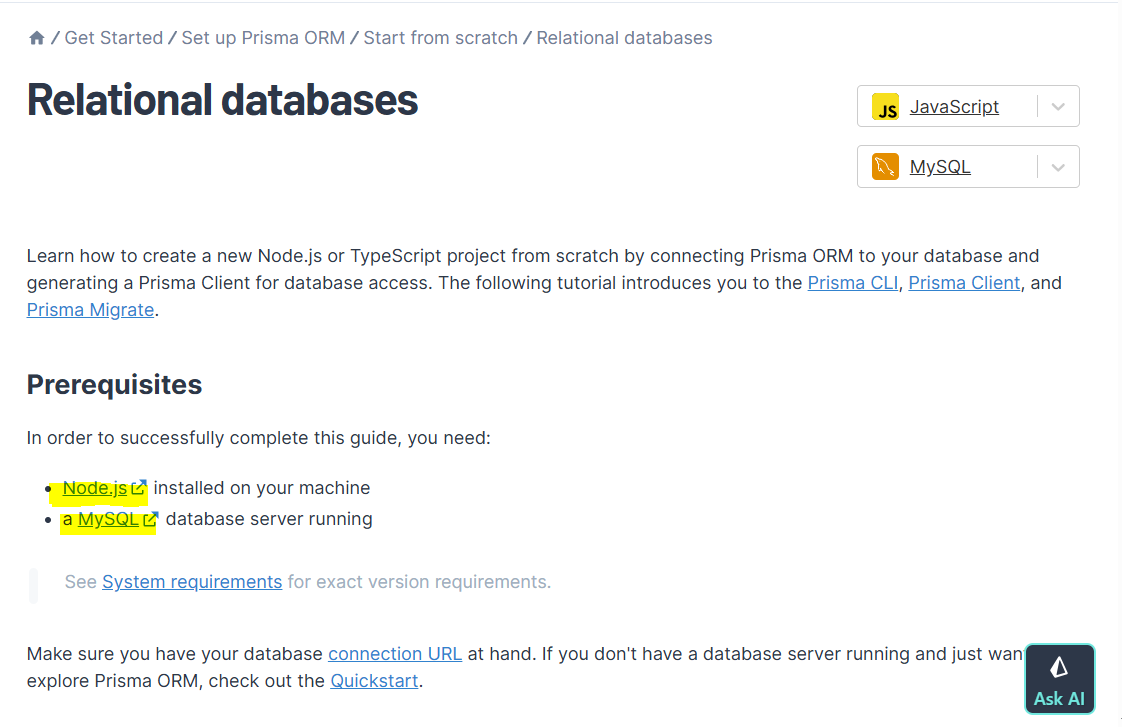
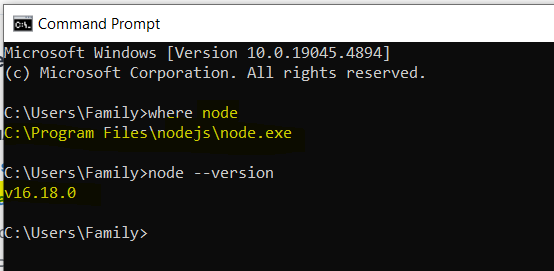
**Prisma**

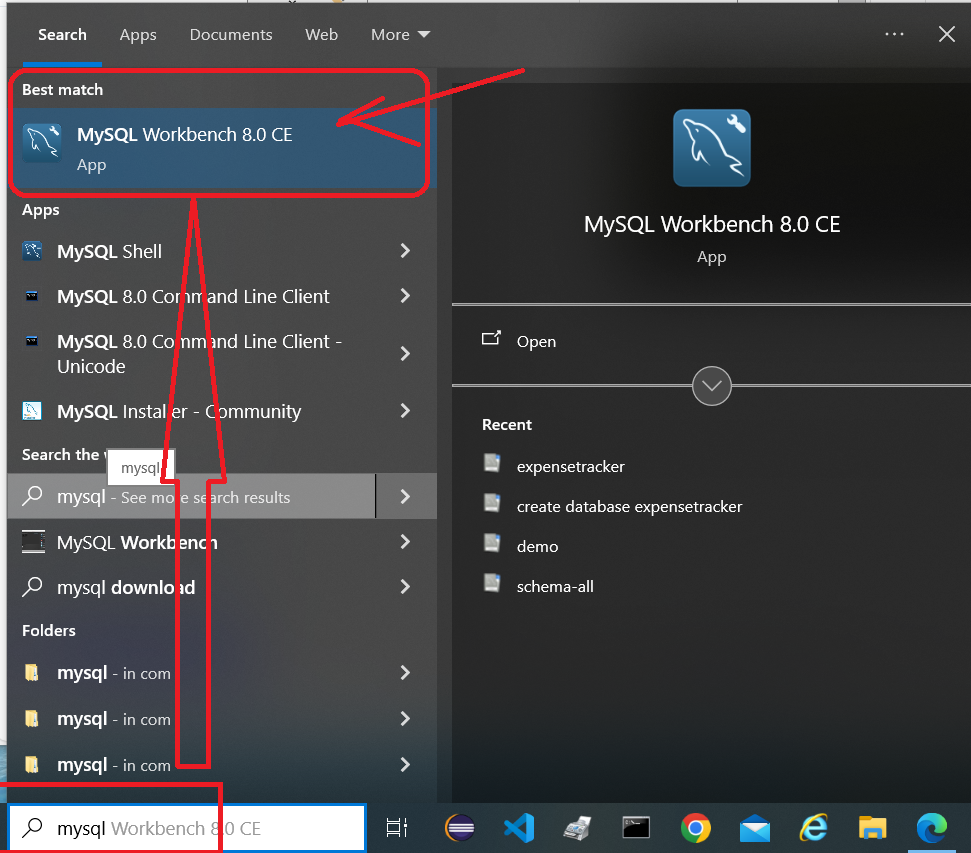
Start from scratch with Prisma ORM using JavaScript and MySQL (15 min) | Prisma Documentation 🡪 <https://www.prisma.io/docs/getting-started/setup-prisma/start-from-scratch/relational-databases-node-mysql>

