

```

1 import pandas as pd
2 import matplotlib.pyplot as plt
3 import os
4
5 if not os.path.exists('./Output'):
6     os.mkdir('Output')
7
8 if not os.path.exists('./Output/csv'):
9     os.mkdir('./Output/csv')
10
11 if os.path.exists('MESURES.csv'):
12
13     headerName = ["flag", "dt", "gravity_x", "gravity_y", "gravity_z", "gyro_x",
14                   "gyro_y", "gyro_z", "mag_x", "mag_y", "mag_z", "accel_x", "accel_y", "accel_z",
15                   "euler_h", "euler_p", "euler_r", "quater_w", "quater_x", "quater_y", "quater_z"]
16
17     df = pd.read_csv('MESURES.csv', sep=';', index_col=False, names=headerName,
18                     lineterminator='\n')
19
20     df.head()
21     df.to_excel(r'./Output/Mesures.xlsx', index = False, header=True)
22
23     df_t = df.transpose()
24
25     flg = pd.DataFrame(df_t.loc['flag'].to_numpy().reshape(1,-1))
26
27     dt = pd.DataFrame(df_t.loc['dt'].to_numpy().reshape(1,-1))
28
29     gravity_x = pd.DataFrame(df_t.loc['gravity_x'].to_numpy().reshape(1,-1))
30     gravity_y = pd.DataFrame(df_t.loc['gravity_y'].to_numpy().reshape(1,-1))
31     gravity_z = pd.DataFrame(df_t.loc['gravity_z'].to_numpy().reshape(1,-1))
32
33     gyro_x = pd.DataFrame(df_t.loc['gyro_x'].to_numpy().reshape(1,-1))
34     gyro_y = pd.DataFrame(df_t.loc['gyro_y'].to_numpy().reshape(1,-1))
35     gyro_z = pd.DataFrame(df_t.loc['gyro_z'].to_numpy().reshape(1,-1))
36
37     mag_x = pd.DataFrame(df_t.loc['mag_x'].to_numpy().reshape(1,-1))
38     mag_y = pd.DataFrame(df_t.loc['mag_y'].to_numpy().reshape(1,-1))
39     mag_z = pd.DataFrame(df_t.loc['mag_z'].to_numpy().reshape(1,-1))
40
41     accel_x = pd.DataFrame(df_t.loc['accel_x'].to_numpy().reshape(1,-1))
42     accel_y = pd.DataFrame(df_t.loc['accel_y'].to_numpy().reshape(1,-1))
43     accel_z = pd.DataFrame(df_t.loc['accel_z'].to_numpy().reshape(1,-1))
44
45     euler_h = pd.DataFrame(df_t.loc['euler_h'].to_numpy().reshape(1,-1))
46     euler_p = pd.DataFrame(df_t.loc['euler_p'].to_numpy().reshape(1,-1))
47     euler_r = pd.DataFrame(df_t.loc['euler_r'].to_numpy().reshape(1,-1))
48
49     quater_w = pd.DataFrame(df_t.loc['quater_w'].to_numpy().reshape(1,-1))
50     quater_x = pd.DataFrame(df_t.loc['quater_x'].to_numpy().reshape(1,-1))
51     quater_y = pd.DataFrame(df_t.loc['quater_y'].to_numpy().reshape(1,-1))
52     quater_z = pd.DataFrame(df_t.loc['quater_z'].to_numpy().reshape(1,-1))
53
54     flg.to_csv(r'./Output/csv/flag.csv', sep=";", index=False, header=False,
55               line_terminator=None)
56     dt.to_csv(r'./Output/csv/dt.csv', sep=";", index=False, header=False,
57              line_terminator=None)
58
59     gravity_x.to_csv(r'./Output/csv/gravity_x.csv', sep=";", index=False, header=False,
60                      line_terminator=None)
61     gravity_y.to_csv(r'./Output/csv/gravity_y.csv', sep=";", index=False, header=False,
62                      line_terminator=None)
63     gravity_z.to_csv(r'./Output/csv/gravity_z.csv', sep=";", index=False, header=False,
64                      line_terminator=None)
65
66     gyro_x.to_csv(r'./Output/csv/gyro_x.csv', sep=";", index=False, header=False,
67                   line_terminator=None)
68     gyro_y.to_csv(r'./Output/csv/gyro_y.csv', sep=";", index=False, header=False,
69                   line_terminator=None)

```

```

