LAPORAN PENGANTAR KECERDASAN BUATAN

TUGAS PEMROGRAMAN 2

Disusun untuk memenuhi salah satu tugas mata kuliah Pengantar Kecerdasan Buatan

Disusun oleh kelompok 6 :

1.	Herjanto Janawisuta	1301200421 / IF-44-08
2.	Hilman Taris Muttaqin	1301204208 / IF-44-08

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY

2022

1. Deskripsi Tugas

Diberikan file bengkel.xlsx berupa himpunan data 100 bengkel mobil yang ada di kota Bandung dengan dua atribut: Kualitas Servis (bilangan real 1-100; semakin tinggi semakin baik) dan Harga (bilangan real 1-10, semakin tinggi semakin mahal). Bangunlah sebuah sistem berbasis Fuzzy Logic untuk memilih 10 bengkel terbaik di kota Bandung. Sistem membaca masukan file bengkel.xlsx dan mengeluarkan output berupa sebuah file peringkat.xlsx yang berisi 10 nomor/ID bengkel terbaik beserta skor-nya (output Defuzzification).

2. Desain dan Analisa

- **Jumlah dan Nama Linguistik setiap atribut**

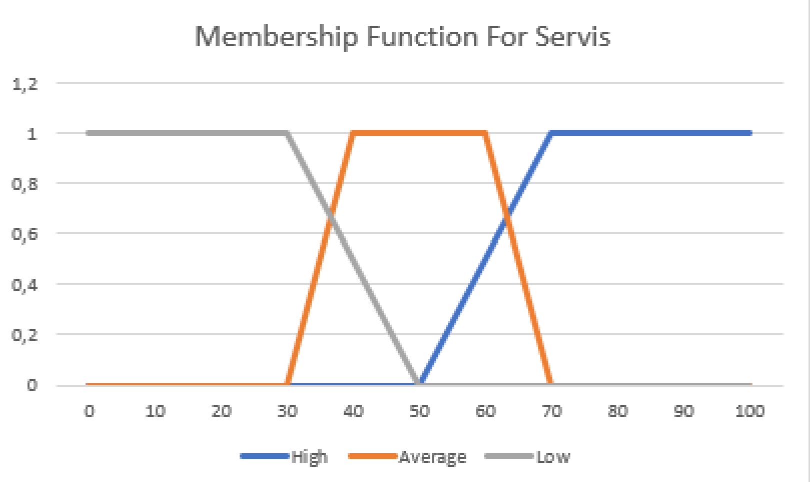
  Kualitas Servis : High, Average, Low

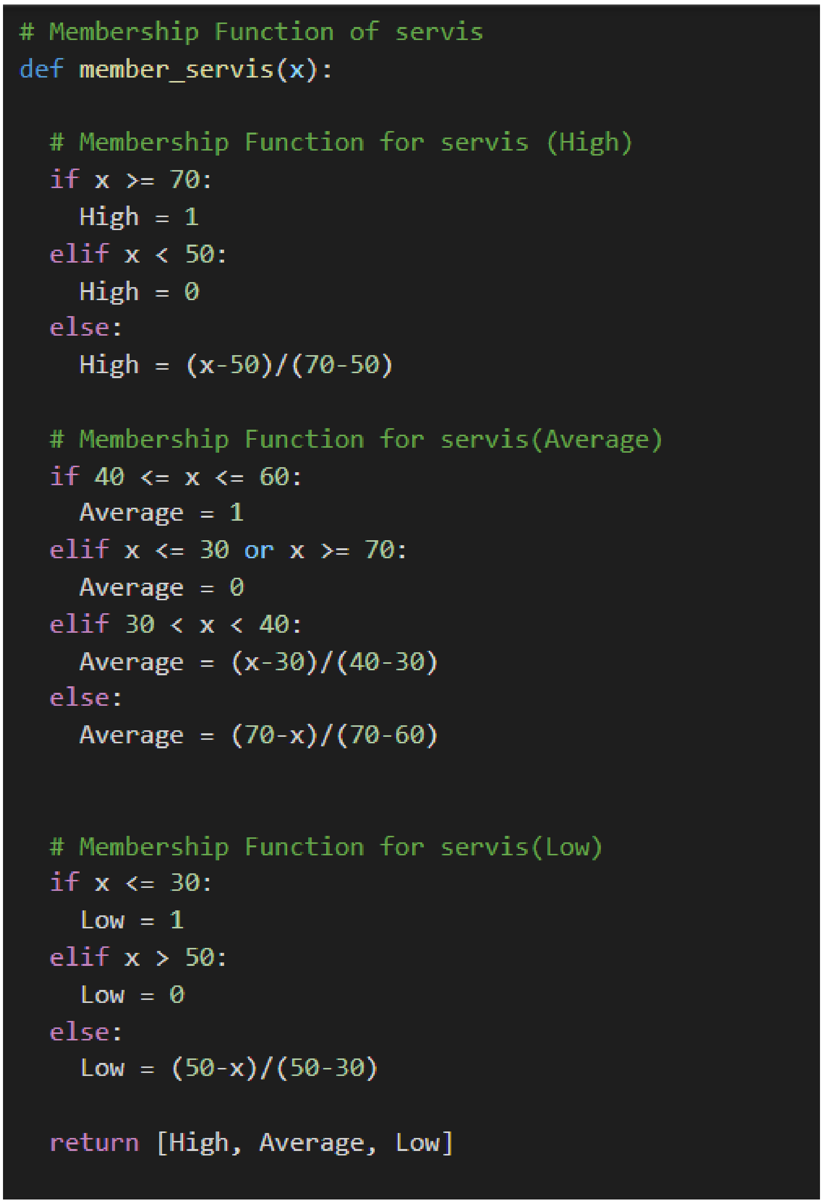
  Harga : Expensive, Affordable, Cheap

  Score : Good, Normal, Bad

- **Bentuk dan Batas Fungsi Keanggotaan Input**
  - Service Quality

    servis >= 70 is definitely High and servis < 50 is definitely not High Average servis is considered between 40 to 60, while servis <= 30 or servis >= 70 is considered not average servis <= 30 is definitely Low and servis > 50 is definitely not Low.

