20_pandas

March 22, 2020

1 Generate Signal Analysis with Pandas.

pandas is a fast, powerful, flexible and easy to use open source data analysis and manipulation tool, built on top of the Python programming language.

- https://pandas.pydata.org/

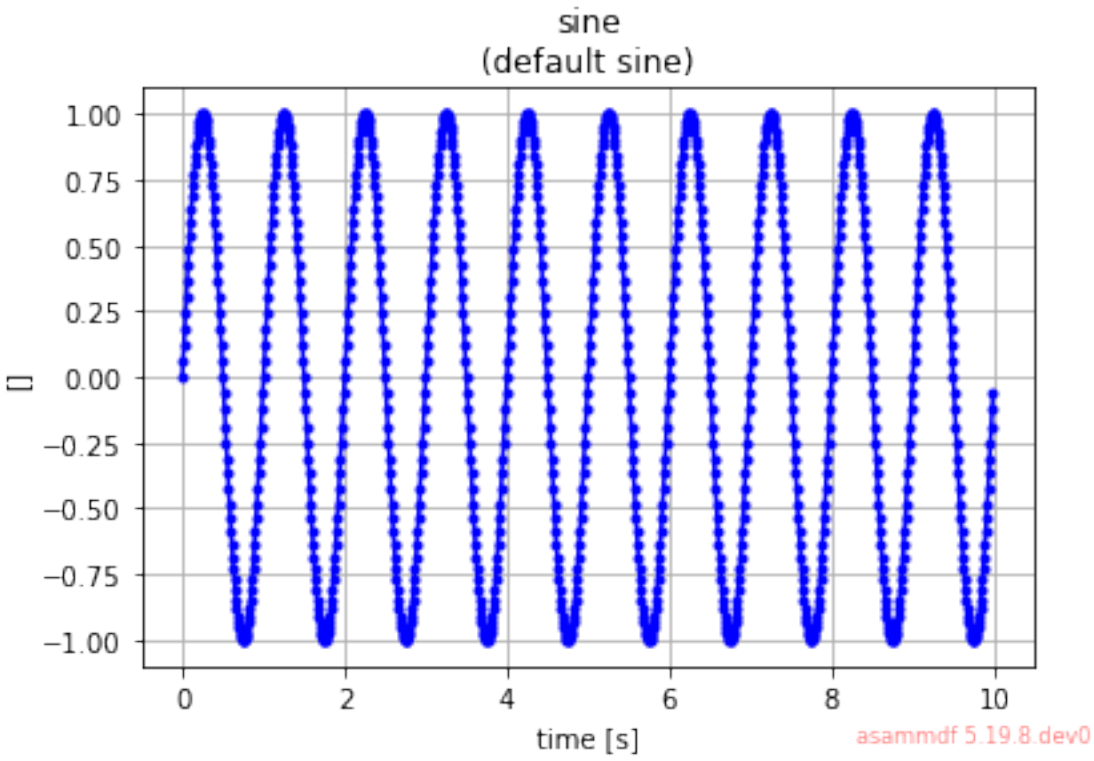
[1]:
```
import IPython.display as display
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import seaborn as sns
```

[2]:
```
from asammdf import MDF
```

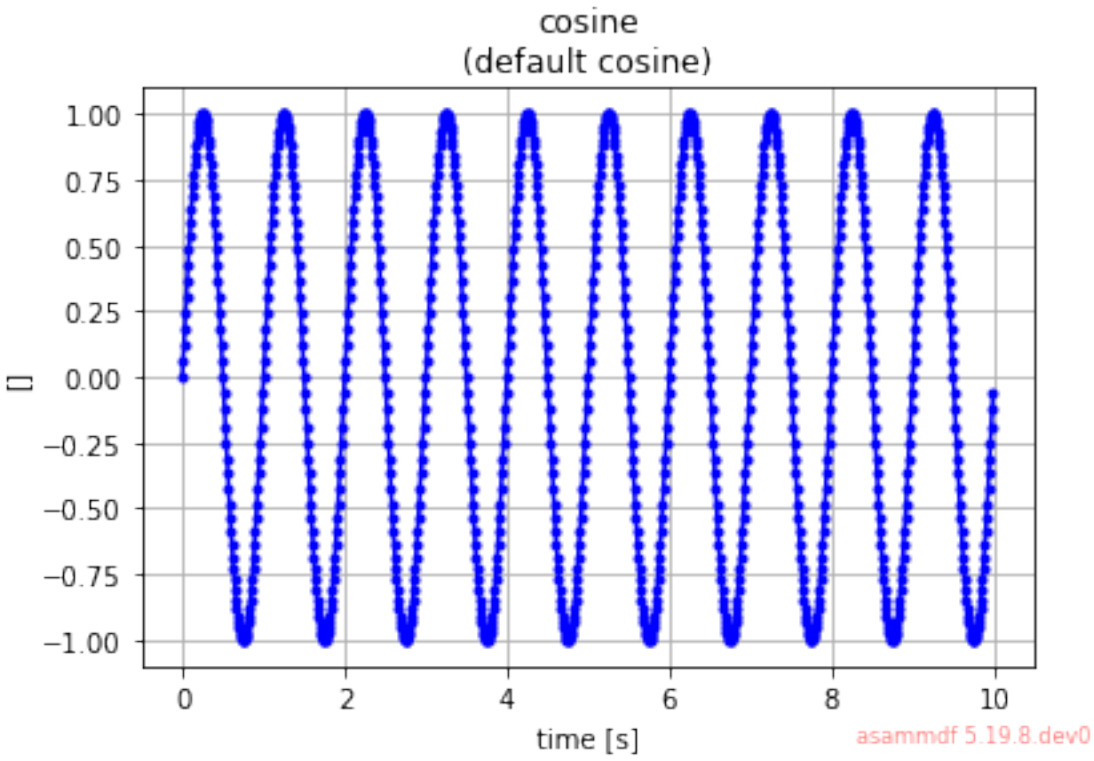
[3]:
```
# Generated in
mdf = MDF("90_BasicSignals.mf4")
```

