

PEMROGRAMAN ANALISIS DATA
“DECISION MAKING”
TUGAS KE – 2



DISUSUN OLEH :
Nama : Andreas Nathanael Priambodo
NIM : 215314043

**TEKNIK INFORMATIKA FAKULTAS SAINS DAN
TEKNOLOGI UNIVERSITAS SANATA DHARMA
YOGYAKARTA 2023**

Exercise 38: Name That Shape

(Solved, 31 Lines)

Write a program that determines the name of a shape from its number of sides. Read the number of sides from the user and then report the appropriate name as part of a meaningful message. Your program should support shapes with anywhere from 3 up to (and including) 10 sides. If a number of sides outside of this range is entered then your program should display an appropriate error message.

Exercise 39: Month Name to Number of Days

(Solved, 18 Lines)

The length of a month varies from 28 to 31 days. In this exercise you will create a program that reads the name of a month from the user as a string. Then your program should display the number of days in that month. Display “28 or 29 days” for February so that leap years are addressed.

Exercise 40: Sound Levels

(30 Lines)

The following table lists the sound level in decibels for several common noises.

Noise	Decibel Level
Jackhammer	130 dB
Gas Lawnmower	106 dB
Alarm Clock	70 dB
Quiet Room	40 dB

Write a program that reads a sound level in decibels from the user. If the user enters a decibel level that matches one of the noises in the table then your program should display a message containing only that noise. If the user enters a number of decibels between the noises listed then your program should display a message indicating which noises the value is between. Ensure that your program also generates reasonable output for a value smaller than the quietest noise in the table, and for a value larger than the loudest noise in the table.

Exercise 41: Classifying Triangles

(Solved, 21 Lines)

A triangle can be classified based on the lengths of its sides as equilateral, isosceles or scalene. All three sides of an equilateral triangle have the same length. An isosceles triangle has two sides that are the same length, and a third side that is a different length. If all of the sides have different lengths then the triangle is scalene.

Write a program that reads the lengths of the three sides of a triangle from the user. Then display a message that states the triangle’s type.

Exercise 42: Note to Frequency

(Solved, 39 Lines)

The following table lists an octave of music notes, beginning with middle C, along with their frequencies.

Note	Frequency (Hz)
C4	261.63
D4	293.66
E4	329.63
F4	349.23
G4	392.00
A4	440.00
B4	493.88

Begin by writing a program that reads the name of a note from the user and displays the note's frequency. Your program should support all of the notes listed previously.

Once you have your program working correctly for the notes listed previously you should add support for all of the notes from C0 to C8. While this could be done by adding many additional cases to your `if` statement, such a solution is cumbersome, inelegant and unacceptable for the purposes of this exercise. Instead, you should exploit the relationship between notes in adjacent octaves. In particular, the frequency of any note in octave n is half the frequency of the corresponding note in octave $n + 1$. By using this relationship, you should be able to add support for the additional notes without adding additional cases to your `if` statement.

Hint: You will want to access the characters in the note entered by the user individually when completing this exercise. Begin by separating the letter from the octave. Then compute the frequency for that letter in the fourth octave using the data in the table above. Once you have this frequency you should divide it by 2^{4-x} , where x is the octave number entered by the user. This will halve or double the frequency the correct number of times.

Exercise 43: Frequency to Note

(Solved, 42 Lines)

In the previous question you converted from a note's name to its frequency. In this question you will write a program that reverses that process. Begin by reading a frequency from the user. If the frequency is within one Hertz of a value listed in the table in the previous question then report the name of the corresponding note. Otherwise report that the frequency does not correspond to a known note. In this exercise you only need to consider the notes listed in the table. There is no need to consider notes from other octaves.

Exercise 44: Faces on Money

(31 Lines)

It is common for images of a country's previous leaders, or other individuals of historical significance, to appear on its money. The individuals that appear on banknotes in the United States are listed in below.