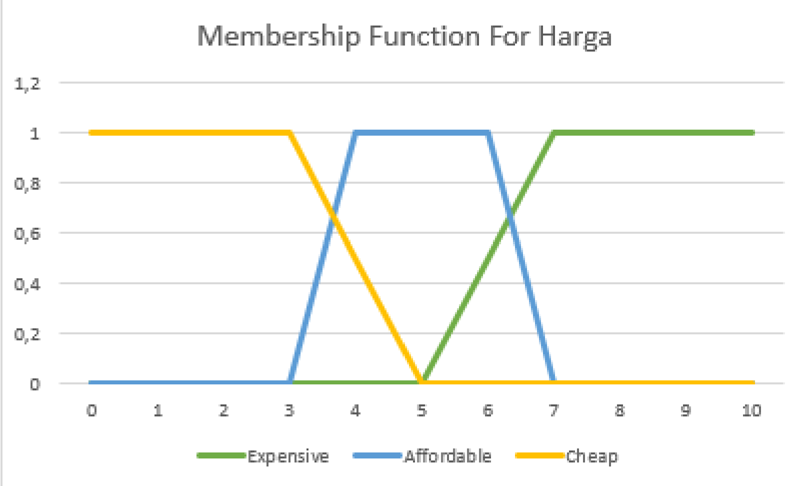


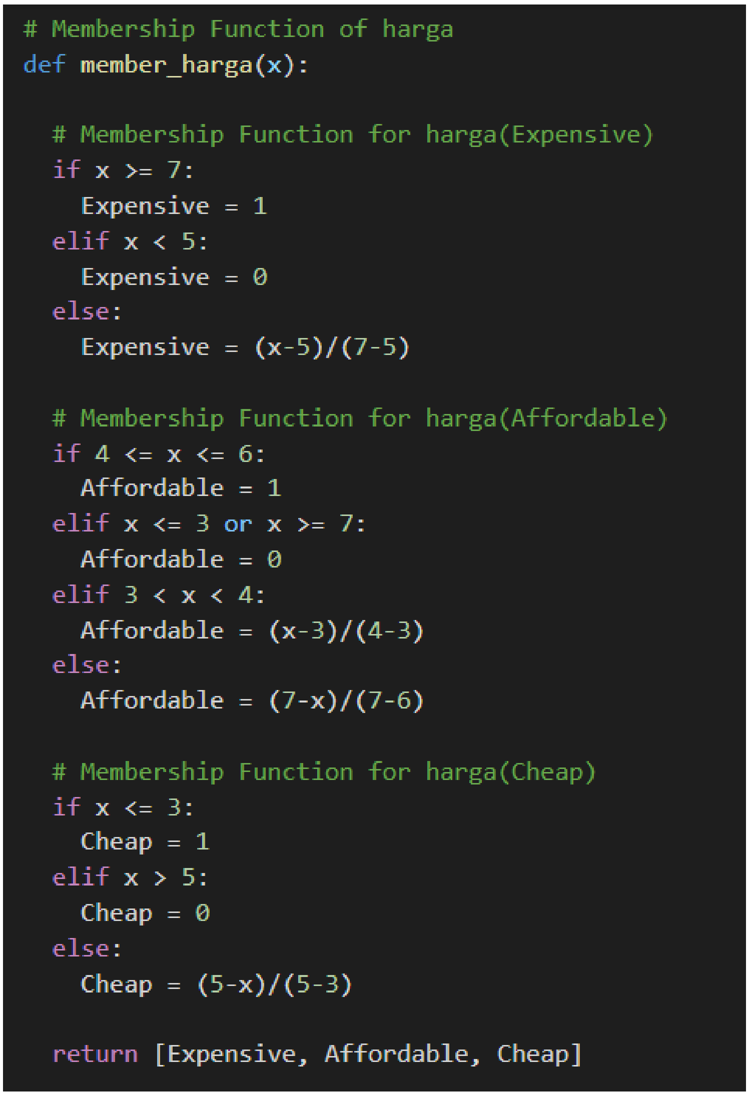

```
# Membership Function of servis
def member_servis(x):

  # Membership Function for servis (High)
  if x >= 70:
    High = 1
  elif x < 50:
    High = 0
  else:
    High = (x-50)/(70-50)

  # Membership Function for servis(Average)
  if 40 <= x <= 60:
    Average = 1
  elif x <= 30 or x >= 70:
    Average = 0
  elif 30 < x < 40:
    Average = (x-30)/(40-30)
  else:
    Average = (70-x)/(70-60)


  # Membership Function for servis(Low)
  if x <= 30:
    Low = 1
  elif x > 50:
    Low = 0
  else:
    Low = (50-x)/(50-30)

  return [High, Average, Low]
```

- Price

harga>=7 is definitely Expensive and harga<5 is definitely not

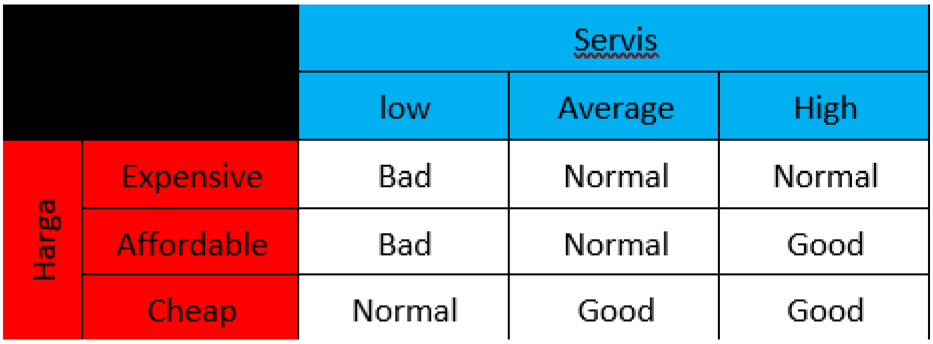
Expensive Affordable harga is considered between 4 to 6, while harga <= 3 or harga >= 7 is considered not Affordable harga <= 3 is definitely Cheap and harga > 5 is definitely not Cheap





```
# Membership Function of harga
def member_harga(x):

  # Membership Function for harga(Expensive)
  if x >= 7:
    Expensive = 1
  elif x < 5:
    Expensive = 0
  else:
    Expensive = (x-5)/(7-5)

  # Membership Function for harga(Affordable)
  if 4 <= x <= 6:
    Affordable = 1
  elif x <= 3 or x >= 7:
    Affordable = 0
  elif 3 < x < 4:
    Affordable = (x-3)/(4-3)
  else:
    Affordable = (7-x)/(7-6)

  # Membership Function for harga(Cheap)
  if x <= 3:
    Cheap = 1
  elif x > 5:
    Cheap = 0
  else:
    Cheap = (5-x)/(5-3)

  return [Expensive, Affordable, Cheap]
```

