

ConMas i-Reporter

ConMas Gateway インストール手順書

Windows Server、Windows10、Windows11 環境

全体の流れ

確認と準備

- ConMas Manager をV7.0以上（推奨は最新版）にアップデートしてください。（バージョンアップの場合）
【ソフトウェアダウンロードページ】

<https://cimtops-support.com/i-Reporter/ja/dl-software>

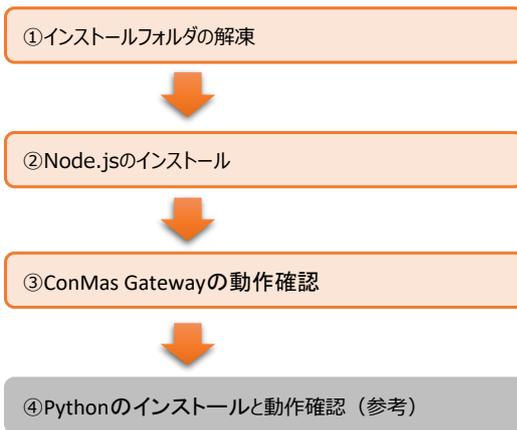
- ConMas Gateway のダウンロードとインストール
【ソフトウェアダウンロードページ】

<https://cimtops-support.com/i-Reporter/ja/dl-software>

【マニュアル】

https://cimtops-support.com/i-Reporter/ir_manuals/ip/gateway-iot/ConMasGateway_Manual.pdf

環境構築作業



① インストールフォルダの配置

ダウンロードした[ConMasGatewayXX.XX.XXXX.ZIP]を任意の場所にコピーして解凍してください。

XX.XX.XXXXにはバージョンが入ります

本手順書ではD:¥に配置した場合で説明します。

解凍したフォルダ内の¥ConMas¥gatewayフォルダをD:¥にコピーします。(例：D:¥ConMas¥gateway)

※解凍したフォルダ内の¥ConMas¥GatewayServer_Readme.txtも本手順書と合わせて確認ください。

上記ファイル内のアップデート手順に関する記述は、直前のバージョンとの差分になっております。



② Node.jsのインストール

ZIPを解凍した ¥ Toolsフォルダ内、以下のファイルを実行し、インストールします。

※2.3.240200 版から Node.js v20 を使用しています。

旧バージョンをご利用の場合は、v20への更新をお願いします。(旧バージョンのアンインストールは不要です。)

ConMas Gateway 導入済の方は、ConMas Gateway (conmas-gateway サービスまたはコマンドプロンプト起動) を停止してから行い完了後に再起動してください。