63 gyro_z.to_csv(r'./Output/csv/gyro_z.csv', sep=";", index=False, header=False,
64 line_terminator=None)
65 mag_x.to_csv(r'./Output/csv/mag_x.csv', sep=";", index=False, header=False,
66 line_terminator=None)
67 mag_y.to_csv(r'./Output/csv/mag_y.csv', sep=";", index=False, header=False,
68 line_terminator=None)
69 mag_z.to_csv(r'./Output/csv/mag_z.csv', sep=";", index=False, header=False,
70 line_terminator=None)
71 accel_x.to_csv(r'./Output/csv/accel_x.csv', sep=";", index=False, header=False,
72 line_terminator=None)
73 accel_y.to_csv(r'./Output/csv/accel_y.csv', sep=";", index=False, header=False,
74 line_terminator=None)
75 accel_z.to_csv(r'./Output/csv/accel_z.csv', sep=";", index=False, header=False,
76 line_terminator=None)
77 euler_h.to_csv(r'./Output/csv/euler_h.csv', sep=";", index=False, header=False,
78 line_terminator=None)
79 euler_p.to_csv(r'./Output/csv/euler_p.csv', sep=";", index=False, header=False,
80 line_terminator=None)
81 euler_r.to_csv(r'./Output/csv/euler_r.csv', sep=";", index=False, header=False,
82 line_terminator=None)
83 quater_w.to_csv(r'./Output/csv/quater_w.csv', sep=";", index=False, header=False,
84 line_terminator=None)
85 quater_x.to_csv(r'./Output/csv/quater_x.csv', sep=";", index=False, header=False,
86 line_terminator=None)
87 quater_y.to_csv(r'./Output/csv/quater_y.csv', sep=";", index=False, header=False,
88 line_terminator=None)
89 quater_z.to_csv(r'./Output/csv/quater_z.csv', sep=";", index=False, header=False,
90 line_terminator=None)
91
92 fig, ((grav, gyro, mag), (acc, euler, quater)) = plt.subplots(2, 3)
93 fig.suptitle('Mesures IMU')
94
95 #flag.grid()
96 #flag.plot(df['flag'])
97 #flag.set_ylabel(r'Flag mesure')
98
99 grav.grid()
100 grav.plot(df['gravity_x'], label="X")
101 grav.plot(df['gravity_y'], label="Y")
102 grav.plot(df['gravity_z'], label="Z")
103 grav.legend(loc=1, prop={'size': 7})
104 grav.set_ylabel(r'Gravity  $[m/s^2]$ ')
105
106 gyro.grid()
107 gyro.plot(df['gyro_x'], label="X")
108 gyro.plot(df['gyro_y'], label="Y")
109 gyro.plot(df['gyro_z'], label="Z")
110 gyro.legend(loc=1, prop={'size': 7})
111 gyro.set_ylabel('Gyroscope  $[^{\circ}/s]$ ')
112
113 mag.grid()
114 mag.plot(df['mag_x'], label="X")
115 mag.plot(df['mag_y'], label="Y")
116 mag.plot(df['mag_z'], label="Z")
117 mag.legend(loc=1, prop={'size': 7})
118 mag.set_ylabel('Magnitude  $[uT]$ ')
119
120 acc.grid()
121 acc.plot(df['accel_x'], label="X")
122 acc.plot(df['accel_y'], label="Y")
123 acc.plot(df['accel_z'], label="Z")
124 acc.legend(loc=1, prop={'size': 7})
125 acc.set_ylabel('Accel.  $[m/s]$ ')
126
127 euler.grid()
128 euler.plot(df['euler_h'], label="H")

```

```
121 euler.plot(df['euler_p'], label="P")
122 euler.plot(df['euler_r'], label="R")
123 euler.legend(loc=1, prop={'size': 7})
124 euler.set_ylabel('Euler')
125
126 quater.grid()
127 quater.plot(df['quater_w'], label="W")
128 quater.plot(df['quater_x'], label="X")
129 quater.plot(df['quater_y'], label="Y")
130 quater.plot(df['quater_z'], label="Z")
131 quater.legend(loc=1, prop={'size': 7})
132 quater.set_ylabel('Quaternion')
133
134
135 manager = plt.get_current_fig_manager()
136 manager.full_screen_toggle()
137 plt.savefig("./Output/MESURES_GRAPH.svg")
138 plt.savefig("./Output/MESURES_GRAPH.png", dpi=300)
139 plt.show()
140
141 else :
142     print("Pas de fichier MESURES.csv")
143
144
```