

LAPORAN PRAKTIKUM PEMROGRAMAN MINGGU KE-11

“String”

Dosen Pembimbing: Tri Hadiah Muliawati,S.ST., M.Kom



Disusun oleh :

Puput Ayu Setiawati	(3322600004)
Hanadia Navaz Kamalia	(3322600007)
Dicky Syarif Renaldi	(3322600009)
Bayu Kurniawan	(3322600019)
Johan Winarwan Nawawi	(3322600023)

PROGRAM STUDI D4 SAINS DATA TERAPAN

DEPARTEMEN TEKNIK INFORMATIKA DAN KOMPUTER

POLITEKNIK ELEKTRONIKA NEGERI SURABAYA

NOVEMBER 2022

Latihan 1

```
[1] #A
    for ch in "abc123XYX":
        if ch.isupper():
            print(ch.lower(), end='')
        elif ch.islower():
            print(ch.upper(), end='')
        else:
            print(ch, end='')
```

ABC123xyx

Analisis: *Syntax* di atas digunakan untuk mencetak perubahan huruf yang awalnya abc123XYX menjadi ABC123xyx, karena pada if jika hurufnya besar maka diubah menjadi huruf kecil, dan pada elif jika hurufnya kecil maka akan diubah ke huruf besar.

```
s1 = 'where are the snows of yesteryear?'
s2 = s1.split()
print(s2[-2])
```

of

Analisis: Program di atas tercetak of karena pada variabel s2 terdapat method split yakni digunakan untuk memisahkan satu string menjadi substring. Dan yang diminta adalah index ke- (-2) maka yang tercetak adalah kata of.

```
the_list = ['where', 'are', 'the', 'snows?']
s = '*'.join(the_list)
print(s)
```

where*are*the*snows?

Analisis: *Syntax* di atas digunakan untuk menambahkan tanda * di antara substring satu ke substring selanjutnya, kemudian substring tersebut digabungkan menjadi satu string maka akan tercetak where*are*the*snows.



```
s = 'it is either easy or impossible'  
s = s.replace('easy', 'hard').replace('im', '')  
print(s)
```

```
it is either hard or possible
```

Analisis: Pada program di atas terdapat method replace yang digunakan untuk mengganti kata easy menjadi kata hard dan merubah kata impossible menjadi possible karena im pada impossible diubah menjadi string kosong.

Latihan 2

1. Puput Ayu Setiawati

```
print("nano block yang dimiliki")
print("="*15)
nano = "harimau gitar dinosaurus piano"
print(nano)
print()

print("terdapat revisi data nano block yang dimiliki")
y = nano.replace('gitar','bunga')
print(y)

x = y.split()
print(x)

jual = "nanoblock yang akan saya jual pertama adalah piano"

if jual.islower():
    print(jual.upper(), end='')

while True:
    a = input("\napakah masih ada yang ingin jual(y/t)? ")
    if a == "y":
        b = input("masukkan bentuk apa yang ingin jual? ")
        print("jadi nano block yang akan dijual adalah piano dan",b)
    elif a == "t":
        print("selesai")
        break
```

```
nano block yang dimiliki
=====
harimau gitar dinosaurus piano

terdapat revisi data nano block yang dimiliki
harimau bunga dinosaurus piano
['harimau', 'bunga', 'dinosaur', 'piano']
NANOBLOCK YANG AKAN SAYA JUAL PERTAMA ADALAH PIANO
apakah masih ada yang ingin jual(y/t)? y
masukkan bentuk apa yang ingin jual? bunga
jadi nano block yang akan dijual adalah piano dan bunga

apakah masih ada yang ingin jual(y/t)? t
selesai
```