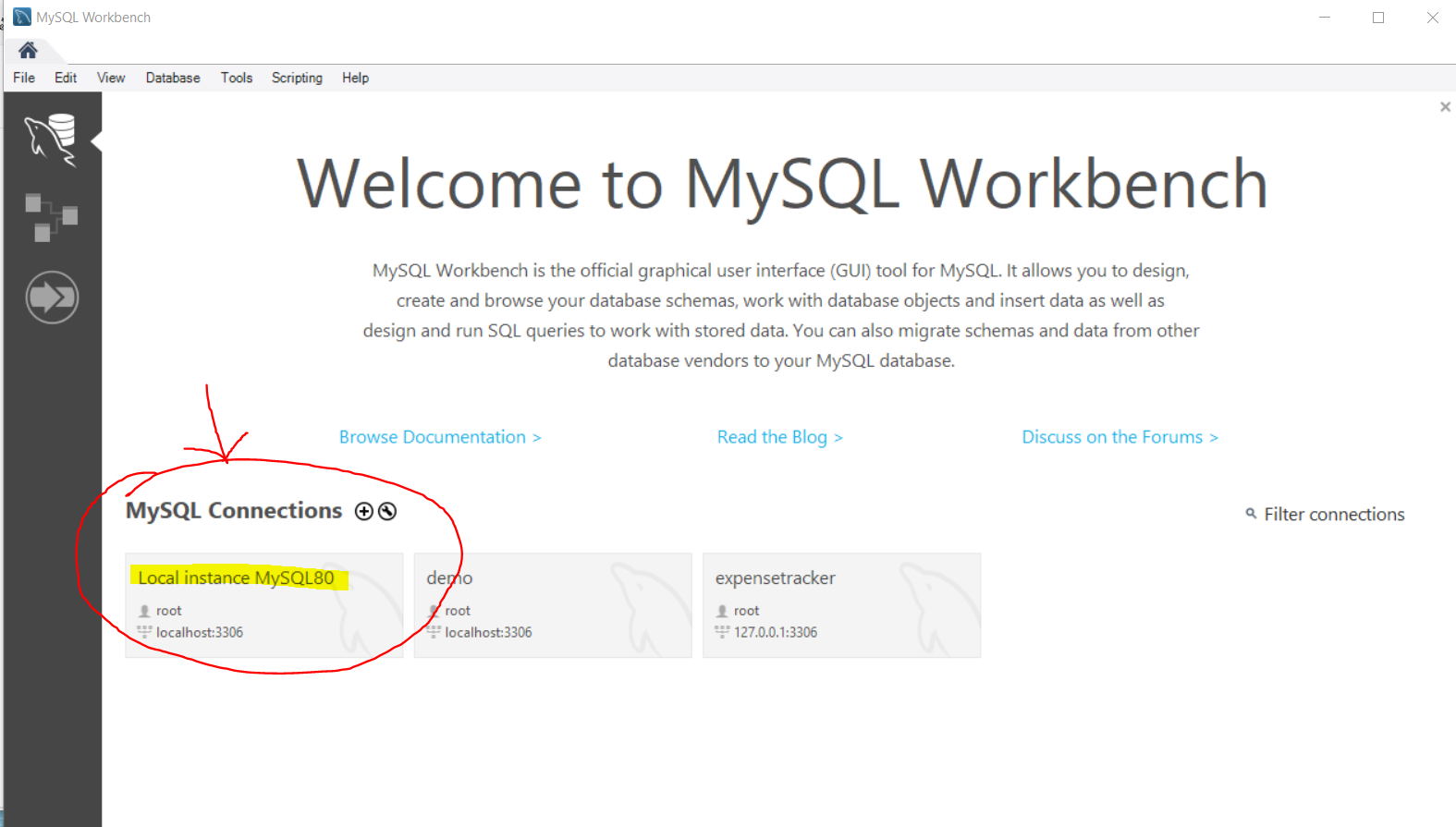


Node.js

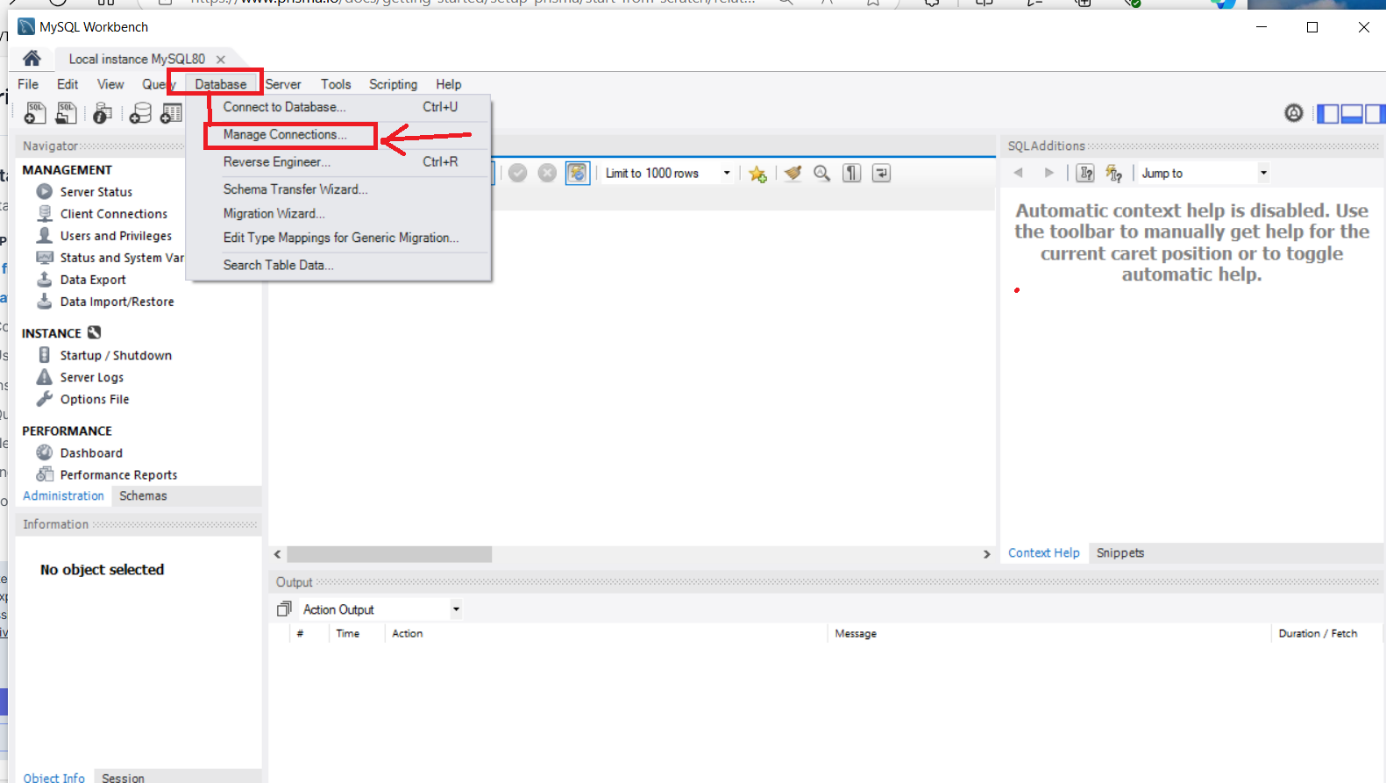


Window 10 🡪 MySQL

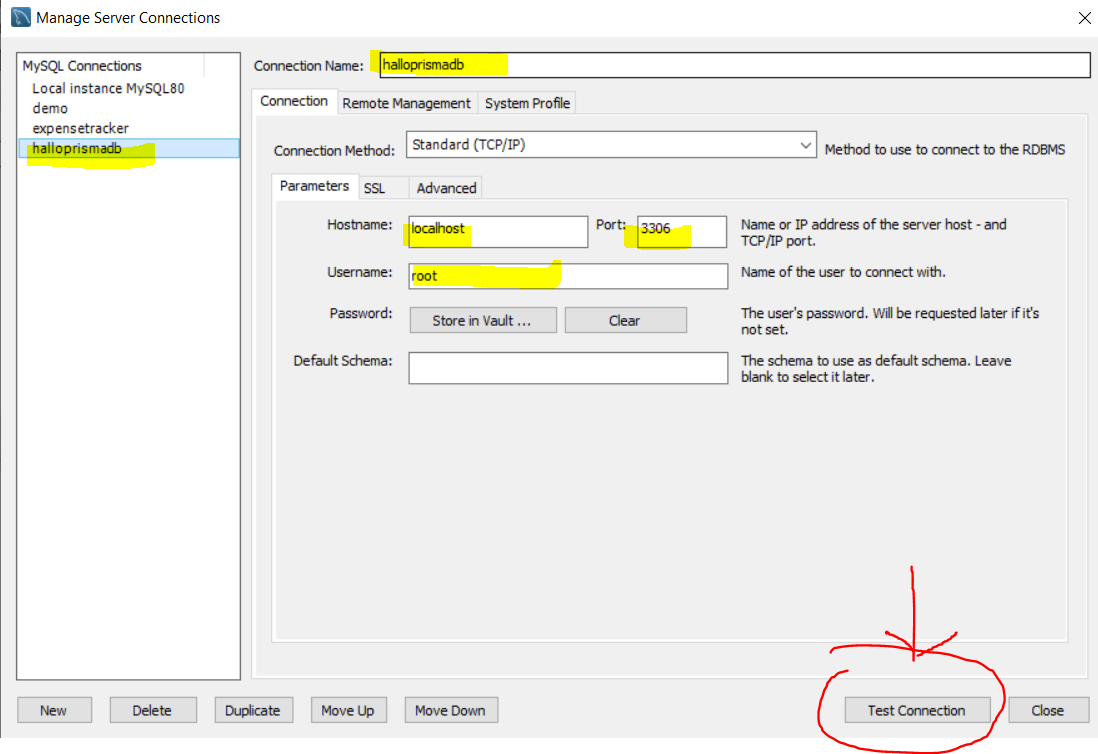




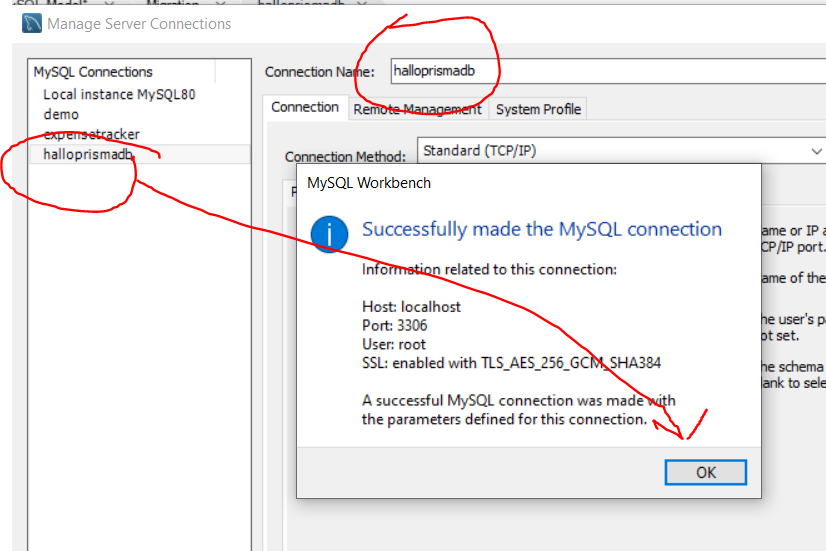
Manage Connections…

