\\athena\IPSA\home\isaori\Desktop\完成版\構成ファイル\dotest.py

```python
 1  # -*- coding:utf-8 -*-
 2  #!/usr/bin/python3
 3  import mariadb
 4  import sys
 5  from bluepy import btle
 6  from omron_env_broadcast import ScanDelegate
 7  import time
 8  import schedule
 9  import slackweb
10  import RPi.GPIO as GPIO
11
12
13  GPIO.setmode(GPIO.BCM)
14  float_SW_GPIO = 4
15
16  GPIO.setup(float_SW_GPIO,GPIO.IN,pull_up_down=GPIO.PUD_DOWN)
17
18  scanner = btle.Scanner().withDelegate(ScanDelegate())
19  scanner.scan(5.0)
20  IotTemperature = scanner.delegate.sensorValue['Temperature']
21  IotHumidity = scanner.delegate.sensorValue['Humidity']
22  IotVoltage = scanner.delegate.sensorValue['BatteryVoltage']
23
24  print("data connect")
25
26  slack_DB_tuuti = "温度：" + str(IotTemperature) + "\n湿度：" + str(IotHumidity) + "\n電池残
    量" + str(IotVoltage)
27
28  slack_SW_tuuti = "水が減っています"
29  slack_SW_tuuti_1 = "水はあります"
30
31  def DBjob():
32      slack = slackweb.Slack(url ="
    https://hooks.slack.com/services/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")
33      print("start")
34      try:
35          # Python Style
36          DBConnector = mariadb.connect(
37          user="ipsa",
38          password="ipsa2221",
39          host="157.13.24.163",
40          port=3306,
41          database="23project_test"
42          )
43          print(f"CONNECT SUCCESSED")
44      except mariadb.Error as e:
45          print(f"CONNECT ERROR: {e}")
46          sys.exit(1)
47
48      # Create DB Object
49      DBobj = DBConnector.cursor()
50
51      # IoT Data
52      scanner = btle.Scanner().withDelegate(ScanDelegate())
53      #スキャンしてセンサ値取得（タイムアウト5秒）
54      scanner.scan(5.0)
55
```

```python
     IotTemperature = scanner.delegate.sensorValue['Temperature']
     IotHumidity = scanner.delegate.sensorValue['Humidity']
     IotVoltage = scanner.delegate.sensorValue['BatteryVoltage']

     # Iot Data to SQL
     nowTemp = str(IotTemperature)
     nowHumit = str(IotHumidity)
     nowVol = str(IotVoltage)

     # SQL Execute
     try:
         DBobj.execute("INSERT INTO test(collecttime,temperature,humidity,BatteryVoltage)
VALUES (NOW(),?,?,?)",
         (nowTemp, nowHumit, nowVol) )
         print("温度")
         print(scanner.delegate.sensorValue['Temperature'])
         print("湿度")
         print(scanner.delegate.sensorValue['Humidity'])
         print("電池残量")
         print(scanner.delegate.sensorValue['BatteryVoltage'])
         print("-------------------")
     except mariadb.Error as e:
         print(f"SQL execute error: {e}")
     # Commit DB
     DBConnector.commit()

     # Close DB Connection
     DBConnector.close()
     slack_DB_tuuti = "DBに値送信しました\n温度：" + str(IotTemperature) + "\n湿度：" +
str(IotHumidity) + "\n電池残量" + str(IotVoltage)
     slack.notify(text = slack_DB_tuuti)

def SWjob():
     slack = slackweb.Slack(url ="
https://hooks.slack.com/services/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")
     time.sleep(1)
     float_switch_status = GPIO.input(float_SW_GPIO)

     if float_switch_status == 0:
         print("水ない")
         slack.notify(text = slack_SW_tuuti)
         print("水不足通知送信")
     else:
         print("水ある")
         slack.notify(text = slack_SW_tuuti_1)
         print("水存在通知送信")

try:
     while True:
         print("hello")
         schedule.every().day.at("09:00").do(DBjob)
         schedule.every().day.at("12:00").do(DBjob)
         schedule.every().day.at("16:00").do(DBjob)

         schedule.every().day.at("16:52").do(DBjob)
         schedule.every().day.at("16:52").do(SWjob)

         schedule.every().day.at("09:10").do(SWjob)
         schedule.every().day.at("12:10").do(DBjob)
         schedule.every().day.at("16:10").do(DBjob)
```

```
114          while True:
115              schedule.run_pending()
116              time.sleep(5)
117
118  except Exception as err_txt:
119      print("----- Error! -----")
120      pass
```