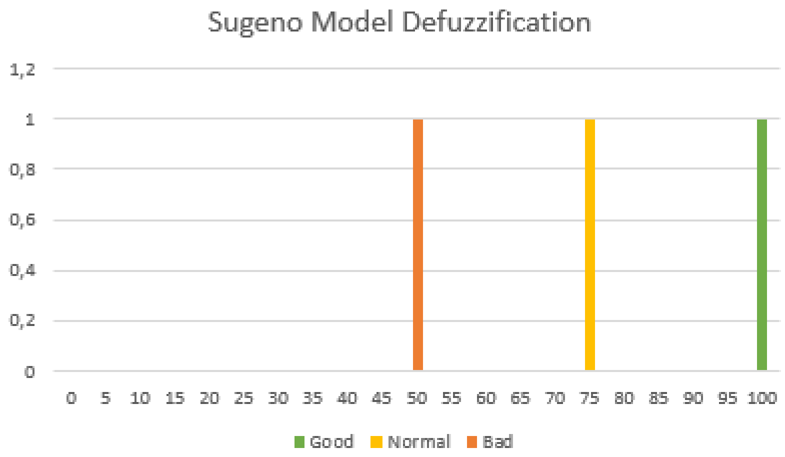
- **Aturan Inferensi**

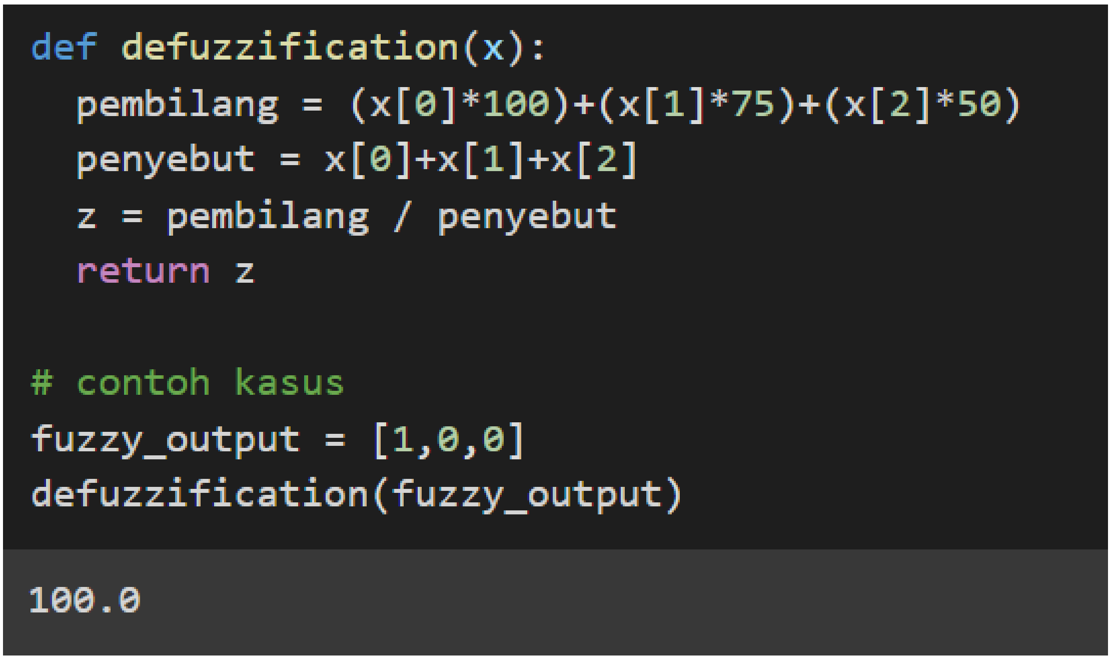


		Servis		
		low	Average	High
Harga	Expensive	Bad	Normal	Normal
	Affordable	Bad	Normal	Good
	Cheap	Normal	Good	Good

- **Metode Defuzzification**

  Metode yang digunakan untuk mendapatkan output yang diinginkan adalah **metode defuzzification sugeno**.