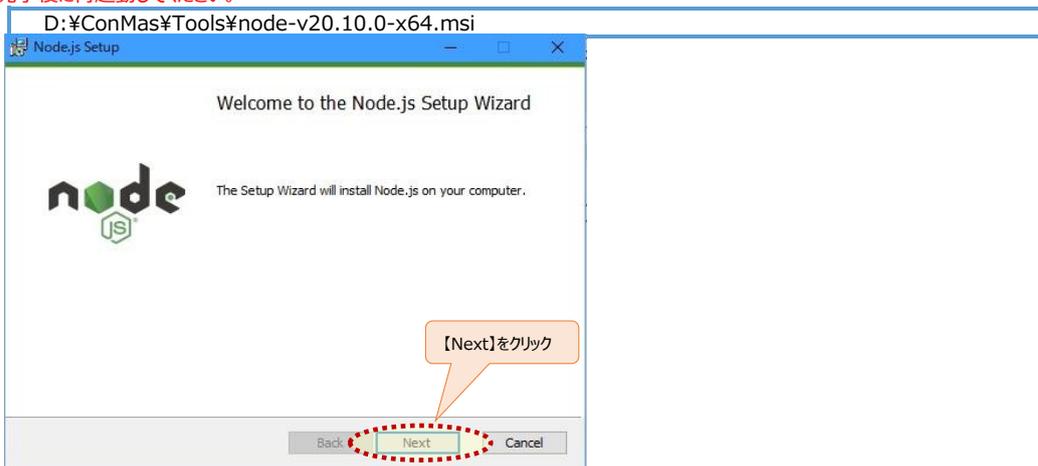


Fig.2-1

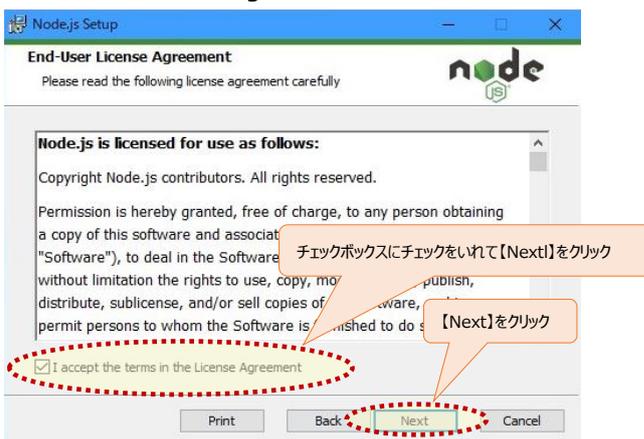


Fig.2-2

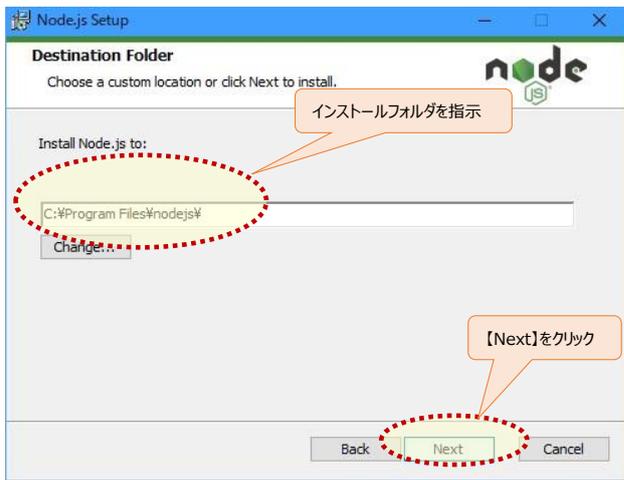


Fig.2-3

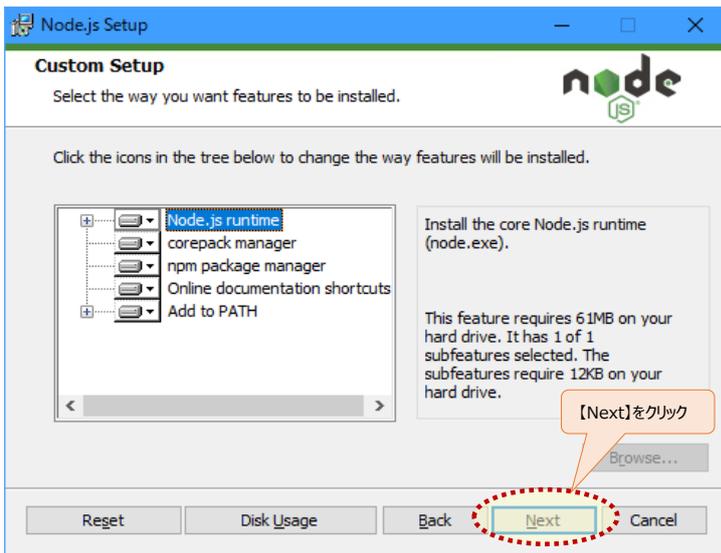


Fig.2-4

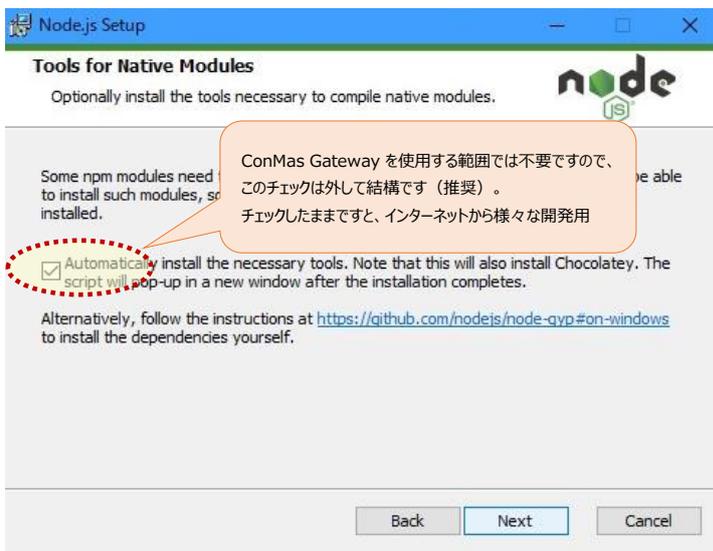


Fig.2-5

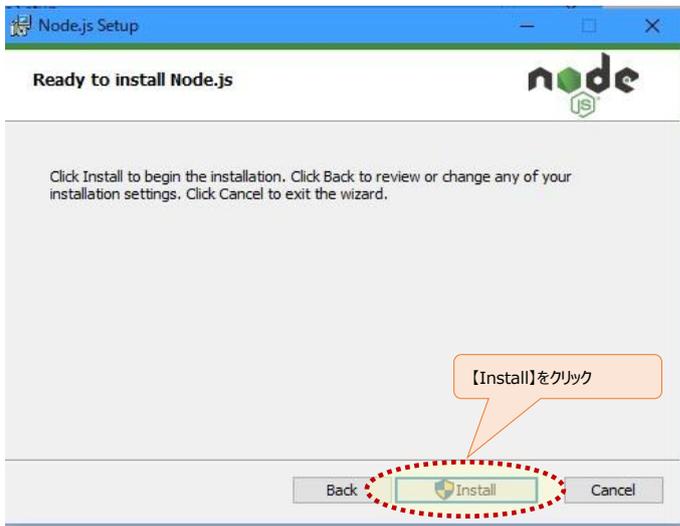


Fig.2-6

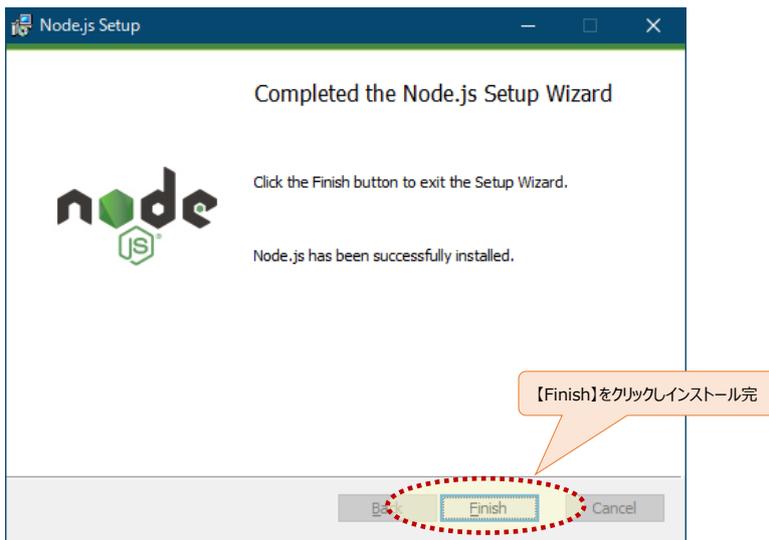


Fig.2-7

③ConMas Gatewayの動作確認

ConMas Gatewayを起動します。

コマンドプロンプトで[¥ConMas¥gateway]フォルダに移動し、以下のコマンドを実行します。

```
> node index.js
```

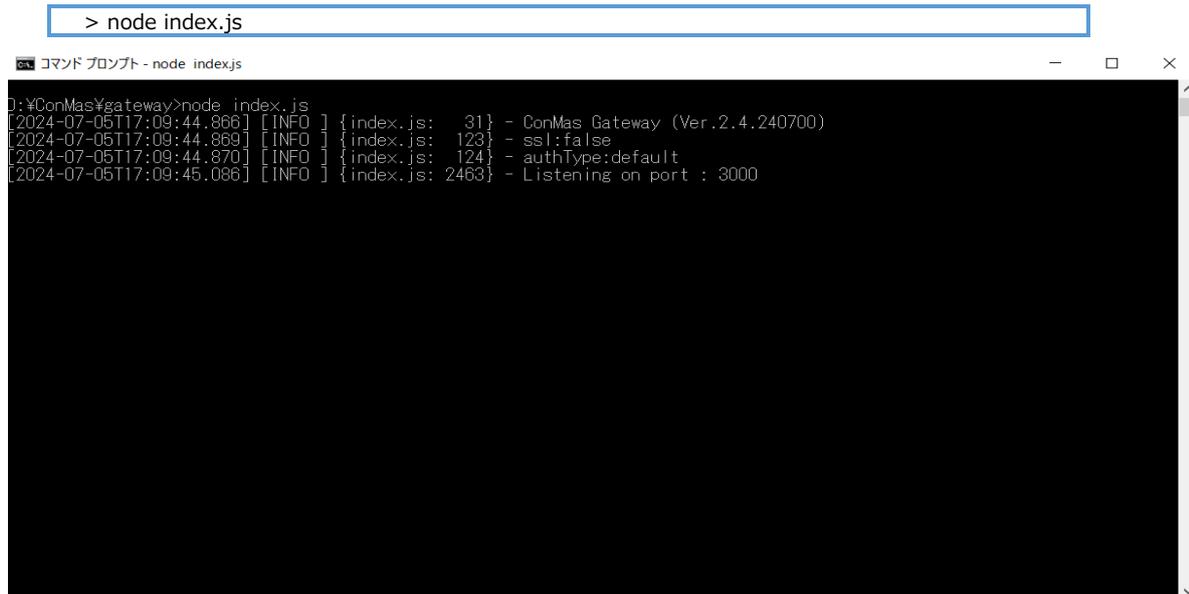


Fig.3-1

(終了するには「Ctrl+C」を入力します)

●WebAPI接続動作確認 (※WebAPI接続を行う場合のみ実施します)

[PowerShell] API実行確認コマンド

【スタート】→【Windows PowerShell】→【Windows PowerShell】よりPowerShellを起動します。

```
Invoke-WebRequest http://localhost:3000/api/v1/getvalue/test? -Headers @{Authorization="Bearer 39aulp41gwcj50"}
```

