

---

# **DB-API GRANTA\_引張試験検索機能追加**

リリース **1.0.0**

**SIP-MI(A4)**

**2020 年 02 月 26 日**

# 目次:

第 1 章	GRANTA 検索 環境設定手順	1
1.1	システム構成	1
1.2	環境設定	1
1.2.1	必要パッケージ	1
1.2.2	パッケージインストール	1
1.2.3	動作確認	2
第 2 章	DB-API nodejs サンプルコード作成手順	3
2.1	概要	3
2.2	システム構成	3
2.3	関連ソース	3
2.4	DB-API nodejs 構成	3
2.5	nodejs 編集	4
2.5.1	関連ファイルデプロイ	4
2.6	サービス再起動	5
2.7	動作確認	5
2.8	参考	5
2.8.1	GRANTA 検索スクリプト	5

## 第 1 章

# GRANTA 検索 環境設定手順

## 1.1 システム構成

OS	CentOS 7.4
python	3.6 以降

## 1.2 環境設定

### 1.2.1 必要パッケージ

python(anaconda) についてはすでにインストール済みとする。  
 その他必要なパッケージは以下

パッケージ	内容
numpy1.17.4	数値計算ライブラリ
pandas0.25.3	データ分析/解析ライブラリ
Pint0.10.1	単位操作ライブラリ
GRANTA ScriptingToolKit 1.4	GRANTA 操作ライブラリ

### 1.2.2 パッケージインストール

numpy, pandas, Pint については以下コマンドにてインストールする。

```
$ pip install numpy==1.17.4
```

- \* 各パッケージのバージョン番号を指定する。
- \* (指定しない場合は最新版がインストールされる)

GRANTA ScriptingToolKit については以下操作する。

```
$ (インストーラファイルを任意のフォルダに置き、解凍しておく)
$ cd MIScriptingToolkit-1.4.0
$ sudo python3.6 setup.py install
```

- \* python のバージョンにより上記コマンドは適宜変更すること。

### 1.2.3 動作確認

以下のテストスクリプトを作成し、実行する。

(test.py)

```
import numpy
import pandas
from pint import UnitRegistry
import GRANTA_MIScriptingToolkit as gdl
```

\* 実行してエラーが出なければ OK.

\* エラーとなる場合は、該当するパッケージを入れなおす。

## 第 2 章

# DB-API nodejs サンプルコード作成手順

## 2.1 概要

本手順書では、GRANTA に対し、DB-API を用いて検索を行うための手順を示す。

## 2.2 システム構成

OS	CentOS 7.4
mySQL	v11.15.0

## 2.3 関連ソース

<<https://github.com/materialsintegration/DB-API/tree/master/sample/sample2/src/GRANTA>>

ファイル	内容
GrantaUtilities.py	GRANTA 操作ユーザ関数スクリプト
PythonUtilities.py	Python 操作ユーザ関数スクリプト
class_iniconf.py	設定ファイルクラス
granta_schema.csv	GRANTA 検索項目
searchGRANTA.conf	GRANTA 検索用メインスクリプト設定ファイル
searchGRANTA<n>.py	GRANTA 検索用メインスクリプト

<<https://github.com/materialsintegration/DB-API/tree/master/sample/sample2/src/DB-API>>

