

# Corrigé

P1C5 - Créez et testez une première route avec NestJS



**1. Modifier le Service** - Dans le fichier `app.service.ts`, créez une liste de projets et une méthode pour en récupérer un par son identifiant :

```
import { Injectable } from '@nestjs/common';

export interface Project {
  id: number;
  titre: string;
  description: string;
}

const projects: Project[] = [
  { id: 1, titre: 'Refonte du site', description: 'Mise à jour du design complet' },
  { id: 2, titre: 'Migration base de données', description: 'Passage vers PostgreSQL en production' },
  { id: 3, titre: 'Création API', description: 'Développement des routes avec NestJS' }
];

@Injectable()
export class AppService {
  getProjectById(id: number): Project {
    // on utilise la méthode find pour trouver un projet avec un id
    // correspondant à celui passé en paramètre
    const project = this.projects.find(project => project.id === id);
    if (!project) {
      // si aucun projet ne correspond on retourne une erreur précise à l'utilisateur
      throw new NotFoundException('Project not found');
    }
    return project;
  }
}
```

**2. Modifier le Controller** - Dans le fichier `app.controller.ts`, ajoutez une route avec un paramètre dynamique :

```
import { Controller, Get, Param } from '@nestjs/common';
import { AppService } from './app.service';

@Controller()
export class AppController {
  constructor(private readonly appService: AppService) {}

  @Get('projects/:id')
  getProject(@Param('id') id: string): Project {
    return this.appService.getProjectById(Number(id));
  }
}
```

**3. Tester la route** - Après avoir lancé votre serveur :

- Méthode : `GET`
- URL : `http://localhost:3000/projects/1`

Résultat attendu :

- Statut **200**
- Réponse JSON :

```
{
  "id": 1,
  "titre": "Refonte du site",
  "description": "Mise à jour du design complet"
}
```