

**LAPORAN UTS KECERDASAN BUATAN  
“AI”**



Oleh:

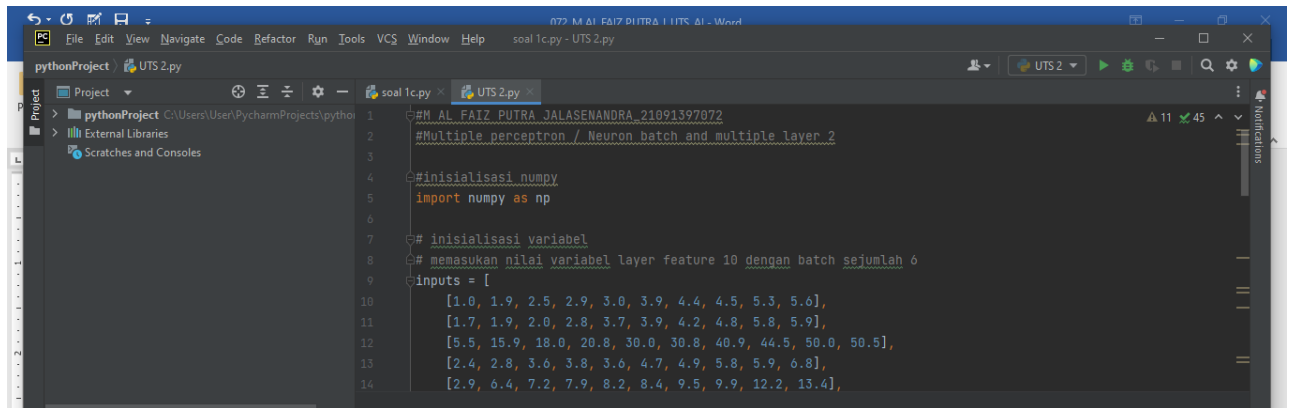
**M AL FAIZ P J**

**21091397072**

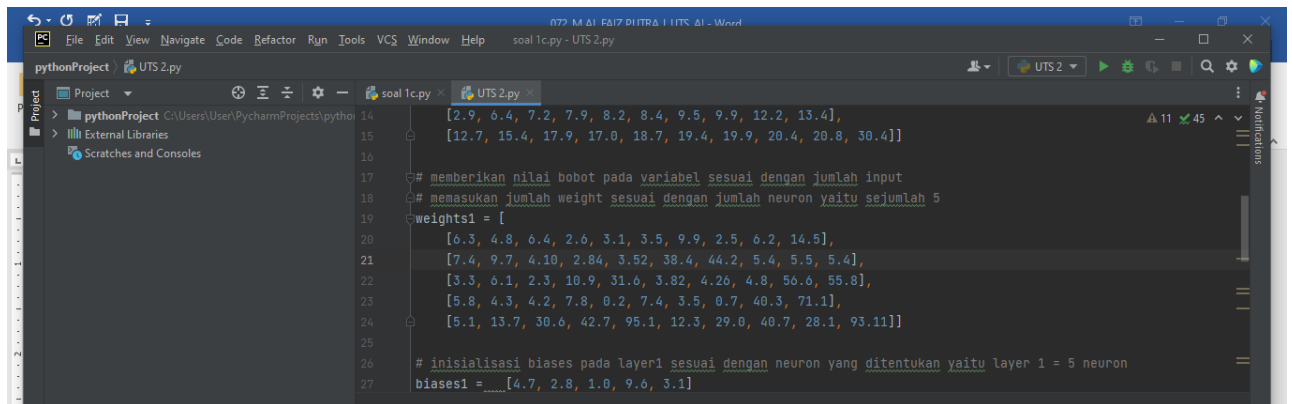
**UNIVERSITAS NEGERI SURABAYA  
FAKULTAS VOKASI  
PRODI D IV MANAJEMEN INFORMATIKA  
TAHUN AJARAN 2022/2023**