[4]:
```
for channel in mdf.iter_channels():
    channel.plot()
```

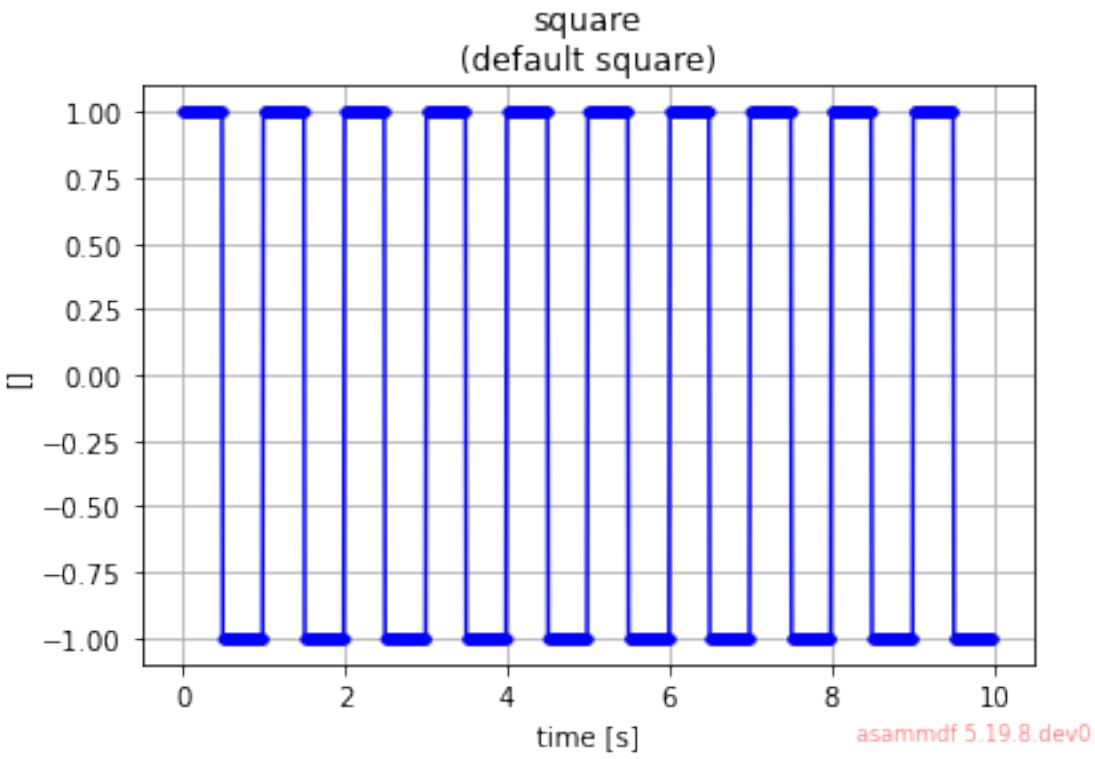
```
WARNING:root:Signal plotting requires pyqtgraph or matplotlib
```



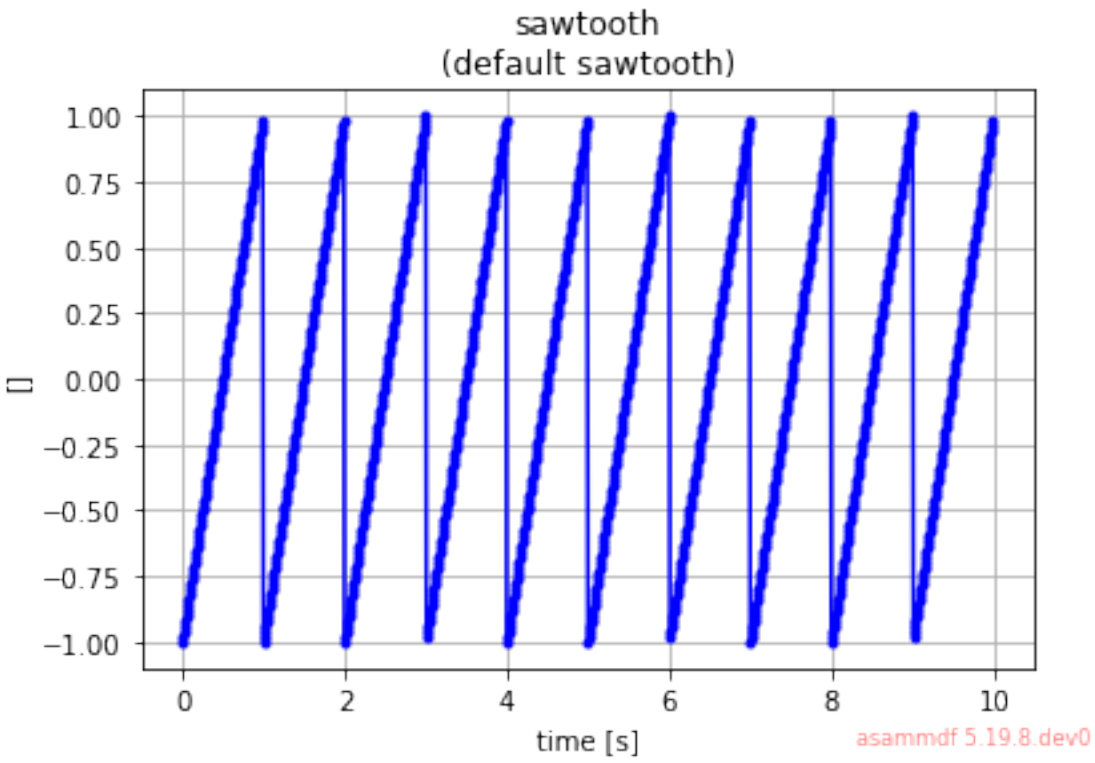

```
WARNING:root:Signal plotting requires pyqtgraph or matplotlib
```