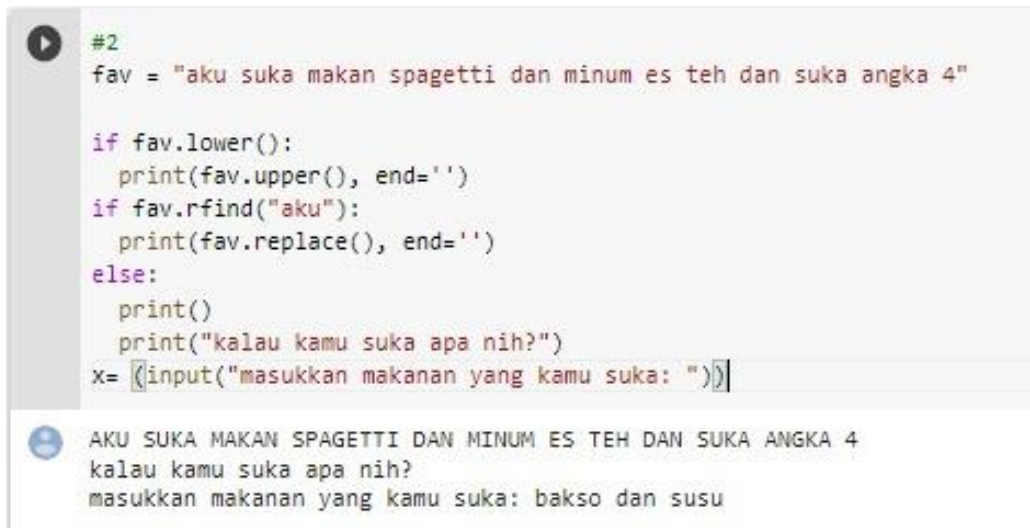
Analisis :

Pada program di atas adalah menampilkan data nanoblock yang dimiliki dan yang ingin jual. Pada awalnya mencetak nanoblock apa saja yang dimiliki, kemudian memasukkan method string yakni replace karena ada data yang perlu diganti atau direvisi, data yang direvisi adalah gitar diubah menjadi bunga, kemudian cetak hasil revisi tersebut. Kemudian terdapat method string yaitu split (y.split()) yang berfungsi sebagai pengubah yang sudah direvisi tadi menjadi substring agar user mudah memahami karena sudah dipecah-pecah, lalu dicetak lagi.

Selanjutnya untuk menjual nanoblock, membuat variabel biasa dengan diberi nama jual. Pada if pertama jika variabel string yang dibuat tadi menggunakan huruf kecil maka diubah menjadi huruf besar semua dengan method upper sebelum itu menggunakan method islower untuk mengetahui huruf kecil atau tidak yang terdapat di variabel jual. Kemudian masuk di while True untuk menanyakan kepada user apakah

ada tambahan nanoblock yang ingin jual, pada if jika y atau berarti ya maka akan diperintahkan untuk memasukkan nanoblock apa lagi yang akan dijual sesuai dengan data yang sudah ada pada substring, lalu cetak apa saja yang jadi dihapus, dan apabila t atau berarti tidak maka program akan di break dan cetak selesai.

2. Hanadia Navaz Kamalia



```
#2
fav = "aku suka makan spaghetti dan minum es teh dan suka angka 4"

if fav.lower():
    print(fav.upper(), end='')
if fav.rfind("aku"):
    print(fav.replace(), end='')
else:
    print()
    print("kalau kamu suka apa nih?")
x= input("masukkan makanan yang kamu suka: ")
```

AKU SUKA MAKAN SPAGETTI DAN MINUM ES TEH DAN SUKA ANGKA 4
kalau kamu suka apa nih?
masukkan makanan yang kamu suka: bakso dan susu

Analisis:

Pada program yang telah dibuat di atas, pada if yang pertama user diminta untuk mengecek apakah string pada variabel fav berisi huruf kecil semua menggunakan method lower(). jika benar maka user diminta untuk mencetak string tersebut dengan huruf besar semua dengan menggunakan perintah print(fav.upper(), end=' '). lalu if yang kedua user diminta untuk mengecek apakah string pada variabel fav terdapat kata "aku" pada awal kalimat menggunakan method rfind(). jika benar maka user diminta untuk mencetak string tersebut dengan mengganti kata di bagian depan dengan kata di bagian belakang menggunakan perintah print(fav.replace(), end=' '). selanjutnya jika semua looping telah dijalankan maka user diminta untuk mencetak perintah yang ada pada else:.

