

Football API

Wassim Lomri A21

Spécification du système

Football API est une API REST en Node.js. Elle a pour objectif de permettre de retrouver plus rapidement les données tout autour du football. Cette API regroupe les entités suivantes : Joueur, Club, Trophée et Championnat.

Contexte :

(Voir le diagramme de classe pour comprendre les liens entre les différentes entités)

1. Le joueur possède un id, un nom, un prénom, un âge, une nationalité et un club.
2. Un club est identifié par son id, son nom, son année et son championnat.
3. Un trophée contient un id, un nom, un vainqueur et une année.
4. Un championnat possède un id, un nom et un pays.
5. Il est possible de limiter le nombre d'éléments d'une liste retournée.
6. Il est possible d'afficher les joueurs appartenant à un club.
7. Il est possible d'afficher le championnat appartenant à un joueur, un trophée ou à un club.
8. Il est possible d'afficher la liste des clubs appartenant à un championnat.
9. Il est possible d'afficher/ajouter/modifier/supprimer chacune de ces entités.
10. Il est possible d'afficher le club/championnat d'un joueur.
11. Il est possible d'afficher le trophée/championnat d'un club.
12. Il est possible d'afficher le club/championnat d'un trophée.
13. Il est possible d'afficher les clubs appartenant à un championnat.

Endpoints disponibles :

- (GET) /: Information de l'API.

Joueur :

- (GET) /joueurs/: Récupère la liste de tous les joueurs.
- (POST) /joueurs/: Ajout d'un nouveau joueur.
- (GET) /joueurs/nationalite/{nationalite}/: Récupère les joueurs qui ont cette nationalité.
- (GET) /joueurs/{nom_joueur}/: Récupère le joueur en fonction de son nom
- (PUT) /joueurs/{nom_joueur}/: Met à jour le joueur
- (DELETE) /joueurs/{nom_joueur}/: Retire un joueur

- (GET)/club/{nom_club}/joueurs/: Récupère tous les joueurs appartenant à ce club.

Club :

- (GET) /club/: Récupère la liste de tous les clubs.
- (POST) /club/: Ajout d'un nouveau club.
- (GET) /joueur/{nom_joueur}/club/: Récupère le club où joue le joueur.
- (GET) /club/championnat/{championnat}/: Récupère les clubs qui ont ce championnat.
- (GET) /club/nom/{nom_club}/: Récupère le club en fonction de son nom
- (PUT) /club/nom/{nom_club}/: Met à jour le club
- (DELETE)/club/nom/{nom_club}/: Retire un club
- (GET)/trophee/{nom_trophee}/club/: Récupère le club qui a gagné ce trophée.

Championnat :

- (GET) /championnat/: Récupère la liste de tous les championnats.
- (POST) /championnat/: Ajout d'un nouveau championnat.
- (GET) /joueur/{nom_joueur}/championnat/: Récupère le championnat où joue le joueur.
- (GET) /club/championnat/{championnat}/: Récupère les clubs qui ont ce championnat.
- (GET) /club/{nom_club}/championnat/: Récupère le championnat où joue le club.
- (PUT) /championnat/{nom_champ}/: Met à jour le championnat
- (DELETE)/championnat/{nom_champ}/: Retire un championnat
- (GET)/championnat/pays/{pays}/: Récupère les championnats présents dans un pays.
- (GET)/championnat/{nom_champ}/: Récupère le championnat en fonction du nom.
- (GET)/trophee/{nom_trophee}/championnat/: Récupère le championnat en fonction du trophée.

Trophée :

- (GET) /trophee/: Récupère la liste de tous les trophées.
- (POST) /trophee/: Ajout d'un nouveau trophée.
- (GET) /club/{nom_club}/trophee/: Récupère le trophée gagné par ce club.
- (GET) /trophee/vainqueur/{vainqueur}/: Récupère le trophée en fonction du vainqueur.
- (GET) /trophee/{nom_trophee}/: Récupère le trophée en fonction de son nom
- (PUT) /trophee/{nom_trophee}/: Met à jour le trophée
- (DELETE)/trophee/{nom_trophee}/: Retire un trophée

Exécuter l'API

Prérequis

Il faut avoir Node.js installé sur sa machine. Vous pouvez télécharger les dernières versions depuis le site

<https://nodejs.org/en> (<https://nodejs.org/en>).

Démarrer le serveur

Dépendance/Démarrage

Ouvrir un terminal, se placer dans la racine du projet et lancer le serveur.

```
cd football_api_Wassim_LOMRI_A21_R5_08  
npm start
```

"*npm start*" va exécuter une fois "*npm install*" si ce dernier n'a pas été exécuté préalablement et va démarrer le serveur.

Vous pouvez aussi faire :

```
cd football_api_Wassim_LOMRI_A21_R5_08  
npm install  
node index.js
```

Accéder

Vous rendre à l'adresse : <http://localhost:8080>

Jeu de données

Football API utilise un jeu de données fictives pour simuler des informations sur les joueurs, les trophées, les clubs et les championnats de football. Ces données sont stockées en mémoire dans des fichiers **.json** dans un répertoire **data** du projet.

Exemple fichier .json :

Championnat :

```
[  
  {  
    "id_champ": 1,  
    "nom_champ": "La Liga",  
    "pays": "Espagne"  
  }  
]
```

Méthodologie

Avez-vous fait la spécification OpenAPI avant le serveur ou l'inverse ?

J'ai fait la spécification OpenAPI avant de faire le serveur, la spécification a été faite sur Swagger.

Pourquoi ?

Pour commencer, Swagger propose la génération d'un serveur en Node.js.

De plus cette méthodologie donne une approche avec plusieurs avantages tel que la clarté de conceptions, avec la définition des endpoints et des paramètres.

De plus Swagger permet de détecter les erreurs potentielles et génère une documentation de l'API