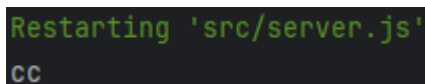


Énoncé travaux pratique - Sujet 1

Charité et Blockchain

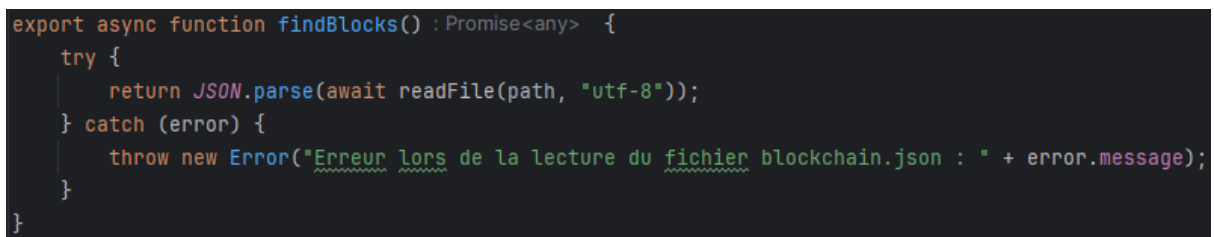
Étape 1 - état des lieux

J'ai ajouté la ligne `console.log("cc")` dans le switch case 'GET:/blockchain'.
Maintenant, quand je me rend sur l'url <http://localhost:3000/blockchain> cela print cc dans la console de l'IDE.



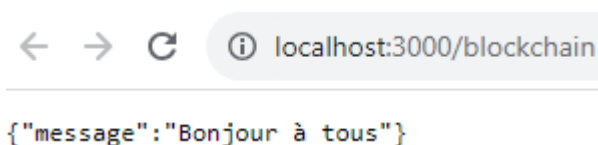
```
Restarting 'src/server.js'  
cc
```

Étape 2 - apprenons à lire



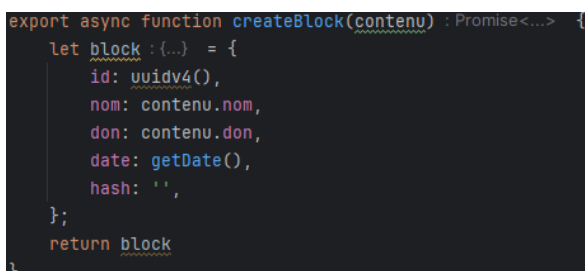
```
export async function findBlocks() : Promise<any> {  
  try {  
    return JSON.parse(await readFile(path, "utf-8"));  
  } catch (error) {  
    throw new Error("Erreur lors de la lecture du fichier blockchain.json : " + error.message);  
  }  
}
```

J'ai développé la méthode `findBlocks()`.
Maintenant, quand je me rend sur l'url <http://localhost:3000/blockchain> cela m'affiche `{"message": "Bonjour à tous"}`.



```
< > ↻ ⓘ localhost:3000/blockchain  
  
{ "message": "Bonjour à tous" }
```

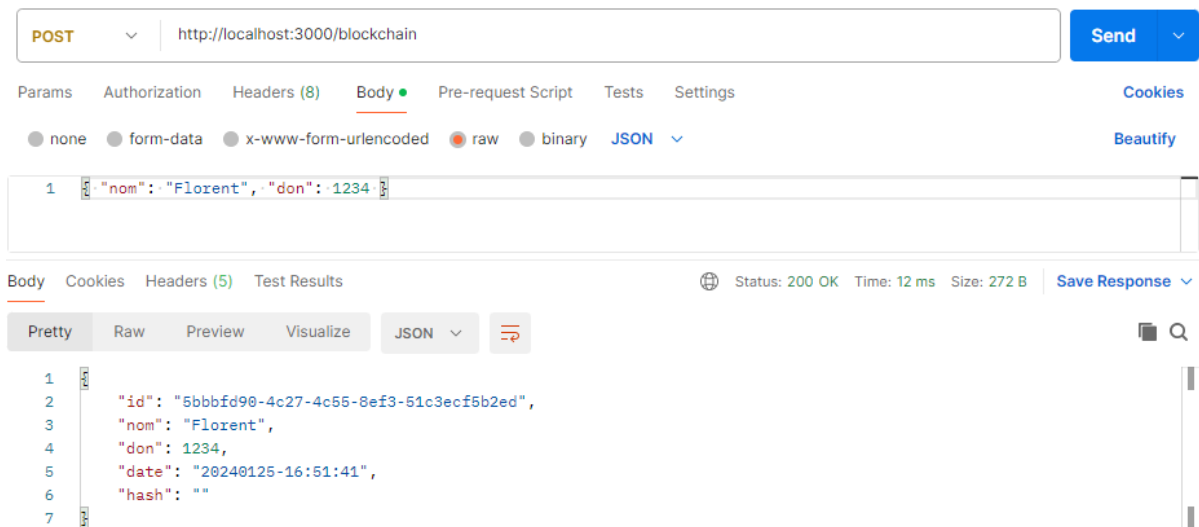
Étape 3 - une brique après l'autre



```
export async function createBlock(contenu) : Promise<...> {  
  let block : {...} = {  
    id: uuidv4(),  
    nom: contenu.nom,  
    don: contenu.don,  
    date: getDate(),  
    hash: '',  
  };  
  return block  
}
```

J'ai développé la méthode `createBlock()` qui crée un nouvel objet bloc en utilisant les données fournies dans le contenu.

Quand je fais cette requête POST, cela me renvoie le bloc avec toutes les bonnes informations dans la réponse.



Étape 4 - « Vers l'infini et au-delà ! »

```
const blocks = await findBlocks();
if (blocks.length === 0) {
  return null;
}
return blocks[blocks.length - 1];
```

Tout d'abord je code la méthode findLastBlock() qui renvoie le dernier bloc ou null s'il n'y a aucun bloc.

```
const lastBlock : (id: string, nom: string, don: ... | null) = await findLastBlock();

if (lastBlock !== null) {
  const hash : Promise<ArrayBuffer> = createHash('sha256').update(JSON.stringify(lastBlock)).digest({ algorithm: 'hex' });
  block.hash = hash;
}
```

Par la suite, dans la méthode createBlock, je récupère ce dernier bloc (si il existe) que je vais ensuite hash pour pouvoir remplir le champ hash du bloc en question (celui qui est en train d'être créé).