How do you seed a database with Prisma?

Written by Seth Corker on Benevolent Bytes

Prisma is great ORM to use but sometimes the documentation can be confusing and/or lacking. One area that I found confusing was seeding a DB with Prisma, luckily it's not difficult. It's just buried in the docs. Here's a few tips on how you can write a seed script to populate your database when using Prisma.

I've made an example repo to go along with this explanation.

A Coffee scenario

To make this a little more concrete, let's imagine a scenario where we are creating a web app to manage coffee orders. As a starting point, we want a list of possible coffees to order and we don't want to have to manually create them each time our DB needs resetting. This is the perfect scenario for seeding the database with coffee data on setup.

The schema

Here's our Prisma schema for the coffee ordering app, it's just got the initial setup and a single Coffee model with a name and id.

```
generator client {
  provider = "prisma-client-js"
}

datasource db {
  provider = "sqlite"
  url      = env("DATABASE_URL")
}

model Coffee {
  id   Int    @id @default(autoincrement())
  name String
}
```

The seed script

The key things to in your seed script are:

1. Import the Prisma client
2. Create the records you want to seed in your DB using the client
3. Await the creation of these records (otherwise node will terminate too soon)

Here's the seed script stored in `/prisma/seed.ts` which will create the coffee records.

```
import { PrismaClient } from '@prisma/client'

const prisma = new PrismaClient()

const COFFEE_NAMES = [
  "Black Coffee",
  "Turkish Coffee",
  "Cold Brew",
  "Iced Coffee",
  "Ristretto",
  "Doppio",
  "Americano",
  "Lungo",
  "Caffé Crema",
  "Café Zorro",
  "Café Cubano",
  "Cappuccino",
  "Latte",
  "Piccolo Latte",
  "Mocha",
  "Macchiato",
  "Cortado",
  "Café Bombon",
  "Cafe Con Leche",
  "Carajillo",
  "Espresso Romano",
  "Espressino",
  "Flat White",
  "Café Au Lait",
  "Caffe Breve",
  "Antoccino",
  "Cafe Affogato",
  "Red Eye",
  "Black Eye",
  "Dripped Eye",
  "Lazy Eye",
  "Vienna",
  "Café Borgia",
  "Ca Phe Sua Da",
  "Galao",
  "Frappe",
  "Mazagran",
  "Irish Coffee",
]

/**
 * For each coffee name, create a Coffee record in the DB
 */
function seedCoffee() {
  Promise.all(COFFEE_NAMES.map(n => prisma.coffee.create({ data: { name: n } })))
    .then(() => console.info('[SEED] Succussfully create coffee records'))
    .catch(e => console.error('[SEED] Failed to create coffee records', e))
}

seedCoffee();
```

So now that we have this, how do we make sure it runs and works with the Prisma CLI?

Using ts-node to run our script

Update the package.json

To tell Prisma that we have a seed, we need to add a property to our package.json:

```
"prisma": {
  "seed": "ts-node prisma/seed.ts"
}
```

This lets us run `prisma db seed` to seed the DB. You'll *also* need to have ts-node added as a dev dependency and we also need some additional configuration to support ES6 imports.

Update tsconfig.json

We need to tell ts-node to compile modules so we can run them (this may not be necessary in future once ES modules are supported fully by node).

Add the rolling to the `tsconfig.json` file:

```
"ts-node": {
    "compilerOptions": {
      "module": "CommonJS"
    }
}
```

Now you should be able to setup the DB `yarn prisma db push` and seed `npx prisma db seed`.

Resources

- Prisma documentation on Seeding your database
- Example repo using Prisma and Next.js