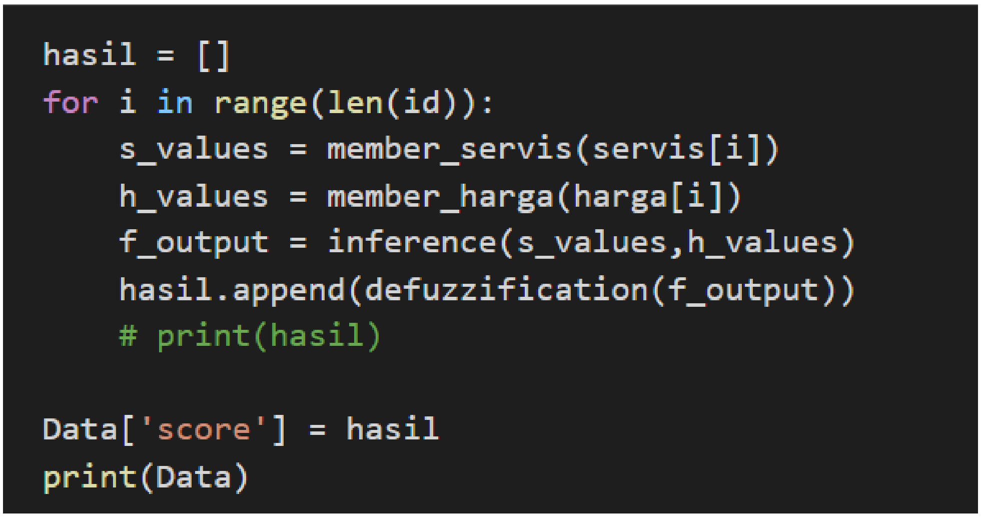




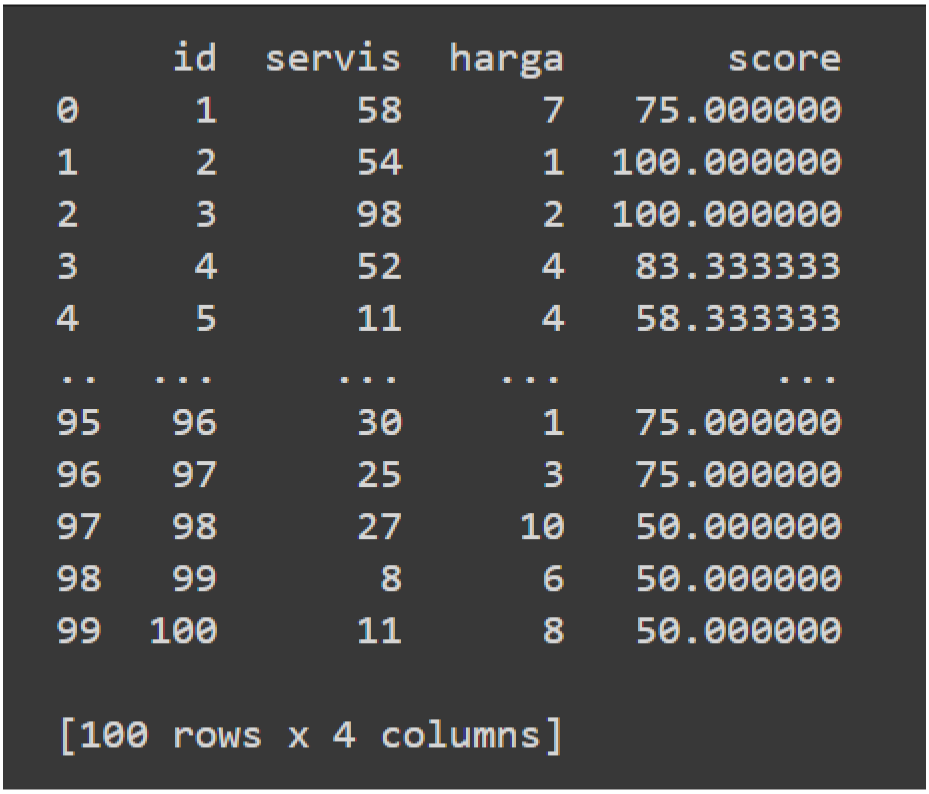
```
def defuzzification(x):
  pembilang = (x[0]*100)+(x[1]*75)+(x[2]*50)
  penyebut = x[0]+x[1]+x[2]
  z = pembilang / penyebut
  return z

# contoh kasus
fuzzy_output = [1,0,0]
defuzzification(fuzzy_output)
```

```
100.0
```

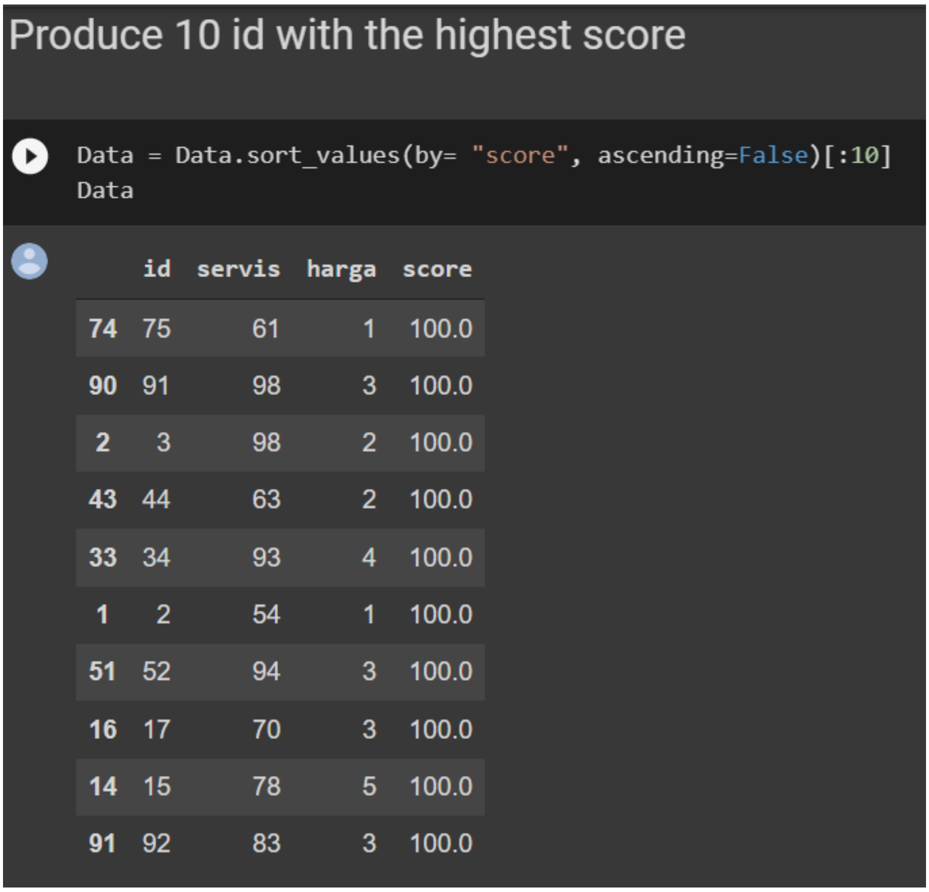
- **Bentuk dan Batas Fungsi Keanggotaan Output (sesuai metode Defuzzification)**



```python
hasil = []
for i in range(len(id)):
    s_values = member_servis(servis[i])
    h_values = member_harga(harga[i])
    f_output = inference(s_values,h_values)
    hasil.append(defuzzification(f_output))
    # print(hasil)

Data['score'] = hasil
print(Data)
```



```
     id  servis  harga       score
0     1      58      7   75.000000
1     2      54      1  100.000000
2     3      98      2  100.000000
3     4      52      4   83.333333
4     5      11      4   58.333333
..  ...     ...    ...         ...
95   96      30      1   75.000000
96   97      25      3   75.000000
97   98      27     10   50.000000
98   99       8      6   50.000000
99  100      11      8   50.000000

[100 rows x 4 columns]
```

