There is already an object named AspNetRoles in the database. (entity-framework-core)

▲

3

▼

★

I have an asp.net core mvc website using entity framework core.

I don't know how it initially happened, but I can't get past the error: "There is already an object named AspNetRoles in the database"

I ended up deleted the database, deleting all my migration .cs files and starting from scratch.

I've then tried `add-migration MyInitialMigration`

same error

I then deleted the database again, and tried a direct `update-database` (without adding a migration)

still the same error

I tried changing the appsettings config string to point to a new, non-existing database.

Still the same error!

Previously on MVC5/EF6 I've encountered this error and have resolved it using the same method as this answer: [There is already an object named in the database](#)

```
 -Add-Migration Initial -IgnoreChanges
```

However -IgnoreChanges is unsupported in EFCore

Or previously on EF6 I've resolved it by adding a migration, then deleting everything in the Up/Down and running an empty migration. This has not worked on EFCore

How can I continue? I've checked the migration table and they're always empty even though it's created the whole database each time. Just doesn't seemed to have logged it in the migrations table

Join Stack Overflow to learn, share knowledge, and build your career.

| Sign up with email | G Sign up with Google | Sign up with Facebook ✕ |

The solution to this problem is to apply the command drop-database. After that remove-migration and then you can start again with add-migration and update-database. It works for me all the time – Marlon Diaz La Hoz Apr 5 '18 at 3:50

2 Answers

7

Found my issue.

I had followed this tutorial to load appsettings from EntityFramework:
https://docs.asp.net/en/latest/fundamentals/configuration.html#example-entity-framework-settings

In the EntityFrameworkConfigurationProvider there is a `public override void Load()` which has a `dbContext.Database.EnsureCreated();` and a `CreateAndSaveDefaultValues`

Between the two of those it looks like it was initializing the database without tracking the migration.

I commented those out, ran a migration and it worked fine. Hopefully I don't have to comment them out every time I want to run a migration :/

answered Sep 21 '16 at 10:00

mejobloggs
3,205   5   21   34

1

I've just had the same issue, and seems like Database.EnsureCreated() doesn't use migrations, you can use Database.Migrate() instead which is what I've just tried and seems to work, but time will tell when I try another migration.

answered Nov 24 '16 at 16:14

RichardB
65   1   1   9

Join Stack Overflow to learn, share knowledge, and build your career.

| Sign up with email | G Sign up with Google | Sign up with Facebook | ✕ |