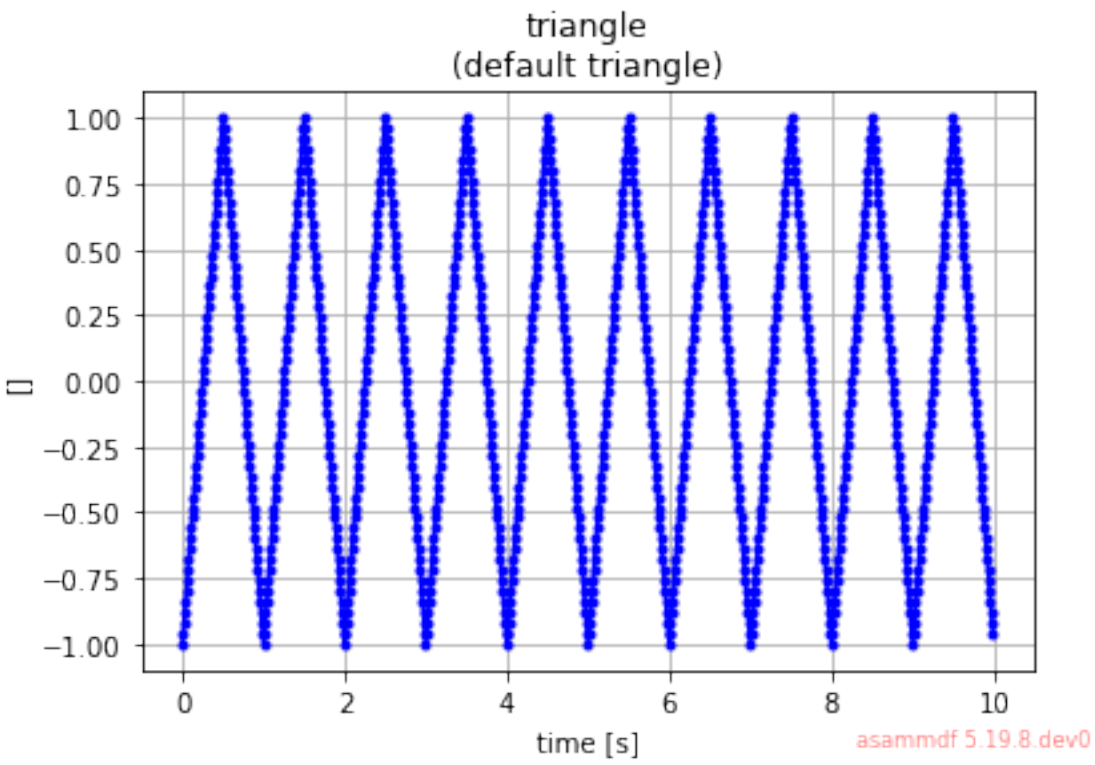

```
WARNING:root:Signal plotting requires pyqtgraph or matplotlib
```



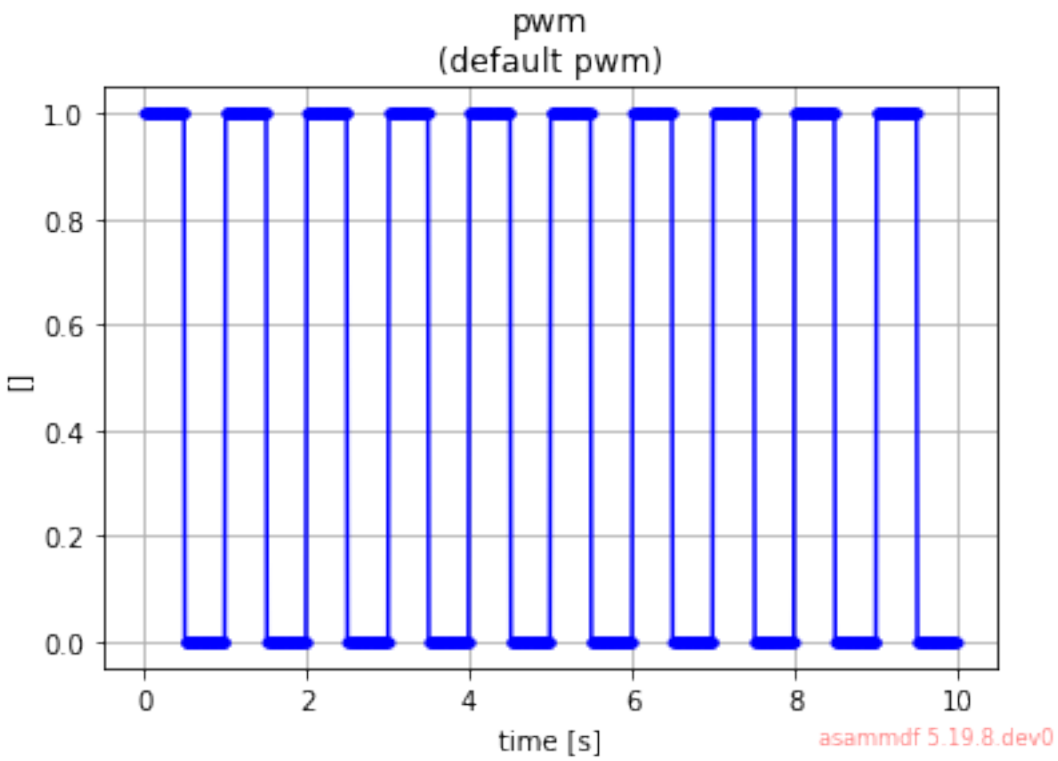

```
WARNING:root:Signal plotting requires pyqtgraph or matplotlib
```



