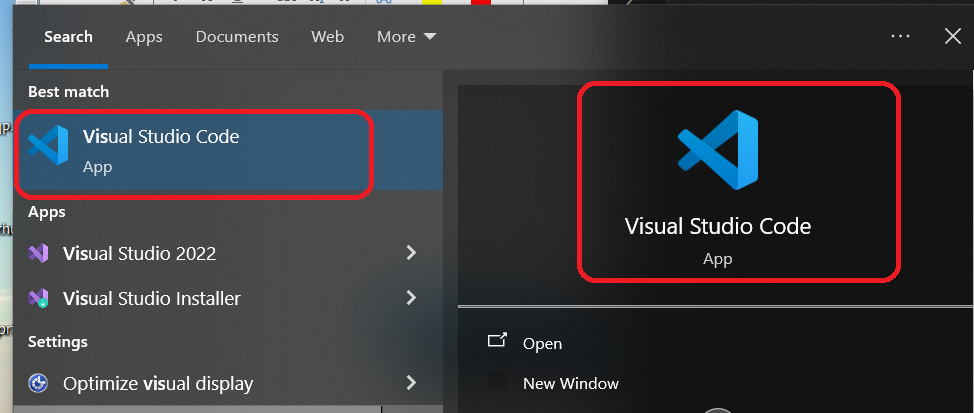
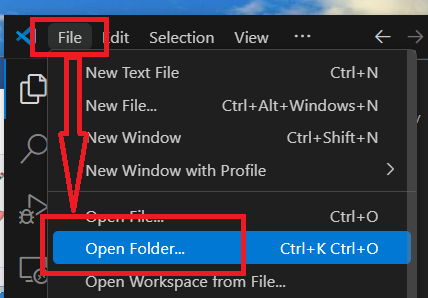
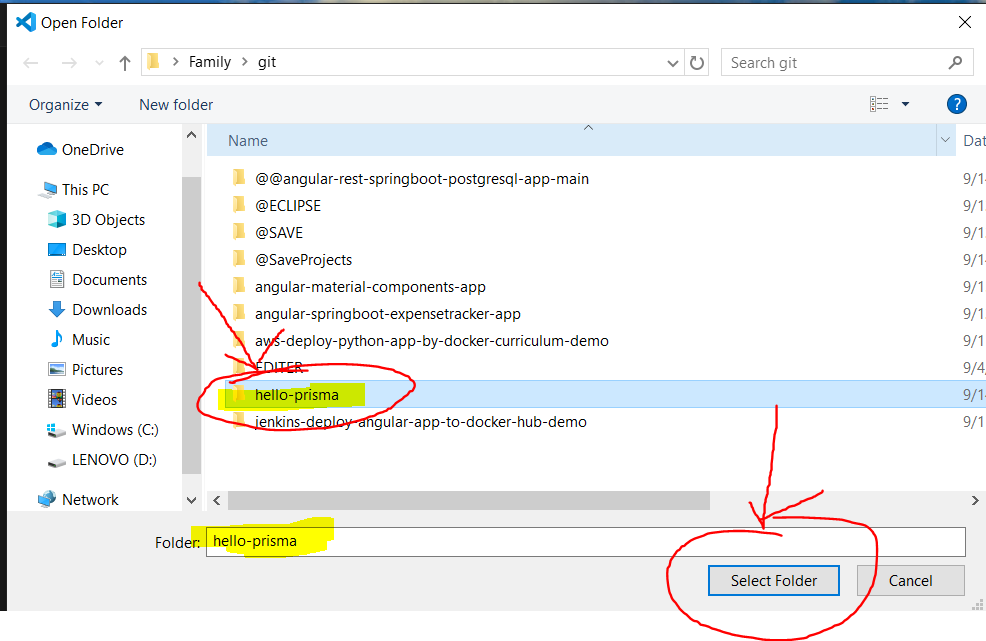
Open 🡪 Visual Studio Code App