コマンドを実行します。

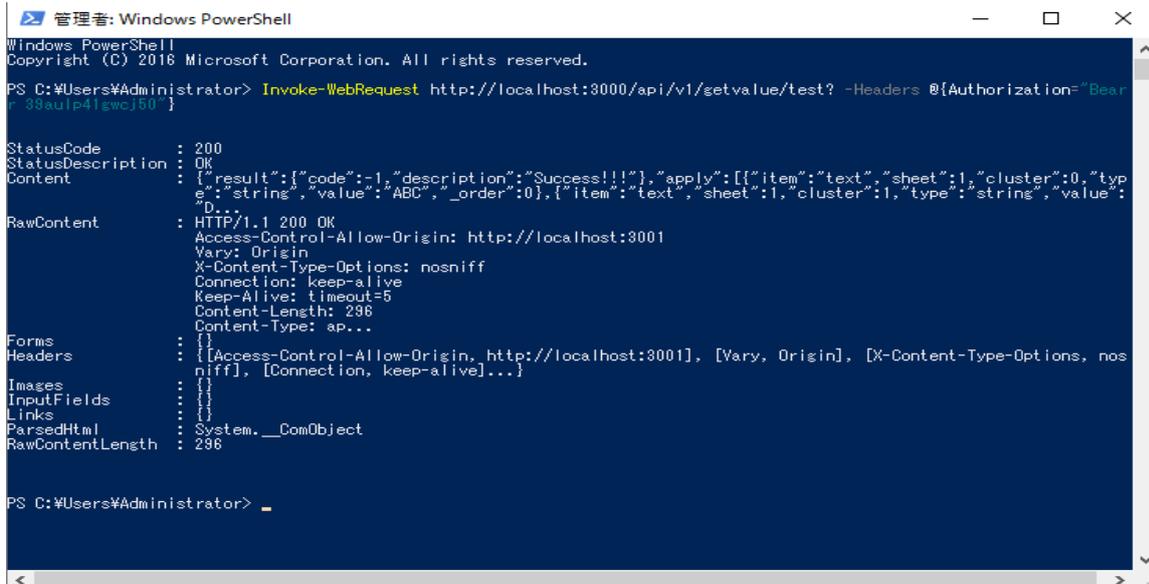


Fig.3-2

上記応答があれば正常です。

Windowsファイアウォールで、ConMasGatewayの受信ポートを許可します。(デフォルトは3000番)

任意のポート番号で利用する場合は以下の設定を変更後、任意のポートをファイアウォールで許可します。

[¥ConMas¥gateway¥config¥default.json]

port: 3000,

● **PostgreSQL接続動作確認（※PostgreSQL接続を行う場合のみ実施します）**

[¥ConMas¥gateway¥config¥datasources¥irepodb.json]ファイルをメモ帳、または、テキストエディタ等で開き、DB接続情報を設定します。

※確認用サンプルではi-Reporterデータベースのユーザーマスターに接続します。

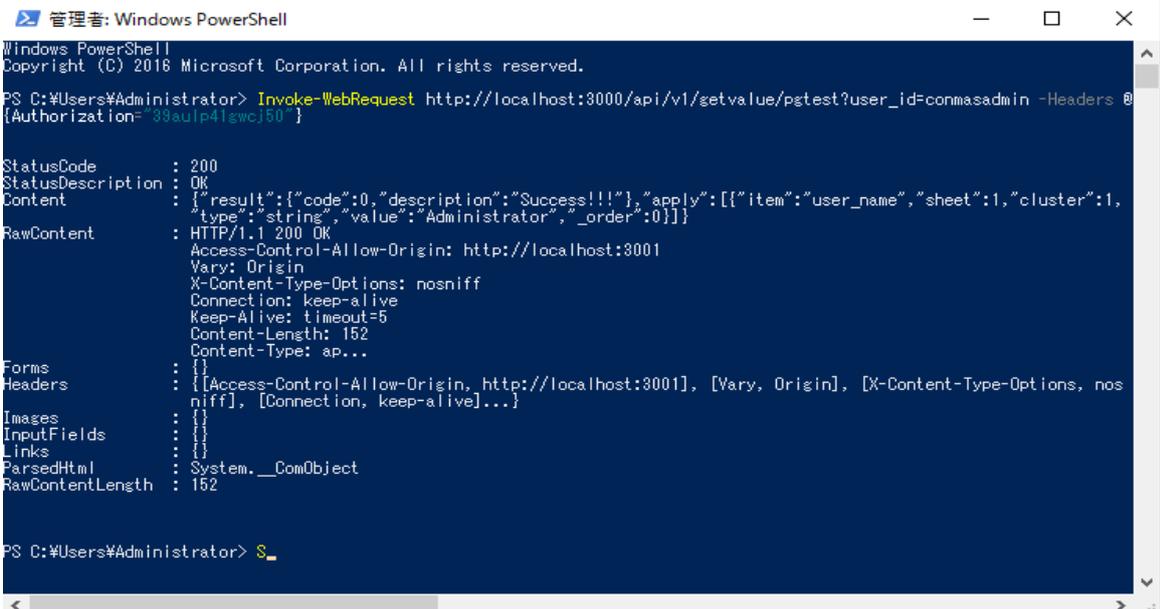
接続情報を環境に合わせて修正するにはirepodb.json内のhost, user, password, database, portの値を変更してください。

※**default.jsonファイルを編集した場合にはConMas Gatewayの再起動が必要です。**

【スタート】→【Windows PowerShell】→【Windows PowerShell】よりPowerShellを起動します。

```
Invoke-WebRequest http://localhost:3000/api/v1/getvalue/pgtest?user_id=conmasadmin -Headers @{Authorization="39aulp41gwcj50"}
```

コマンドを実行します。



```
管理: Windows PowerShell
Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\Users\Administrator> Invoke-WebRequest http://localhost:3000/api/v1/getvalue/pgtest?user_id=conmasadmin -Headers @{Authorization="39aulp41gwcj50" }

StatusCode      : 200
StatusDescription : OK
Content         : [{"result":{"code":0,"description":"Success!!!"},"apply":[{"item":"user_name","sheet":1,"cluster":1,"type":"string","value":"Administrator","_order":0}]}]
RawContent      : HTTP/1.1 200 OK
                  Access-Control-Allow-Origin: http://localhost:3001
                  Vary: Origin
                  X-Content-Type-Options: nosniff
                  Connection: keep-alive
                  Keep-Alive: timeout=5
                  Content-Length: 152
                  Content-Type: ap...
Forms           : {}
Headers         : {[Access-Control-Allow-Origin, http://localhost:3001], [Vary, Origin], [X-Content-Type-Options, nosniff], [Connection, keep-alive]...}
Images          : {}
InputFields     : {}
Links           : {}
ParsedHtml      : System.__ComObject
RawContentLength : 152

PS C:\Users\Administrator> S_
```

Fig.3-3

上記応答があれば正常です。

●SQL Server接続動作確認（※SQL Server接続を行う場合のみ実施します）

SQL Server に対し接続を行うには、下記に記したサーバー側の設定変更が必要となりますので、ご注意ください。

1. 認証モードをSQL Server認証に変更してください。（Windows認証は未対応です）
2. SQL Serverへのログインユーザーとデータベースユーザーを作成する
3. TCP/IPプロトコルの有効化
4. ポートの固定・開放
5. SQL Server Browser サービスの有効化

上記の詳細は SQL Server の公式サイトなどをご確認ください。

[¥ConMas¥gateway¥config¥datasources¥gwdb.json]ファイルをメモ帳、または、テキストエディタ等で開き、DB接続情報を設定します。

