import numpy **as** np

import matplotlib.pyplot **as** plt

#Puntos

a = np.array([0, 100])

b = np.array([20, 150])

c = np.array([30, 200])

**d** = np.array([40, 180])

**e** = np.array([50, 250])

**g** = np.array([60, 230])

xy = a[0]\*a[1] + b[0]\*b[1] + c[0]\*c[1] + **d**[0]\***d**[1] + **e**[0]\***e**[1] + **g**[0]\***g**[1]

#Aplico las ecuaciones clasicas para calcular **la** regresion lineal

Sum\_x = a[0] + b[0] + c[0] + **d**[0] + **e**[0] + **g**[0]

Sum\_y = a[1] + b[1] + c[1] + **d**[1] + **e**[1] + **g**[1]

Sum\_x2 = a[0]\*\*2 + b[0]\*\*2 + c[0]\*\*2 + **d**[0]\*\*2 + **e**[0]\*\*2 + **g**[0]\*\*2

**n** = 6

pendiente = (**n**\*xy – sum\_x \* sum\_y) / (**n**\*sum\_x2 – sum\_x\*\*2)

intercepto = sum\_y/**n** - (A1\*sum\_x/**n**)

def f(x):

**return** pendiente \* x + intersepto;

**N**=1000

x = np.linspace(0,(60), num=**N**)

y = f(x)

**print**("y =",intercepto," + ", pendiente ,"x")

ECM = ((f(0)-a[1]) + (f(20)-b[1]) + (f(30)-c[1]) + (f(40)-**d**[1]) + (f(50)-**e**[1]) +(f(60)-**g**[1]))\*\*2/**n**

# Aplico **la** formula del **error** cuadratico medio

# y^ viene de **la** funcion f(x) evaluada **en** cada punto x

**print**("Error cuadratico medio, ECM = ", ECM)

fig, ax = plt.subplots()

ax.**plot**( x ,y , **label**='y = 106.43 + 2.36 X')

ax.**scatter**(a[0], a[1], c='red')

ax.**scatter**(b[0], b[1], c='red')

ax.**scatter**(c[0], c[1], c='red')

ax.**scatter**(**d**[0], **d**[1], c='red')

ax.**scatter**(**e**[0], **e**[1], c='red')

ax.**scatter**(**g**[0], **g**[1], c='red')

ax.set\_xlabel('Gasto **en** publicidad')

ax.set\_ylabel('Ventas')

ax.legend()

ax.grid()

Chart, line chart, scatter chart

Description automatically generated