

Rapport du projet Nodes Js, Express Js

Introduction

Dans ce doc, je vais vous raconter comment j'ai fait pour documenter mon API qui gère des infos sur les Pokémon. C'est un truc que j'ai codée en ExpressJS et qui est assez cool, à mon avis.

Installation

Pour commencer, j'ai écrit comment installer et lancer l'API. Genre, il faut avoir Node.js, cloner mon repo GitHub, et puis lancer `npm install` pour avoir toutes les dépendances. Après ça, un simple `node app.js` et le tour est joué !

Les Endpoints

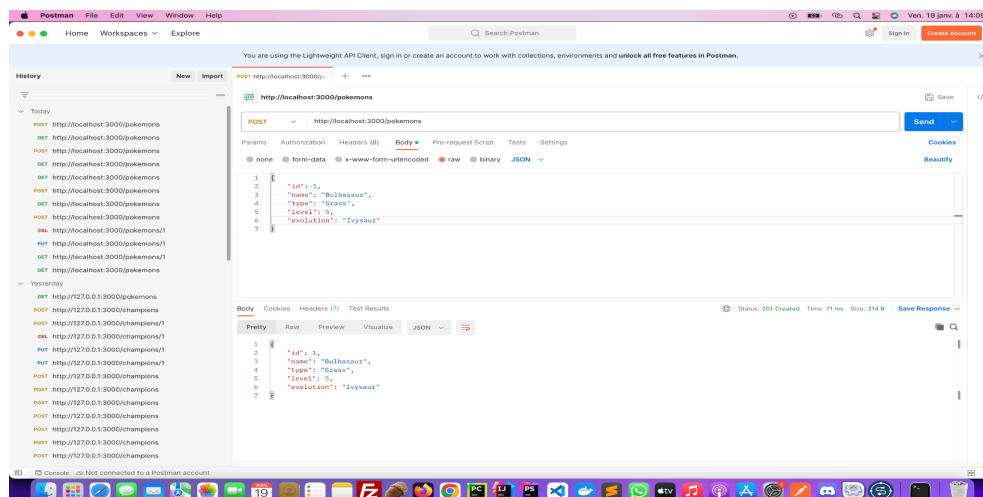
C'est la partie principale ! Pour chaque endpoint de mon API, j'ai noté :