※確認用サンプルではテスト用にgwdbデータベースを作成し、その上にplcdata1テーブルを作成した上で行っていきます。

またgwdb.json内の host, user, password などの値を修正し、接続テスト用に使用できます。

※SQL Server 認証にのみ対応しております。

※接続できない場合、デフォルトインスタンスを使用している可能性があるため、不要であるinstanceName行を削除してください。

※default.jsonファイルを編集した場合にはConMas Gatewayの再起動が必要です。

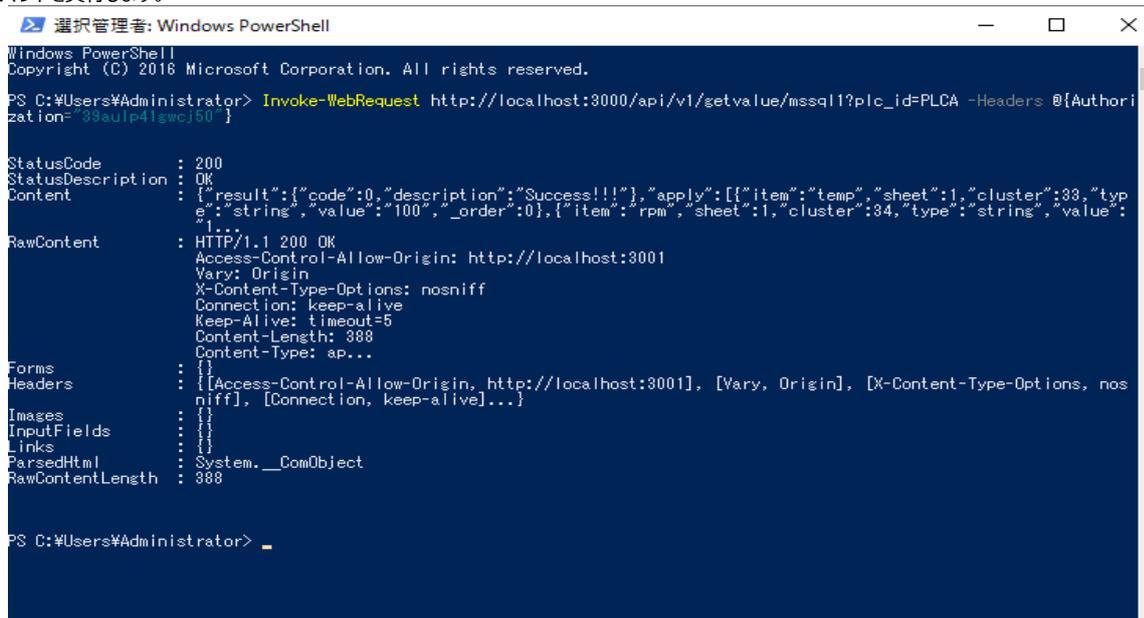
SQL Server Management Studio などで、テーブルを作成するための gwdb データベースを作成します。

[¥ConMas¥samples¥samplepy_graph¥db¥mssql_table.sql]ファイルをDB上で実行し、plcdata1テーブルを作成します。

[スタート]→[Windows PowerShell]→[Windows PowerShell]よりPowerShellを起動します。

```
Invoke-WebRequest http://localhost:3000/api/v1/getvalue/mssql1?plc_id=PLCA -Headers @{Authorization="39aulp41gwcj50"}
```

コマンドを実行します。



```
Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\Users\Administrator> Invoke-WebRequest http://localhost:3000/api/v1/getvalue/mssql1?plc_id=PLCA -Headers @{Authorization="39aulp41gwcj50"}

StatusCode      : 200
StatusDescription : OK
Content          : [{"result":{"code":0,"description":"Success!!!"},"apply":[{"item":"temp","sheet":1,"cluster":33,"type":"string","value":"100","_order":0},{"item":"rpm","sheet":1,"cluster":34,"type":"string","value":
1
RawContent      : HTTP/1.1 200 OK
                  Access-Control-Allow-Origin: http://localhost:3001
                  Vary: Origin
                  X-Content-Type-Options: nosniff
                  Connection: keep-alive
                  Keep-Alive: timeout=5
                  Content-Length: 388
                  Content-Type: ap...
Forms           :
Headers         : [{"Access-Control-Allow-Origin, http://localhost:3001}, [Vary, Origin], [X-Content-Type-Options, nosniff], [Connection, keep-alive]...}
Images          :
InputFields     :
Links           :
ParsedHtml      : System.__ComObject
RawContentLength : 388

PS C:\Users\Administrator> _
```

Fig.3-4

上記応答があれば正常です。

●Oracle接続動作確認（※Oracle接続を行う場合のみ実施します）

[¥ConMas¥gateway¥config¥datasources¥oragwdb.json]ファイルをメモ帳、または、テキストエディタ等で開き、DB接続情報を設定します。

※確認用サンプルではテスト用にplcdata1テーブルを作成した上で行っていきます。

またoragwdb.json内の host, user, password, connectionString などの値を修正し、接続テスト用に使用できます。

※接続用クライアント Oracle Instant Clientのインストールが必要です。Oracle 18c Express Edition には含まれていません。

※Oracle Instant Clientへのパスは、サービス実行ユーザー(Local System Account)が参照できるように、システム環境変数へ追加してください。

※対応 Oracle DBMS サーバー: Oracle 10g(10.2.0.2)以降

※必要な Oracle クライアント: Oracle Instant Client 11.2 以降

※Oracleによれば、Windows11ではOracle Instant Client12.3以上を動作サポートしており、

そのクライアントに対し、サーバー側はOracle 19c(19.13)以降において動作確認しているとのこと。

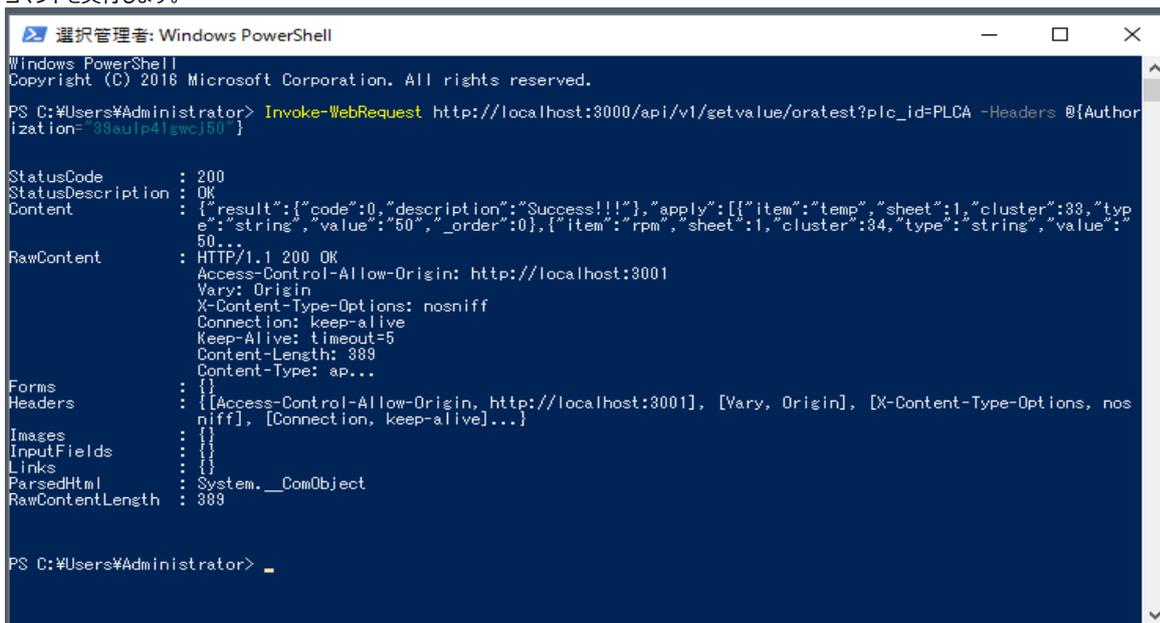
※default.jsonファイルを編集した場合にはConMas Gatewayの再起動が必要です。

[¥ConMas¥samples¥samplepy_graph¥db¥oracle_table.sql]ファイルをDB上で実行し、plcdata1テーブルを作成します。

[スタート]→[Windows PowerShell]→[Windows PowerShell]よりPowerShellを起動します。

```
Invoke-WebRequest http://localhost:3000/api/v1/getvalue/oratest?plc_id=PLCA -Headers @{Authorization="39aulp41gwcj50"}
```