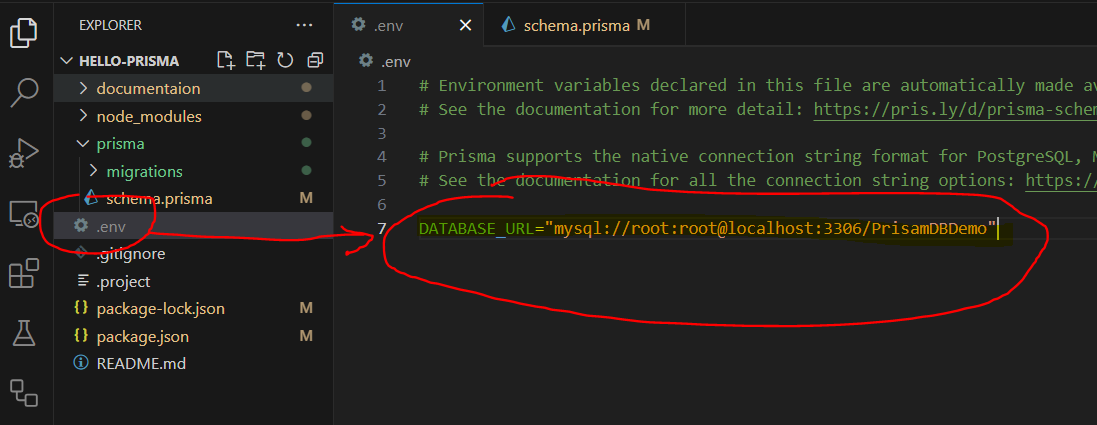




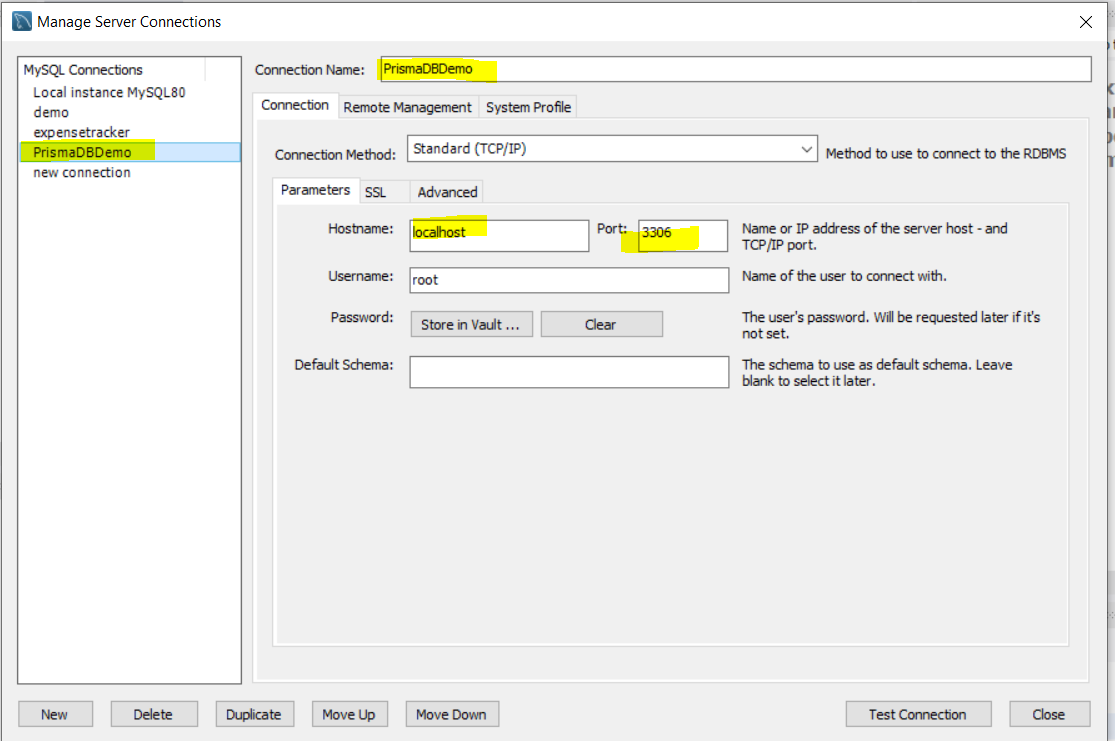


