Milan Jovanović • Following
Practical .NET and Software Architecture Tips | Microsoft MVP
6h •

5 EF Core features you need to know

1. Query splitting

- Splits JOINs into multiple queries

- Could improve query performance

2. Bulk updates

- Uses ExecuteUpdate to update many entities

- Directly goes to DB, isn't part of a unit of work

3. Raw SQL queries

- Load entities from DB using raw SQL

- Allows you to use SQL features

4. Bulk deletes

- Uses ExecuteUpdate to delete many entities

- Directly goes to DB, isn't part of the unit of work

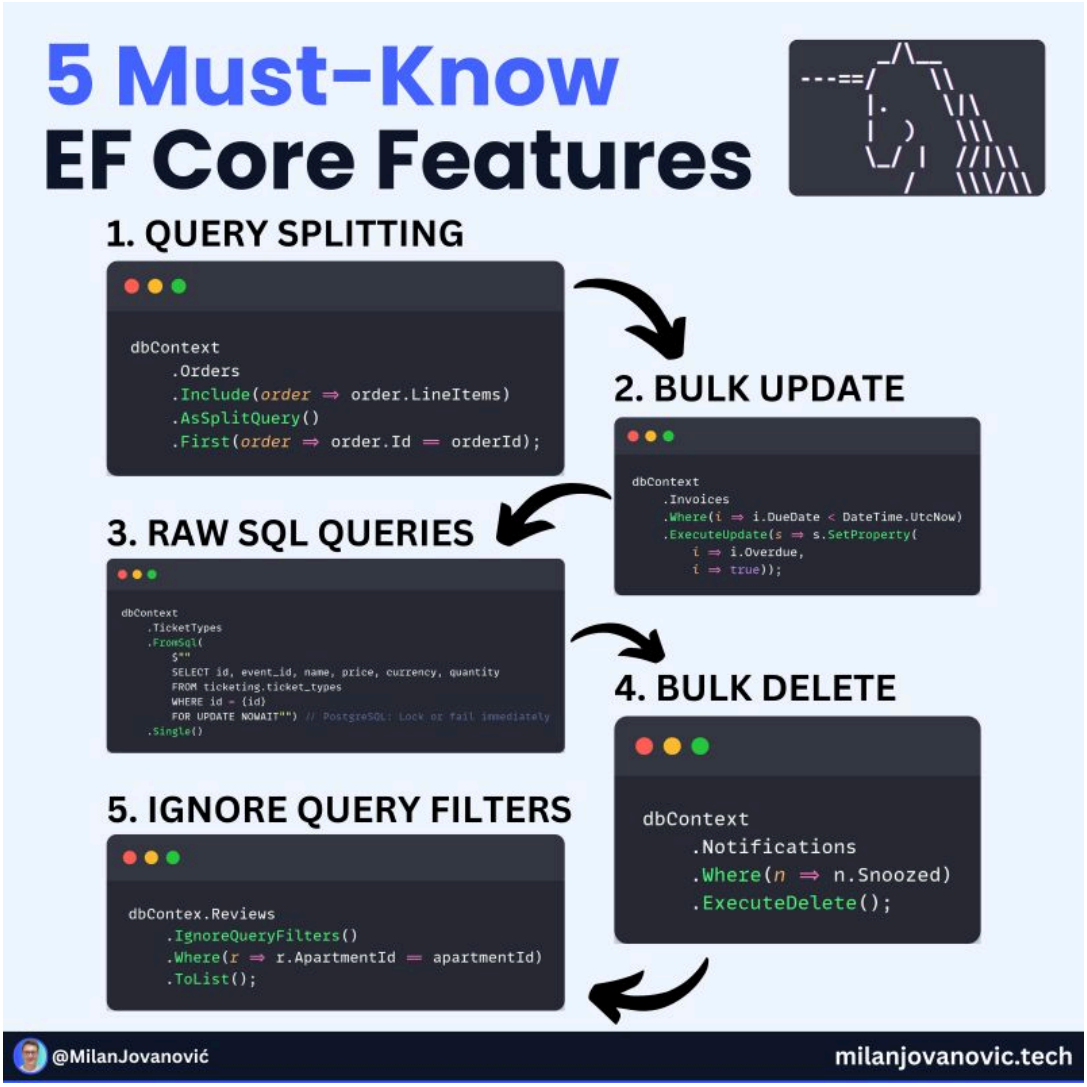
5. Ignoring query filters

- Query filter allows you to automatically filter LINQ queries

- Ignoring the query filter can be useful sometimes

Bonus - Here are five more EF Core features you should know: https://lnkd.in/eijGi98W

What is your favorite thing about EF Core?



SERKUT YILDIRIM and 301 others 22 comments 33 reposts

Like Comment Repost Send

Most relevant

Tatenda P. Nyamuda • 3rd+
.NET Developer | C# | Vue.js
5h

Just used AsSplitQuery a few minutes ago in my project. I had three .Include() statements in my query. I was getting a warning from EF Core and the response from the query was slow. After doing some digging, that's when I realized I should be using query splitting in this case.

So from what I understand, query splitting allows EF Core to load each related entity using separate SQL queries instead of trying to join everything into one big query, which is what could cause a decrease in performance.

I haven't used the other 4 features yet, but I'm interested in Bulk Update and Raw Queries.

Like | Reply 1 reply

Micael Uthas • 3rd+
Freelance SQL Server Specialist | 20+ Years in Jeeves ERP Adaptations, SQL Dev. With Extensive En...
1h

Tatenda P. Nyamuda This is one thing I use as an argument for not using ORM's.

Transfer more data then needed to gain performance(you must recognise how stupid that sounds).

Investigate the original query and determine why it performs bad before you move optimization stages to backend!

Databasengines was built for this kind of tasks.

Like | Reply

Habeeb Yakubu • 2nd
Microsoft MVP | Architecting Enterprise Solutions on Azure, Power Platform & D365 | Senior Consultant ...
6h

Favorite thing about EF Core: I can still write raw SQL when needed. Both FromSqlRaw and FromSqlInterpolated return IQueryable, which I can add a LINQ to. It also keeps my SQL sharp. No one should ever forget how to write some raw queries.

Like 1 | Reply 4 replies

Milan Jovanović Author
Practical .NET and Software Architecture Tips | Microsoft MVP
6h

What I don't like there is that they do a nested query in that case, instead of somehow modifying the original one. This could lead to some strange query plans.

Like 1 | Reply

Habeeb Yakubu • 2nd
Microsoft MVP | Architecting Enterprise Solutions on Azure, Power Platform & D365 | Senior C...
6h

Milan Jovanović Valid, something to consider while using.