コマンドを実行します。



```
Windows PowerShell
Copyright (C) 2018 Microsoft Corporation. All rights reserved.

PS C:\Users\Administrator> Invoke-WebRequest http://localhost:3000/api/v1/getvalue/oratest?plc_id=PLCA -Headers @{Authorization="39aulp41gwcj50"}

StatusCode      : 200
StatusDescription : OK
Content         : [{"result":{"code":0,"description":"Success!!!"},"apply":[{"item":"temp","sheet":1,"cluster":33,"type":"string","value":"50","_order":0}, {"item":"rpm","sheet":1,"cluster":34,"type":"string","value":"50..."}]}]
RawContent      : HTTP/1.1 200 OK
                  Access-Control-Allow-Origin: http://localhost:3001
                  Vary: Origin
                  X-Content-Type-Options: nosniff
                  Connection: keep-alive
                  Keep-Alive: timeout=5
                  Content-Length: 389
                  Content-Type: ap...
Forms           : {}
Headers         : [{"Access-Control-Allow-Origin, http://localhost:3001}, [Vary, Origin], [X-Content-Type-Options, nosniff], [Connection, keep-alive]...}
Images          : {}
InputFields     : {}
Links           : {}
ParsedHtml      : System.__ComObject
RawContentLength : 389

PS C:\Users\Administrator> _
```

Fig.3-5

上記応答があれば正常です。

● **MySQL接続動作確認 (※MySQL接続を行う場合のみ実施します)**

[¥ConMas¥gateway¥config¥datasources¥mysqlgwdb.json]ファイルをメモ帳、または、テキストエディタ等で開き、DB接続情報を設定します。

※確認用サンプルではテスト用にgwdbデータベースを作成し、その上にplcdata1テーブルを作成した上で行っています。

またmysqlgwdb.json内の user, password, host, port, database の値を修正し、接続テスト用に使用できます。

※動作確認 MySQL DBMS サーバー: MySQL 8.0以降

MySQL Command Line Client などで、テーブルを作成するための gwdb データベースを作成します。

[¥ConMas¥samples¥samplepy_graph¥db¥mysql_table.sql]ファイルをDB上で実行し、plcdata1テーブルを作成します。

[スタート]→[Windows PowerShell]→[Windows PowerShell]よりPowerShellを起動します。

```
Invoke-WebRequest http://localhost:3000/api/v1/getvalue/mysqltest?plc_id=PLCA -Headers @{Authorization="39aulp41gwcj50"}
```

コマンドを実行します。

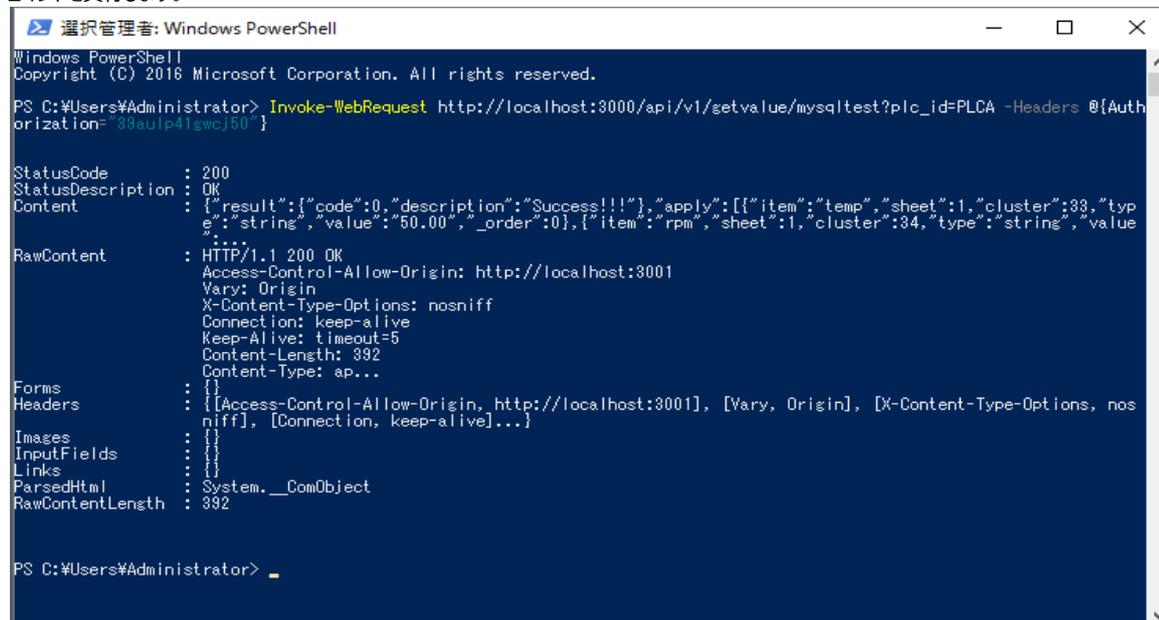


Fig.3-6

上記応答があれば正常です。

●サービス登録/解除 (※Windows)

ConMas Gatewayをサービス登録します。

※プロンプトでConMasGatewayが起動中の場合は「Ctrl+C」で事前に停止しておきます。

コマンドプロンプトを管理者権限で起動します。

コマンドプロンプトで「%ConMas%gateway」フォルダに移動し、以下のコマンドを実行します。

```
> npm run install-service
```

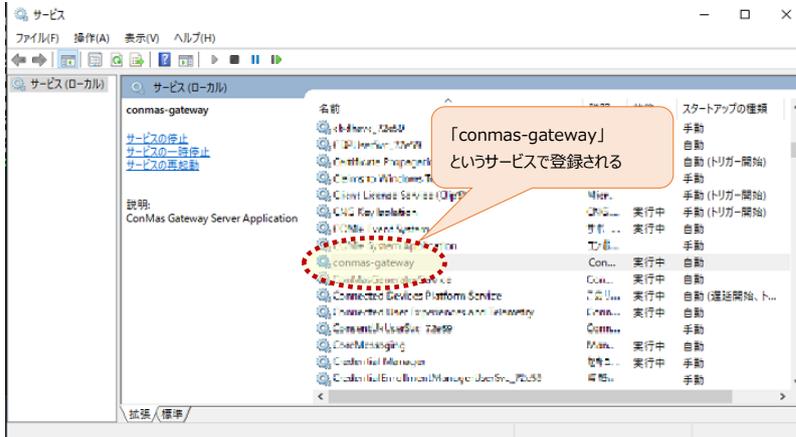


Fig.3-7

※ 連携先でセキュリティーのアクセス権が必要な場合は、【conmas-gateway】サービスのログオンアカウントを適切なアカウントに変更をお願いします。
例：Pythonスクリプト内のコードで共有フォルダなどへアクセスする場合など。

サービス登録を解除する際は、以下のコマンドを実行します。

```
> npm run uninstall-service
```

●ログの確認

ログ出力には「log4js」を使用しています。

<https://github.com/log4js-node/log4js-node>

《フォルダー構成例》	
D:¥	
└─ ConMas	
├─ gateway	
│ └─ actions アクションファイル格納フォルダー
│ └─ config 設定ファイル格納フォルダー
│ │ └─ log4js.json log4js設定ファイル
│ └─ ex Pythonスクリプト連携サンプル利用フォルダー
│ └─ logs ログ格納フォルダー
│ │ └─ access.log アクセスログ
│ │ └─ system.log システムログ
│ └─ modules 共通モジュール格納フォルダー
│ └─ node_modules ライブラリ
│ └─ scripts Python サンプルスクリプト
│ └─ sql SQL格納フォルダ
│ └─ uploads POSTリクエストされた画像ファイル格納フォルダ
│ └─ stock_ex Pythonスクリプト連携サンプル利用フォルダ
└─ samples サンプル
└─ Tools ツール
└─ updater バージョンアップ用ツール