George Washington	\$1
Thomas Jefferson	\$2
Abraham Lincoln	\$5
Alexander Hamilton	\$10
Andrew Jackson	\$20
Ulysses S. Grant	\$50
Benjamin Franklin	\$100

Write a program that begins by reading the denomination of a banknote from the user. Then your program should display the name of the individual that appears on the banknote of the entered amount. An appropriate error message should be displayed if no such note exists.

While two dollar banknotes are rarely seen in circulation in the United States, they are legal tender that can be spent just like any other denomination. The United States has also issued banknotes in denominations of \$500, \$1,000, \$5,000, and \$10,000 for public use. However, high denomination banknotes have not been printed since 1945 and were officially discontinued in 1969. As a result, we will not consider them in this exercise.

B. Algoritma

- Exercise 38

Masukan daftar sisi dan nama bentuk ke dalam kamus

Masukan Jumlah sisi

Untuk kamus dalam item

Jika sisi sama dengan item maka nama sama dengan item kamus

Jika nama kosong maka cetak tidak ada daftar nama

Jika tidak cetak bentuk dan nama bentuk

- Exercise 39

Masukan nama bulan ke dalam bulan

Hari sama dengan 30

jika bulan sama dengan februari maka hari sama dengan 28/29

jika bulan sama dengan April atau June atau septermbet atau November maka
hari sama dengan 31

cetak hari

- Exercise 40

Masukan frekuensi suara ke dalam freq

Jika freq kurang dari sama dengan 130 dan lebih dari 106 maka decide sama
dengan jackhammer

Jika freq kurang dari sama dengan 106 dan lebih dari 70 maka decide sama
dengan Gas lawnmower

Jika freq kurang dari sama dengan 70 dan lebih dari 40 maka decide sama
dengan alarm clock

Jika freq kurang dari sama dengan 40 dan freq lebih dari 0 maka decide sama
dengan quiet room

Jika freq kurang dari sama dengan 0 maka decide sama dengan kosong

Cetak decide

- Exercise 41

Masukan sisi 1 ke dalam sisi1

Masukan sisi 2 ke dalam sisi2

Masukan sisi 3 kedalam sisi3

Jika sisi1 sama dengan sisi2 sama dengan sisi3 maka segi sama dengan
segitiga sama sisi

Lain jika sisi1 sama dengan sisi2 tidak sama dengan sisi3 atau sisi2 sama dengan sisi4 tidak sama dengan sisi1 atau sisi3 sama dengan sisi1 tidak sama dengan sisi2 maka segi sama dengan segitiga sama kaki
jika tidak maka segi sama dengan segitiga sembarang
cetak segi

- Exercise 42

Number sama dengan kosong

Masukan data table ke dalam kamus

Untuk dict dalam item

Jika data sama dengan item maka number sama dengan kampus item

Jika number sama dengan kosong maka keluar

Jika tidak maka result sama dengan rumus

- Exercise 43

Fetch_data sama dengan kosong

Masukan data table ke dalam kamus

Untuk note_freq dan freq_hz dalam kamus

Jika data sama dengan freq_hz maka fetch_data sama dengan
note_freq

Jika fetch_data sama dengan kosong

Jika tidak maka cetak fetch_data

- Exercise 44

Fetch_data sama dengan kosong

Masukan data table ke dalam kamus

Jika individual dan amount dalam kampus

Jika data sama dengan amount maka fetch_data sama dengan
individual

C. ScreenShot

- Exercise 38

```
Laporan2 > exercise38.py > ...
1 name = None
2 dict = {3:"Triangle", 4:"Quadrangle", 5:"Pentagon", 6:"Hexagon", 7:"Heptagon", 8:"Octagon", 9:"Nonagon", 10:"Decagon"}
3 print("=====")
4 sisi = int(input("masukan jumlah sisi : "))
5
6 for item in dict.keys() :
7     if sisi == item :
8         name = dict.get(item)
9
10 if name is None :
11     print("No data name Found!")
12 else :
13     print("The shape name is : ",name)
14 print("=====")
```