3. Dicky Syarif Renaldi

```
#2
while True:
    a = input("Masukkan kalimat yang ingin anda eja: ")
    b = a.split()
    print(b)
    konfirm1 = input("Apakah kalimatnya ingin anda satukan kembali? (y/t)")
    if konfirm1 == "y":
        b = " ".join(b)
        print(b)
    elif konfirm1 == "t":
        print(b)
    konfirm2 = input("Apakah anda ingin mencoba lagi? (y/t)")
    if konfirm2 == "t":
        break
```

Masukkan kalimat yang ingin anda eja: pens adalah politeknik terbaik
['pens', 'adalah', 'politeknik', 'terbaik']
Apakah kalimatnya ingin anda satukan kembali? (y/t)y
pens adalah politeknik terbaik
Apakah anda ingin mencoba lagi? (y/t)y
Masukkan kalimat yang ingin anda eja: sains data terapan
['sains', 'data', 'terapan']
Apakah kalimatnya ingin anda satukan kembali? (y/t)t
['sains', 'data', 'terapan']
Apakah anda ingin mencoba lagi? (y/t)t

Analisis:

Pada program di atas user diminta untuk menginput kalimat yang kemudian kalimat tersebut akan dieja dengan menggunakan method split. Lalu user diminta untuk konfirmasi apakah user ingin kalimatnya disatukan Kembali atau tidak. Lalu user diminta untuk konfirmasi apakah user ingin mengeja kalimat lain atau tidak

4. Bayu Kurniawan

```
nama = {"1": "Bayu", "2": "Kurniawan", "3": "Erion", "4": "Kei", "5": "Joo",
        "6": "Ren", "7": "Zeoo", "8": "Bob", "9": "Livy", "10": "Rob"}
kalimat = "adalah bukan"
while True:
    data = str(input("Masukkan Nomer peserta yang kamu cari: "))
    cari = nama.get(data)
    if data in nama:
        print('Nama yang kamu cari', cari.upper(), kalimat.rstrip('bukan') + "peserta perlomabaan ini")
    else:
        print('Nomer yang kamu cari', data + kalimat.lstrip('adalah'), "peserta perlomabaan ini")
    hasil = input('Apakah masih ada Nomer peserta yang kamu cari (y/t):')
    if hasil == "y":
        continue
    elif hasil == "t":
        break
```

```
➤ Masukkan Nomer peserta yang kamu cari: 1
Nama yang kamu cari BAYU adalah peserta perlomabaan ini
Apakah masih ada Nomer peserta yang kamu cari (y/t):y
Masukkan Nomer peserta yang kamu cari: 3
Nama yang kamu cari ERION adalah peserta perlomabaan ini
Apakah masih ada Nomer peserta yang kamu cari (y/t):y
Masukkan Nomer peserta yang kamu cari: 11
Nomer yang kamu cari 11 bukan peserta perlomabaan ini
Apakah masih ada Nomer peserta yang kamu cari (y/t):t
```

Analisis:

Program tersebut untuk mencari nama dari nomor peserta yang diinput oleh user, pada program tersebut ada dictionary untuk key value dari nomor peserta, terdapat juga looping yang berisikan perintah input dari user untuk mencari nomor peserta, ketika nomor dari user termasuk di dalam dictionary maka akan ada perintah untuk mencetak “nama yang kamu cari, value dari dictionary dengan huruf kapital, dan variabel kalimat yang (bukan) dibagaimanakan terhapus, ketika nomor dari user tidak termasuk ada juga perintah mencetak dengan data serta variabel kalimat yang (adalah) di bagian kirinya terhapus, terakhir terdapat input untuk menentukan looping lanjut atau berhenti.