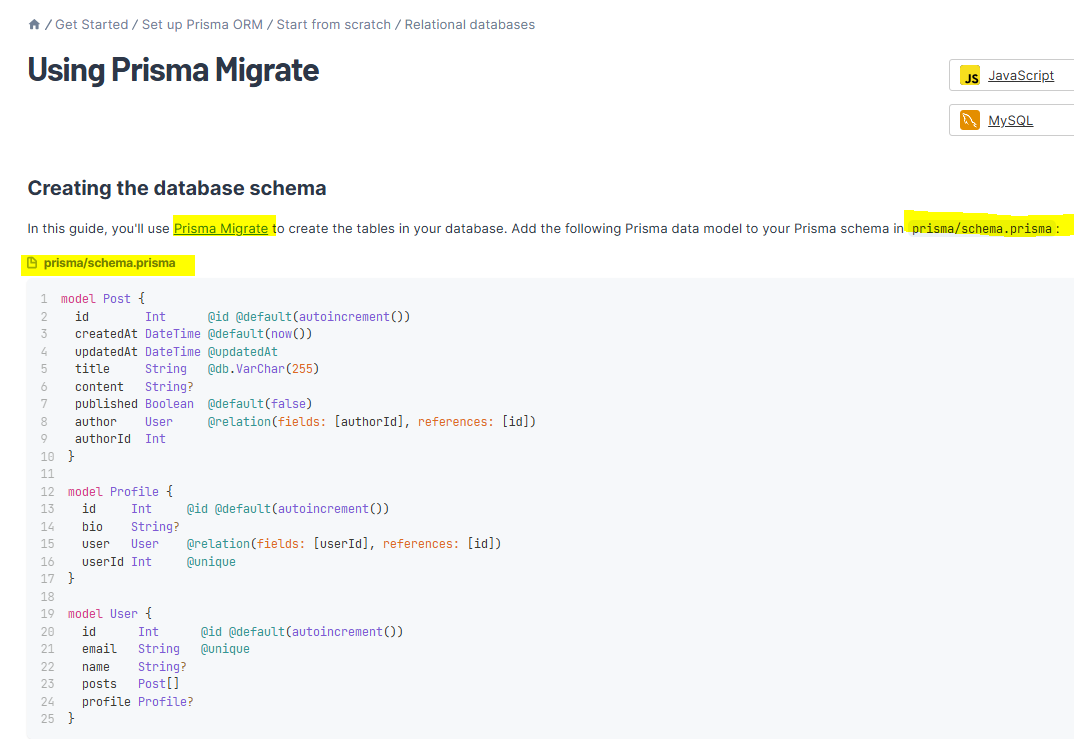


DATABASE\_URL="mysql://root:root@localhost:3306/PrisamDBDemo"