「システムログ」にコンソール出力と同様のログ情報が出力されます。

サービス化後はログファイルで動作状況の確認を行います。

④Pythonのインストールと動作確認 (参考)

- ※ Python環境については、基本的にはお客様側に必要な環境を、Gatewayと同じマシンに構築していただけます。本資料のインストール手順はPython環境を構築する一例です。サンプルについては、Python3.10で動作確認しております。

以前、PATHに Python 用のパスを追加（【Add Python x.x to Path】にチェック） するようにお伝えしましたが、それは一旦アンインストールして頂きたいです。
PATHに加え、Py ランチャーに実行ファイルの選択を任せられた方が、環境としての自由度が高いことが判明したからです。
また同じ理由から、システム環境変数に PYTHONPATH を追加していれば、それも削除して頂きたいです。

Python のダウンロードは、<https://www.python.org/downloads/> から可能です。

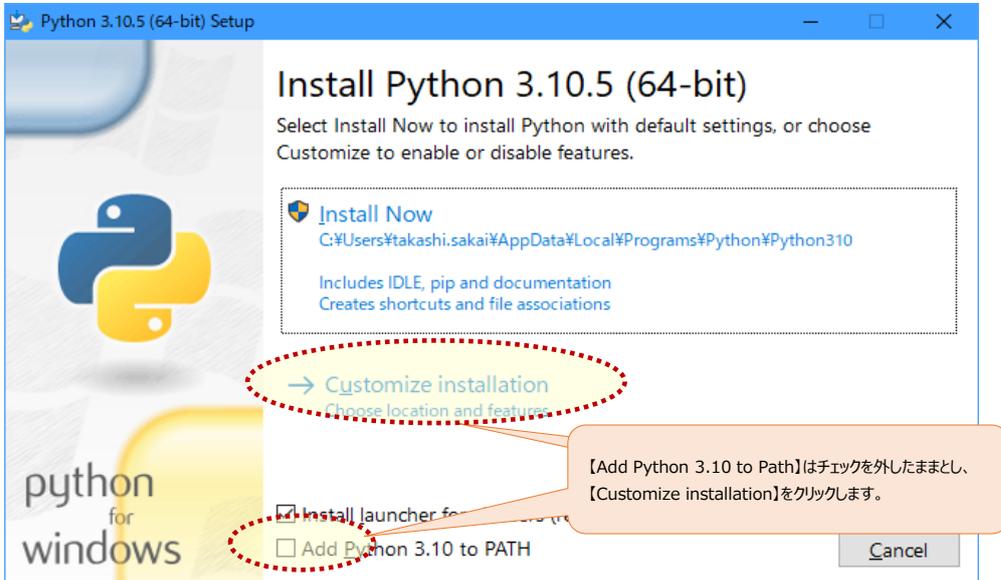


Fig.4-1

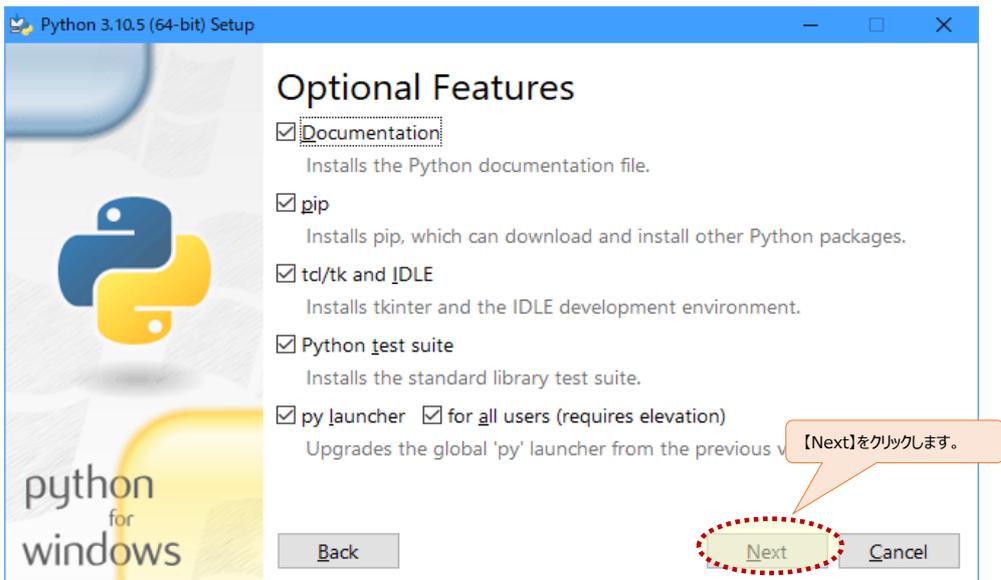


Fig.4-2

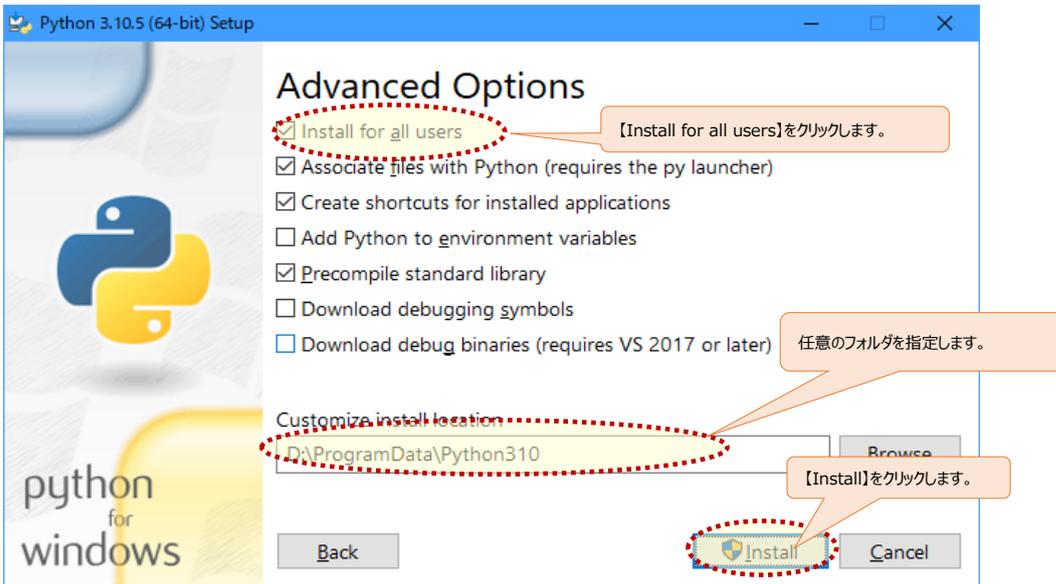


Fig.4-3

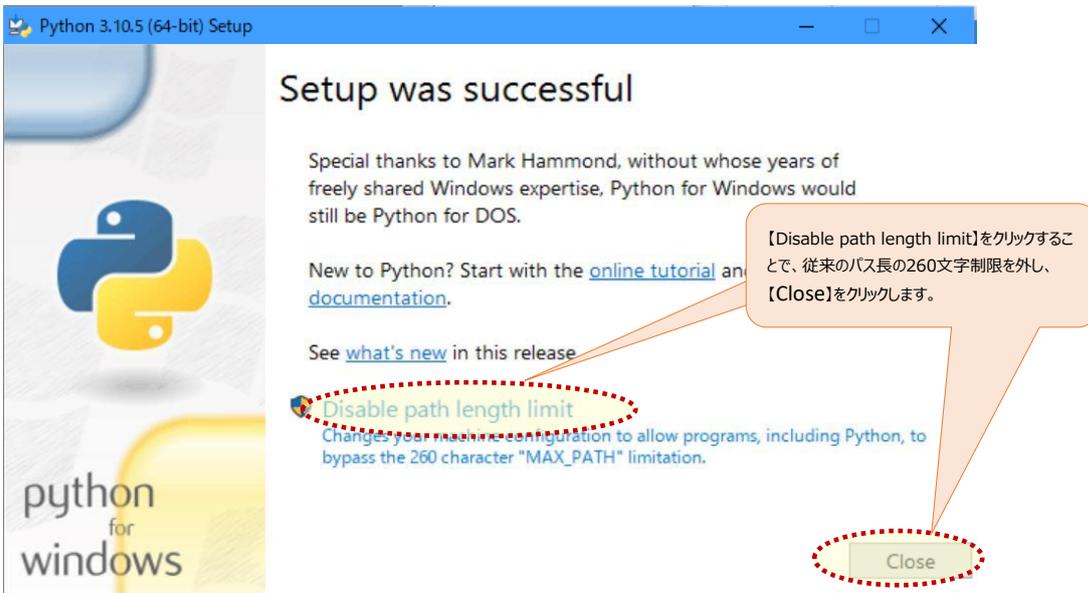


Fig.4-4

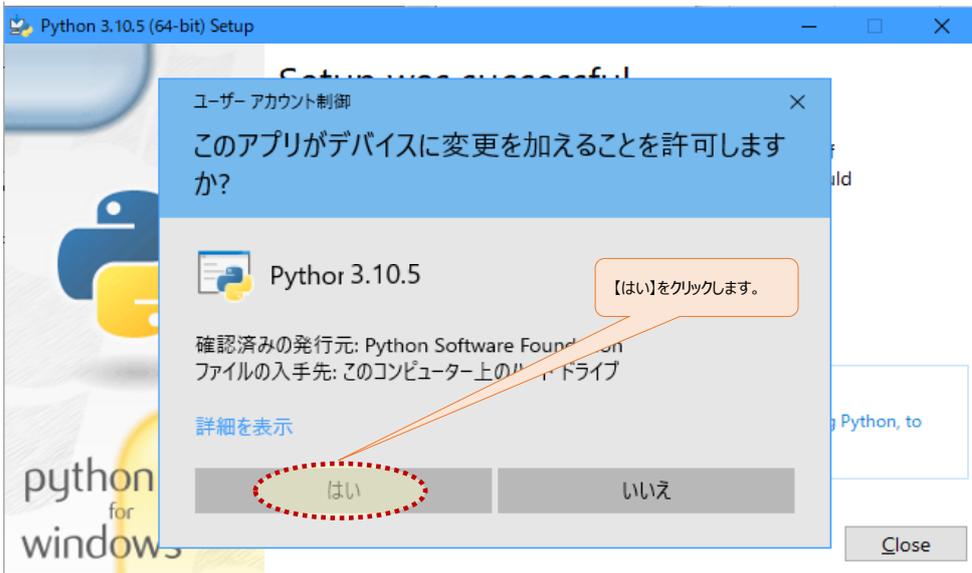


Fig.4-5

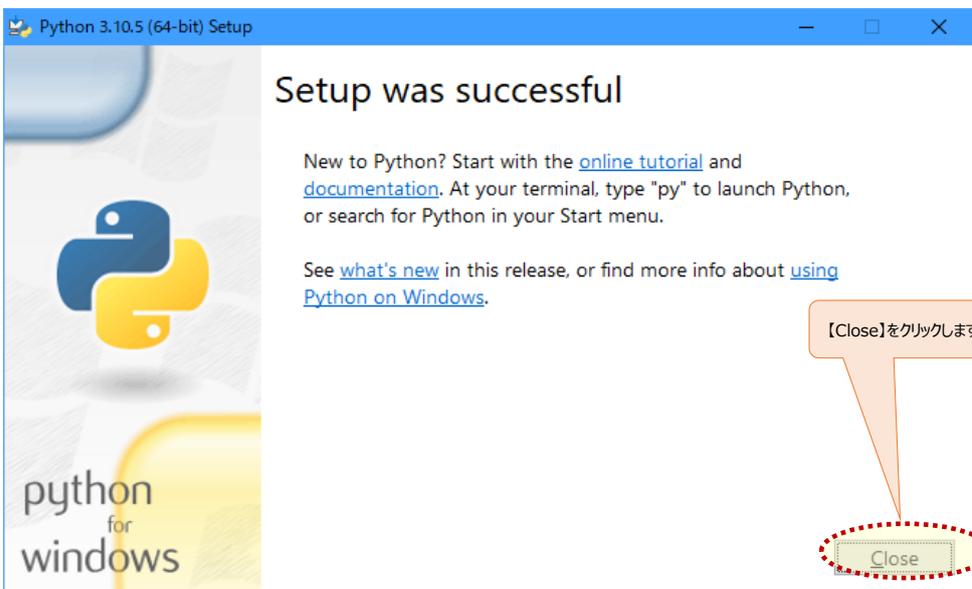


Fig.4-6

コマンドプロンプトを起動しPython インストール先を確認します。(py --list-paths)