5. Johan Winarwan Nawawi

```
1 def hitung_hvokal(teks):
2     hvokal = ['a', 'i', 'u', 'e', 'o']
3     total_hvokal = 0
4     tc_lower = teks.lower()
5     jum_hvokal = {'a': tc_lower.count('a'),
6                   'i': tc_lower.count('i'),
7                   'u': tc_lower.count('u'),
8                   'e': tc_lower.count('e'),
9                   'o': tc_lower.count('o')}
10    for char in teks.lower():
11        if char in hvokal:
12            total_hvokal += 1
13    print()
14    print(["+\"Hasil Dari Program Menghitung Huruf Vokal\".center(70, \"=\")+"])
15    print("Teks yang dimasukkan adalah: ", teks)
16    print("Jumlah karakter pada teks yang dimasukkan adalah: ", len(teks))
17    print("Total huruf vokal pada teks yang dimasukkan adalah: ", total_hvokal)
18    print()
19    print("Rincian jumlah huruf vokal pada teks yang dimasukkan sebagai berikut: ")
20    for i in jum_hvokal.keys():
21        print("    +\"Huruf vokal\", i, \"sebanyak: \", jum_hvokal[i], \"huruf\")
22    print(["+\"Program Menghitung Huruf Vokal Selesai Dijalankan\".center(70, \"=\")+"])
23
24    inputUser = input("Masukkan teks yang ingin dicek jumlah huruf vokalnya: ")
25    hitung_hvokal(inputUser)
```

Analisis Program:

Program yang telah saya buat di atas adalah program untuk menghitung jumlah dari huruf vokal pada satu teks berupa kata atau kalimat yang di-input oleh *user*. Langkah pertama pada baris ke-1 hingga baris ke-22 merupakan ***user-defined function***. Fungsi yang dibuat ini adalah **fungsi untuk menghitung huruf vokal pada teks** yang dimasukkan *user*. Untuk penjelasan per baris, sebagai berikut:

1. Pada baris ke-1, terdapat nama fungsi yaitu '**hitung_hvokal()**' dengan parameter formalnya berupa teks.
2. Pada baris ke-2, terdapat list yang berisi huruf vokal a, i, u, e, dan o yang tersimpan pada variabel **hvokal**.
3. Pada baris ke-3, berupa variabel **total_hvokal** menyimpan nilai 0 (nol) yang nantinya digunakan sebagai penampung jumlah huruf vokal pada teks yang dimasukkan.
4. Pada baris ke-4, variabel **tc_lower** menyimpan type case **lower()** dari teks yang diinputkan oleh *user*. Pada baris ke-4 tersebut terdapat method yang berupa **lower()**.

5. Pada baris ke-5 sampai dengan baris ke-9 merupakan *dictionary* yang menyimpan *key* berupa huruf vokal dan *value* berupa jumlah huruf vokal pada teks. Untuk mengetahui jumlah huruf vokal pada teks digunakan method **count()**.
6. Selanjutnya pada baris ke-10 hingga baris ke-12 terdapat perulangan *for* yang mana untuk menghitung jumlah huruf vokal yang terdapat pada teks yang di-*input* oleh *user* dan diberi method **lower()**. Perulangan tersebut memuat *instruction* berupa *conditional statement* *if* guna mengecek apakah setiap huruf yang tersimpan pada variabel iterasi **char** pada perulangan *for* dari teks sama dengan salah satu huruf vokal yang ada pada variabel *hvokal*? Jika 'iya', maka *total_hvokal* ditambah dengan 1.
7. Pada baris ke-14 terdapat method **center()** yang berfungsi untuk mencetak *string* yang kita mau menjadi rata tengah.
8. Pada baris ke-15 terdapat perintah untuk menampilkan teks yang telah dimasukkan dan ingin dihitung jumlah huruf vokalnya.
9. Pada baris ke-16 menampilkan perintah untuk mencetak jumlah karakter dari teks yang dimasukkan oleh pengguna, untuk mencari jumlahnya dengan menggunakan fungsi **len()**.
10. Pada baris ke-17 menampilkan perintah untuk mencetak total huruf vokal dari teks yang dimasukkan pengguna.
11. Pada baris ke-19 hingga baris ke-21 menampilkan perintah untuk mencetak rincian jumlah huruf vokal pada teks yang telah dimasukkan. Untuk menampilkannya ada baris ke-20 sampai baris ke-21 terdapat perulangan *for* dari *dictionary keys* dengan *instruction* mencetak jumlah dari masing-masing huruf vokal yang terdapat pada teks yang dimasukkan pengguna.
12. Pada baris ke-22 terdapat method **center()** sama dengan baris ke-14 yang fungsinya untuk mencetak *string* menjadi rata tengah.
13. Pada baris ke-24 terdapat fungsi *input* yang tersimpan pada variabel **inputUser** yang mana digunakan sebagai parameter aktual pemanggilan fungsi **hitung_hvokal()** pada baris ke-25.