- Exercise 39

```
Laporan2 > exercise39.py > ...
1 print("=====")
2 bulan = input("Masukan nama Bulan : ")
3 hari = "30"
4 if bulan == "februari" :
5     hari = "28/29"
6 if bulan == "april" or bulan == "june" or bulan == "september" or bulan == "november" :
7     hari = "31"
8 print(hari)
9 print("=====")
```

- Exercise 40

```
Laporan2 > exercise40.py > ...
1 print("=====")
2 freq = int(input("Insert Sound Frequency : "))
3 print("Deciding Which one is the sound...")
4 if freq <= 130 and freq > 106 :
5     decide = "Jackhammer"
6 if freq <= 106 and freq > 70 :
7     decide = "Gas Lawnmower"
8 if freq <= 70 and freq > 40 :
9     decide = "Alarm Clock"
10 if freq <= 40 and freq > 0 :
11     decide = "Quiet Room"
12 if freq <= 0 :
13     decide = "what is this shit ?"
14 print("The sound of ", decide)
15 print("=====")
```

- Exercise 41

Laporan2 > exercise41.py > ...

```
1 print("=====")
2 sisi1 = int(input("Masukan panjang sisi 1 : "))
3 sisi2 = int(input("Masukan panjang sisi 2 : "))
4 sisi3 = int(input("Masukan panjang sisi 3 : "))
5
6 if sisi1 == sisi2 == sisi3 :
7     segi = "Segitiga Sama sisi"
8 elif sisi1 == sisi2 != sisi3 or sisi2 == sisi3 != sisi1 or sisi3 == sisi1 != sisi2 :
9     segi = "Segitiga sama kaki"
10 else :
11     segi = "Segitiga Sembarang"
12 print(segi)
13 print("=====")
```

- Exercise 42

Laporan2 > exercise42.py > ...

```
1 import math
2 print("=====")
3 number = None
4 dict = {"C4":261.63, "D4":293.66, "E4":329.63, "F4":349.23, "G4":392.00, "A4":440.00, "B4":493.88}
5 print("{:<10} {:<}".format('Note', 'Frequency (Hz)'))
6
7 for value in dict.items() :
8     note, freq = value
9     print("{:<10} {:<10}".format(note, freq))
10
11 data = str(input("Masukan data yang akan dicari : "))
12 for item in dict.keys() :
13     if data == item :
14         number = dict.get(item)
15
16 if number is None :
17     quit()
18 else :
19     result = float(number) / float(math.pow(2,(4 - float(number))))
20     print(result)
21 print("=====")
```

- Exercise 43

Laporan2 > exercise43.py > ...

```
1 print("=====")
2 fetch_data = None
3 dict = {"C4":261.63, "D4":293.66, "E4":329.63, "F4":349.23, "G4":392.00, "A4":440.00, "B4":493.88}
4 print("{:<10} {:<}".format('Note', 'Frequency (Hz)'))
5
6 for value in dict.items() :
7     note, freq = value
8     print("{:<10} {:<10}".format(note, freq))
9
10 data = float(input("Masukan data yang akan dicari : "))
11 for note_freq, freq_hz in dict.items() :
12     if data == freq_hz :
13         fetch_data = note_freq
14
15 if fetch_data is None :
16     print("Null data Fetched")
17     quit()
18 else :
19     print("The Note is ", fetch_data)
20 print("=====")
```


- Exercise 44

```
Laporan2 > exercise44.py > ...
1 print("=====")
2 fetch_data = None
3 dict = {"George Washington":1, "Thomas Jefferson":2, "Abraham Lincoln":5, "Alexander Hamilton":10, "Andrew
4 Jackson":20, "Ulysses S. Grant":50, "Benjamin Franklin":100}
5 print("{:<30} {:<30}".format('Individual', 'Amount'))
6
7 for value in dict.items() :
8     indv, amount = value
9     print("{:<30} {:<30}".format(indv, amount))
10
11 data = float(input("Masukan data yang akan dicari : "))
12 for individual, amnt in dict.items() :
13     if data == amnt :
14         fetch_data = individual
15
16 if fetch_data is None :
17     print("No Data Found!")
18 else :
19     print("The name of the denomination of a banknote ",fetch_data)
20 print("=====")
```

