

```
1 #-----
2 # Name:          This script is for reading and writing data from Arduino to
  Blockchain
3 # Author:       Rejith Reghunathan
4 # Created:      18/04/2022
5 # Licence:      MIT
6 #-----
7
8 from brownie import accounts, config, network, IOT_Web3
9 from datetime import datetime
10 import time
11 from typing import Iterator
12 from pyfirmata import Arduino,util
13 import os.path
14
15 def read_arduino(channel):
16     # Reading data from Arduino Channel 01
17     time.sleep(20)
18     board = Arduino('/dev/ttyACM0')
19     Iterator = util.Iterator(board)
20     Iterator.start()
21     ai_channel = board.get_pin(channel)
22     time.sleep(20)
23     ai_voltage_integer = int(ai_channel.read()*1000)
24     time.sleep(20)
25     print (ai_voltage_integer)
26     return ai_voltage_integer
27
28
29 def read_contract():
30     account = get_account()
31     if(network.show_active() == "development"):
32         IOT_Web3.deploy({"from": account})
33
34     #Writing backup
35     directory = '/home/rejithr/Desktop/IoT/IOT_Web3_Storage'
36     filename = "eth_backup.txt"
37     file_path = os.path.join(directory, filename)
38     if not os.path.isdir(directory):
39         os.mkdir(directory)
40
41
42
43     # Sensor_01
44     ai_voltage_integer_01 = read_arduino('a:0:i')
45     tagname_01 = "TEMP_SENSOR_001"
46     time_stamp_01 = datetime.now().strftime("%m/%d/%Y,%H:%M:%S")
47     tagname_01_unit = "degC"
48     iot_data_storage = IOT_Web3[-1]
49     transaction = iot_data_storage.create_iot_data(tagname_01,
50 time_stamp_01, round(ai_voltage_integer_01/2.048), tagname_01_unit, {"from":
51 account})
52     transaction.wait(1)
53
54     # Sensor_02
55     ai_voltage_integer_02 = read_arduino('a:1:i')
56     tagname_02 = "VIBR_SENSOR_001"
```

```
55     time_stamp_02 = datetime.now().strftime("%m/%d/%Y,%H:%M:%S")
56     tagname_02_unit = "Hz"
57     iot_data_storage = IOT_Web3[-1]
58     transcation = iot_data_storage.create_iot_data(tagname_02,
time_stamp_02,round(ai_voltage_integer_02),tagname_02_unit,{"from":
account})
59     transcation.wait(1)
60
61
62     iot_data = iot_data_storage.getiot_datas()
63     print("Eth Data stored to file!")
64     with open(file_path, 'w') as f:
65         for data in iot_data:
66             #print(data)
67             f.write(str(data) + '\n')
68     print("-----\n")
69
70     f.close()
71
72
73
74 def get_account():
75     if(network.show_active() == "ganache-zero"):
76         return accounts[0]
77     else:
78         return accounts.add(config["wallets"] ["from_key"])
79
80
81 def main():
82     while(True):
83         read_contract()
84         print('transcation completed!')
85         time.sleep(1)
86
```