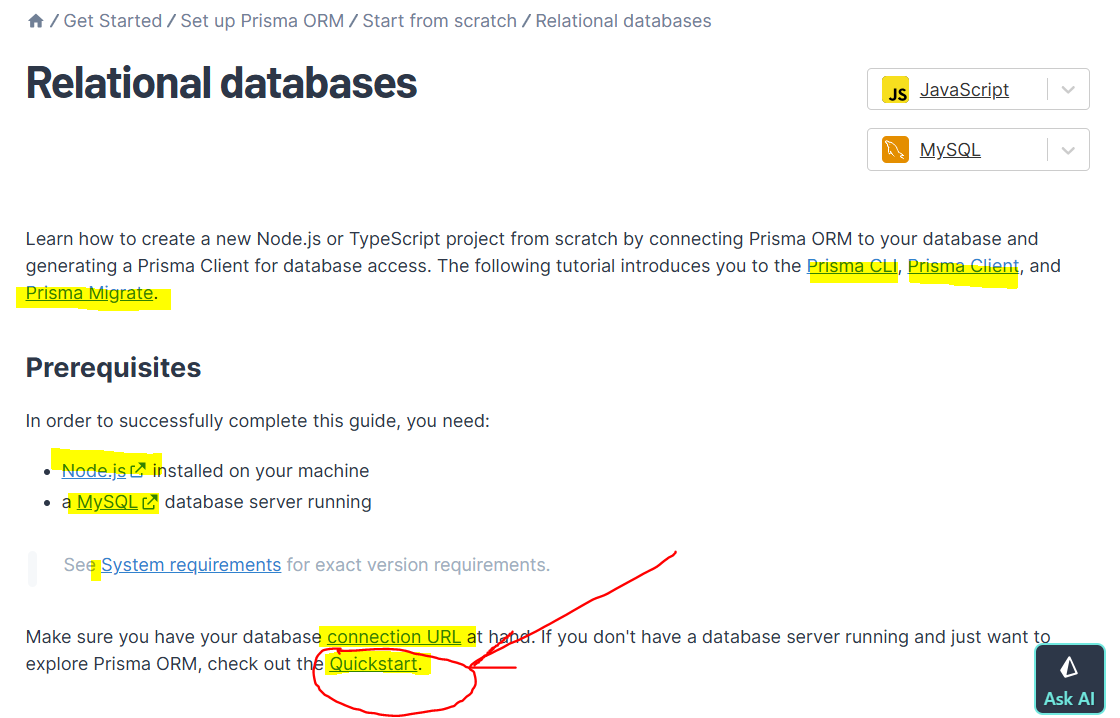


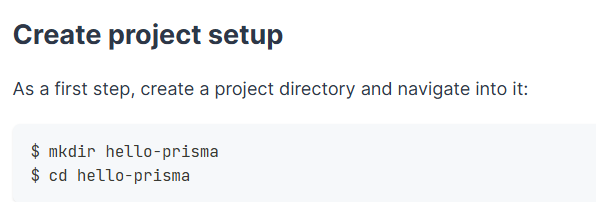
Click New 🡪 Connection Nam=PrimaDBDemo 🡪 Click Test Connection



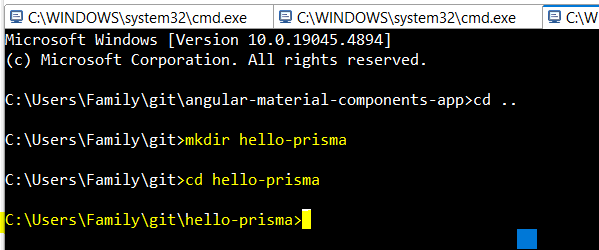
Password: root 🡪 Click OK

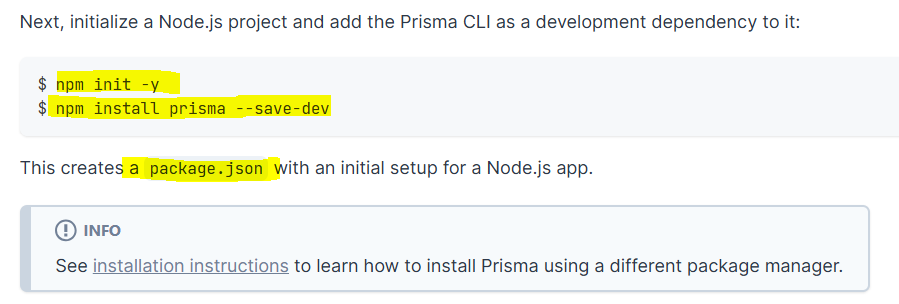