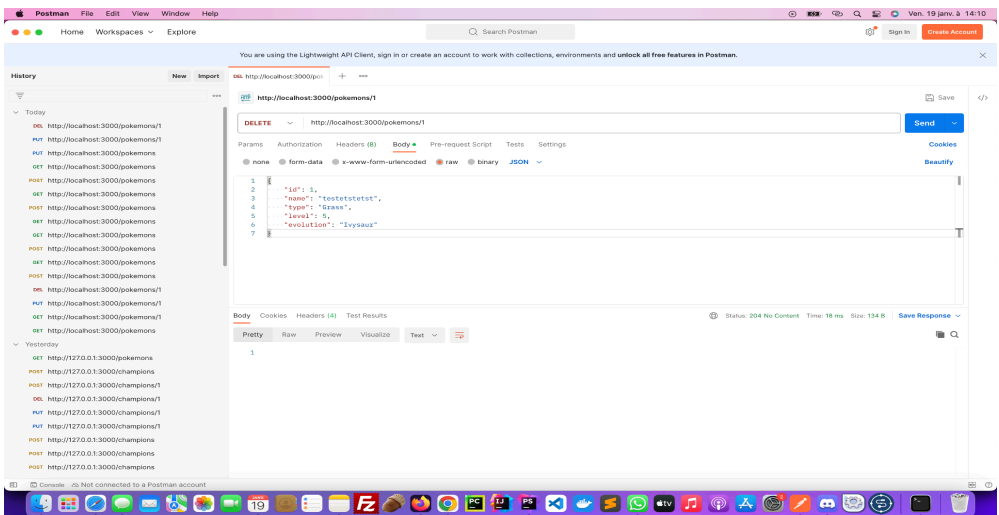
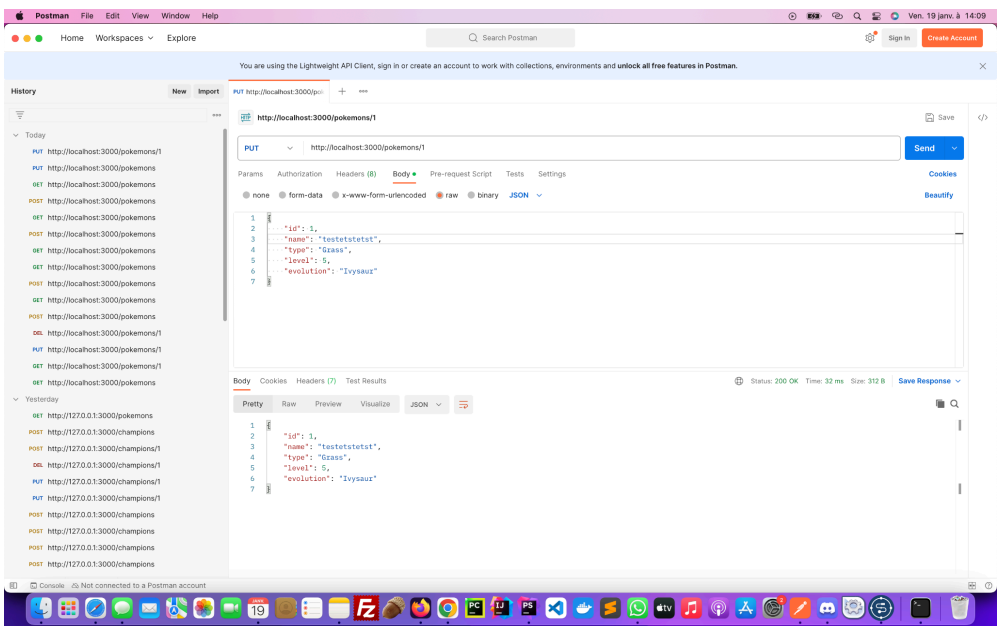
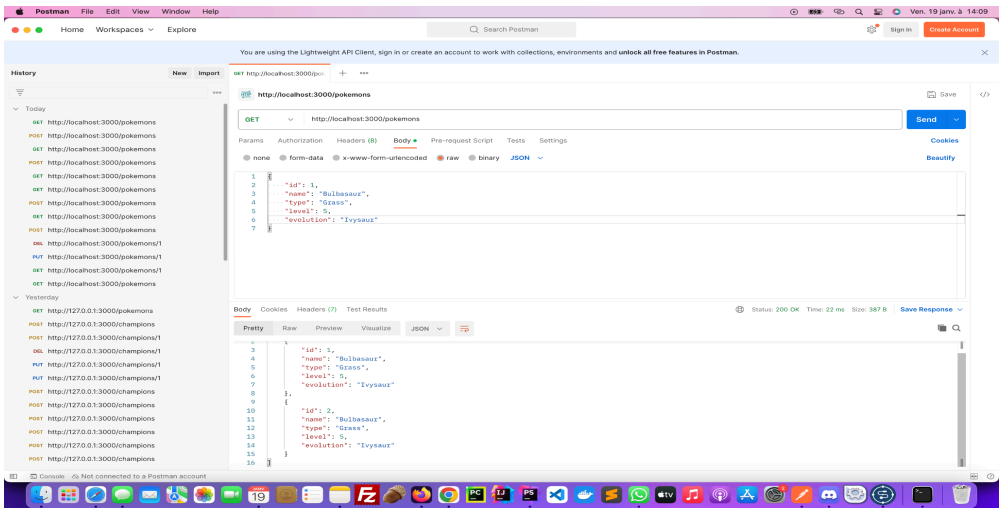
- **L'URL** : Où envoyer la requête.
- **La Méthode** : GET pour récupérer des données, POST pour en ajouter, etc.
- **Les Paramètres et le Corps de la Requête** : Ce qu'il faut envoyer avec la requête.
- **Les réponses** : À quoi ressemblent les données quand tout se passe bien.
- **Les Messages d'Erreur** : Pour savoir ce qui n'a pas marché si ça tourne mal.

Mon approche des Tests :

Pour garantir la fiabilité et la solidité de mon API Pokémon développée en ExpressJS, j'ai Testé toutes ses fonctionnalités essentielles. J'ai particulièrement veillé à ce que les opérations CRUD – c'est-à-dire créer, lire, mettre à jour et supprimer des données – fonctionnent sans accroc. En passant en revue chaque fonctionnalité et en m'assurant qu'elles répondaient correctement aux requêtes :

Voici les captures d'écrans des tests :





Conclusion

En finalisant ce projet d'API Pokémon avec ExpressJS, j'ai non seulement renforcé mes compétences en développement web, mais j'ai également acquis une compréhension approfondie des opérations CRUD et des principes RESTful. L'expérience de construire une API depuis le début, de la conceptualiser jusqu'à la tester minutieusement, a été incroyablement enrichissante.