ファイル	内容
get_test_GRANTA.js	エンドポイントスクリプト

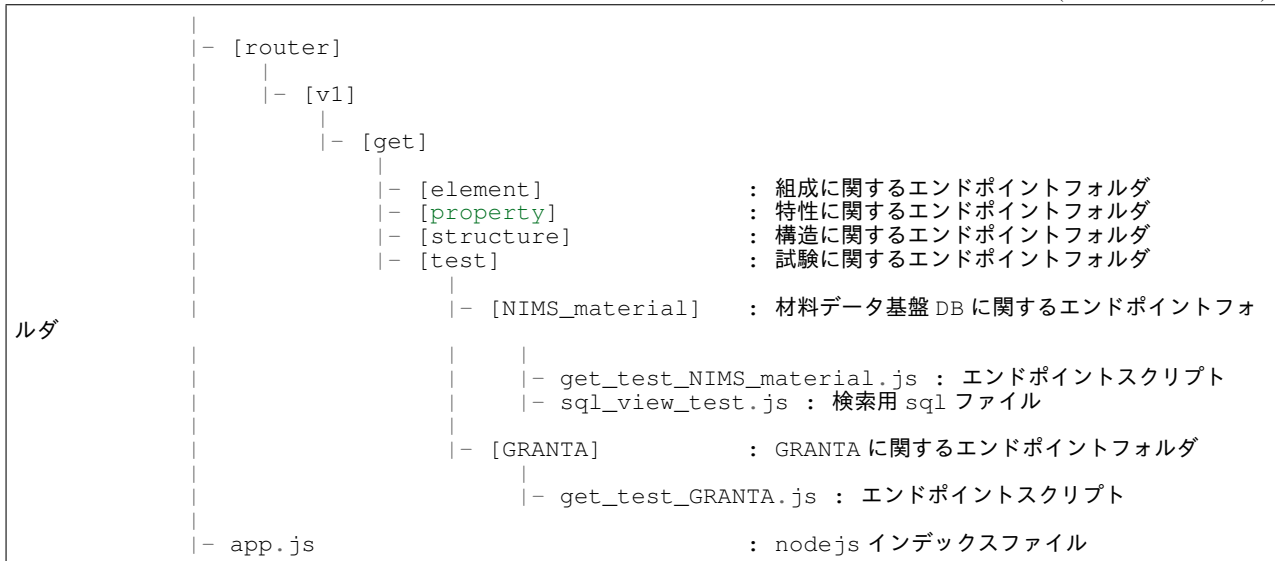
## 2.4 DB-API nodejs 構成

DB-API の nodejs コード構成ファイルは以下。

[nodejs]	
- [DBAPI]	
- [app]	
- [db]	: DB に関するソースフォルダ
- [granta]	: GRANTA に関するソースフォルダ
- dic_dblist.js	: 検索対象の DB のリスト定義ファイル
- Mysql.js	: mysql のモジュール定義ファイル
- Postgresql.js	: postgresql のモジュール定義ファイル

(次のページに続く)

(前のページからの続き)



- \* DB に関する情報は [db] フォルダ以下に格納する。
- \* エンドポイントに関するコードは [get] フォルダ以下にサブフォルダを切って配置する。
- \* routing 情報を app.js に記述する。

## 2.5 nodejs 編集

### 2.5.1 関連ファイルデプロイ

検索処理を登録するためのエンドポイントを作成する。

上記 nodejs 構成にて、以下のフォルダを作成する。

- (1) nodejs/DBAPI/app/db/granta
- (2) nodejs/DBAPI/app/router/v1/get/test/GRANTA

作成したフォルダにそれぞれ以下ファイルを配置する。

- (1) 関連ソースの GRANTA スクリプトファイル群
- (2) 関連ソースの DB-API にある get\_test\_GRANTA.js

その後、以下ファイルを編集する。

```

$ vi ~/nodejs/DBAPI/app/app.js
> (63 行目以降) 追記
> var getTest_GRANTA = require(router_get_path +
>   '/test/GRANTA/get_test_GRANTA.js');
>
> // routing
> app.use(rooturl + '/test/GRANTA/', getTest_GRANTA);
  
```

- \* getTest\_GRANTA にエンドポイントのパスとスクリプトを記述。
- \* app.use にて追加したエンドポイントスクリプトを定義する。

```
$ vi ~/nodejs/DBAPI/app/db/granta/searchGRANTA.conf
> [authorize]
> user=hogehoge
> pass=xxxx
```

- GRANTA の登録ユーザーについて、ユーザー名とパスワードを指定する。
- 詳細は GRANTA 管理者に確認すること。

## 2.6 サービス再起動

nodejs のコード修正、ファイル追加等を行った場合、サービスを再起動する。

```
# systemctl restart nodejs
```

## 2.7 動作確認

ブラウザを起動し、以下 url をリクエストする。

```
http://<db-api サーバ>/db-api/v1/get/test/GRANTA/?mimetype=csv&test=tensile_test
```

DB 検索結果が表示されれば OK。

注釈: GRANTA 検索については、検索処理に数分かかる。

## 2.8 参考

### 2.8.1 GRANTA 検索スクリプト

GRANTA における検索ルールは、以下のファイルにて設定している。

(granta\_schema.csv)

	Powder Metallurgy	MI-21
out_format	Design_data	Materials
Ni[%]	Ni (nickel)	Ni (Nickel)
Al[%]	Al (aluminum)	Al (Aluminium)
...(省略)...		
Tensile_strength[MPa]	Ultimate tensile strength	Ultimate Tensile Strength (UTS)
...(省略)...		

- \* 1 列目の out\_format 列以下に出力するカラム名とその単位をカッコ入力で表記する。
- \* 2 列目以降に検索対象の DB 名、テーブル名、出力カラム名に該当する属性名を記述する。
- \* 該当属性がない場合は値を空にする。
- \* 2 列目以降の検索対象 DB(テーブル) は複数指定可能。
- \* 検索対象データは、GRANTA スカラ型に限定される (グラフデータ、ファイルは非対応)