

Nombre: Logan Daniel Beltran Ruiz

cc: 1000382874

▼ Asignación Clase 6

1. Convertir en coordenadas polares los datos de de y d1
2. Graficar un cardioide en coordenadas polares en Python

```
1 import pandas as pd
2 import numpy as np
3
4 df1 = pd.DataFrame({
5     'de': np.sort(np.random.normal(loc = 4, scale = 1, size=96)),
6     'd1': np.sort(np.random.normal(loc=4.5, scale=1.2, size=96)),
7
8 })
9
10 df1
```

	de	d1
0	1.648792	1.832723
1	1.660988	2.186220
2	1.687970	2.301449
3	1.777549	2.498528
4	2.338802	2.565662
...
91	5.619444	6.533989
92	5.733197	7.025748
93	5.774850	7.148563
94	5.946676	7.552742
95	6.848235	8.296262

96 rows × 2 columns

```
1 x = df1['de']
2 y = df1['d1']
3 x2 = x**2
4 y2 = y**2
5
6 df1['r'] = np.sqrt(x2+y2)
7 df1['th'] = np.arctan(y/x)
8 df1
9
```

	de	d1	r	th
0	1.648792	1.832723	2.465237	0.838180
1	1.660988	2.186220	2.745621	0.921082
2	1.687970	2.301449	2.854104	0.937979
3	1.777549	2.498528	3.066320	0.952435
4	2.338802	2.565662	3.471688	0.831621
...
91	5.619444	6.533989	8.618072	0.860506
92	5.733197	7.025748	9.068113	0.886359
93	5.774850	7.148563	9.189714	0.891297
94	5.946676	7.552742	9.612849	0.903815
95	6.848235	8.296262	10.757615	0.880722

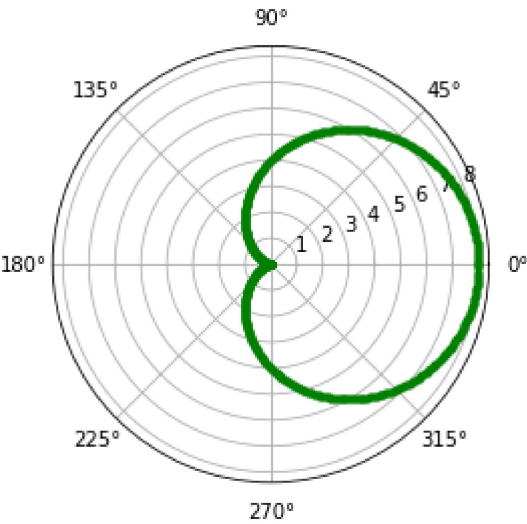
96 rows × 4 columns

```

1 import numpy as np
2 import matplotlib.pyplot as plt
3 import math
4
5
6 plt.axes(projection = 'polar')
7
8 a=4
9
10 rads = np.arange(0,(2 * np.pi), 0.01)
11
12 for rad in rads:
13     r = a + (a*np.cos(rad))
14     plt.polar(rad,r,'g.')
15
16 plt.show()

```





✓ 2 s se ejecutó 17:06 ● ✕