Output Program:

a. Menggunakan huruf kapital semua

Masukkan teks yang ingin dicek jumlah huruf vokalnya: JOHAN WINARWAN NAWAWI

[=====Hasil Dari Program Menghitung Huruf Vokal=====]

Teks yang dimasukkan adalah: JOHAN WINARWAN NAWAWI

Jumlah karakter pada teks yang dimasukkan adalah: 21

Total huruf vokal pada teks yang dimasukkan adalah: 8

Rincian jumlah huruf vokal pada teks yang dimasukkan sebagai berikut:

Huruf vokal a sebanyak: 5 huruf

Huruf vokal i sebanyak: 2 huruf

Huruf vokal u sebanyak: 0 huruf

Huruf vokal e sebanyak: 0 huruf

Huruf vokal o sebanyak: 1 huruf

[=====Program Menghitung Huruf Vokal Selesai Dijalankan=====]

b. Menggunakan huruf campur

Masukkan teks yang ingin dicek jumlah huruf vokalnya: jOhan WinaRwAn NawAwI

[=====Hasil Dari Program Menghitung Huruf Vokal=====]

Teks yang dimasukkan adalah: jOhan WinaRwAn NawAwI

Jumlah karakter pada teks yang dimasukkan adalah: 21

Total huruf vokal pada teks yang dimasukkan adalah: 8

Rincian jumlah huruf vokal pada teks yang dimasukkan sebagai berikut:

Huruf vokal a sebanyak: 5 huruf

Huruf vokal i sebanyak: 2 huruf

Huruf vokal u sebanyak: 0 huruf

Huruf vokal e sebanyak: 0 huruf

Huruf vokal o sebanyak: 1 huruf

[=====Program Menghitung Huruf Vokal Selesai Dijalankan=====]

Analisis Output:

Walaupun input yang dimasukkan user berbeda jenis hurufnya, seperti pada poin a yang menggunakan huruf kapital semua dan pada poin b yang menggunakan huruf campuran. Akan tetapi, hasil yang didapatkan sama penjumlahan huruf vokalnya. Hal ini dikarenakan pada baris ke-5 hingga ke-10 terdapat *type case* ke *lower case* dari teks yang dimasukkan *user*.

