Projet Crypto Arbitrage :

1. A propo de l'API

ToDo

- Créer une BDD init symboles ( 1 requête pour filtrer les assets 🡪 puis une requpête pour filtrer les symbols) :

- Les exclusions se feront sur la pair directement.

Notes :

* Api : <https://docs.coinapi.io/?javascript>
* La sand-box possède un nombre de requêtes journalières + élevé
* Les symbols possède des types ( FUTURE , PERPETUAL , **SPOT**, … )

Points de terminaison

| **Enviroment** | **Chiffrement** | **Valeur** |
| --- | --- | --- |
| Production | Oui | https://rest.coinapi.io/ |
| Production | Non | http://rest.coinapi.io/ |
| Sand-boc | Oui | https://rest-sandbox.coinapi.io/ |
| Sand-box | Non | http://rest-sandbox.coinapi.io/ |

Sand-box :

L'environnement sandbox est fourni pour les cas d'utilisation de développement et de non-production, il présente peu de différences par rapport à celui de production:

* Vous avez toujours besoin d'une clé API active pour accéder au bac à sable; cela peut être gratuit.
* Nous ne fournissons aucun support ou SLA pour cet environnement.
* Vous ne pouvez pas demander les données historiques il y a plus d'un jour.
* Le paramètre de limite par défaut et la valeur maximale sont définis sur 10.
* Les données en temps réel et historiques sont limitées à des échanges spécifiques: COINBASE, GEMINI, LMAX Digital UAT, Binance Futures Testnet et ECB (Banque centrale européenne).
* Les données peuvent être invalides / fausses ou retardées.
* Les modifications de l'API pourraient être visibles plus rapidement sur le bac à sable que dans l'environnement de production.

Requêtes http

* Autorisation

X-CoinAPI-Key: 85B572BB-58D2-4CC8-BF82-319D44B66D25

* Chaque requête HTTP doit contenir l'en-tête suivante **car toutes nos réponses sont au format JSON.**

Accept: application/json

* Nous vous encourageons à utiliser l'en-tête de requête HTTP pour toutes les requêtes.

Accept-Encoding: deflate, gzip

Cela nous indiquera que nous pouvons vous fournir des données compressées qui, de votre côté, doivent être **décompressées** de manière transparente.

En autorisant la compression des données, vous réduisez les besoins en bande passante d'environ 80%

Appliquer des filtres à la recherche :

GET /v1/symbols?filter\_symbol\_id={filter\_symbol\_id}&filter\_exchange\_id={filter\_exchange\_id}&filter\_asset\_id={filter\_asset\_id}

1. BID & ASK:

Ask & Bid : BTC/USDT

**ASK** : Est la valeur en **quote**(USDT) que les vendeurs(de type "maker") **demandent** aux acheteurs(de type "taker") pour leurs vendre **1** **base**(BTC)

Ex : Les vendeurs demandent 9100 USDT pour vendre 1 BTC

**BID :** Sont les **offres** en **quote**(USDT) que sont prêt à faire acheteurs(de type "maker") aux vendeurs(de type "taker") afin de leurs acheter **1** **base**(BTC)

Ex : Les acheteurs offrent 9000 USDT pour acheter 1 BTC

L'**offre** et la **demande** d'un carnet d'ordre sont toujours exprimé en **QUOTE**

Dans le cadre du projet :

* Nous achèterons toujours en tant que **taker** au prix de l'**ASK** !
* Nous revendrons toujours en tant que **taker** au prix du **BID** !

Le volume : BTC/USDT

Le volume d'un carnet d'ordre est toujours exprimé en **BASE**.

1. Système d'exclusion :

* Pour une pair : je peux définir une liste de markets dans lesquels la pair sera exclus. Si j'exclus la pair de **tous les markets** alors c'est la pair elle-même qui est exclus.
* Pour un market : Je peux simplement l'exclure ou pas .
* Pour un asset : je peux définir une liste de pairs dans lesquelles l'asset sera exclus. Si j'exclus l'asset de **toutes les pairs** alors c'est l'asset lui-même qui est exclus.

Une pair, un market ou un asset sera **exclus** seulement s'il contient un code d'exclusion avec la sévéritée "**éliminatoire**".