mkdir hello-prisma  
cd hello-prisma

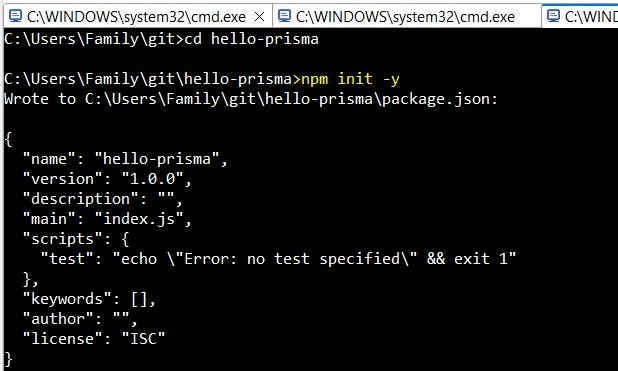


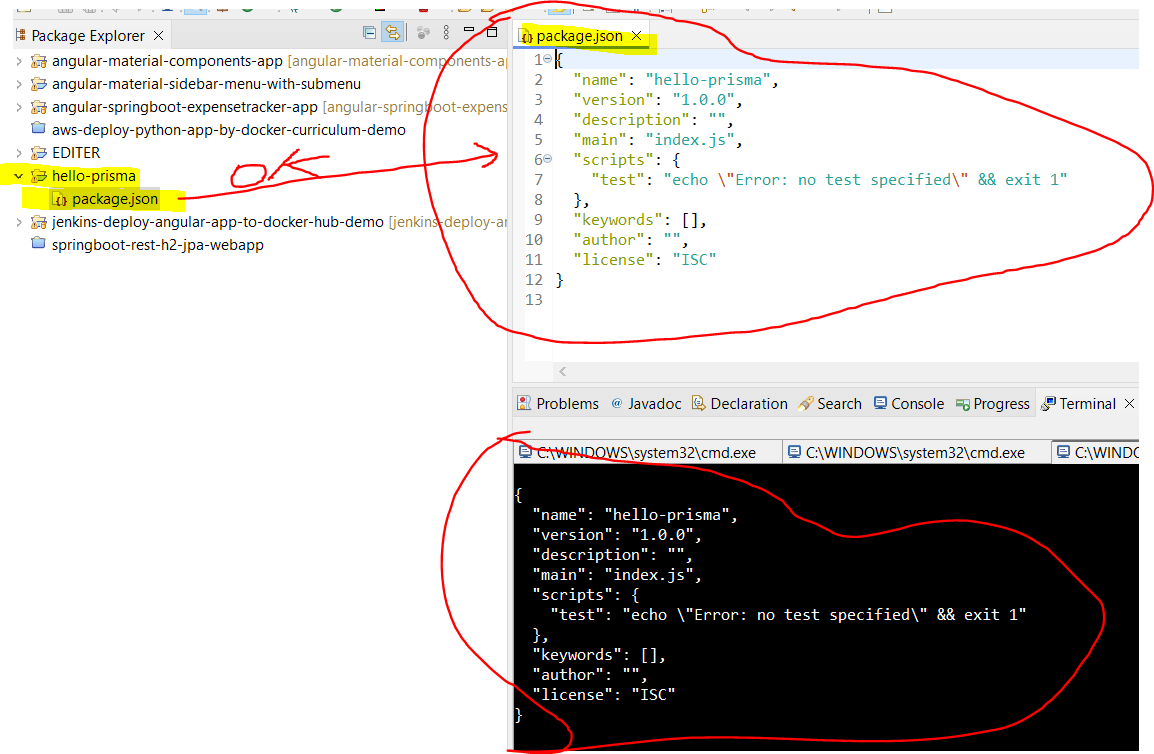


npm init -y

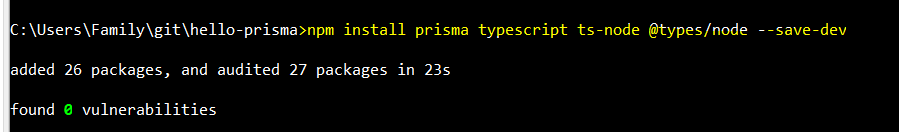
npm install prisma typescript ts-node @types/node --save-dev