D. Analisis Hasil

- Exercise 38

```
PS C:\Users\natha\OneDrive\Documents\Python> & C:/Users/natha/AppData/L
Laporan2/exercise38.py
=====
masukan jumlah sisi : 3
The shape name is : Triangle
=====
PS C:\Users\natha\OneDrive\Documents\Python>

PS C:\Users\natha\OneDrive\Documents\Python> & C:/Users/natha/AppData/L
Laporan2/exercise38.py
=====
masukan jumlah sisi : 11
No data name Found!
=====
PS C:\Users\natha\OneDrive\Documents\Python>
```

Menggunakan variable nama yang diberikan data kosong lalu ada dictionary python yang digunakan sebagai penyimpanan nilai data salam pasangan untuk kunci : nilai yang disini dilakukan adalah untuk berapa jumlah sisi sama dengan nama dari bentuk tersebut sebagai contoh 3 sama dengan triangle dan seterusnya selanjutnya menggunakan input dan memasukanya didalam variable sisi selanjutnya ada for yang looping untuk kamus dan mengambil keys atau kunci yang ada di kamus sebelumnya dan disimpan didalam variable item selanjutnya dibawah ada if untuk mengecek apakah sisi untuk inputan sebelumnya apakah sama dengan item dimana item sebelumnya berasal dari kamus jika iya maka nama sama dengan data kamus yang diambil nilai dari item tadi lalu cetak

- Exercise 39

```
/Python/Python311/python.exe c:/Users/natha/OneDrive/Documents/Python/Lap
ise39.py
```

```
=====
Masukan nama Bulan :
```

```
=====
Masukan nama Bulan : february
28/29
=====
PS C:\Users\natha\OneDrive\Documents\Python>
```

Menggunakan inputan dan memasukkannya didalam variable bulan dan memasukan data string 30 untuk hari lalu selanjutnya mengecek bulan sama dengan february maka hari sama dengan 28/29 lalu ada if lagi untuk mencetak apakah bulan sama dengan April atau June atau september atau November jika iya maka hari menjadi 31 lalu cetak hari

- Exercise 40

```
Laporan2/exercise40.py
=====
Insert Sound Frequency :
```

```
Laporan2/exercise40.py
=====
Insert Sound Frequency : 2
Deciding Which one is the sound...
The sound of Quiet Room
=====
PS C:\Users\natha\OneDrive\Documents\Python>
```

```
PS C:\Users\natha\OneDrive\Documents\Python> & C:/Users/natha/AppData/Local/Programs/Py
Laporan2/exercise40.py
=====
Insert Sound Frequency :
```

```

Laporan2/exercise10.py
=====
Insert Sound Frequency : -1
Deciding Which one is the sound...
The sound of what is this shit ?
=====
PS C:\Users\natha\OneDrive\Documents\Python>

```

Pada awal program menggunakan input untuk dimasukan kedalam freq yang inputanya menggunakan integer lalu selanjutnya mengecek apakah freq itu merupakan kurang dari sama dengan 130 atau lebih dari 106 jika iya maka akan mengisi decide menjadi jackhammer jika tidak maka akan langsung masuk ke perintah if setelahnya yang didalam isinya mengisi variable decide menjadi sesuai dengan freq dengan contohnya lalu cetak

- Exercise 41

```

PS C:\Users\natha\OneDrive\Documents\Python> cd C:\Users\natha\AppData\Local\Microsoft\Windows\OneDrive\OneDrive\Documents\Python
Laporan2/exercise41.py
=====
Masukan panjang sisi 1 :