## Uts ke 2

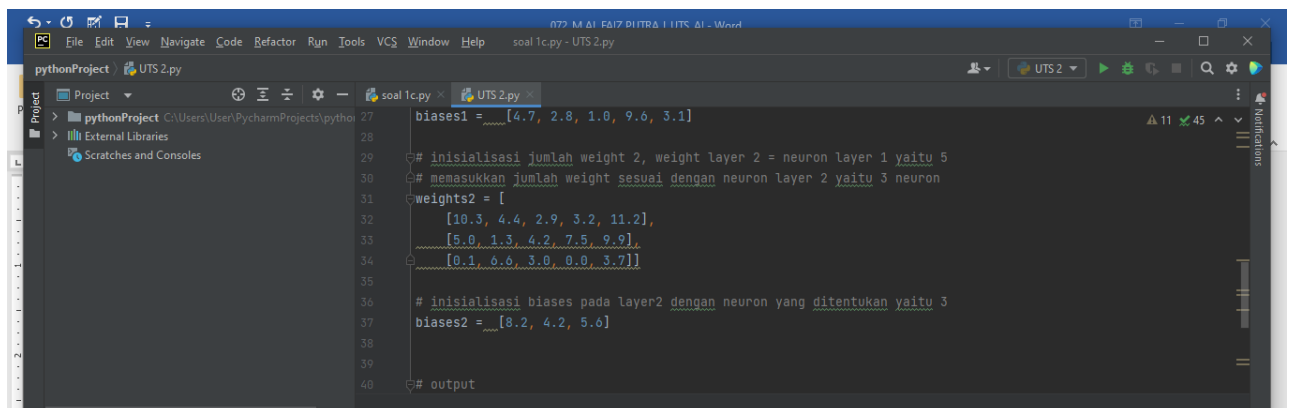
1.



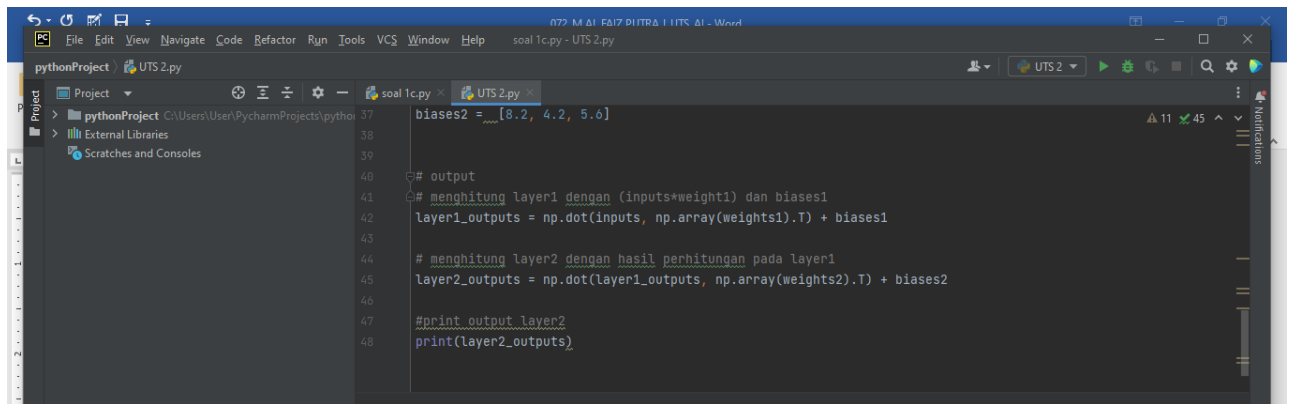
```
1 #M AL FAIZ PUTRA JALASENANDRA 21091397072
2 #Multiple perceptron // Neuron batch and multiple layer 2
3
4 #inisialisasi numpy
5 import numpy as np
6
7 # inisialisasi variabel
8 # memasukkan nilai variabel layer feature 10 dengan batch sejumlah 6
9 inputs = [
10     [1.0, 1.9, 2.5, 2.9, 3.0, 3.9, 4.4, 4.5, 5.3, 5.6],
11     [1.7, 1.9, 2.0, 2.8, 3.7, 3.9, 4.2, 4.8, 5.8, 5.9],
12     [5.5, 15.9, 18.0, 20.8, 30.0, 30.8, 40.9, 44.5, 50.0, 50.5],
13     [2.4, 2.8, 3.6, 3.8, 3.6, 4.7, 4.9, 5.8, 5.9, 6.8],
14     [2.9, 6.4, 7.2, 7.9, 8.2, 8.4, 9.5, 9.9, 12.2, 13.4],
```



```
14     [2.9, 6.4, 7.2, 7.9, 8.2, 8.4, 9.5, 9.9, 12.2, 13.4],
15     [12.7, 15.4, 17.9, 17.0, 18.7, 19.4, 19.9, 20.4, 20.8, 30.4]]
16
17 # memberikan nilai bobot pada variabel sesuai dengan jumlah input
18 # memasukkan jumlah weight sesuai dengan jumlah neuron yaitu sejumlah 5
19 weights1 = [
20     [6.3, 4.8, 6.4, 2.6, 3.1, 3.5, 9.9, 2.5, 6.2, 14.5],
21     [7.4, 9.7, 4.10, 2.84, 3.52, 38.4, 44.2, 5.4, 5.5, 5.4],
22     [3.3, 6.1, 2.3, 10.9, 31.6, 3.82, 4.26, 4.8, 56.6, 55.8],
23     [5.8, 4.3, 4.2, 7.8, 0.2, 7.4, 3.5, 0.7, 40.3, 71.1],
24     [5.1, 13.7, 30.6, 42.7, 95.1, 12.3, 29.0, 40.7, 28.1, 93.11]]
25
26 # inisialisasi biases pada layer1 sesuai dengan neuron yang ditentukan yaitu layer 1 = 5 neuron
27 biases1 = [4.7, 2.8, 1.0, 9.6, 3.1]
```