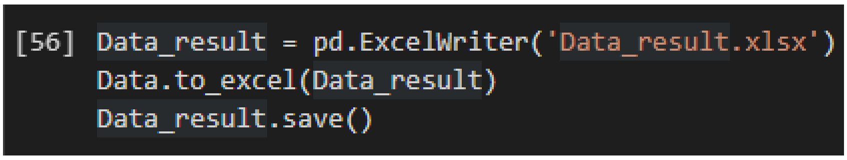


Produce 10 id with the highest score

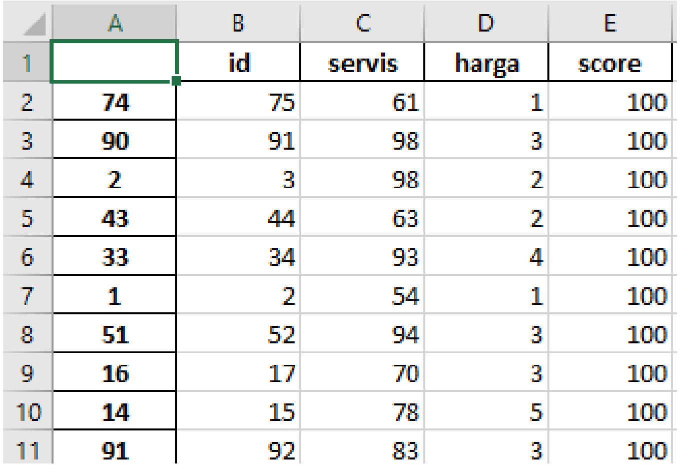
```
Data = Data.sort_values(by= "score", ascending=False)[:10]
Data
```

	id	servis	harga	score
74	75	61	1	100.0
90	91	98	3	100.0
2	3	98	2	100.0
43	44	63	2	100.0
33	34	93	4	100.0
1	2	54	1	100.0
51	52	94	3	100.0
16	17	70	3	100.0
14	15	78	5	100.0
91	92	83	3	100.0

- **Menyimpan ouput ke file excel**



```
[56] Data_result = pd.ExcelWriter('Data_result.xlsx')
     Data.to_excel(Data_result)
     Data_result.save()
```



	A	B	C	D	E
1		id	servis	harga	score
2	74	75	61	1	100
3	90	91	98	3	100
4	2	3	98	2	100
5	43	44	63	2	100
6	33	34	93	4	100
7	1	2	54	1	100
8	51	52	94	3	100
9	16	17	70	3	100
10	14	15	78	5	100
11	91	92	83	3	100

3. Pembagian Tugas

a. Herjanto : Fuzzification, membership function service, defuzzification, implementasi (playground)

b. Hilman : Membership function price, inference, rekam video

4. Link

- Google Colab
  https://colab.research.google.com/drive/14Gk0RyR03ryfjAzvWdTkPAHWL_UntARt#scrollTo=nBo-IWR4q7Ll
- Video Rekaman
  https://youtu.be/bm5si5BKjbM