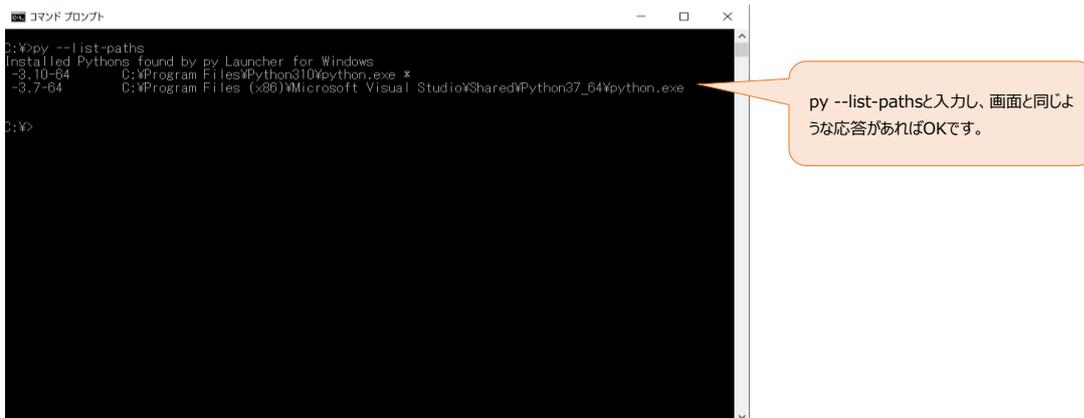


Fig.4-7

● Python動作確認 (※Pythonスクリプトを使用する場合のみ実施します)

動作確認用のPythonサンプルスクリプトで使用するライブラリをインストールします。

コマンドプロンプトを**管理者権限**で起動して下記コマンドを実行します。

今後、追加されていくサンプルをご使用の際はソースのimport文をご確認の上、必要モジュールをインストールしてください。

```
> py -m pip install matplotlib==3.8.2
> py -m pip install qrcode
> py -m pip install Image
> py -m pip install python-barcode
> py -m pip install pandas
> py -m pip install openpyxl
> py -m pip install requests
> py -m pip install Japanera==1.1.0
> py -m pip install pycopg2
> py -m pip install pyodbc
> py -m pip install cx-Oracle
```

※画面は matplotlib の例です。

※SQL Server接続用にpyodbc(あるいはpymssql)、Oracle接続用にcx_Oracle(あるいはoracledb)が必要です。

※py -m pip list -v コマンドにてインストール先を確認できます。

※モジュールバージョンを指定する場合は、P.18 Pythonモジュールバージョンについて補足説明を参照してください。

なお、タイムアウトエラーが出る場合は、組織内プロキシサーバーを経由していない可能性があります。pip install の オプション --proxy でプロキシ設定をした上で、インストールをお試しください。