```

```

=====
Masukan panjang sisi 1 : 12
Masukan panjang sisi 2 :

```

```

=====
Masukan panjang sisi 1 : 12
Masukan panjang sisi 2 : 12
Masukan panjang sisi 3 :

```

```

Laporan2/exercise41.py
=====
Masukan panjang sisi 1 : 12
Masukan panjang sisi 2 : 12
Masukan panjang sisi 3 : 12
Segitiga Sama sisi
=====
PS C:\Users\natha\OneDrive\Documents\Python>

```

```

Laporan2/exercise41.py
=====
Masukan panjang sisi 1 : 12
Masukan panjang sisi 2 : 12
Masukan panjang sisi 3 : 3
Segitiga sama kaki
=====
PS C:\Users\natha\OneDrive\Documents\Python>

```

Memasukan data inputan yang berupa integer untuk 3 inputan yang dimasukan ke masing masing variable sisi1 sisi2 sisi3 setelah itu mengecek apakah sisi1 sisi2 sisi3 sama jika iya maka akan mengisikan variable segi menjadi segitiga sama sisi jika tidak maka akan mencoba mencari lagi apakah berbentuk sama kaki atau segitiga sembarang kemudian segi tadi dicetak

- Exercise 42

```
PS C:\Users\natha\OneDrive\Documents\Python> & C:/Users/natha/AppData/Local/Programs/Python/Python39-6/Python.exe C:/Users/natha/AppData/Local/Programs/Python/Python39-6/Python.exe Laporan2/exercise42.py
=====
Note      Frequency (Hz)
C4        261.63
D4        293.66
E4        329.63
F4        349.23
G4        392.0
A4        440.0
B4        493.88
Masukan data yang akan dicari :
```

```
PS C:\Users\natha\OneDrive\Documents\Python> & C:/Users/natha/AppData/Local/Programs/Python/Python39-6/Python.exe C:/Users/natha/AppData/Local/Programs/Python/Python39-6/Python.exe Laporan2/exercise42.py
=====
Note      Frequency (Hz)
C4        261.63
D4        293.66
E4        329.63
F4        349.23
G4        392.0
A4        440.0
B4        493.88
Masukan data yang akan dicari : C4
9.376598584835433e+79
=====
PS C:\Users\natha\OneDrive\Documents\Python> |
```

```

PS C:\Users\natha\OneDrive\Documents\Python> & C:/Users/natha/AppData/Local/Programs/Python/Python38-32/Python.exe C:/Users/natha/AppData/Local/Programs/Python/Python38-32/Python.exe Laporan2/exercise42.py
=====
Note      Frequency (Hz)
C4        261.63
D4        293.66
E4        329.63
F4        349.23
G4        392.0
A4        440.0
B4        493.88
Masukan data yang akan dicari : 4
PS C:\Users\natha\OneDrive\Documents\Python>

```

Untuk pertama memasukan data kosong kedalam number dan juga memasukan data yang ada didalam table ke dalam kamus lalu di cetak datanya yang pertama dicetak untuk menuliskan judul format table nya yang kedua mencetak note dan freq lalu ada inputa data berbentuk string yang dimasukan kedalam var data lalu ada for untuk mengambil kunci dari kamus yang dimasukan kedalam item lalu ada if statement untuk mengecek apakah var item tadi sama dengan data atau inputan sebelumnya jika iya maka number akan diisi dengan data yang ada di dalam kamus untuk item nya lalu selanjutnya cetak hasil perhitugangan

- Exercise 43

```

PS C:\Users\natha\OneDrive\Documents\Python> & C:/Users/natha/AppData/Local/Programs/Python/Python38-32/Python.exe C:/Users/natha/AppData/Local/Programs/Python/Python38-32/Python.exe Laporan2/exercise43.py
=====
Note      Frequency (Hz)
C4        261.63
D4        293.66
E4        329.63
F4        349.23
G4        392.0
A4        440.0
B4        493.88
Masukan data yang akan dicari :