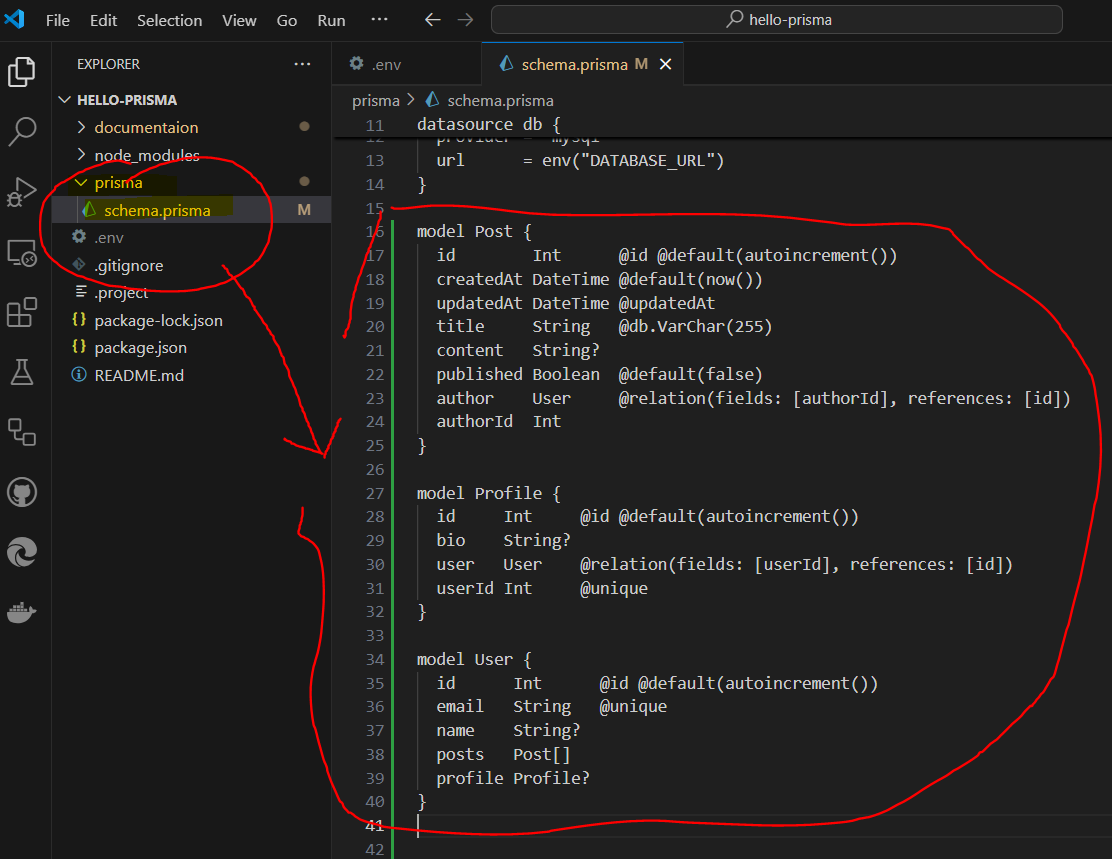
MySQL 🡪 PrismaDBDemo

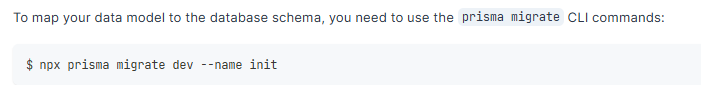




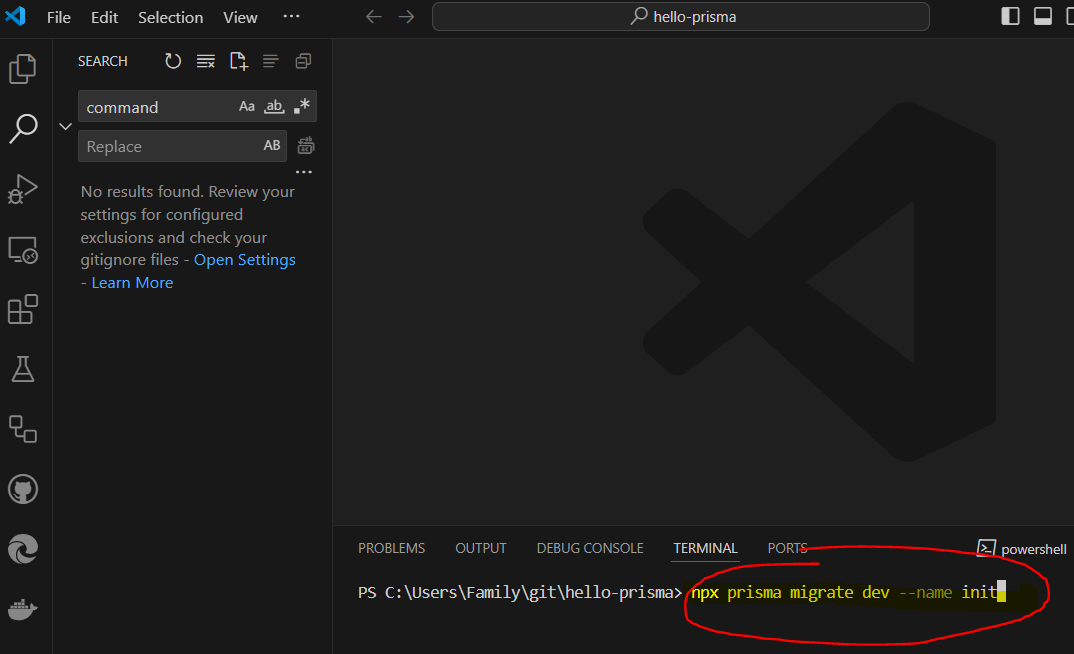
prisma/schema.prisma

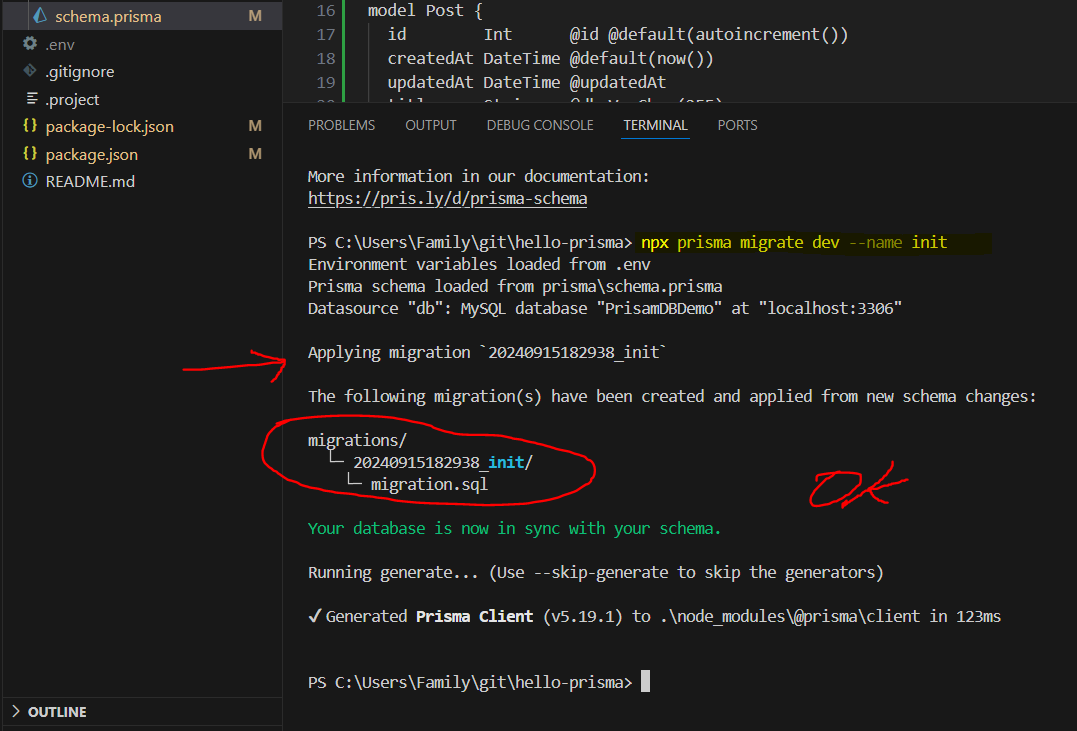
model Post {  
 id Int @id @default(autoincrement())  
 createdAt DateTime @default(now())  
 updatedAt DateTime @updatedAt  
 title String @db.VarChar(255)  
 content String?  
 published Boolean @default(false)  
 author User @relation(fields: [authorId], references: [id])  
 authorId Int  
}  
  
model Profile {  
 id Int @id @default(autoincrement())  
 bio String?  
 user User @relation(fields: [userId], references: [id])  
 userId Int @unique  
}  
  
model User {  
 id Int @id @default(autoincrement())  
 email String @unique  
 name String?  
 posts Post[]  
 profile Profile?  
}