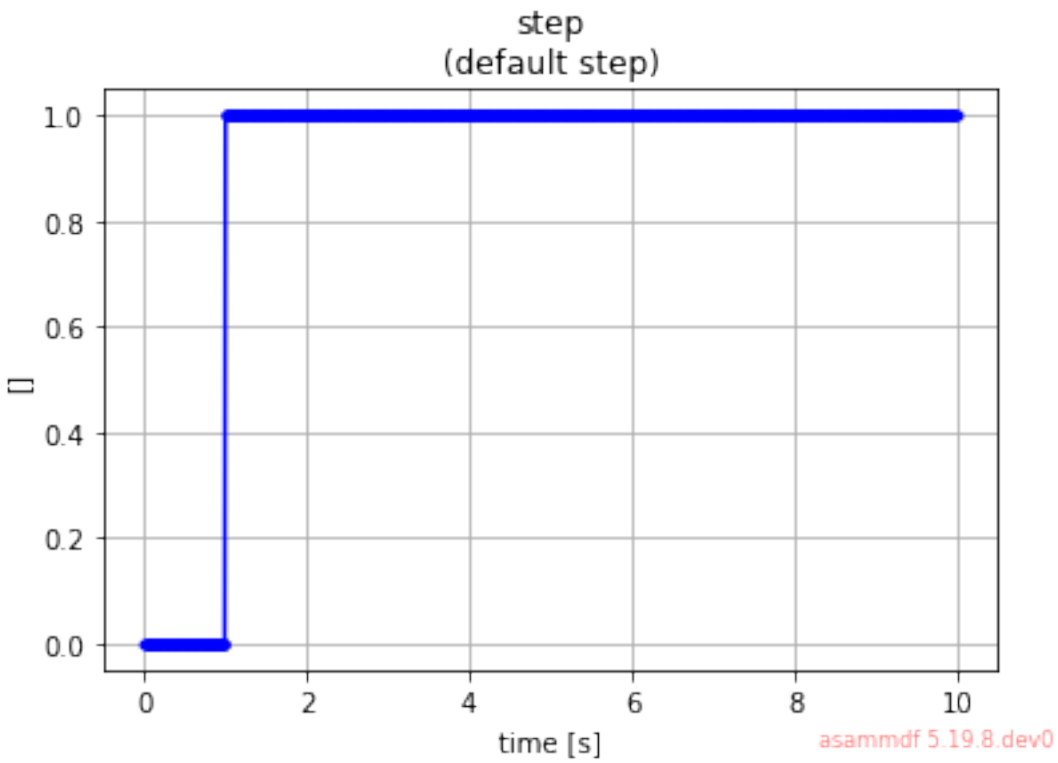

```
WARNING:root:Signal plotting requires pyqtgraph or matplotlib
```




```
WARNING:root:Signal plotting requires pyqtgraph or matplotlib
```




```
WARNING:root:Signal plotting requires pyqtgraph or matplotlib
```



[5]:
```
df = mdf.to_dataframe()
```

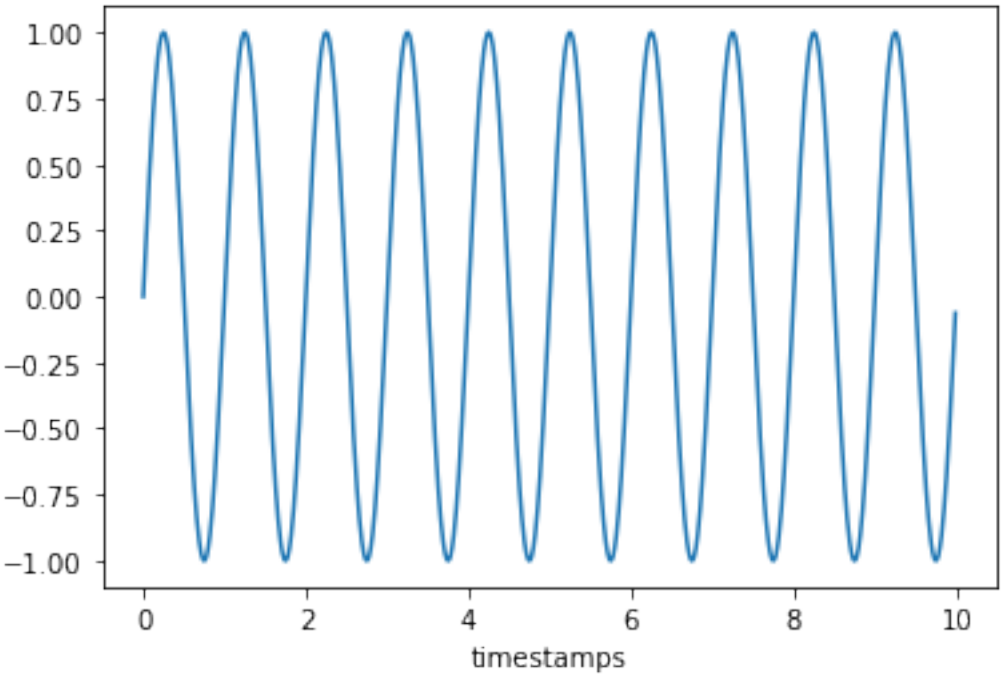
[6]:
```
df
```

[6]:

timestamps	sine	cosine	square	sawtooth	triangle	pwm	step
0.000000	0.000000	0.000000	1.0	-1.000000	-1.000000	1.0	0.0
0.010000	0.062791	0.062791	1.0	-0.980000	-0.960000	1.0	0.0
0.020000	0.125333	0.125333	1.0	-0.960000	-0.920000	1.0	0.0
0.030000	0.187381	0.187381	1.0	-0.940000	-0.880000	1.0	0.0
0.040000	0.248690	0.248690	1.0	-0.920000	-0.840000	1.0	0.0
…	…	…	…	…	…	…	…
9.950000	-0.309018	-0.309018	-1.0	0.899999	-0.799998	0.0	1.0
9.960000	-0.248687	-0.248687	-1.0	0.920000	-0.840001	0.0	1.0
9.969999	-0.187382	-0.187382	-1.0	0.939999	-0.879998	0.0	1.0
9.980000	-0.125334	-0.125334	-1.0	0.959999	-0.919998	0.0	1.0
9.990000	-0.062792	-0.062792	-1.0	0.979999	-0.959998	0.0	1.0

[1000 rows x 7 columns]

[7]:
```
ax1 = df.sine.plot()
```



[8]:
```
df.sine.min()
```

[8]:
```
-1.0
```

[9]:
```
df.sine.max()
```

[9]:
```
1.0
```

[10]:
```
df.sine.mean()
```

[10]:
```
1.9073487e-09
```

[11]:
```
df.describe()
```

[11]:

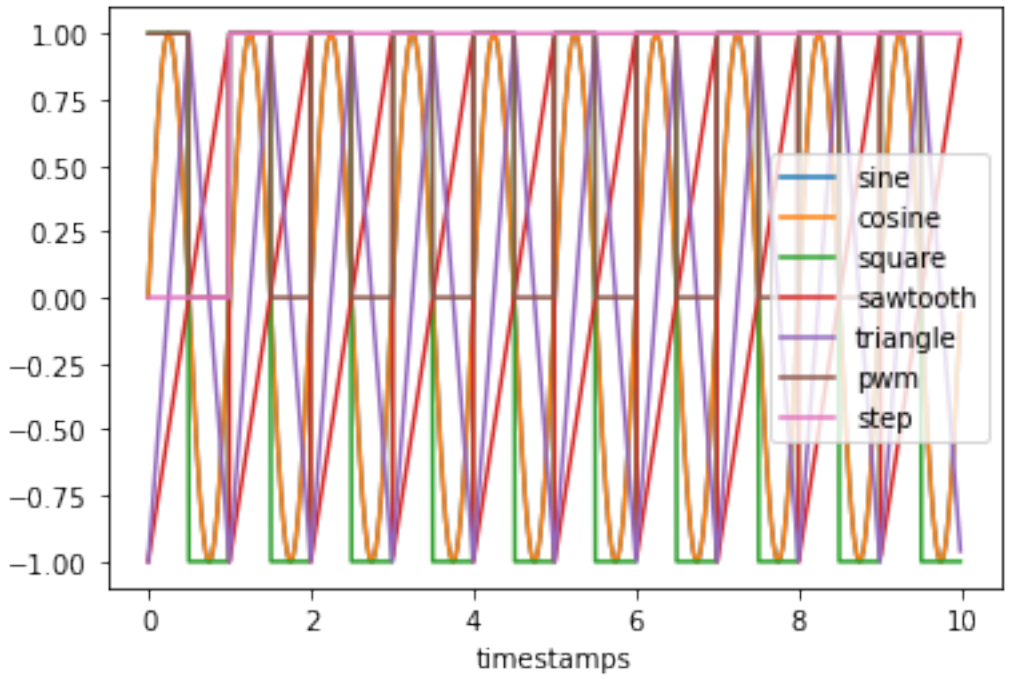
	sine	cosine	square	sawtooth	triangle	\
count	1.000000e+03	1.000000e+03	1000.000000	1.000000e+03	1.000000e+03	
mean	1.907349e-09	1.907349e-09	0.002000	-4.000228e-03	-9.597974e-09	
std	7.074606e-01	7.074606e-01	1.000498	5.776830e-01	5.778701e-01	
min	-1.000000e+00	-1.000000e+00	-1.000000	-1.000000e+00	-1.000000e+00	
25%	-6.956536e-01	-6.956536e-01	-1.000000	-5.000001e-01	-4.900004e-01	
50%	-1.192488e-08	-1.192488e-08	1.000000	-5.960464e-08	0.000000e+00	
75%	6.956536e-01	6.956536e-01	1.000000	4.999998e-01	4.900009e-01	
max	1.000000e+00	1.000000e+00	1.000000	9.999999e-01	9.999999e-01	

	pwm	step

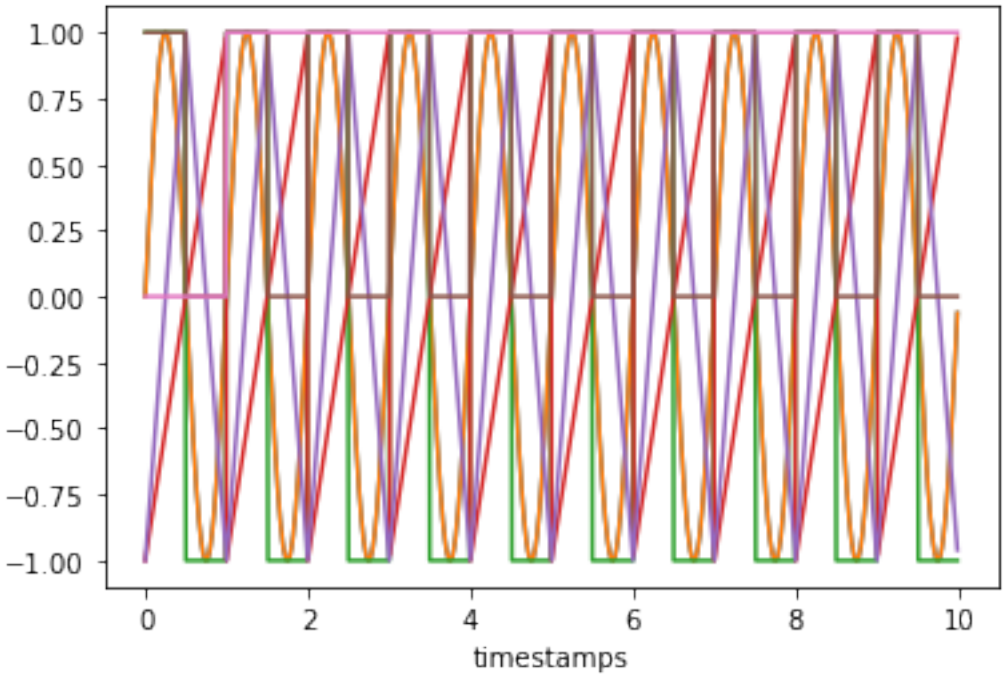
```
count  1000.000000  1000.00000
mean      0.501000     0.90000
std       0.500249     0.30015
min       0.000000     0.00000
25%       0.000000     1.00000
50%       1.000000     1.00000
75%       1.000000     1.00000
max       1.000000     1.00000
```