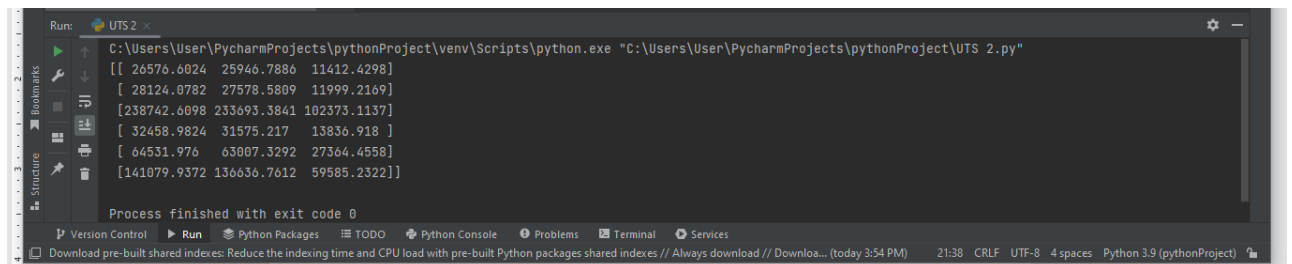


```
27 biases1 = [4.7, 2.8, 1.0, 9.6, 3.1]
28
29 # inisialisasi jumlah weight 2, weight layer 2 = neuron layer 1 yaitu 5
30 # memasukkan jumlah weight sesuai dengan neuron layer 2 yaitu 3 neuron
31 weights2 = [
32     [10.3, 4.4, 2.9, 3.2, 11.2],
33     [5.0, 1.3, 4.2, 7.5, 9.9],
34     [0.1, 6.6, 3.0, 0.0, 3.7]]
35
36 # inisialisasi biases pada layer2 dengan neuron yang ditentukan yaitu 3
37 biases2 = [8.2, 4.2, 5.6]
38
39
40 # output
```



```
37 biases2 = [8.2, 4.2, 5.0]
38
39
40 # output
41 # menghitung layer1 dengan (inputs*weight1) dan biases1
42 layer1_outputs = np.dot(inputs, np.array(weights1).T) + biases1
43
44 # menghitung layer2 dengan hasil perhitungan pada layer1
45 layer2_outputs = np.dot(layer1_outputs, np.array(weights2).T) + biases2
46
47 #print output layer2
48 print(layer2_outputs)
```

HASIL OUTPUT :



```
Run: UTS 2
C:\Users\User\PycharmProjects\pythonProject\venv\Scripts\python.exe "C:\Users\User\PycharmProjects\pythonProject\UTS 2.py"
[[ 26576.6024  25946.7886  11412.4298]
 [ 28124.0782  27578.5809  11999.2169]
 [238742.6098 235693.3841 102373.1137]
 [ 32458.9824  31575.217  13836.918 ]
 [ 64531.976  63007.3292  27364.4558]
 [141079.9372 136636.7612  59585.2322]]
Process finished with exit code 0
```

Penjelasan :

1. Inisialisasi numpy sebagai method perhitungan
2. Masukkan variabel untuk inputs, weights, dan bias sesuai dengan ketentuan  
Inputs = 10  
Batch = 6  
Weights1 = 5\*10  
Biases1 = 5  
Weights2 = 3\*5  
Biases2 = 3
3. Buat output untuk menghitung variabel yang telah kita masukkan/buat  
np.dot = untuk penghitungan vektor weight dan input  
kemudian hasil penghitungan vektor ditambahkan dengan biases
4. Untuk output yang diinginkan adalah output layer2 yang berasal dari hasil perhitungan layer1 lalu dihitung kembali dengan weights2 dan biases2
5. Buat command print untuk menampilkan hasil perhitungan output

