

Fichier : 04 - mock_llm_claude.mock.js

Tutoriel Node.js : Comprendre la fonction `reply`

Dans ce tutoriel, nous allons decomposer une fonction Node.js appelee `reply`. Cette fonction est concue pour generer une reponse formatee a partir d'un certain type de demande et des donnees associees.

Code de la fonction `reply`

```
function reply(type, data) {
  const name = (data.name || 'Inconnu').replace('-', ' ');
  const style = data.style || 'neutral';
  const length = data.length || 'medium';
  const llm = data.llm || 'claude';
  const validType = ['biography', 'filmography', 'summary'].includes(type) ? type : 'contenu';

  return `Mock Backend - Demande envoyee a ${llm} pour une ${validType} de "${name}", avec un
style "${style}" et une longueur "${length}".`;
}

export default reply;
```

Explication du code

La fonction `reply` prend deux arguments : `type` et `data`. `type` est une chaine de caracteres qui indique le type de contenu demande, et `data` est un objet contenant des informations supplementaires sur la demande.

Definition des constantes

```
const name = (data.name || 'Inconnu').replace('-', ' ');
const style = data.style || 'neutral';
const length = data.length || 'medium';
const llm = data.llm || 'claude';
```

Dans cette section, nous definissons plusieurs constantes a partir des donnees fournies. Si une certaine valeur n'est pas fournie, nous utilisons une valeur par default. Par exemple, si `data.name` n'est pas fourni, nous utilisons 'Inconnu' comme valeur par default. De plus, si `data.name` contient un tiret, nous le remplacons par un espace.

Validation du type de contenu

```
const validType = ['biography', 'filmography', 'summary'].includes(type) ? type : 'contenu';
```

Ici, nous verifions si le type de contenu demande est valide. Si `type` est 'biography', 'filmography' ou 'summary', nous utilisons cette valeur. Sinon, nous utilisons 'contenu' comme valeur par default.

Generation de la reponse

```
return `Mock Backend - Demande envoyee a ${llm} pour une ${validType} de "${name}", avec un style "${style}" et une longueur "${length}".`;
```

Enfin, nous generons la reponse en utilisant une chaine de caracteres formatee. Cette chaine contient toutes les informations que nous avons recueillies et validees.

Conclusion

La fonction ``reply`` est un exemple de la facon dont nous pouvons traiter et valider des donnees en Node.js. En comprenant comment cette fonction fonctionne, vous pouvez mieux comprendre comment manipuler et valider les donnees dans vos propres applications Node.js.