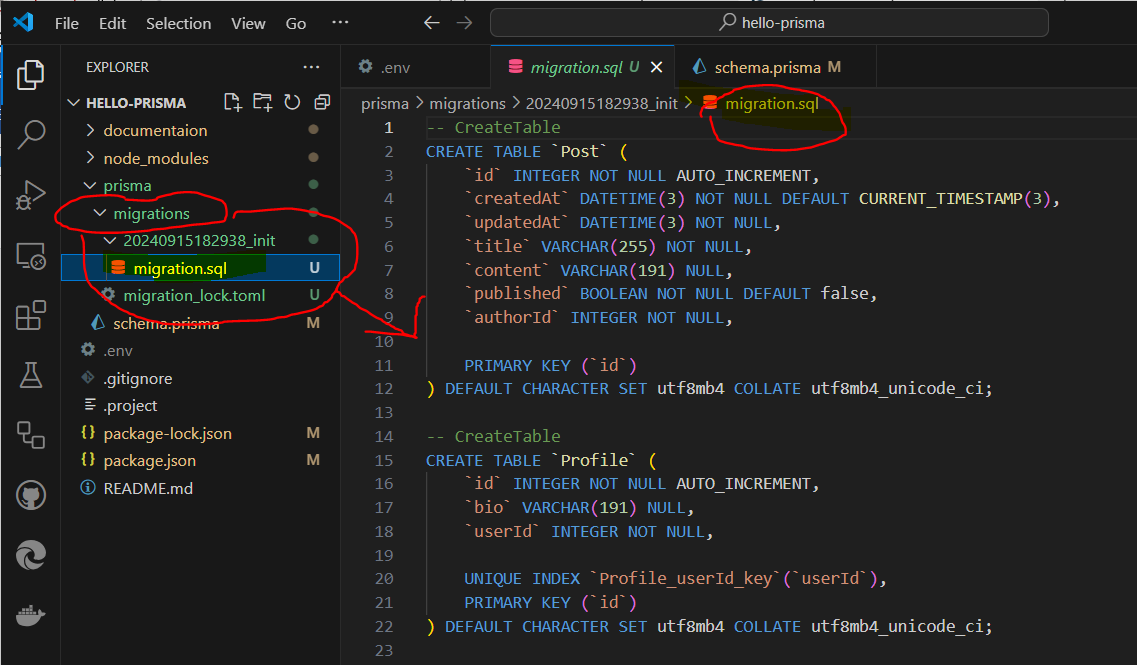


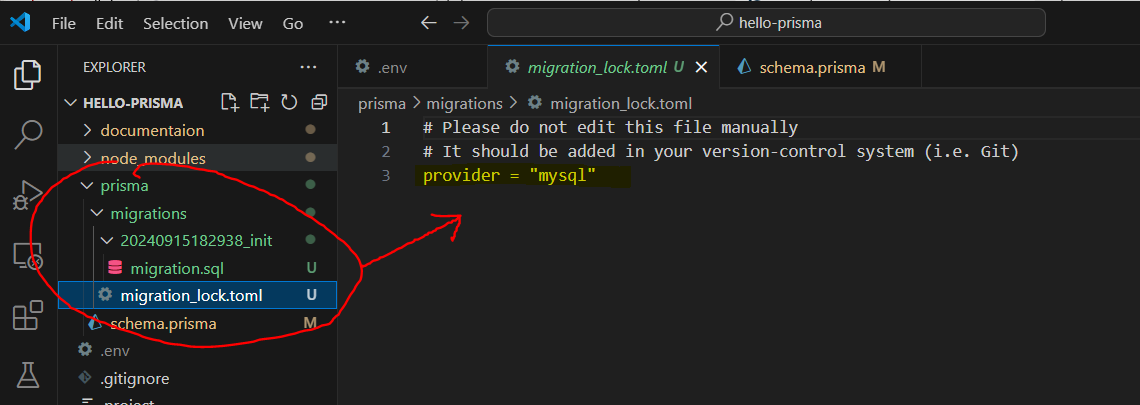
C:\Users\Family\git\hello-prisma> npx prisma migrate dev --name init

