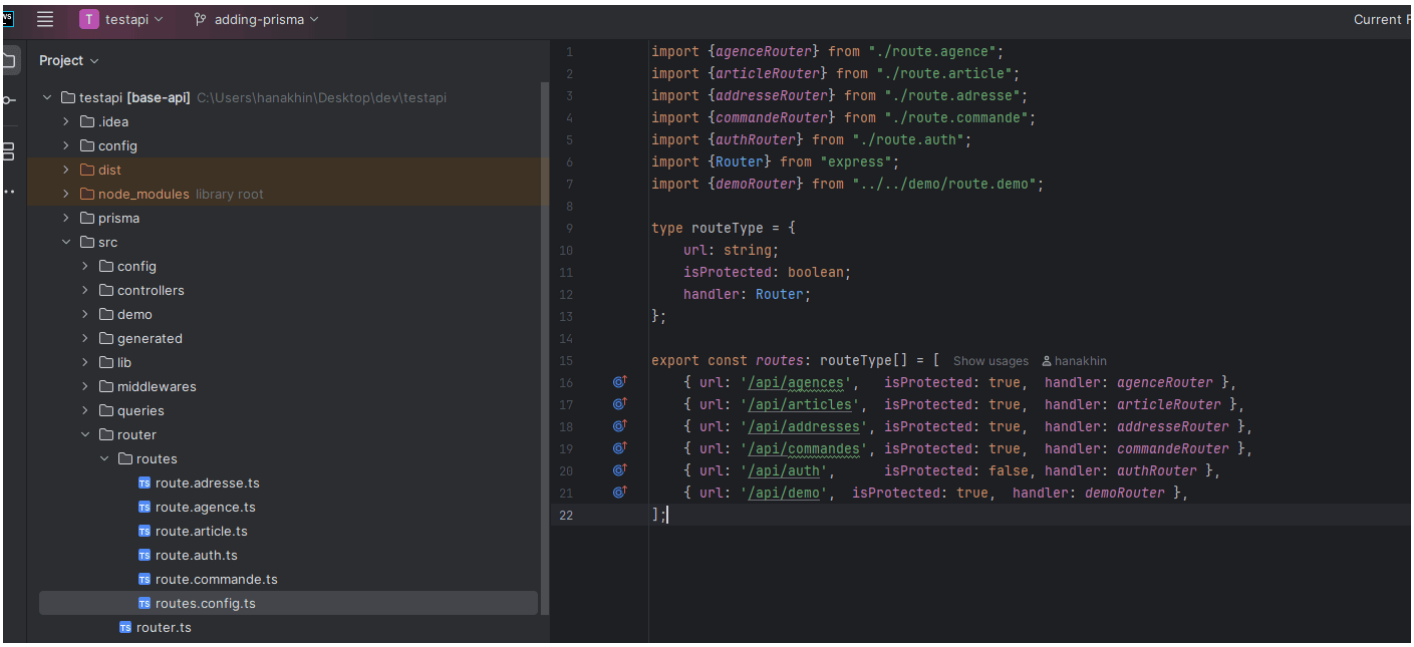
Créer une route

Pour creer une route il faut aller dans src/router/routes/routes.config.ts et y ajouter sa route dans le tableau de routes. Une route attends 3 paramètres:

L'url en string, Le Booleen "isProtected" qui définit si la route est accessible hors connexion ou non , et le router qui lui est associé.

Le paramétré "url" définit la route de base du router associé , donc par exemple si on met "/api/test" dans url et le router "testRouter" dans le handler; toutes les routes du router commenceront par "/api/test"



une fois que c'est fait , il faut aller créer le router associé , on va utiliser le "DemoRouter" pour cet exemple.

dans le fichier "src/router/routes" on crée un ficher ts qu'on appelle route.[le nom du router].ts

```ts
import {Router} from "express";
import {CommandeController} from "../../controllers/commande.controller";

export const commandeRouter :Router = Router(); Show usages hanakhin

commandeRouter
    .get( path: "/", CommandeController.getAllCommandesByCcli)
    .post( path: "/",CommandeController.createPcde)

```

ensuite on crée une instance de Router qui est importé depuis "express"

puis , on définit les routes du router, donc comme on a dit avant toutes les routes du demoRouter on par défaut le prefix "/api/demo", donc il nous suffit de lui mettre l'endpoint qu'on veut précédé de la méthode de la route.

dans notre cas ici on a une get et une post , suivi du controlleur, et de la méthode qu'on veut utiliser.

