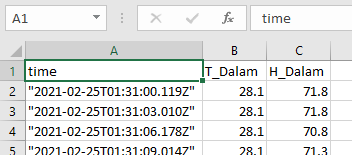
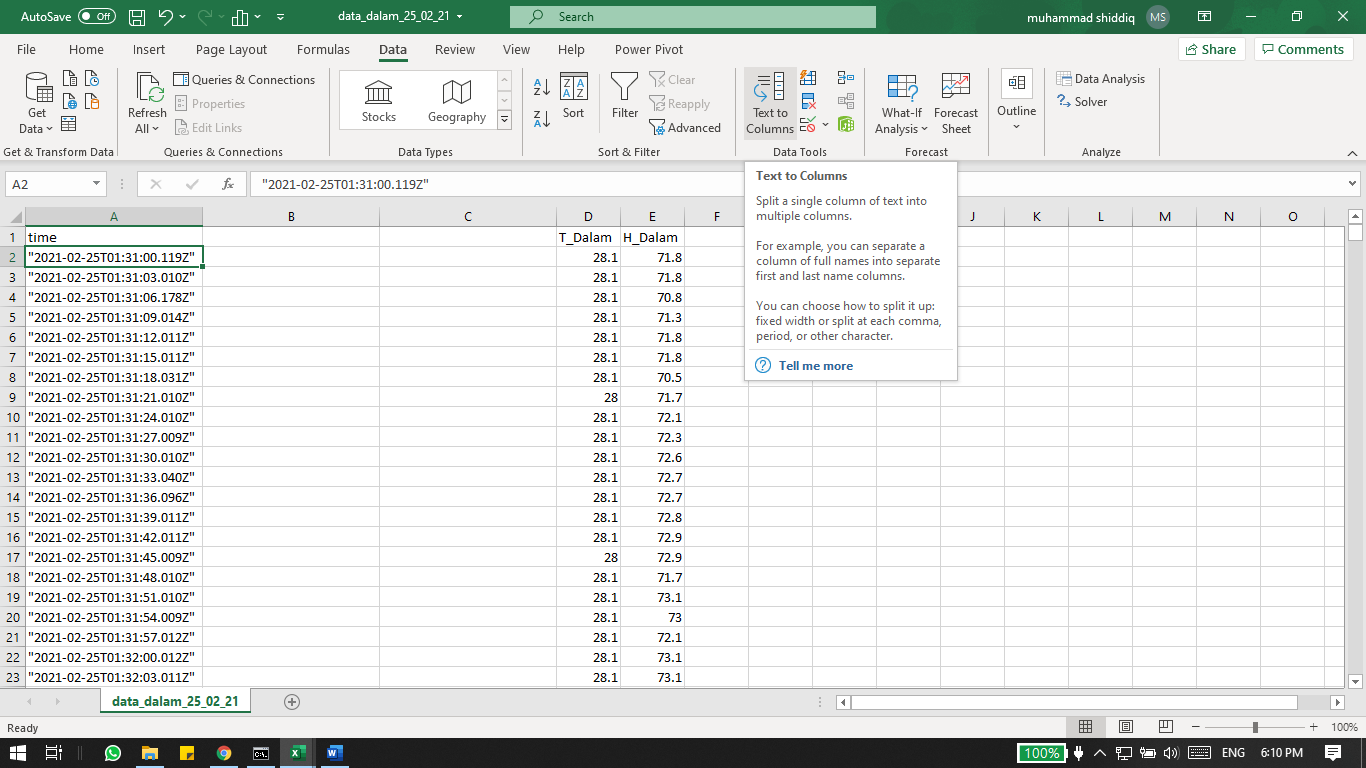
Buka data mentah csv data\_dalam\_25\_02\_21, didapatkan dari github



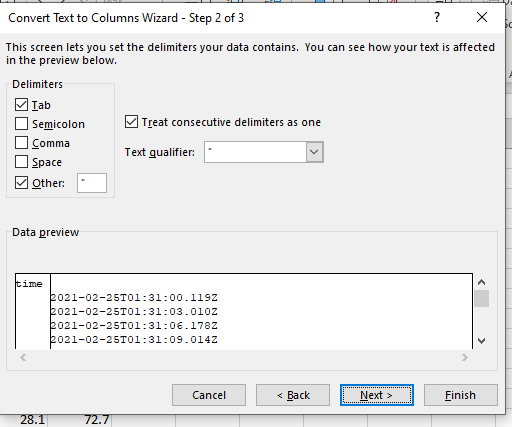
Didalam data ini ada 3 prarameter , waktu, Temperatur dalam ruangan dan Kelembapan dalam ruangan

