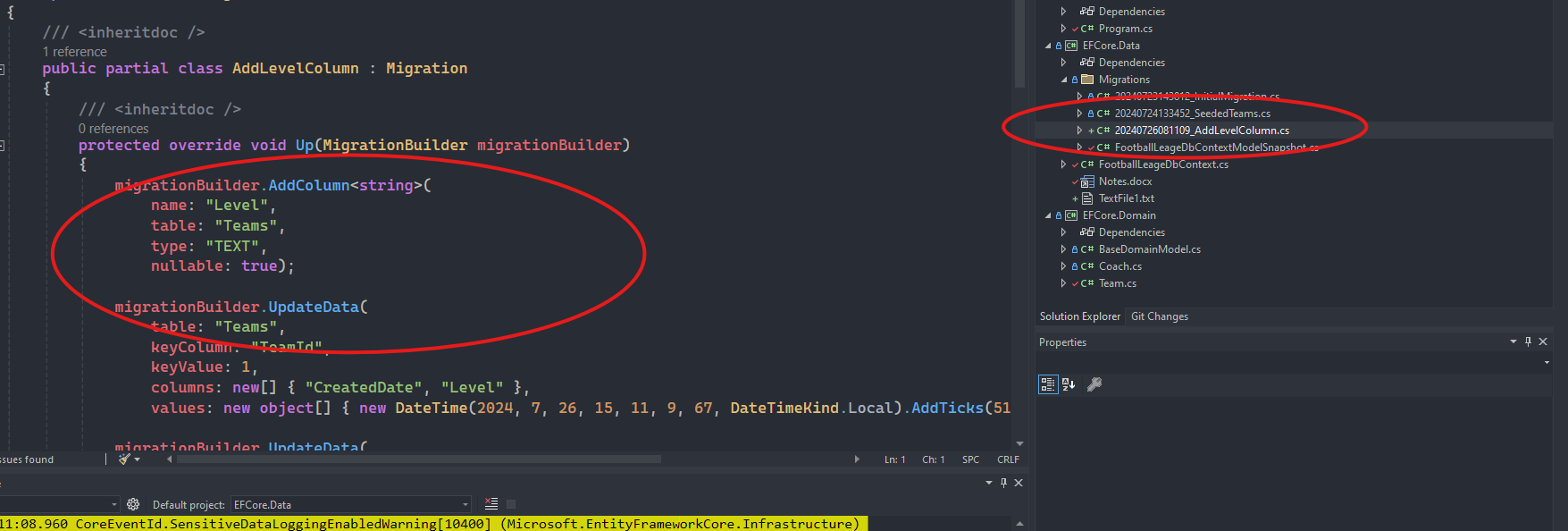


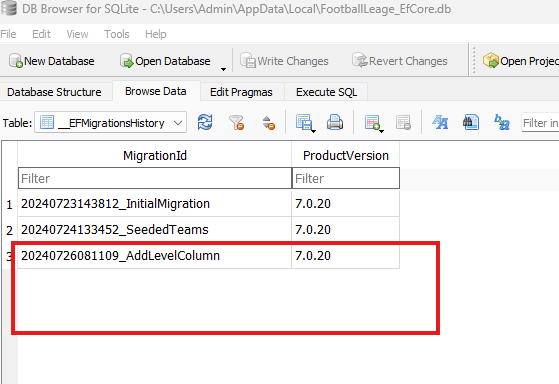


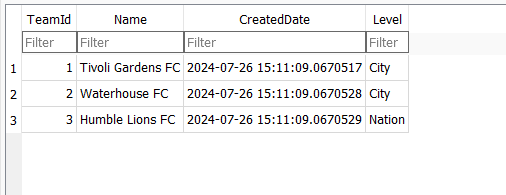
# Nhóm dữ liệu với GroupBy

Để trực quan hơn chúng ta thêm một cấp độ cho đội bóng và sau đó cập nhật database để thêm