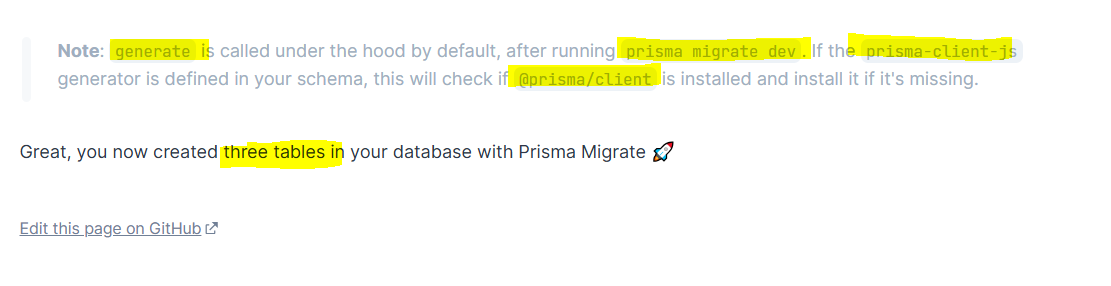




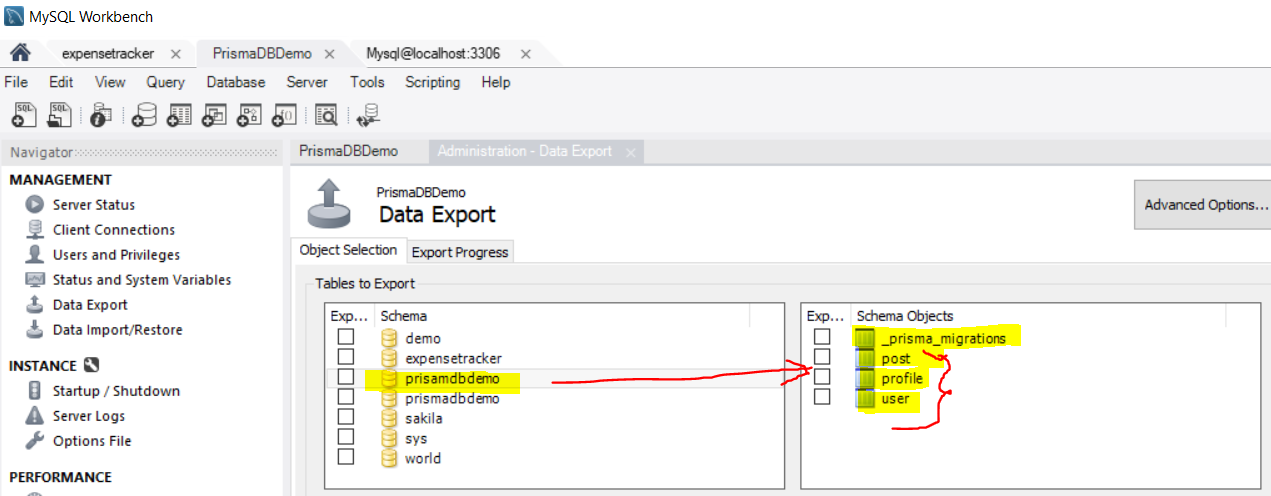
Generated migrations folder

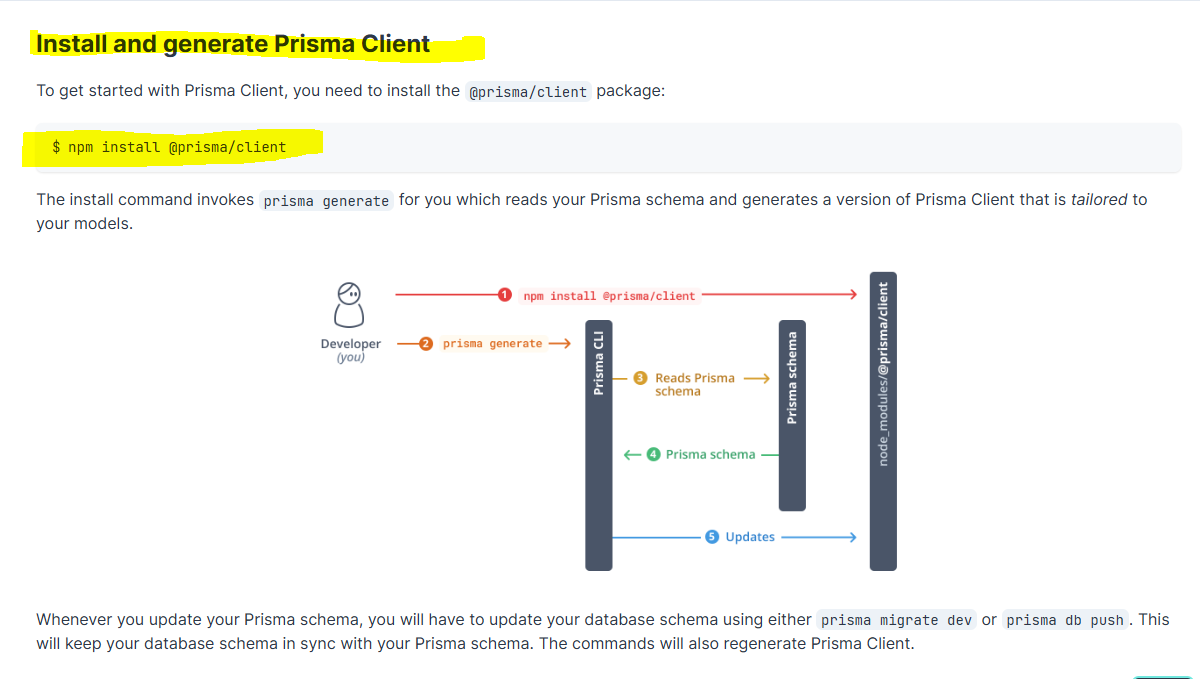






post, profile and user tables can be found in the MySQL PrismaDBDemo database





Whenever you update your Prisma schema, you will have to update your database schema using either prisma migrate dev or prisma db push. This will keep your database schema in sync with your Prisma schema. The commands will also regenerate Prisma Client.

C:\Users\Family\git\hello-prisma> npm install @prisma/client