## Penghitungan layer 1

perhitungan dot product

weight 10 <sup>5</sup>										input 6*10									
6.3	4.8	6.4	2.6	3.1	3.5	9.9	2.5	6.2	14.5	1.0	1.9	2.5	2.9	3.0	3.9	4.4	4.5	5.3	5.6
7.4	9.7	4.10	2.84	3.52	38.4	44.2	5.4	5.5	5.4	1.7	1.9	2.0	2.8	3.7	3.9	4.2	4.8	5.8	5.9
3.3	6.1	2.3	10.9	31.6	3.82	4.26	4.8	56.6	55.8	5.5	15.9	18.0	20.8	30.0	30.8	40.9	44.5	50.0	50.5
5.8	4.3	4.2	7.8	0.2	7.4	3.5	0.7	40.3	71.1	2.4	2.8	3.6	3.8	3.6	4.7	4.9	5.8	5.9	6.8
5.1	13.7	30.6	42.7	95.1	12.3	29.0	40.7	28.1	93.11	2.9	6.4	7.2	7.9	8.2	8.4	9.5	9.9	12.2	13.4
										12.7	15.4	17.9	17.0	18.7	19.4	19.9	20.4	20.8	30.4

weight\*input :

26576	6024	25946	7886	11412	4298
28124	0782	27578	5809	11999	2169
238742	6098	233693	3841	102373	1137
32458	9824	31575	217	13836	918
64531	976	63007	3292	27364	4558
141079	9372	136636	7612	59585	2322

kemudian np.dot + bias

26576	6024	25946	.7886	11412	4298
28124	0782	27578	.5809	11999	2169
238742	6098	233693	.3841	102373	1137
32458	9824	31575	.217	13836	918
64531	.976	63007	.3292	27364	4558
141079	9372	136636	.7612	59585	2322

+

[4.7, 2.8, 1.0, 9.6, 3.1]

230.78	482.806	814.752	706.85	1545.826
240.12	485.266	882.57	749.16	1644.789
2039.46	4211.902	7484.65	6351.88	14020.985
287.28	580.694	968.508	848.68	1900.288
559.17	1099.74	1968.718	1684.58	3812.184
1256.33	2444.114	4083.642	3578.57	8388.784

## Penghitungan layer 2

$$\begin{array}{c} \text{weights 2} \\ 5 \times 3 \end{array} \begin{bmatrix} 10.3 & 4.4 & 2.9 & 3.2 & 11.2 \\ 5.0 & 1.3 & 4.2 & 7.5 & 9.9 \\ 0.1 & 6.6 & 3.0 & 0.0 & 3.7 \end{bmatrix} * \begin{array}{c} \text{output layer 1} \\ 5 \times 6 \end{array} = \begin{bmatrix} 230.78 & 482.806 & 814.752 & 706.85 & 1545.826 & 240.12 & 485.266 & 882.57 & 749.16 & 1644.789 \\ 2039.46 & 4211.902 & 7484.65 & 6351.88 & 14020.985 & 287.28 & 580.694 & 968.508 & 848.68 & 1900.288 \\ 559.17 & 1099.74 & 1968.718 & 1684.58 & 3812.184 & 1256.33 & 2444.114 & 4083.642 & 3578.57 & 8388.784 \end{bmatrix}$$

$$\text{weight2* output layer1} = \begin{bmatrix} 26568.4024 & 25942.5886 & 11406.8298 \\ 28115.8782 & 27574.3809 & 11993.6169 \\ 238734.4098 & 233689.1841 & 102367.5137 \\ 32450.7824 & 31571.017 & 13831.318 \\ 64523.776 & 63003.1292 & 27358.8558 \\ 141071.7372 & 136632.5612 & 59579.6322 \end{bmatrix}$$

$$\begin{array}{c} \text{kemudian np.dot + bias} \\ \swarrow \quad \searrow \end{array} \begin{bmatrix} 26568.4024 & 25942.5886 & 11406.8298 \\ 28115.8782 & 27574.3809 & 11993.6169 \\ 238734.4098 & 233689.1841 & 102367.5137 \\ 32450.7824 & 31571.017 & 13831.318 \\ 64523.776 & 63003.1292 & 27358.8558 \\ 141071.7372 & 136632.5612 & 59579.6322 \end{bmatrix} + [8.2, 4.2, 5.6]$$

$$= \begin{array}{c} 26576.6024 & 25946.7886 & 11412.4298 \\ 28124.0782 & 27578.5809 & 11999.2169 \\ 238742.6098 & 233693.3841 & 102373.1137 \\ 32458.9824 & 31575.217 & 13836.918 \\ 64531.976 & 63007.3292 & 27364.4558 \\ 141079.9372 & 136636.7612 & 59585.2322 \end{array}$$