L'étape suivante est de créer le controlleur associé comme ci dessous.

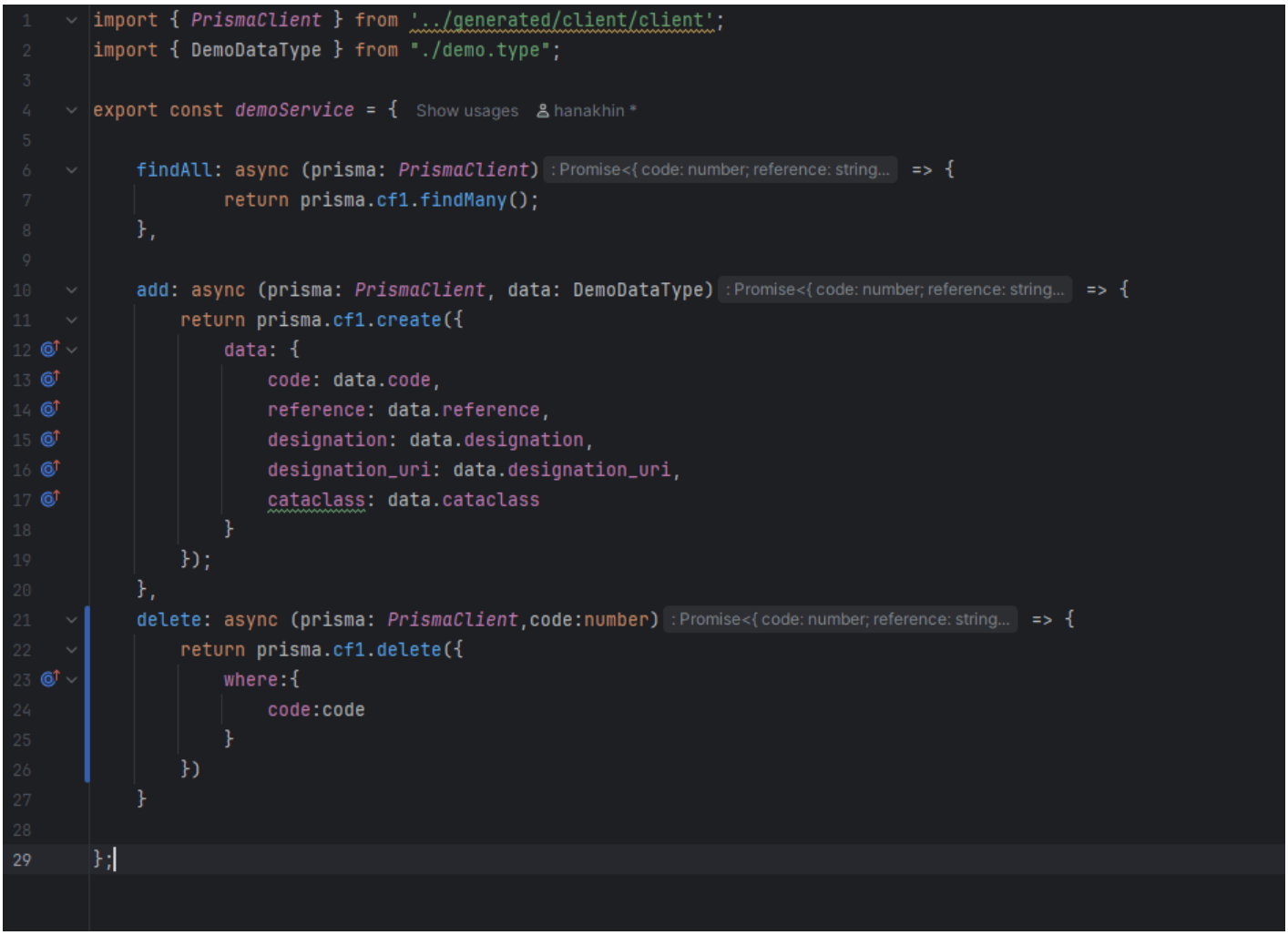
```ts
import { RequestHandler } from "express";
import { demoService } from "./demo.service";

export const DemoController :Record<string, RequestHandler>= { Show usages hanakhin *
    getAllData: async (req: any, res: any) :Promise<any> => {
        try {
            const prisma :any = req.prisma;
            const data :{ code: number; reference: string | null;... = await demoService.findAll(prisma);
            return res.status(200).json(data);
        } catch (err: any) {
            console.error( message: "Error fetching items:", err);
            return res.status(500).json({ error: err.message });
        }
    },

    addData: async (req: any, res: any) :Promise<any> => {...},
    deleteData: async (req: any, res: any) :Promise<any> => {...}
};
```