Tambah 2 kolom di sebelah waktu

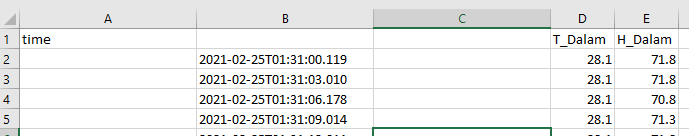
Hilangkan tanda “ dengan Data, text to colomn



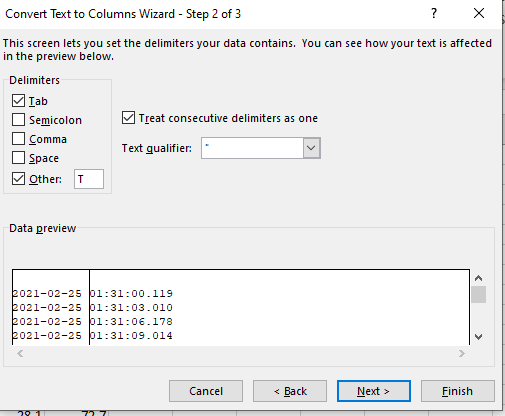
Blok kolom A, pilih delimited dan centang other pilih petik dua (“) ,



Ulangi dengan menghilangkan huruf Z

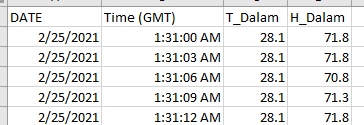


Ulangi lagi dengan menghilangkan huruf T

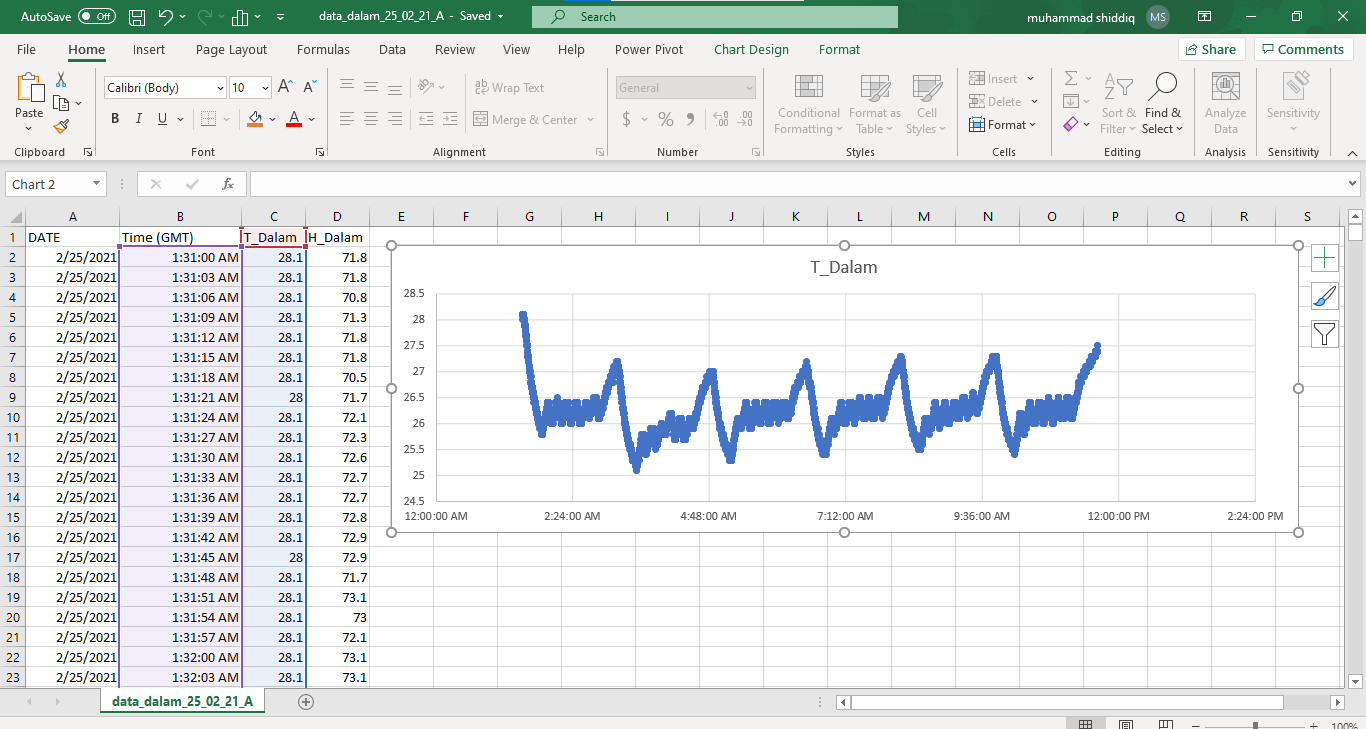


Hasilnya didapatkan

Ubah jenis data kolom C dengan waktu kemudian dirapihkan didapat data:



Lakukan plotting grafik temperatur terhadap waktu



6

5

44

3

2

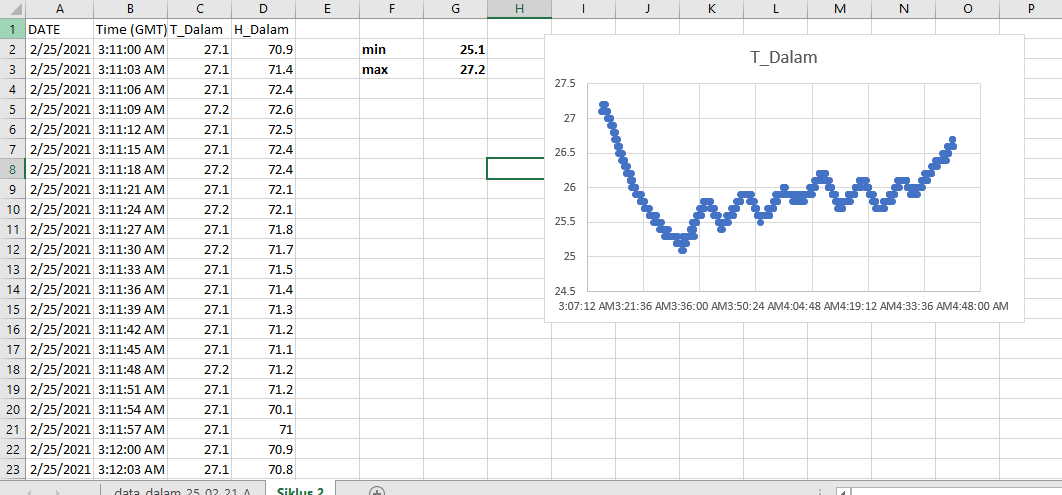
1

Disini terlihat ada 6 siklus AC

Pilih salah satu siklus AC

Pada contoh ini saya ambil sikllus ke 2

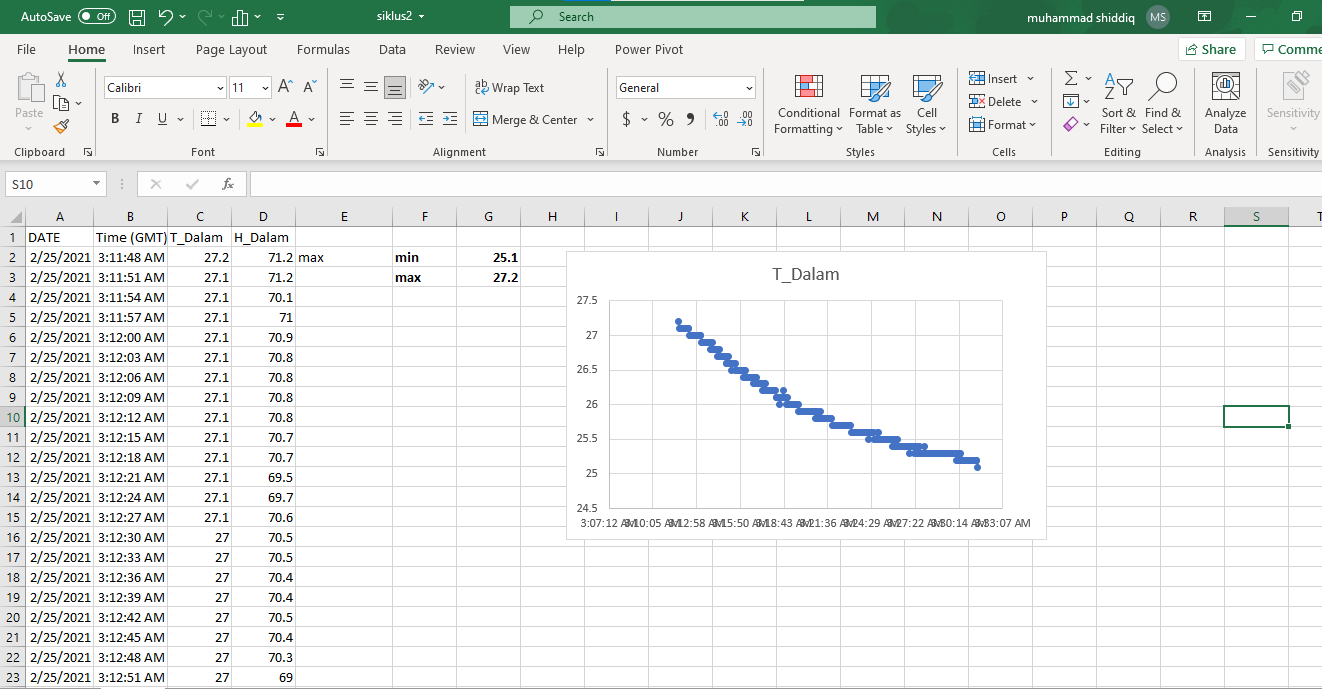
Cari min max dari T dalam pada saat temperatur turun



