

▼ Asignación 4

1. Convertir en coordenadas polares los datos de de y d1 asumiendo que son x y y

```
1 import pandas as pd
2 import numpy as np

1 def prog(r,n,a1):
2     an=a1+r*(n-1)
3     seq = np.arange(start=a1, stop=an, step=r)
4     return seq
5 prog(r=7,n=20,a1=15)

array([ 15,  22,  29,  36,  43,  50,  57,  64,  71,  78,  85,  92,  99,
        106, 113, 120, 127, 134, 141])

1 x = df1["de"]
2 y = df1["d1"]
3
4 r=(x**2+y**2)**(1/2)
5 t= np.arctan2(y,x)
6
7 print(f"En donde, r respresenta la distancia desde el origen hasta el punto: {r}")
8 print(f"Y t representa el ángulo formado con respecto al eje x:{t}")

En donde, r respresenta la distancia desde el origen hasta el punto: 0      2.232817
1      2.799227
2      2.990203
3      3.217707
4      3.418258
...
91      8.545790
92      8.684503
93      8.722801
94      9.171851
95      9.574623
Length: 96, dtype: float64
Y t representa el ángulo formado con respecto al eje x:0      0.964623
1      0.757025
2      0.744431
3      0.741757
4      0.726052
...
91      0.856053
92      0.841870
93      0.840393
94      0.831207
95      0.837644
Length: 96, dtype: float64
```

2. Graficar un cardiode en coordenadas polares en Python

```
1 from matplotlib import pyplot as plt
```

```
1 t= np.linspace(0,2*np.pi,1000)
```

```
2 r=5-5*np.sin(t)
```

```
3 plt.polar(t,r)
```

```
4 plt.show ()
```