例) py -m pip install --proxy http://<user>:<password>@<proxy_host>:<proxy_port> matplotlib

※<>で囲まれたのは変数であり、お使いの環境に合わせ読み替えてください。

それでもインストールできない場合は、"--trusted-host"オプションにファイルダウンロード先を加えてみることを試してください。

詳細については、バージョンなどにより違いが生じる可能性がありますので、ネット上から検索をお願い致します。

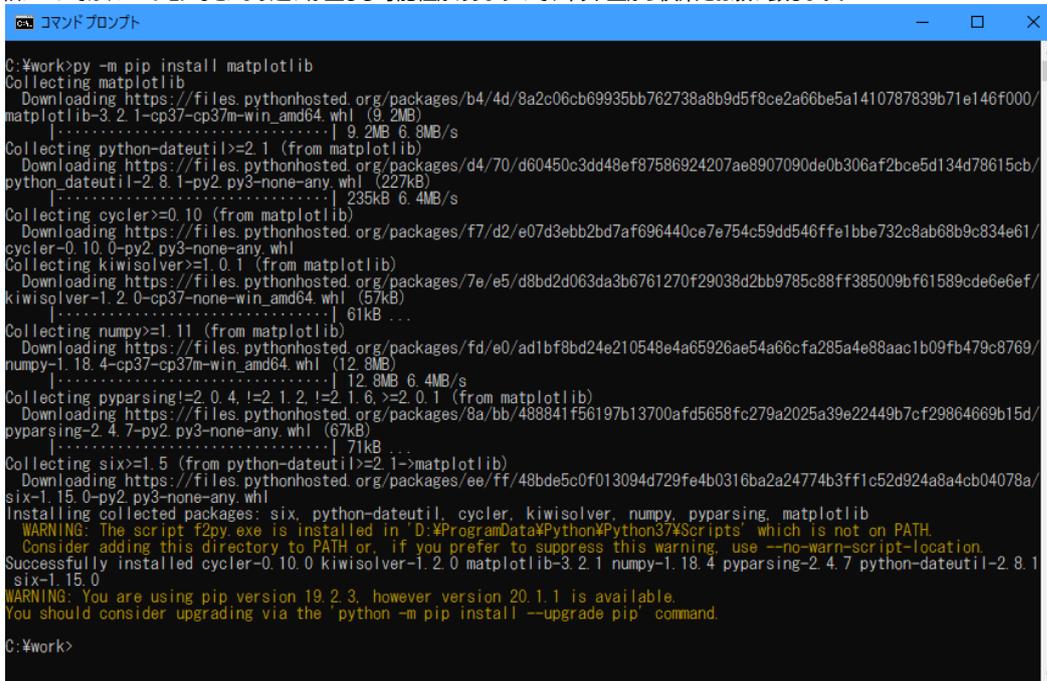


Fig.4-8

● Python動作確認 (※Pythonスクリプトを使用する場合のみ実施します)

Pythonスクリプト動作確認のため、【スタート】→【Windows PowerShell】→【Windows PowerShell】よりPowerShellを起動します。

```
Invoke-WebRequest `
"http://localhost:3000/api/v1/getvalue/chart1?x1=PLC-A&x2=PLC-B&x3=PLC-C&x4=PLC-D&x5=PLC-E&y1=10&y2=20&y3=30&y4=20&y5=10" `
-Headers @{Authorization="39aulp41gwj50"}
```

※行末は ` (バックティク) Shift+@です。

コマンドを実行します。

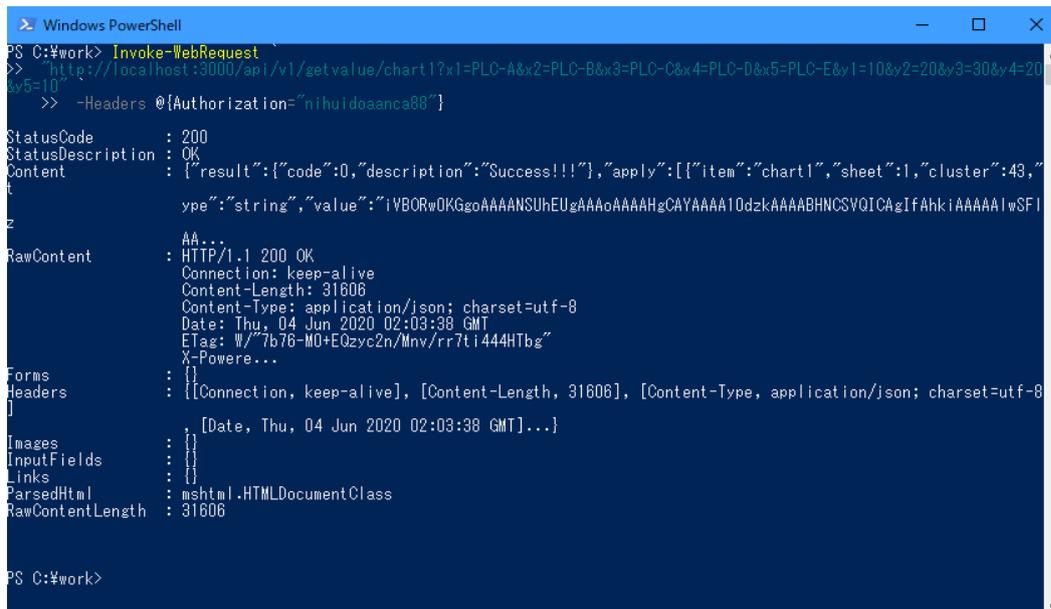


Fig.4-9

上記応答があれば正常です。

● Pythonモジュールバージョンについて補足 (※マニュアルのサンプルスクリプトが正常動作しなかった場合)

モジュール名の後に "==3.8.2" などと付加しているのは、指定バージョンをインストールするようpipコマンドに指示するためです。特定のモジュールではバージョンを指定しない場合、エラーが発生することが確認されており、回避するためにバージョン指定をしています。

すべてのモジュールをバージョン指定で使いたい場合は、下記Pythonサンプルスクリプトの動作確認を行なったモジュールバージョンを参考にしてください。
(※Python 3.10.5との組み合わせで動作確認しております。)

お使いの環境において、弊社提供のサンプルで不具合動作が生じ、その原因がモジュールに起因する疑いがある場合は、インストールされているモジュールバージョンを次のコマンドで出力し、動作確認バージョンと比較いただき確認をお願いします。

```
> py -m pip freeze > requirements.txt
```

上記コマンドにより、現在のディレクトリにrequirements.txtという名前でもジュールリストが出力されます。

【Python3.10.5で動作確認に使用したモジュールバージョン】

```
asgiref==3.8.1
certifi==2024.7.4
charset-normalizer==3.3.2
colorama==0.4.6
contourpy==1.2.1
cx-Oracle==8.3.0
cyclerc==0.12.1
Django==5.0.7
et-xmlfile==1.1.0
fonttools==4.53.1
idna==3.7
image==1.5.33
Japanera==1.1.0
kanjize==1.3.0
kiwisolver==1.4.5
matplotlib==3.8.2
mysql-connector-python==9.0.0
numpy==1.26.4
openpyxl==3.1.5
packaging==24.1
pandas==2.2.2
pillow==10.4.0
psycpg2==2.9.9
pyodbc==5.1.0
pyparsing==3.1.2
pypng==0.20220715.0
python-barcode==0.15.1
python-dateutil==2.9.0.post0
pytz==2024.1
qrcode==7.4.2
requests==2.32.3
six==1.16.0
sqlparse==0.5.0
typing_extensions==4.12.2
tzdata==2024.1
urllib3==2.2.2
```