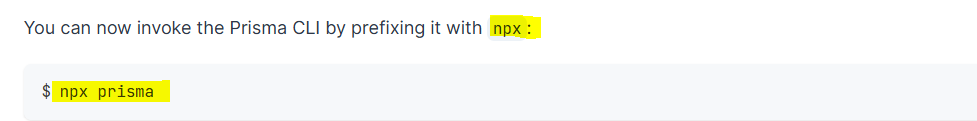
C:\Users\Family\git\hello-prisma>npm init -y



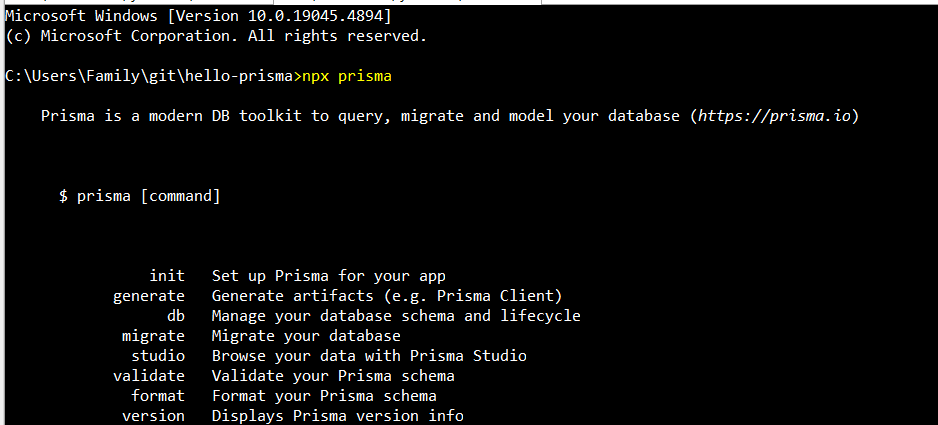


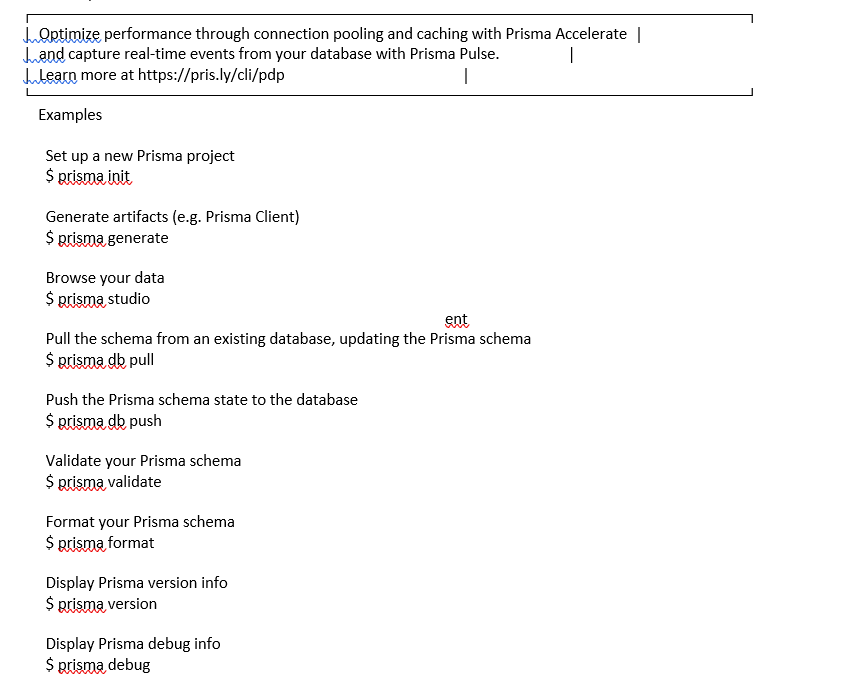
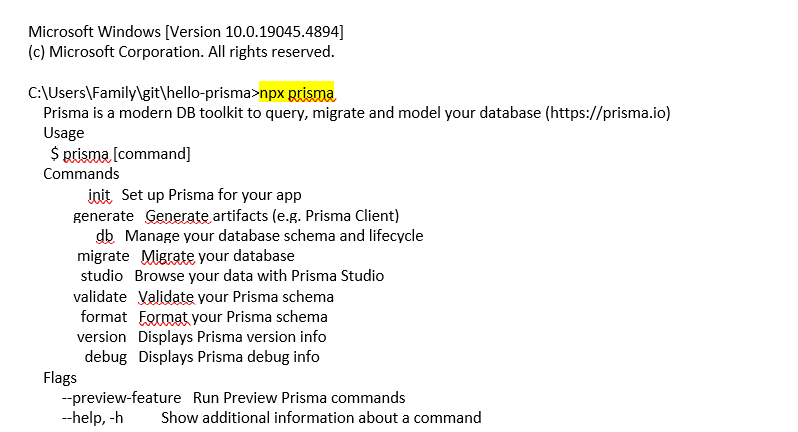
C:\Users\Family\git\hello-prisma>npm install prisma typescript ts-node @types/node --save-dev





