FOR BACKEND WE ARE USING HONO INSTEAD OF EXPRESS BECAUSE THE ISSUE IS HONO WORKS BETTER WITH CLOUDWORKS SO THATS WHY
Hono is built for serverless functions

creating be folder and for installing hono using this
npm create hono@latest

creating the endpoints

To begin with, our backend will have 4 routes

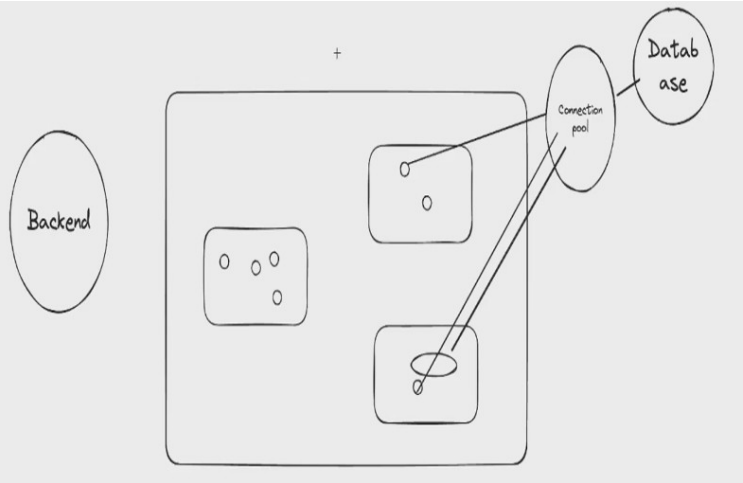
1. POST /api/v1/user/signup
2. POST /api/v1/user/signin
3. POST /api/v1/blog
4. PUT /api/v1/blog
5. GET /api/v1/blog/:id
6. GET /api/v1/blog/bulk

in hono

```
app.post('/api/v1/signup', (c) ⇒ {

})
```

it gives c= context instead of res,req,next so we can access that using this only
Setting up prisma for serverless state
you can use aiven.io , neon.tech
create a pg server using aiven
as the code is ran on small machines in mutliple areas thats why its called as serverless functions(aws lamda, cloudfare workers,vercel edge)



so the issue is in postgress it cannot multiple so many request send by the small machines so to fix that in between we have connection pools
PRISMA ON CLOUDFARE WORKERS

so to run backend code in a serverless env we need a connection pool
so with the db url we also need connection pool url
this url is now no need as prisma consider this thing by default
Connection Pooling = A receptionist managing a fixed number of phone lines ☎

Prisma Accelerate = Receptionist + call center + regional branches + memory of common questions 📞

the prisma uses the original database from the .env file only and just the user should connect to the prisma acclerate we put that url in the wrangler.jsonc file

```
  "vars": {
    "DATABASE_URL": "url"
  },
```

defining the schema and working on that

uuid == popular format to create very long string with very low prob of collision

Understand the relationships in prisma

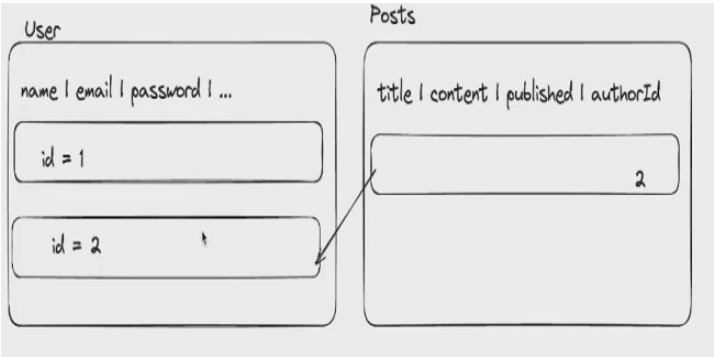
```
model User {
  id       String   @id @default(uuid())
  email    String   @unique
  name     String?
  password String
  posts    Post[]
}

model Post {
  id        String   @id @default(uuid())
  title     String
  content   String
  published Boolean  @default(false)
  author    User     @relation(fields: [authorId], references: [id])
  authorId  String
}
```

in this the author is related to author id and id and both the author id and id are connected also

using the authorid only the tables are connected this authorId tells who is the owner of the post published

the line puts a constraint like if there is not user with id of author id the post will not be made



sometime the system shows does not connect to server so try changing the wifi

you can also do this thing if wifi not working do this from a aws server easy fix

npx prisma migrate dev --name init_schema

npx prisma generate –no-engine (for workers its required specific to cloudfare)

prisma client is like help us use the function such as findOne, post, create etc

npm install @prisma/extension-accelerate

copied the prisma code from the other docker compose project

in this the c also has env var also

```
const app = new Hono<{
    Bindings: {
        DATABASE_URL: string
    }
}>();//this above part is to remove the ts errors
const dbUrl = c.env.DATABASE_URL;
```

or just use //@ts-ignore

```
app.post('/api/v1/signup', async (c) => {
    //@ts-ignore
    const dbUrl = c.env.DATABASE_URL;
    const body = await c.req.json();
    await prisma.user.create({
        data:{
            email: body.email,
            password: body.password,
        }
    })

    return c.text("Hello Hono")

})
```

we can get the data from the user using this syntax

create syntax is also done here

the data can be accessed in the function only not anywhere else

this project is cloudflare specific so yaa