Latihan 3

```
#3
data = {}
while True:
    pilih = int(input("Apakah anda ingin 1. Login atau 2. Daftar: "))
    if pilih == 1:
        print("="*50)
        print("Halaman Login")
        while True:
            id2 = input("Masukkan id anda: ")
            pw2 = input("Masukkan password anda: ")
            data_key = data.keys()
            if id2 not in data_key:
                print("Id tidak terdaftar!!!")
                continue
            data_cekpw = data.get(id2)[0]
            data_nama = data.get(id2)[1]
            data_telp = data.get(id2)[2]
            if pw2 != data_cekpw:
                print("Password anda salah!!!")
                continue
            elif id2 in data_key and pw2 == data_cekpw:
                print("="*50)
                print("Data Anda:")
                print("Nama Anda:", data_nama)
                print("No Telp: ", data_telp)
                break

    konfirmasi = input("Apakah anda ingin melanjutkan? (y/t) ")
    if konfirmasi == "t":
        break
    elif pilih == 2:
        print("Formulir Pendaftaran")
        print("="*50)
        while True:
            inp_id = input("Masukkan id yang ingin didaftarkan: (Tambahkan @gmail.com) ")
            if "@gmail.com" not in inp_id:
                print("Format id salah!!!")
            elif "@gmail.com" in inp_id:
                break
            inp_pw = input("Masukkan password yang ingin anda daftarkan: ")
            while True:
                inp_nama = input("Masukkan nama anda: (Harus berupa huruf!!!) ")
                b = inp_nama.replace(" ", "")
                if not b.isalpha():
                    print("Format anda salah!!!")
                    continue
                else:
                    break
            while True:
                inp_telp = input("Masukkan nomor telepon anda: (Harus berupa angka dan tidak boleh berspasi!!!) ")
                if not inp_telp.isdigit():
                    print("Format anda salah!!!")
                else:
                    break
            data.update({inp_id : [inp_pw, inp_nama, inp_telp]})
        else:
            print("Opsi yang anda pilih tidak ada!!!")
```

▶ Apakah anda ingin 1. Login atau 2. Daftar: 2
Formulir Pendaftaran
=====

Masukkan id yang ingin didaftarkan: (Tambahkan @gmail.com) sarah@gmail.com
Masukkan password yang ingin anda daftarkan: pens22
Masukkan nama anda: (Harus berupa huruf!!!) sarah
Masukkan nomor telepon anda: (Harus berupa angka dan tidak boleh berspasi!!!) 08343459683
Apakah anda ingin 1. Login atau 2. Daftar: 1
=====

Halaman Login
Masukkan id anda: sarah@gmail.com
Masukkan password anda: pens22
=====

Data Anda:
Nama Anda: sarah
No Telp: 08343459683
Apakah anda ingin melanjutkan? (y/t) t

Analisis :

Pada awal program user diminta untuk memilih, bila ingin login masukkan 1 bila ingin mendaftar masukkan 2. Apabila user memilih 2, maka user akan dialihkan ke dalam laman pendaftaran. User akan diminta memasukkan id yang ingin didaftarkan dengan syarat harus ada @gmail.com dibelakang idnya, apabila tidak menyertakan @gmail.com dibelakang maka akan tercetak kalimat “Format Id salah!!!” dan user akan diminta memasukkan ulang id yang ingin didaftarkan. Id yang sudah diinputkan akan disimpan ke dalam variabel `inp_id`. Apabila input dari user sudah memenuhi syarat maka program akan keluar dari looping.

Lalu user diminta memasukkan password yang akan didaftarkan dan password yang telah diinput akan disimpan di dalam variabel `inp_pw`. Setelah itu user diminta untuk menginputkan nama dengan syarat nama harus berupa huruf dan nama yang telah diinput akan disimpan ke dalam variabel `input_nama`. Nama yang telah diinput akan dicek dengan cara mengganti spasi dengan string kosong yang akan menghilangkan spasi dan disimpan ke dalam variabel `b`. Lalu variabel `b` dicek dengan menggunakan method `isalpha()`, jika isi dari variabel `b` tidak sepenuhnya alphabet maka akan tercetak kalimat “Format anda salah!!!” lalu user akan diminta menginput ulang nama. Apabila input dari user sudah memenuhi syarat maka program akan keluar dari looping.

Lalu user diminta menginput nomor telepon dengan syarat harus berupa angka dan tidak mengandung spasi yang kemudian disimpan ke dalam variabel `inp_telp`. Lalu akan dicek dengan menggunakan method `isdigit()`, yang dimana apakah isi dari variabel `inp_telp` sepenuhnya berisi digit angka, bila tidak maka akan tercetak kalimat “Format anda salah!!!” dan user akan diminta menginputkan ulang nomor teleponnya. Apabila input dari user sudah memenuhi syarat maka program akan keluar dari looping.

