import pandas as pd

data1= pd.DataFrame({ "M(gm)":[168.28] ,"m(gm)":[50]})

data = pd.DataFrame( {"X(cm)":[20,40,60,80,100,120], "t1(msec)":[214,416.63,572.65,755.37,959.59,1076],"t2(msec)":[173,436.8,542.16,726.26,994.79,971.33],"t(msec)":[193.5,426.72,557.41,740.82,977.19,1023.67],"t(sec)":[0.19,0.43,0.56,0.74,0.98,1.02], 't^2(sec^2)':[0.04,0.18,0.31,0.55,0.96,1.04]})

table1=pd.DataFrame(data1)

table=pd.DataFrame(data)

headers = {

"selector": "th:not(.index\_name)",

"props": "background-color: #FFC0CB; color: black; text-align: center" #FFC0CB is the color of the pink background head and index

}

properties = {"border": "2px solid black", "width": "65px", "text-align": "center"}

data1.style.format(precision=2).set\_table\_styles([cell\_hover, index\_names, headers]).set\_properties(\*\*{'border':'2px solid black','color': 'gray', "text-align": "center"})

headers = {

"selector": "th:not(.index\_name)",

"props": "background-color: #FFC0CB; color: black; text-align: center" #FFC0CB is the color of the pink background head and index

}

properties = {"border": "2px solid black", "width": "65px", "text-align": "center"}

data.style.format(precision=2).set\_table\_styles([cell\_hover, index\_names, headers]).set\_properties(\*\*{'border':'2px solid black','color': 'gray', "text-align": "center"})

#using matplotlib.pyplot lib to plot a graph

import matplotlib.pyplot as plt

import seaborn as sns

plt.rc('font', family='serif') # new code

plt.figure(figsize=(14,6)) #must be at first to exceed right # new code

x=table['t^2(sec^2)']

y=table["X(cm)"]

plt.grid(True)

plt.plot(x,y,color='k')

sns.set()

plt.xlabel('$t^2(sec^2)$',fontsize=12, color='m') # new code

plt.ylabel("$X(cm)$",fontsize=12 , color='m') # new code

plt.title("Relationship between $x$ versus $t^2$",fontsize=18 , color='c') # new code

plt.show()

from scipy.stats import linregress

x=table['t^2(sec^2)']

y=table["X(cm)"]

slop=linregress(x,y) # we used a new code we didnt study to find the slop from the gragh and it also finds the intercept and other values

print(slop)

import sympy as sp

from sympy import symbols

slop=symbols("slop")

a\_exp=2\*slop

a\_exp

print("a exp =",a\_exp,"cm/s^2")

#using sympy for symbolic computation to write the expertion of the formula

M,m,g = symbols("M m g")

expr = (m\*g)/(M + m)

expr

#the calculate of the theoretical acceleration ath vlue

M = 168.28

m = 50

g = 980

ath = (m\*g)/(M + m)

print("a\_th =", round(ath,2),'cm/$s^2$')