[12]:
```
df.plot()
```

[12]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fef87a78eb0>
```

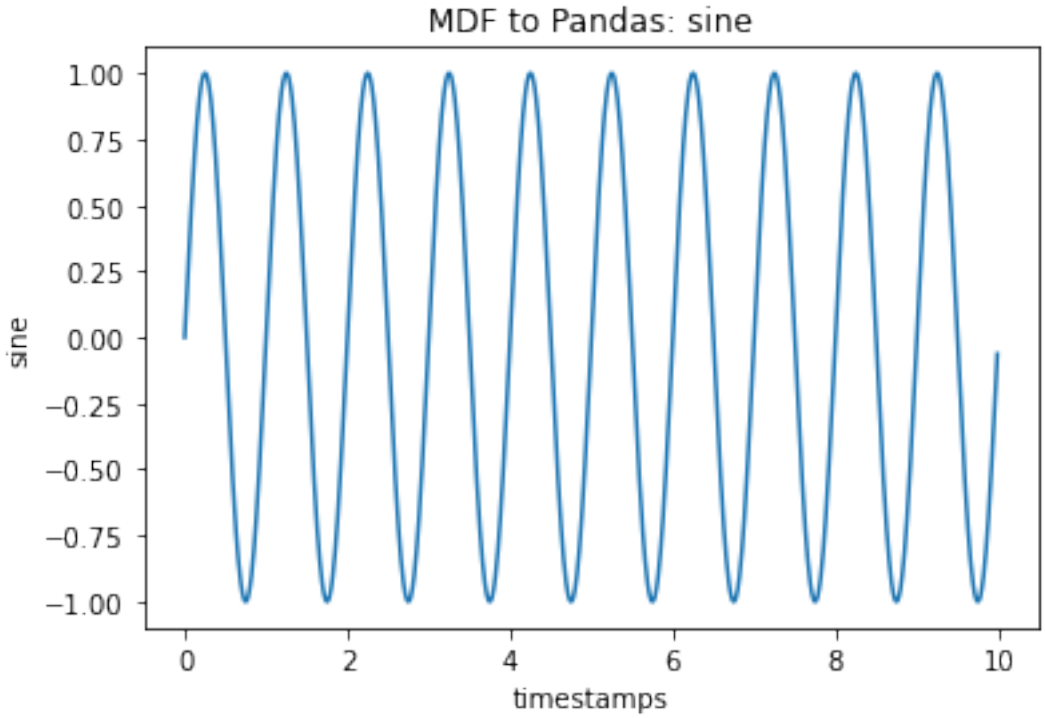


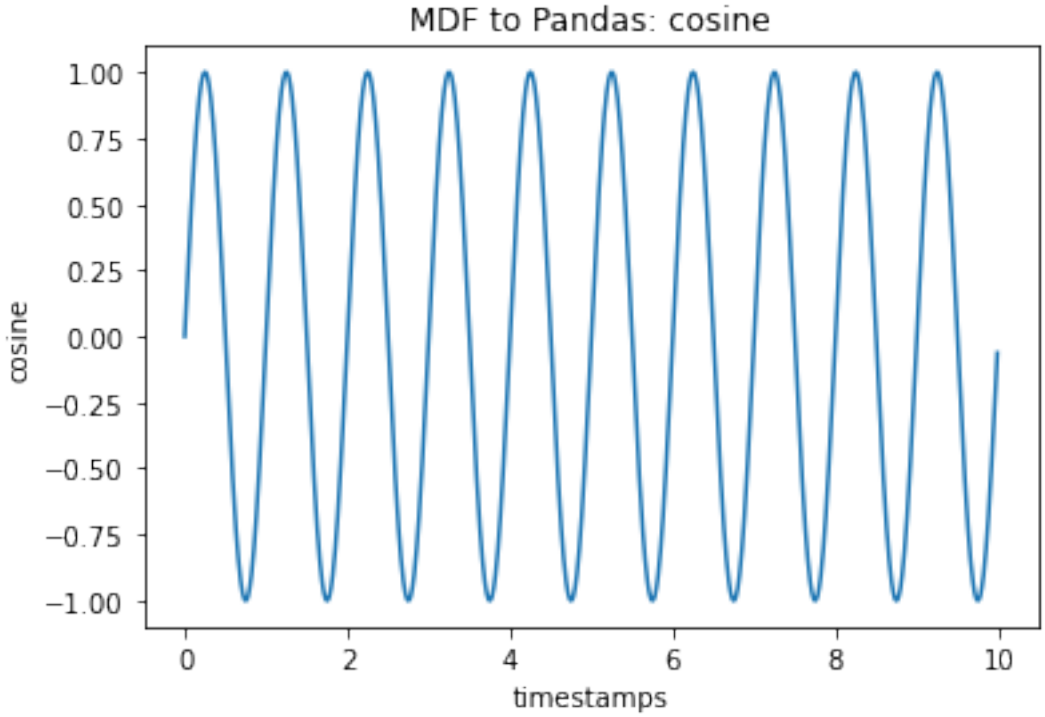
[13]:
```
for signal_name in df.columns:
    signal = getattr(df, signal_name)
    signal.plot()
```

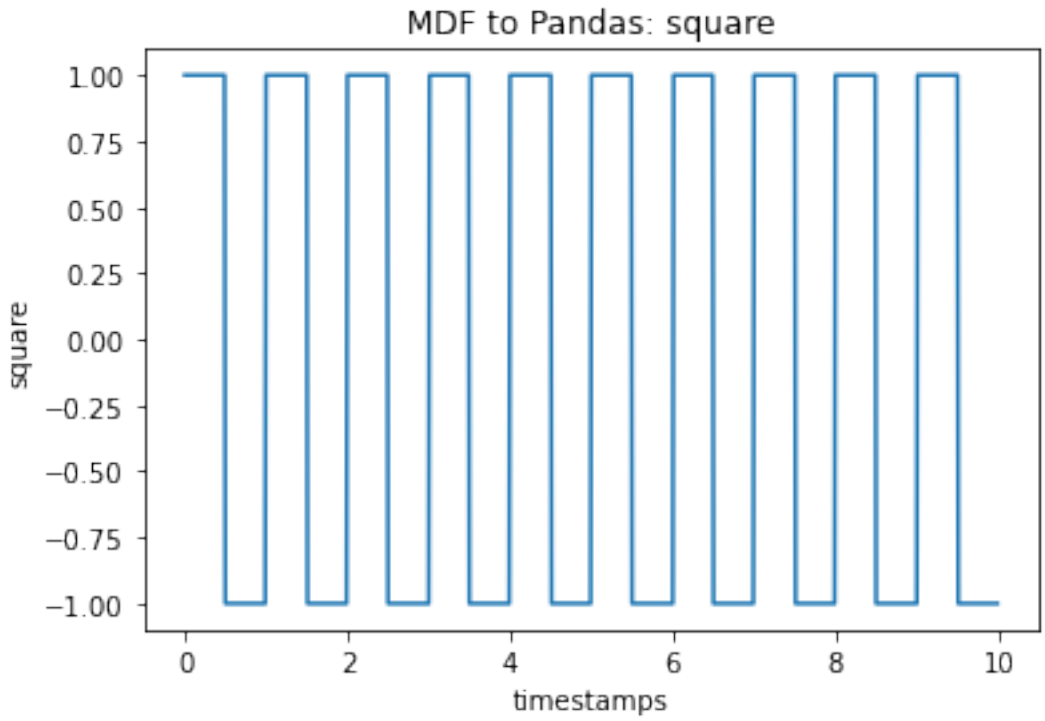


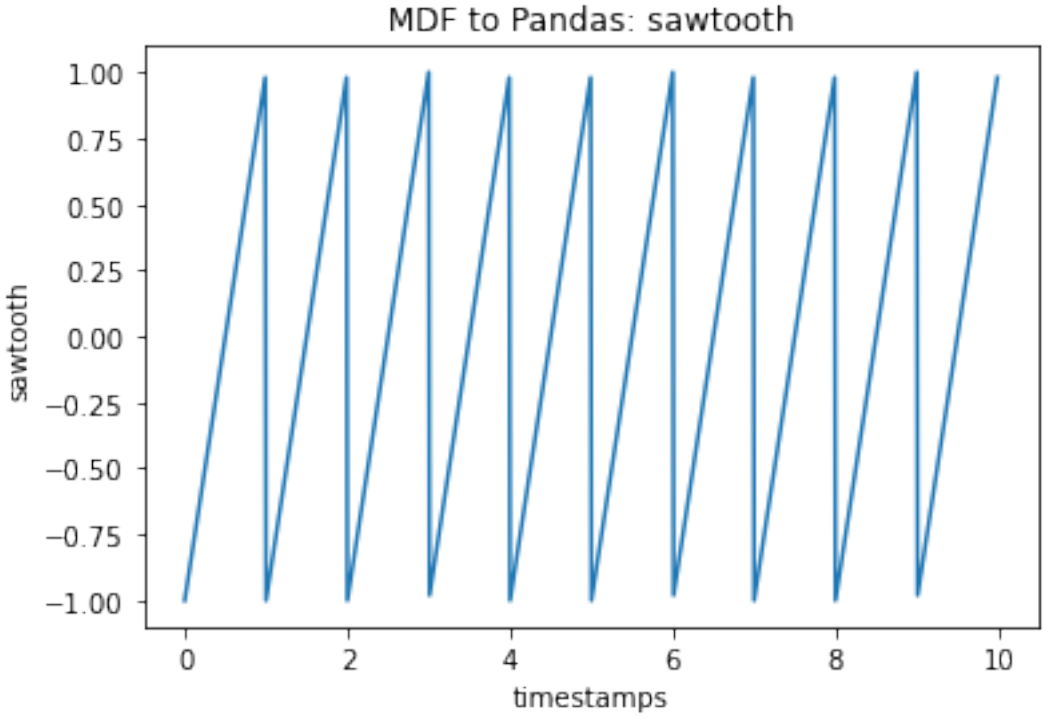
[14]:
```
from IPython import display
```

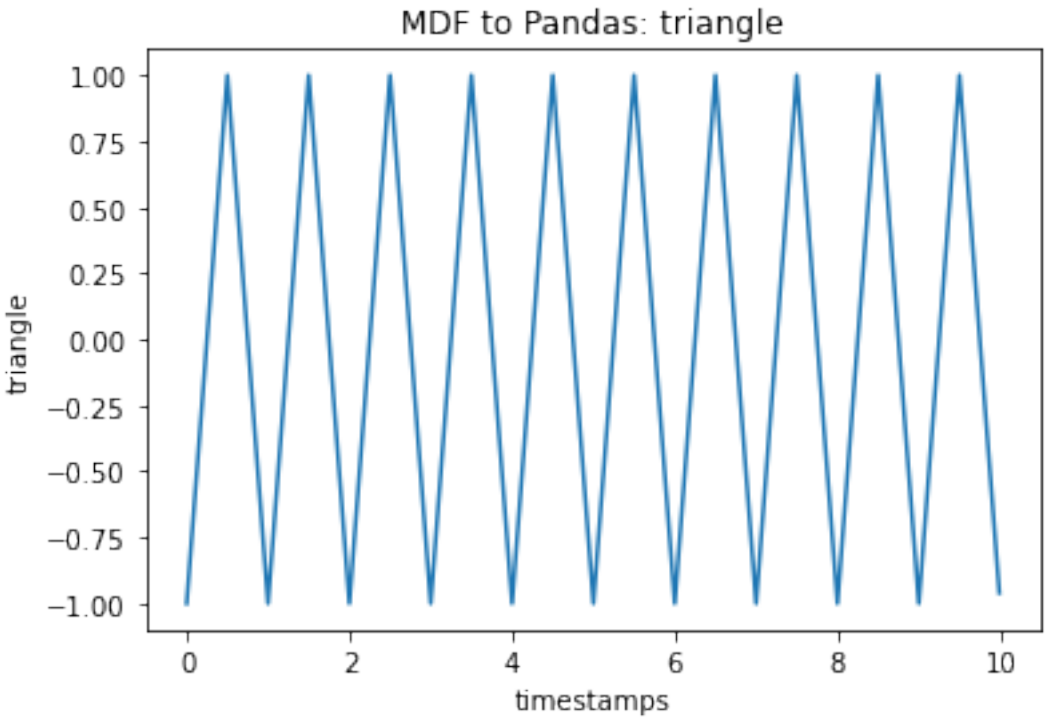
[15]:
```
