a)

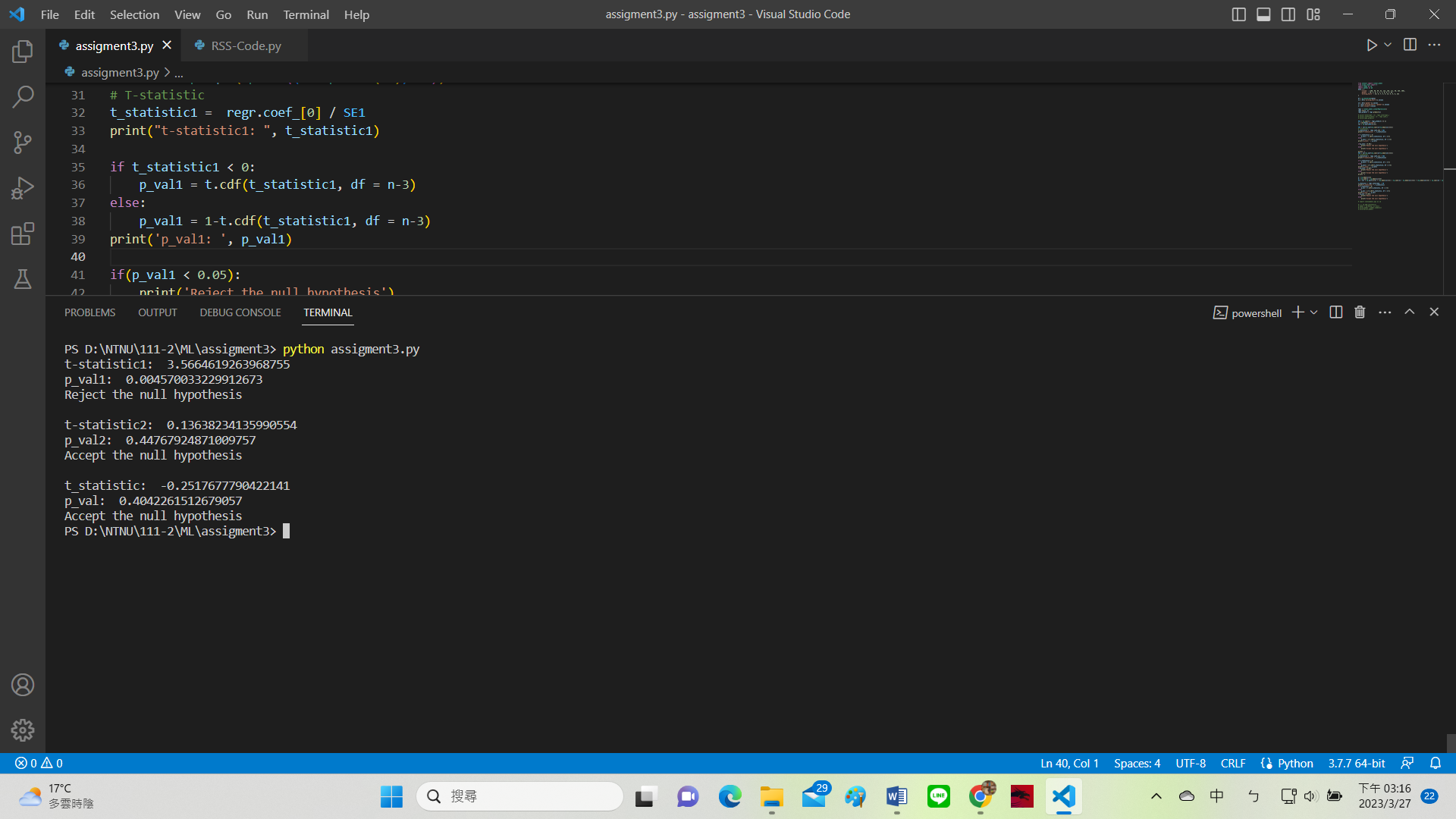
x1 = work hours

x2 = gender

x = [work hours, gender]

y = income

t-statistic and p-value are showed on the following figure.



b)

the result shows that there is some relationship between x1 and y but there is no relationship between x2 and y.

the last t-statistic on figure is t-statistic of whole model. And the result shows there is no relationship between x and y.

Code:

from sklearn import linear\_model

from scipy.stats import t

import numpy as np

import pandas as pd

data = {

    'income' : [100, 90, 99, 62, 123, 165, 111, 50, 165, 200],

    'gender' : [0, 1, 0, 1, 1, 1, 0, 0, 1, 0],

    'working\_hours' : [8, 6, 7, 5, 9, 13, 9, 5, 8, 10]

}

df = pd.DataFrame(data)

x1 = df[['working\_hours']].values

x2 = df[['gender']].values

x = df[['working\_hours','gender']].values

y = df['income'].values

regr = linear\_model.LinearRegression()

regr.fit(x, y)

regr\_predict = regr.predict(x)

RSS = np.sum((y - regr\_predict) \*\* 2)

n = len(data['income'])

var = np.sqrt(RSS/(n-3))

SE1 = var/np.sqrt(np.sum((x1-np.mean(x1))\*\*2))

# T-statistic

t\_statistic1 =  regr.coef\_[0] / SE1

print("t-statistic1: ", t\_statistic1)

if t\_statistic1 < 0:

    p\_val1 = t.cdf(t\_statistic1, df = n-3)

else:

    p\_val1 = 1-t.cdf(t\_statistic1, df = n-3)

print('p\_val1: ', p\_val1)

if(p\_val1 < 0.05):

    print('1st variable \'Reject\' the null hypothesis')

else:

    print('1st variable \'Accept\' the null hypothesis')

print('')

SE2 = var/np.sqrt(np.sum((x2-np.mean(x2))\*\*2))

# T-statistic

t\_statistic2 =  regr.coef\_[1] / SE2

print("t-statistic2: ", t\_statistic2)

if t\_statistic2 < 0:

    p\_val2 = t.cdf(t\_statistic2, df = n-3)

else:

    p\_val2 = 1-t.cdf(t\_statistic2, df = n-3)

print('p\_val2: ', p\_val2)

if(p\_val2 < 0.05):

    print('2st variable \'Reject\' the null hypothesis')

else:

    print('2st variable \'Accept\' the null hypothesis')

print('')

a = np.mean(x)\*\*2

b = np.sum((x - np.mean(x))\*\*2)

SE = var \* np.sqrt((1/n) + (np.mean(x1)\*\*2) / np.sum((x1 - np.mean(x1))\*\*2) + (np.mean(x2)\*\*2) / np.sum((x2 - np.mean(x2))\*\*2))

t\_statistic = regr.intercept\_ / SE

print("t\_statistic: ", t\_statistic)

if t\_statistic < 0:

    p\_val = t.cdf(t\_statistic, df = n-3)

else:

    p\_val = 1-t.cdf(t\_statistic, df = n-3)

print('p\_val: ', p\_val)

if(p\_val < 0.05):

    print('Reject the null hypothesis')

else:

    print('Accept the null hypothesis')