C:\Users\Family\git\hello-prisma>npx prisma





Push the Prisma schema state to the database

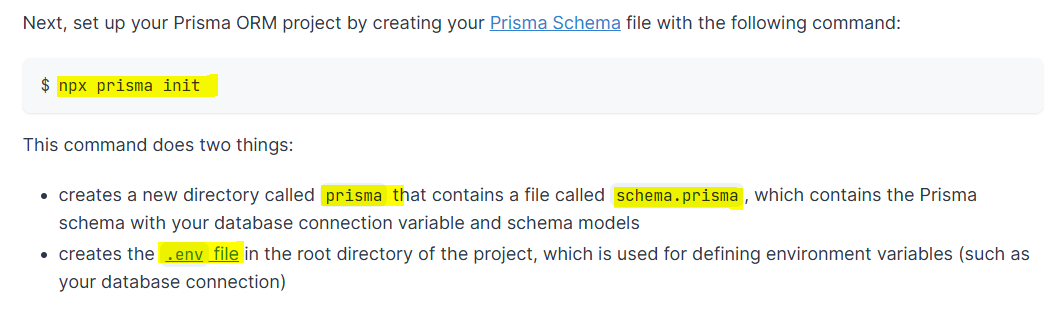
$ prisma db push

Validate your Prisma schema

$ prisma validate

Format your Prisma schema

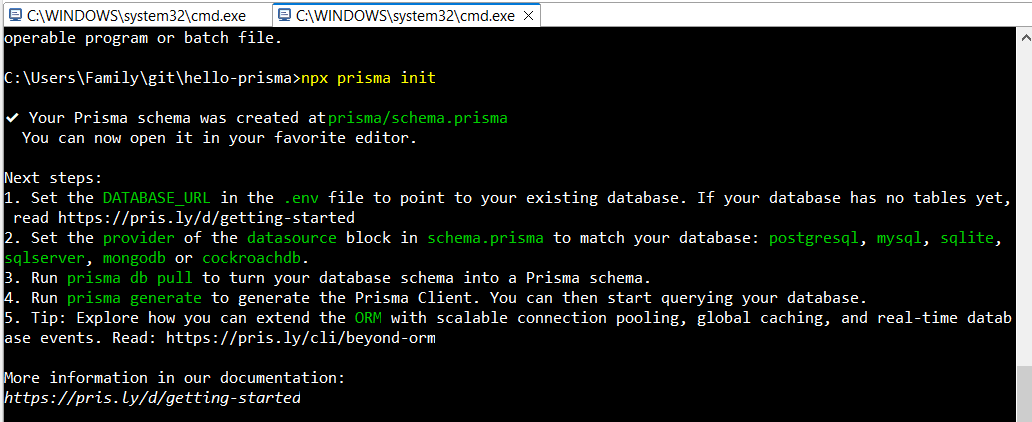
$ prisma format



This command does two things:

- creates a new directory called prisma that contains a file called schema.prisma, which contains the Prisma schema with your database connection variable and schema models

- creates the [.env file](https://www.prisma.io/docs/orm/more/development-environment/environment-variables/env-files) in the root directory of the project, which is used for defining environment variables (such as your database connection)



C:\Users\Family\git\hello-prisma>npx prisma init

✔ Your Prisma schema was created at prisma/schema.prisma

You can now open it in your favorite editor.

Next steps:

1. Set the DATABASE\_URL in the .env file to point to your existing database. If your database has no tables yet,

read <https://pris.ly/d/getting-started>

2. Set the provider of the datasource block in schema.prisma to match your database: postgresql, mysql, sqlite,

sqlserver, mongodb or cockroachdb.

3. Run prisma db pull to turn your database schema into a Prisma schema.

4. Run prisma generate to generate the Prisma Client. You can then start querying your database.

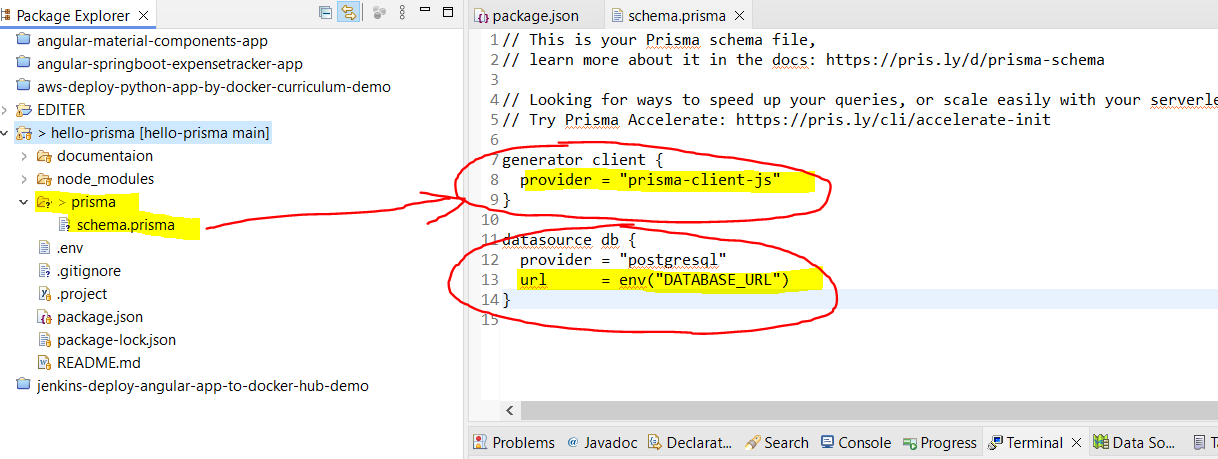
5. Tip: Explore how you can extend the ORM with scalable connection pooling, global caching, and real-time datab ase events. Read: <https://pris.ly/cli/beyond-orm>

More information in our documentation:

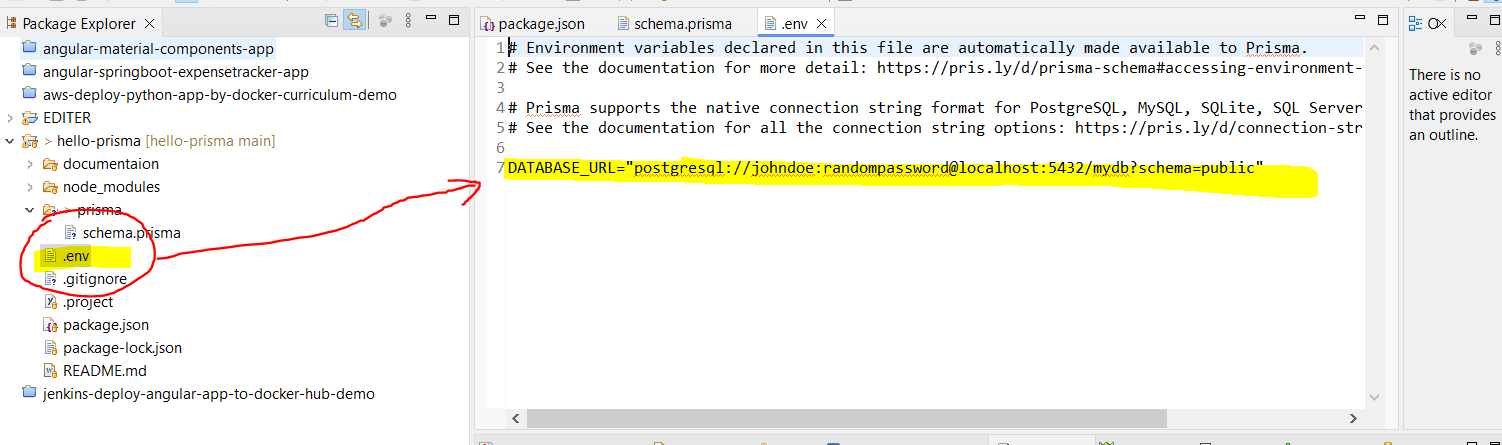
<https://pris.ly/d/getting-started>

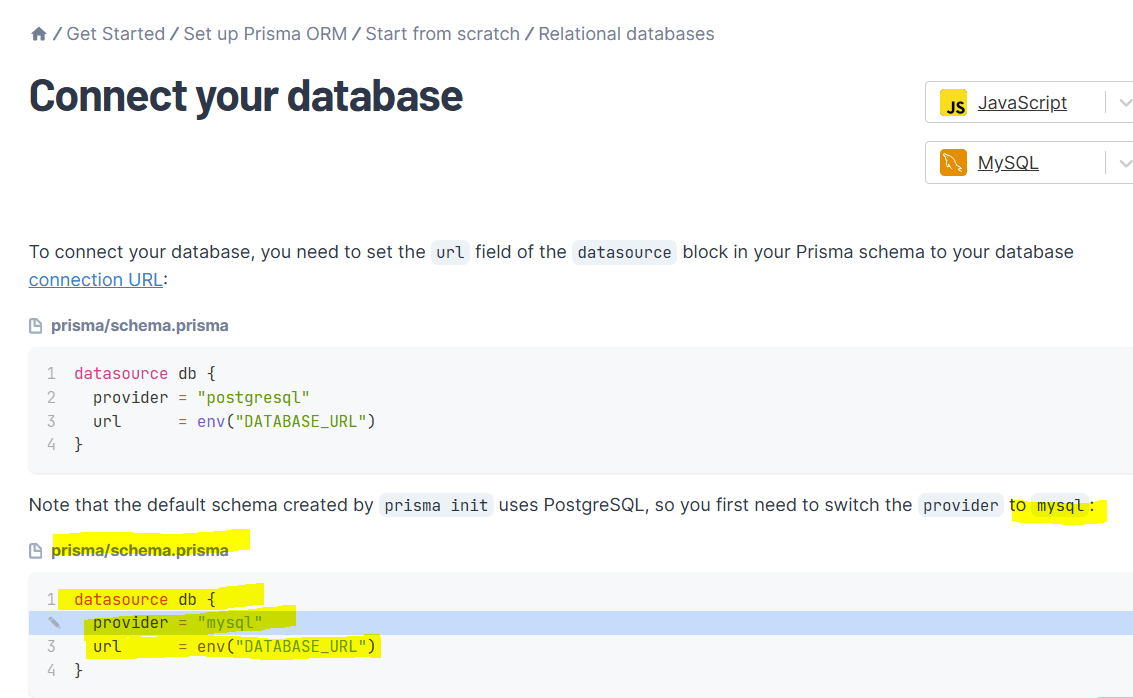
FOLOWING TWO FILES WERE GENERATED.

schema.prisma



[.env file](https://www.prisma.io/docs/orm/more/development-environment/environment-variables/env-files)