for signal_name in df.columns:
    signal = getattr(df, signal_name)
    signal.plot()
    plt.ylabel(signal_name)
    plt.title(f"MDF to Pandas: {signal_name}")
    plt.show()
```

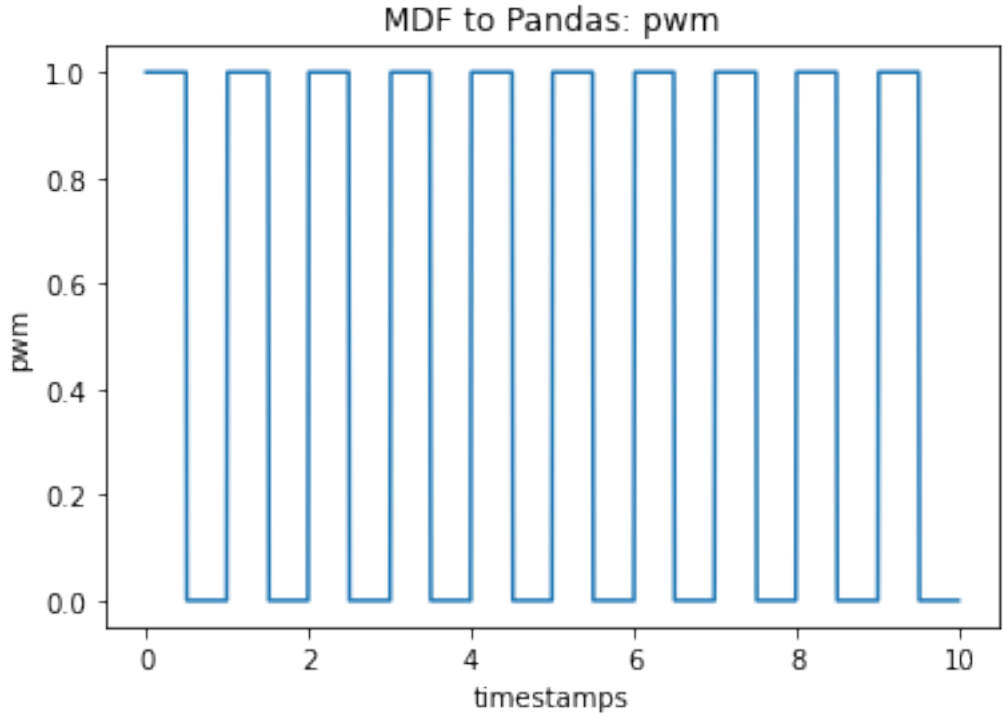

MDF to Pandas: sine
sine
timestamps
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
2
4
6
8
10

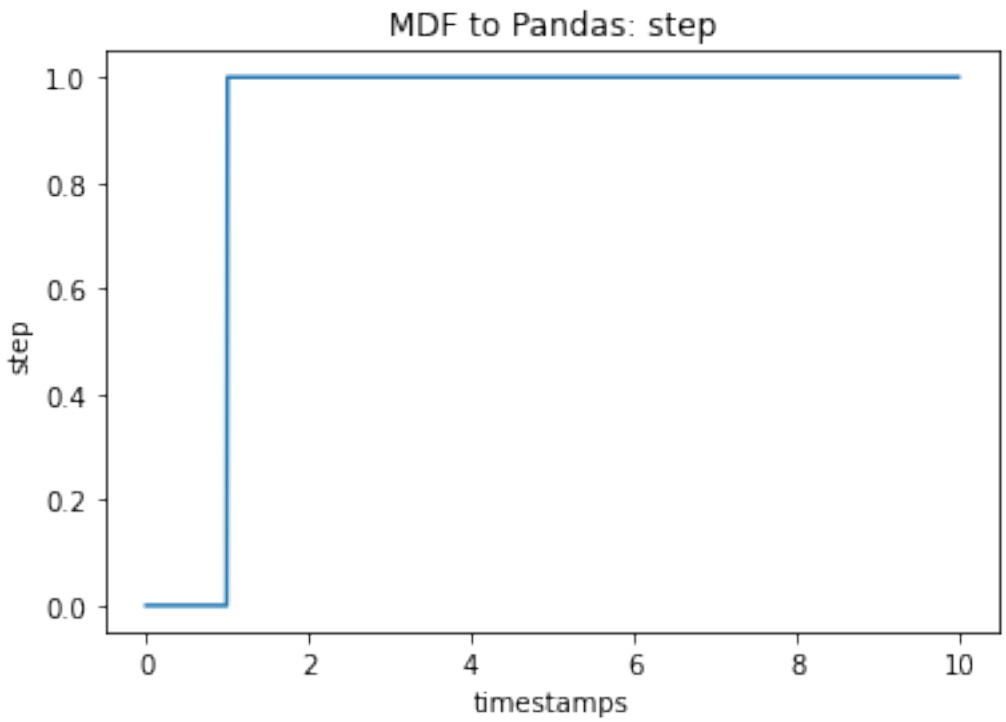

MDF to Pandas: cosine
cosine
timestamps
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
2
4
6
8
10


MDF to Pandas: square
square
timestamps
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
2
4
6
8
10


MDF to Pandas: sawtooth
sawtooth
timestamps
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
2
4
6
8
10


MDF to Pandas: triangle
triangle
timestamps
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
2
4
6
8
10


MDF to Pandas: pwm
pwm
timestamps
1.0
0.8
0.6
0.4
0.2
0.0
0
2
4
6
8
10


MDF to Pandas: step
1.0
0.8
0.6
0.4
0.2
0.0
step
0
2
4
6
8
10
timestamps