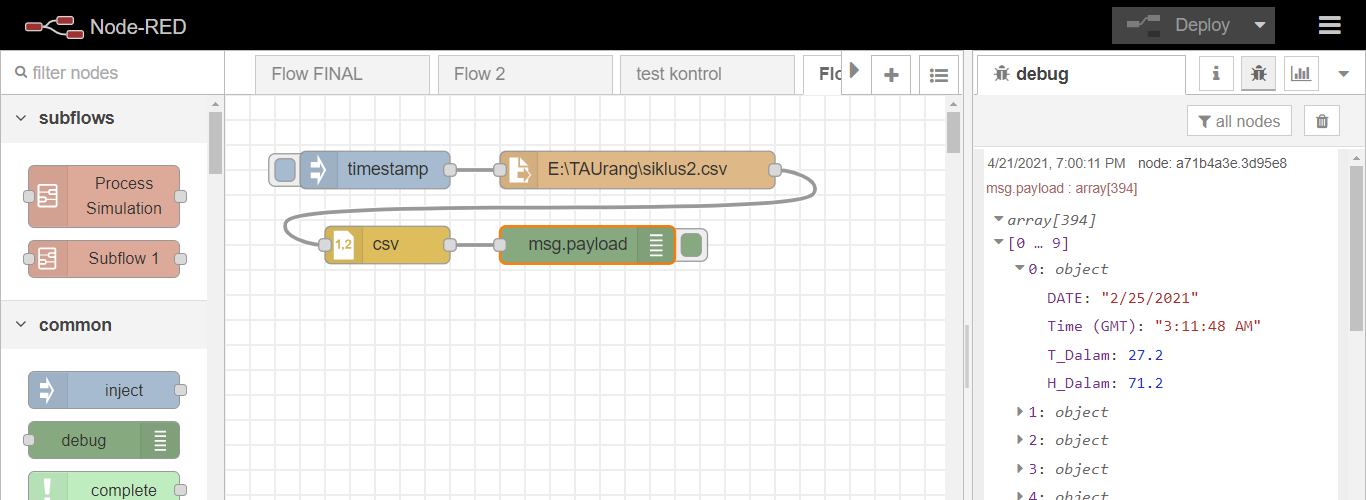
**min**

**max**

Buat file baru yg berisikan hanya data yg berada diantara titik min dan max

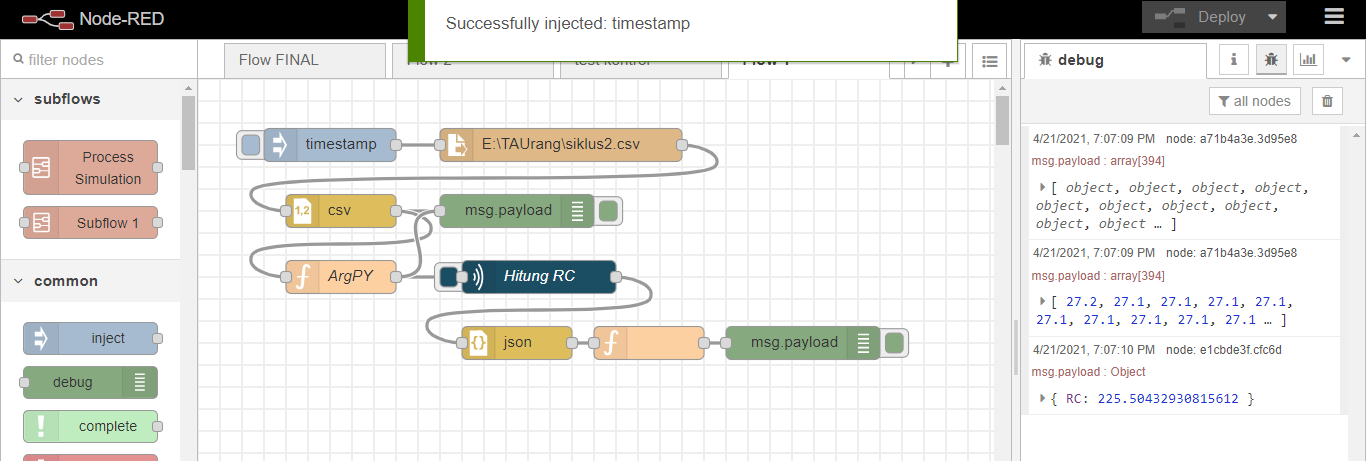


Buka node-red, dan coba import data dari file csv



Jika data sudah dapat dibaca,

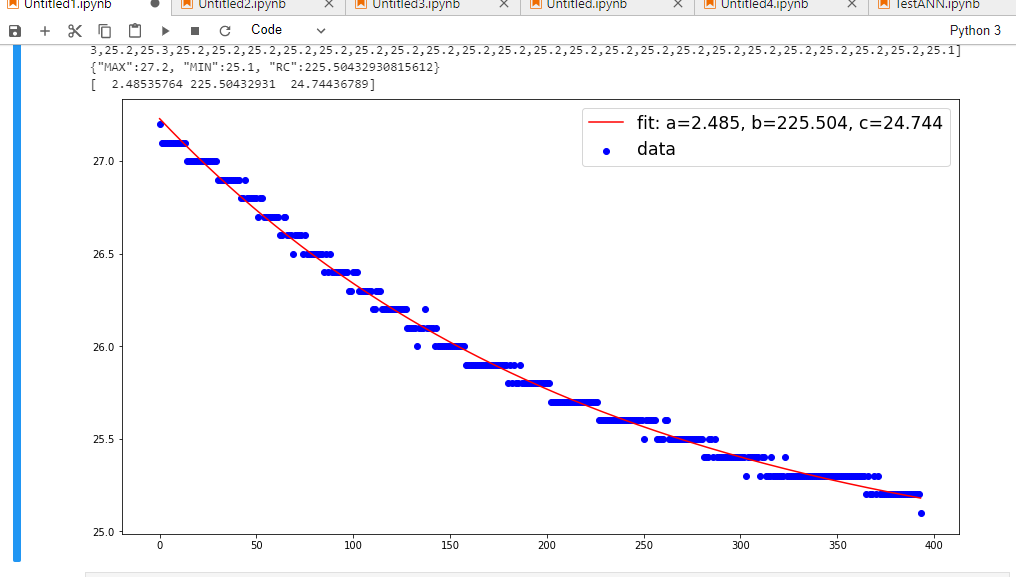
Lakukan perhitungan RC



Didapatkan RC pada siklus 2 ini sebesar **225.50432930815612**

Atau langsung manual dengan menggunakan jupyter

|  |
| --- |
| import sys  import matplotlib.pyplot as plt  import numpy as np  import scipy.optimize as so  def baseFunction(x,a,b,c):  return a \* np.exp(-(x / b)) + c  myString1 = input();  myString2 = myString1.replace("[","")  myString1 = myString2.replace("]","")  myArray1 = myString1.split(",")  myArray2 = []  lengthOfArray = len(myArray1)  if lengthOfArray > 2:  for y in myArray1:  myFloat = float(y)  myArray2.append(myFloat)  valueOfY = myArray2  myArray3 = []  for x in range(0,lengthOfArray):  myArray3.append(x)  valueOfX = myArray3  valueYMIN = min(valueOfY)  valueYMAX = max(valueOfY)  OM = '{"TMIN":' + str(valueYMIN) + ', "TMAX":' + str(valueYMAX) + '}'  fit\_param, pcov = so.curve\_fit(baseFunction,valueOfX,valueOfY)  #RQIN = fit\_param[1]\*(valueYMAX-valueYMIN)  OMR = '{"MAX":' + str(valueYMAX) + ', "MIN":' + str(valueYMIN) + ', "RC":' + str(fit\_param[1]) + '}'  print(OMR)  print(fit\_param)  fig=plt.figure(figsize=(15,8))  plt.scatter(valueOfX, valueOfY, label='data', color='blue')  plt.plot(valueOfX, baseFunction(valueOfX, \*fit\_param), 'r-',label='fit: a='+str(round(fit\_param[0],3))+', b='+str(round(fit\_param[1],3))+', c='+str(round(fit\_param[2],3))+'')  plt.legend(fontsize='xx-large')  plt.show()  else:  print(myString1)  print("null") |



// lampiran



File python untuk menghitung RC gunakan py plot ON di github