Dan selanjutnya semua input user tadi akan diupdate ke dalam dictionary data menggunakan method `update()`. Yang dimana `inp_id` berfungsi sebagai key dalam dictionary. Valuenya berupa list, `inp_pw` berada pada index ke 0, `inp_nama` berada pada index ke 1, `inp_telp` berada pada index ke 2.

Apabila user memilih 1, maka user akan dialihkan ke dalam laman login. User akan diminta menginput id yang sudah terdaftar dan id yang telah diinput akan dimasukkan ke dalam variabel `id2`. Lalu user akan diminta menginput password yang sudah terdaftar dan password yang telah diinput akan dimasukkan ke dalam `pw2`.

Variabel `data_key` berisi semua keys yang ada di dalam dictionary, untuk mengambil semua keys digunakan method `keys()`. Yang kemudian akan dicek, jika isi dari variabel `id2` tidak ada di dalam variabel `data_key` maka akan tercetak kalimat “Id tidak terdaftar!!!” dan perintah di bawahnya akan diskip menggunakan `continue`.

Jika kondisi di atas tidak terpenuhi yang artinya isi dari variabel `id2` ada di dalam variabel `data_key` maka program akan memunculkan:

- a. Variabel `data_cekpw` yang berisi value index ke 0 dari keys berdasarkan isi dari variabel `id2`
- b. Variabel `data_nama` yang berisi value index ke 1 dari keys berdasarkan isi dari variabel `id2`
- c. Variabel `data_telp` yang berisi value index ke 2 dari keys berdasarkan isi dari variabel `id2`

Lalu akan dicek, jika isi dari variabel `pw2` tidak sama dengan isi dari variabel `data_cekpw` maka akan tercetak kalimat “Password anda salah!!!” dan perintah di bawahnya akan diskip menggunakan `continue`.

Jika kondisi di atas tidak terpenuhi yang artinya isi dari variabel `pw2` sama dengan isi dari variabel `data_cekpw` maka akan dicek, jika isi dari variabel `id2` ada di dalam variabel `data_key` dan isi dari variabel `pw 2` sama dengan isi dari variabel `data_cekpw` maka akan tercetak isi dari variabel `data_nama` dan isi dari variabel `data_telp`. Lalu looping akan berhenti.

User akan diminta untuk mengkonfirmasi apakah ingin melanjutkan program atau tidak. Jika user menginput `y` maka program akan berlanjut, jika user menginput `t` maka program akan berhenti

Latihan 4

```
#4
while True:
    str1 = input("Masukkan string 1: ")
    str2 = input("Masukkan string 2: ")
    a = sorted(str1)
    b = sorted(str2)
    if b == a:
        print(str1, "dan", str2, "merupakan anagram")
    else:
        print(str1, "dan", str2, "bukan anagram")
    konfirm = input("Apakah anda ingin mencoba lagi? (y/t) ")
    if konfirm == "t":
        break
    elif konfirm == "y":
        print()
```

```
Masukkan string 1: three
Masukkan string 2: there
three dan there merupakan anagram
Apakah anda ingin mencoba lagi? (y/t) y

Masukkan string 1: read
Masukkan string 2: red
read dan red bukan anagram
Apakah anda ingin mencoba lagi? (y/t) t
```

Analisis:

Program di atas digunakan untuk mengetahui anagram atau bukan pada karakter yang dimasukkan. Pada syntax di atas menggunakan while True dan membuat 2 variabel input yang digunakan user agar bisa memasukkan string pertama dan kedua dan nantinya bisa diperiksa apakah kedua bilangan itu anagram atau bukan anagram. Kemudian membuat variabel a dan b yang terdapat fungsi sorted() yang digunakan untuk mengurutkan kata yang diinputkan pada variabel str1 dan str2.