le controlleur contient la méthode qui permet d'aller chercher le service qui communique avec la base de donnée , lui il se charge de faire l'appel du service et de retourner une réponse .

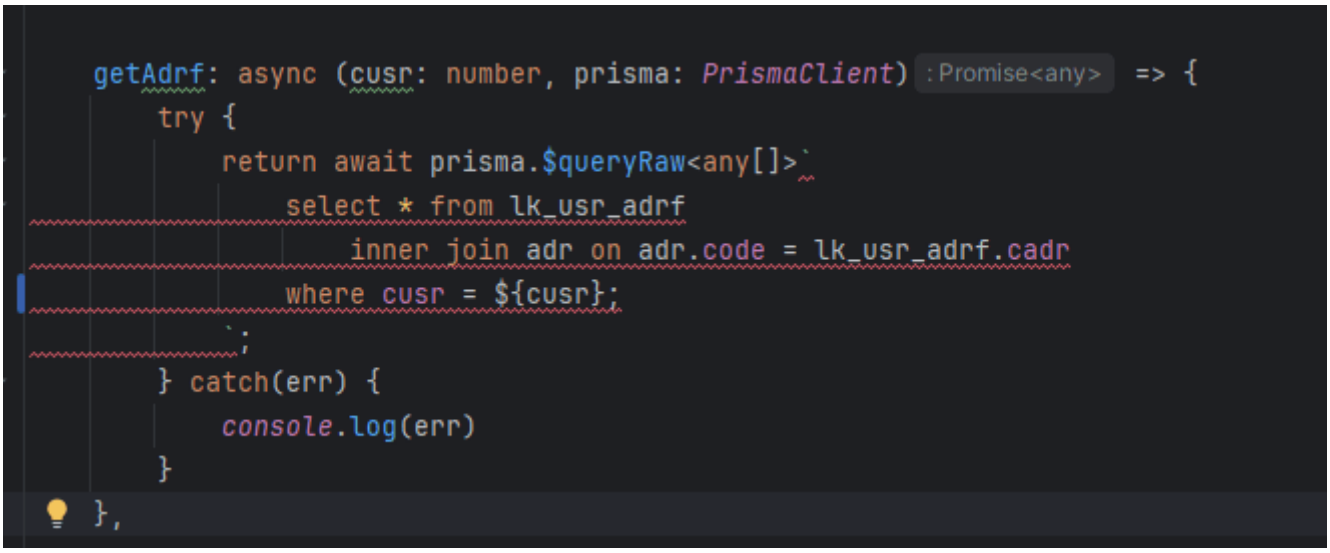
soit une 200 si la requête est acceptée , soit une 500 si il y a un souci, soit une 401 si pas autorisé ... etc , ici on gère uniquement les services.

on crée ce controlleur dans "src/controller" et on l'importe dans le router.

La dernière étape est de créer le service.



Le service ce charge des appel en base de donnée , ici on utilise prisma , mais il est possible de faire du sql ou autres directement .



on crée ce fichier dans "src/services"

pour faire une méthode , il faut suivre une syntaxe claire :

[nom de la méthode] : async ([les paramètres]) =>{[La méthode]}

Donc une fois que tout ceci est fait , la route devrais être accessible .

Le process est le suivant :

Route->Router->Controlleur->Service->Appel db

Revision #1
Created 2026-04-10 15:05:04 UTC by Hanakhin
Updated 2026-04-10 15:06:08 UTC by Hanakhin