

```
1  #!/usr/bin/env python
2  # -*- coding: utf-8 -*-
3
4
5  import sys
6  import json
7
8  from fastapi import APIRouter
9  from pydantic import BaseModel
10 from typing import Union, List, Optional
11
12 from LIB import db
13 from LIB import ethereum
14
15 DEBUG = False # True
16
17 router = APIRouter(
18     prefix="/bchain_data",
19     tags=["bchain_data"],
20 )
21
22 # Ethereum へのアクセス情報取得
23 s = db.DB()
24 eth_rec = s.select(db.DsEthereum, "Ethereum")
25 if DEBUG : print("eth_rec=",eth_rec)
26 if eth_rec != None:
27     url = "http://" + eth_rec.address + "://" + str(eth_rec.port)
28     account = eth_rec.account
29     password = eth_rec.password
30     private_key = eth_rec.private_key
31 else:
32     url = "http://192.168.3.3:8545"
33     account = "0x70f9e0445a572d62eabf6d6669f69283558d7e2f"
34     password = "dataprovider"
35     private_key = "c2f7b10487844a8d0f708a2397a63e0aa2b2695351775fe5548557bc71280e28"
36
37
38
39 if DEBUG:
40     print("url = ", url)
```

```
41     print("account = ", account)
42     print("password = ", password)
43     print("private_key = ", private_key)
44
45
46     eth = ethereum.Ethereum(url, account, password, private_key)
47
48     # コントラクト設定
49     func_rec = s.select(db.DsFunction, "MyDataStore")
50     if DEBUG : print("func_rec = ", func_rec)
51     eth.set_contract(func_rec.address, func_rec.abi)
52
53     """
54     Ethereum
55     """
56
57     class DList(BaseModel):
58         timestamp: Optional[str]
59         source: str
60         data: List[dict]
61
62     @router.get("/")
63     async def get_data(channel_id: str, measurement: str, start: int = 0, stop: int = 100):
64         print("get_data", channel_id, measurement, start, stop)
65         """
66         データを取得する
67         """
68         res = eth.get_data(channel_id, measurement, start, stop, None)
69         if DEBUG : print(res)
70
71         return {"Status": "OK", "data":res}
72
73     @router.get("/length")
74     async def get_data_number(channel_id: str, measurement: str):
75         print("get_data_number", channel_id, measurement)
76         """
77         データ数を取得する
78         """
79         res = eth.get_data_number(channel_id, measurement)
80         if DEBUG : print(res)
```

```
81
82     return {"Status": "OK", "Result":res}
83
84
85 @router.post("/{channel_id}/{measurement}")
86 async def put_data(channel_id: str, measurement: str, data : DList):
87     print("put_data:", channel_id, measurement, data)
88     """
89     データを登録する
90     """
91     eth.put_data(channel_id, measurement, data.json())
92     return {"Status": "OK"}
93
94
```