Prisma

Prisma is an open-source Node.js ORM that provides support for PostgreSQL, MySQL, SQL Server, SQLite, MongoDB, and CockroachDB

Prisma Schema:

Prisma relies on the Prisma schema, i.e. schema.prisma file. In this file we specify our data sources, generators, and data model definition (the app’s models and their relations).

It mainly consists of three tools:

Prisma Client**:** lightweight database client for Node.js and Typescript

Prisma Migrate: migration system that generates migration files automatically, generate SQL migrations from the Prisma schema and executes them against the databases

Prisma Studio: A GUI to view and edit data in your database.

Strengths:

* Good CLI, capable of generating migrations and boilerplate files
* Excellent documentation
* Easy to import in existing project
* Makes it easy for developers by providing a GraphQL API for submitting database queries.
* Has its own server – Prisma Server, all request is made to Prisma Server for carrying out any operations on database so Prisma Server makes request from database and gives your data back.
* Support for RAW queries

Weakness:

* Can be intimidating to beginners

### Sequelize

Sequelize is an open-source Node.js ORM that provides support for Postgres, MySQL, MariaDB, SQLite, SQL Server, and more. Also supports migrations and has a CLI tool for generating and seeding data.

Strengths:

* Strong CLI, capable of generating migrations, helping with seeds and boilerplate files
* Support for RAW queries

Weakness:

* Lack of high quality documentation

**Why Prisma over Sequelize?**

Sequelize can work only with relational databases but Prisma can also work with non-relational databases.

Speed of development faster in Prisma

Auto Migration (Automatically create Migration files) possible in Prisma but not in sequelize

Prisma Studio for viewing and editing your database

Prisma has Excellent documentation compared to Sequelize

Prisma mitigates many problems of traditional ORMs, such as bloated model instances, mixing business with storage logic, lack of type-safety or unpredictable queries caused e.g. by lazy loading.

Prisma has strong Typescript support while sequelize has strong JavaScript support instead of Typescript

Prisma encounters lesser errors while working with Typescript as compared to sequelize.

Prisma makes it easy for developers by providing a GraphQL API for submitting database queries.

Prisma Has its own server – Prisma Server, all request is made to Prisma Server for carrying out any operations on database so Prisma Server makes request from database and gives your data back.