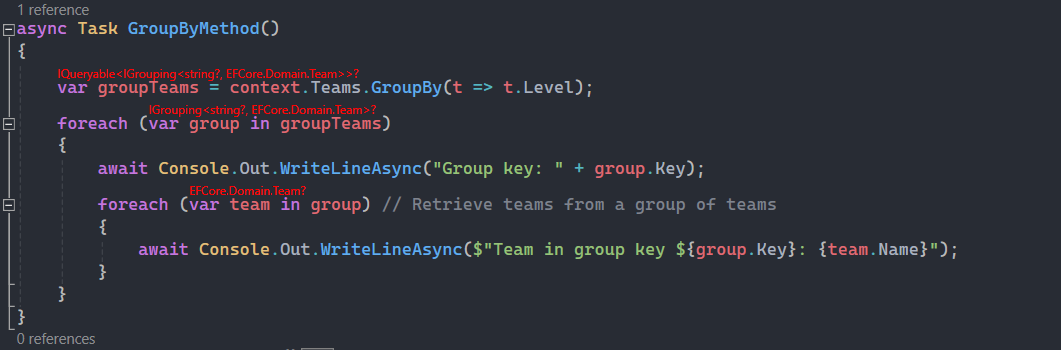




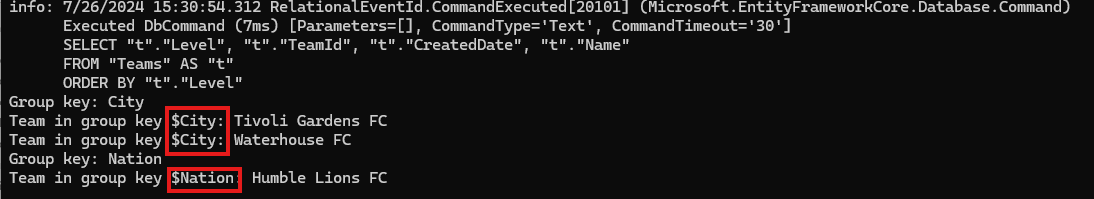




Ở đây có thêm Level cho các đội bóng chúng ta sẽ nhóm theo Level



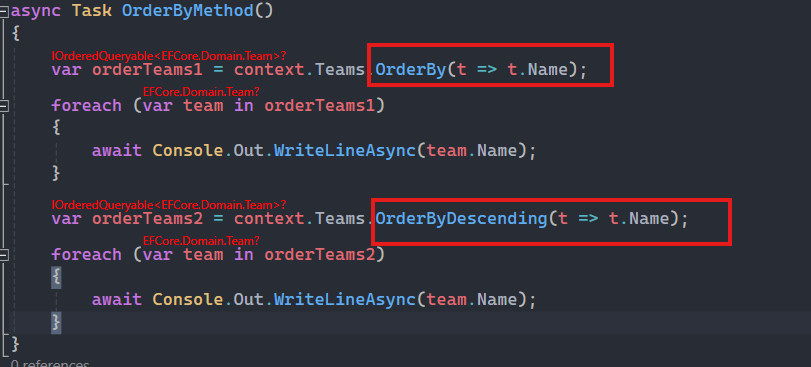
Kết quả sẽ có hai nhóm đội bóng được nhóm theo Level



# Sắp xếp dữ liệu với OrderBy

Một ví dụ với OrderBy (tăng dần)và OrderByDescending(giảm dần)

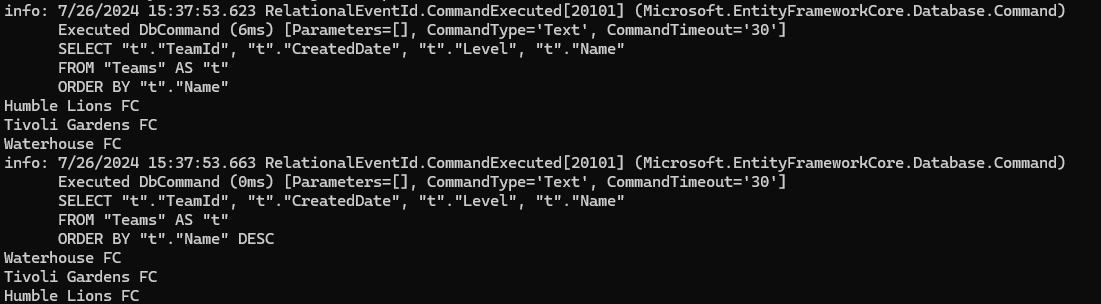
Sắp xếp ở đây áp dụng cho bản chữ cái



Kêt quả ta thấy

Tăng dần bảng chữ cái sẽ là HTW

Giảm dần bảng chữ cái sẽ là WTH



Chúng ta có thể dùng OrderBy để tìm giá trị lớn nhất và bé nhất tương tự như Max Min

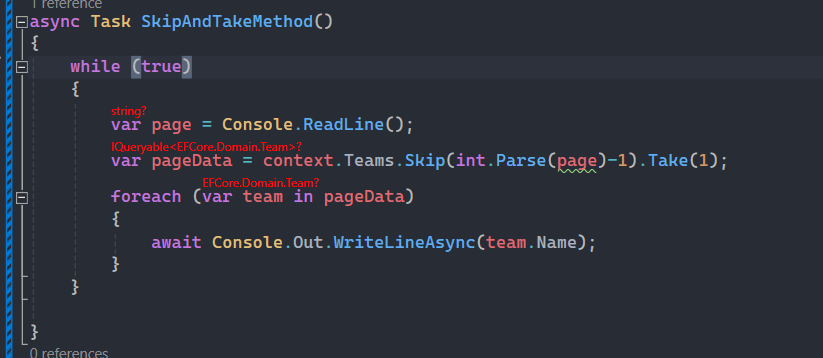
# Chọn lọc dữ liệu với Skip và Take

Skip : bỏ qua số phần tử

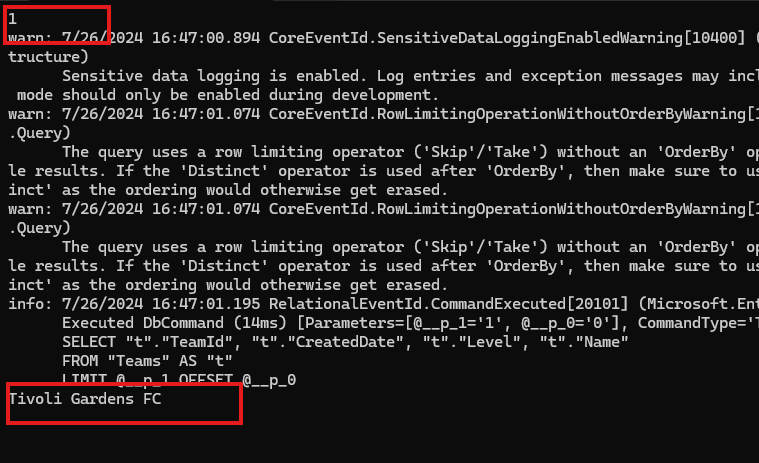
Take : lấy bao nhiêu phần tử

Ví dụ đoạn chuongw trình sau

Đoạn chương trình nhập vào số trang , mỗi trang là một đội bóng



Kêt quả



Nhập vào 1 sẽ hiện đội bóng thứ nhất

Quay trở lại với phương thức lấy tất cả tên đội ở phần trước. Với cách này