Pada if pertama jika hasil sorted b sama dengan a maka kata tersebut termasuk anagram dan akan mencetak input string 1 dan string 2 merupakan anagram. Dan jika tidak memenuhi if atau a dan b hasil sortednya tidak sama maka kata yang dimasukkan pada string 1 dan string 2 bukan anagram dan akan mencetak str1 dan str2 bukan anagram.

Kemudian membuat variabel input lagi untuk menanyakan user apakah ingin memasukkan string lagi atau tidak, dengan menggunakan if sama dengan t atau tidak maka program akan berhenti atau selesai, dan jika if tidak memenuhi dan user menjawab y atau yang berarti ya maka program akan looping ke variabel input untuk masukkan string1 dan string2 lagi.

Latihan 5

```
▶ def baris1(number):  
    if number=='1':  
        return ' #'  
    elif number=='4':  
        return '# #'  
    else:  
        return '###'  
def baris2(number):  
    if number=='1' or number=='2' or number=='3' or number=='7':  
        return ' #'  
    elif number=='5' or number=='6':  
        return '# '  
    else:  
        return '# #'  
def baris3(number):  
    if number=='1' or number=='7':  
        return ' #'  
    elif number=='0':  
        return '# #'  
    else:  
        return '###'  
def baris4(number):  
    if number=='2':  
        return '# '  
    elif number=='6' or number=='8' or number=='0':  
        return '# #'  
    else:  
        return ' #'  
def baris5(number):  
    if number=='1' or number=='4' or number=='7':  
        return ' #'  
    else:  
        return '###'  
lst1 = []  
lst2 = []  
lst3 = []  
lst4 = []  
lst5 = []
```

```

while True:
    pilihan = input('Masukkan digit angka yang akan dicetak kelayar: ')
    cek=pilihan.isdigit()
    if cek == False:
        print('masukkan angka')
        continue
    else:
        pilihan = str(pilihan)
        break

for i in pilihan:
    lst1.append(baris1(i))
    lst2.append(baris2(i))
    lst3.append(baris3(i))
    lst4.append(baris4(i))
    lst5.append(baris5(i))
a = ' '.join(lst1)
b = ' '.join(lst2)
c = ' '.join(lst3)
d = ' '.join(lst4)
e = ' '.join(lst5)
print(a)
print(b)
print(c)
print(d)
print(e)

```

```

Masukkan digit angka yang akan dicetak kelayar: 68543590
###  ###  ###  # #  ###  ###  ###  ###
#   # #  #   # #   #   # #   # #  # #
###  ###  ###  ###  ###  ###  ###  # #
# #  # #   #   #   #   #   #   # #
###  ###  ###   #  ###  ###  ###  ###

```

Analisis :

Program tersebut untuk mencetak angka yang kita tulis menjadi angka dengan symbol pagar, pada program tersebut terdapat lima definisi fungsi, dimana setiap satu fungsi itu digunakan untuk membuat satu baris. Fungsi yang pertama dinamakan baris1, fungsi tersebut untuk membuat baris ke-1. Fungsi tersebut menggolongkan setiap angka yang akan menghasilkan bentuk pagar yang jika=1 kosong, kosong, pagar jika=2 pager, kosong, pager jika=3 pagar, pagar, pagar. Dan begitu juga untuk fungsi-fungsi lainnya.

Setelah itu terdapat 5 list kosong untuk menyimpan data dari looping, setelah itu terdapat looping yang berisi perintah input, jika yang diinput oleh user merupakan angka 0-9, maka setiap angka yang dimasukkan akan dimasukkan ke lima fungsi baris tersebut, dan dimasukan ke dalam list. Tiap satu list digunakan untuk menyimpan satu baris data input dari user. Dan jika yang diinput bukan angka, maka akan menginput ulang.

Setelah itu ada perintah print untuk menampilkan apa saja data yang telah disimpan pada list dan terdapat juga jarak spasi yang menggunakan method join untuk menampilkannya.