```

```
Laporan2/exercise43.py
```

```
=====
```

Note	Frequency (Hz)
C4	261.63
D4	293.66
E4	329.63
F4	349.23
G4	392.0
A4	440.0
B4	493.88

```
Masukan data yang akan dicari : 261.63
```

```
The Note is C4
```

```
=====
```

```
PS C:\Users\natha\OneDrive\Documents\Python>
```

```
PS C:\Users\natha\OneDrive\Documents\Python> & C:/Users/natha/AppData/L
```

```
Laporan2/exercise43.py
```

```
=====
```

Note	Frequency (Hz)
C4	261.63
D4	293.66
E4	329.63
F4	349.23
G4	392.0
A4	440.0
B4	493.88

```
Masukan data yang akan dicari : 5
```

```
Null data Fetched
```

```
PS C:\Users\natha\OneDrive\Documents\Python>
```

Masukan data kosong ke dalam fetch_data Masukan data table ke dalam kamus Cetak format awal untuk note dan frequency dan cetak isi dari kamusnya kemudian ada inputan yang bertipe float kedalam data dan dengan menggunakan loop for untuk yang pertama itu menggunakan kamus item yang artinya semua item diambil dan disimpan ke dalam note_freq untuk kuncinya dan untuk data nya dimasukan kedalam freq_hz lalu dibawahnya ada if statement untuk menhecek apakah data nya itu untuk freq_hz sama dengan data untuk inputan sebelumnya dan jika iya maka akan langsung memasukan fetch_data dengan data yang ada didalam variable note_freq lalu ada if statement lagi untuk mengecek apakah fetch_data masih kosong atau tidak, jika kosong langsung berhenti jika tidak maka fetch_data akan tercetak

- Exercise 44

```
PS C:\Users\natha\OneDrive\Documents\Python> & C:/Users/natha/AppData/Local/Python/Python39-64/Laporan2/exercise44.py
```

```
=====
Individual          Amount
George Washington   1
Thomas Jefferson    2
Abraham Lincoln     5
Alexander Hamilton  10
Andrew Jackson      20
Ulysses S. Grant    50
Benjamin Franklin   100
Masukan data yang akan dicari :
```

```
Laporan2/exercise44.py
```

```
=====
Individual          Amount
George Washington   1
Thomas Jefferson    2
Abraham Lincoln     5
Alexander Hamilton  10
Andrew Jackson      20
Ulysses S. Grant    50
Benjamin Franklin   100
Masukan data yang akan dicari : 1
The name of the denomination of a banknote George Washington
=====
PS C:\Users\natha\OneDrive\Documents\Python> █
```

```
PS C:\Users\natha\OneDrive\Documents\Python> & C:/Users/natha/AppData/Local/Python/Python39-64/Laporan2/exercise44.py
```

```
=====
Individual          Amount
George Washington   1
Thomas Jefferson    2
Abraham Lincoln     5
Alexander Hamilton  10
Andrew Jackson      20
Ulysses S. Grant    50
Benjamin Franklin   100
Masukan data yang akan dicari : 3
No Data Found!
=====
PS C:\Users\natha\OneDrive\Documents\Python> █
```

Masukan fetch_data kosong lalu masukan kamus dengan data yang ada di table dan cetak format judul untuk tablenya lalu cetak isi dalam tablenya selanjutnya ada inputan yang bertipe float yang dimasukan kedalam var data lalu ada looping for dan menyimpan item dari

kamusnya kedalam individual dan amount untuk individual sendiri ada key dan amount ada datanya lalu dibawahnya jika data sama dengan amount maka fetch data akan diisikan dengan key atau individual sebelumnya dan melihat apakah fetch_data kosong jika iya maka akan langsung skip ke else untuk mencetak ke fetch_data

E. Daftar Pustaka

<https://docs.python.org/3/library/math.html>

https://www.w3schools.com/python/module_math.asp