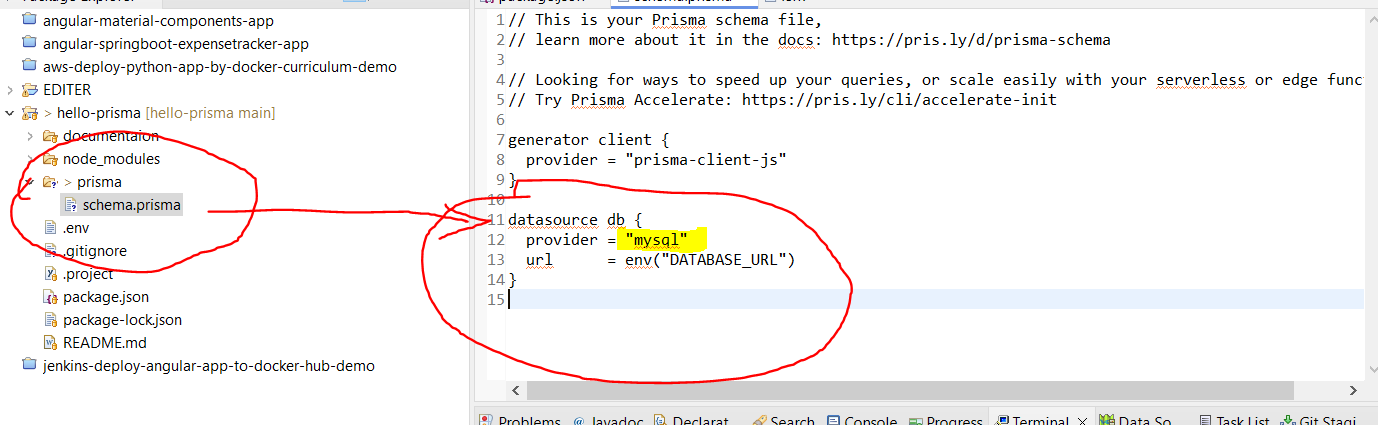




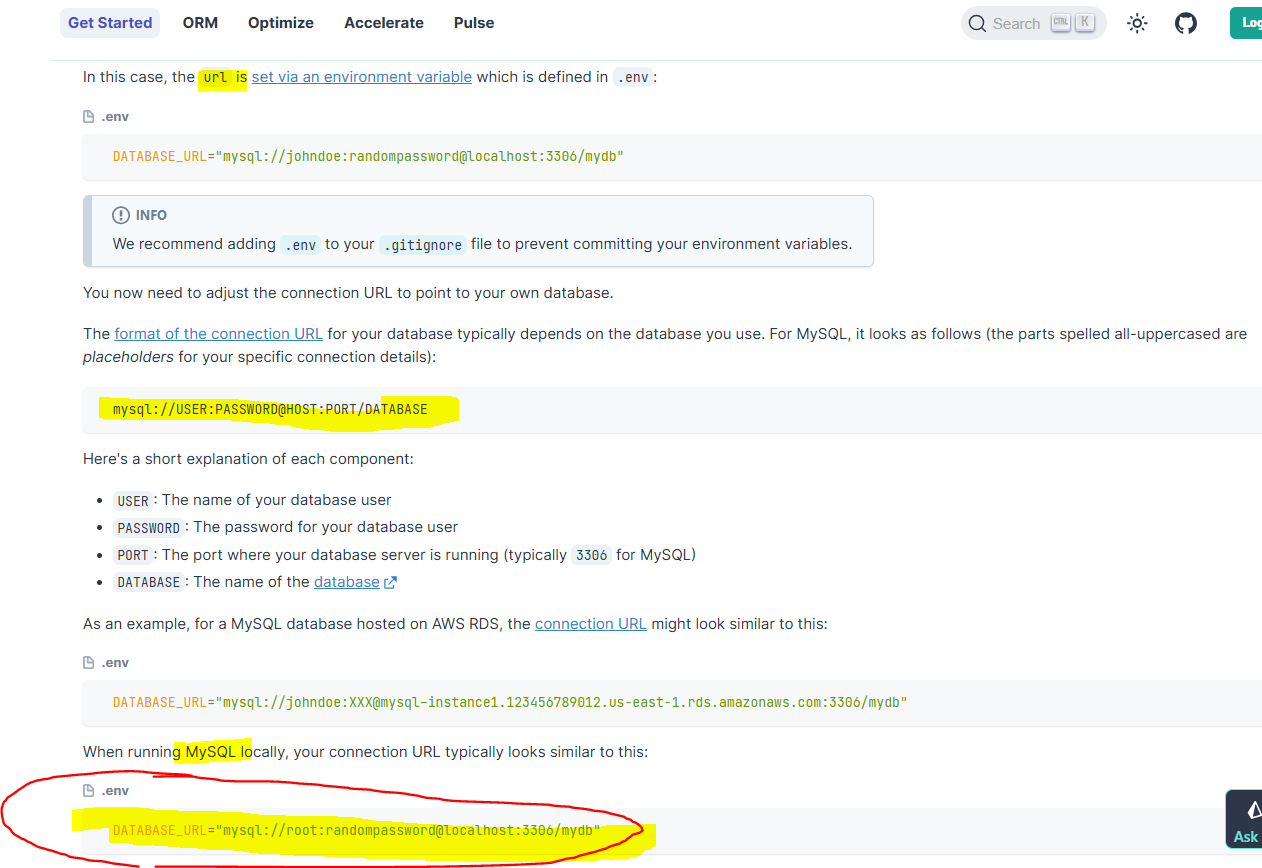
Change provider to mysql

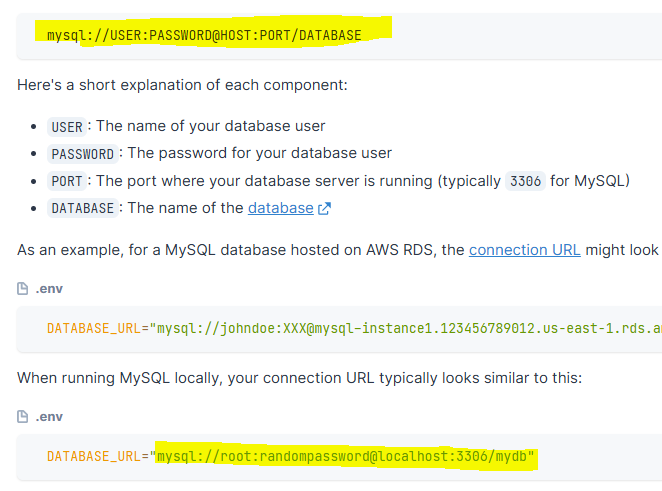
prisma/schema.prisma

datasource db {  
 provider = "mysql"  
 url = env("DATABASE\_URL")  
}

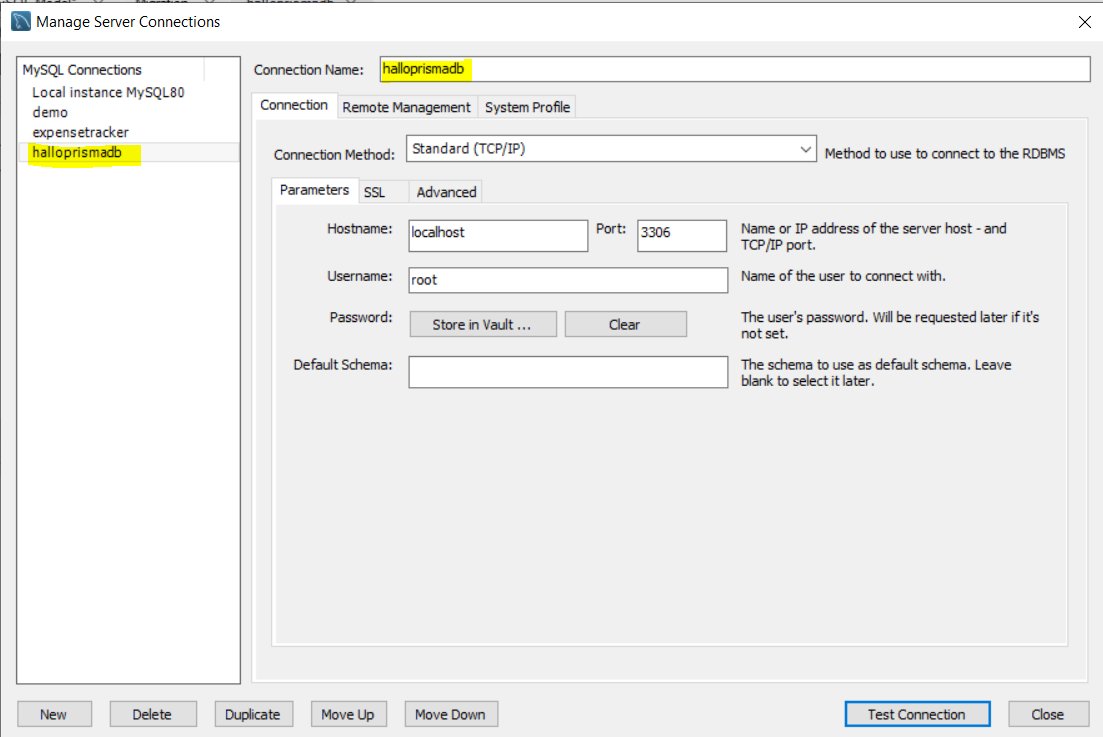


Change database name to halloprismadb





MySQL Database name is halloprismadb



Change database name to halloprismadb

.env

DATABASE\_URL="mysql://root:root@localhost:3306/halloprismadb"

