

Python

Sebuah pengenalan



Daftar Isi

01

**Instalasi
VSCode**

03

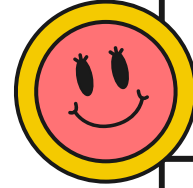
Hello World!

02

**Instalasi
Python 3**

04

**Variabel dan
Tipe Data**



Daftar Isi

05

Input Output

07

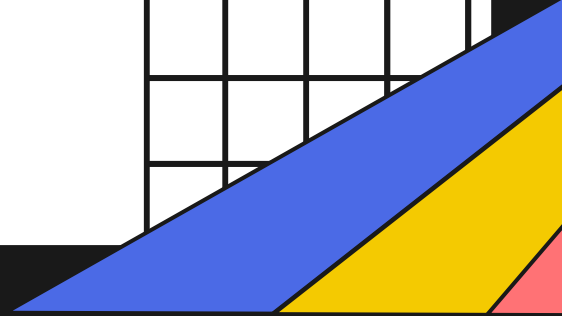
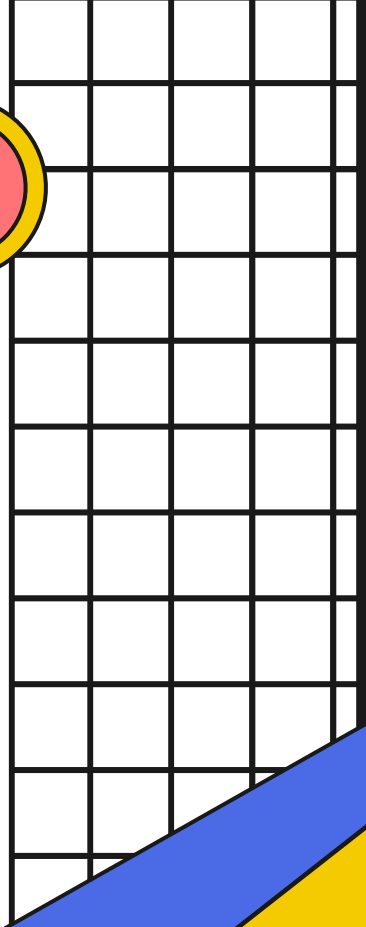
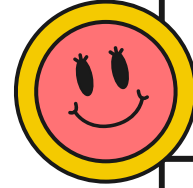
Percabangan

06

**Aritmatika
Sederhana**

08

Perulangan



Daftar Isi

09

List

10

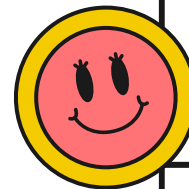
Tuple

11

Dictionary

12

Set



Daftar Isi

13

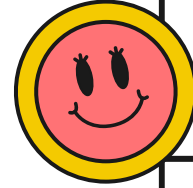
Function

14

Module

15

Read & Write File



01

VSCode

Instalasi, pengenalan, setup



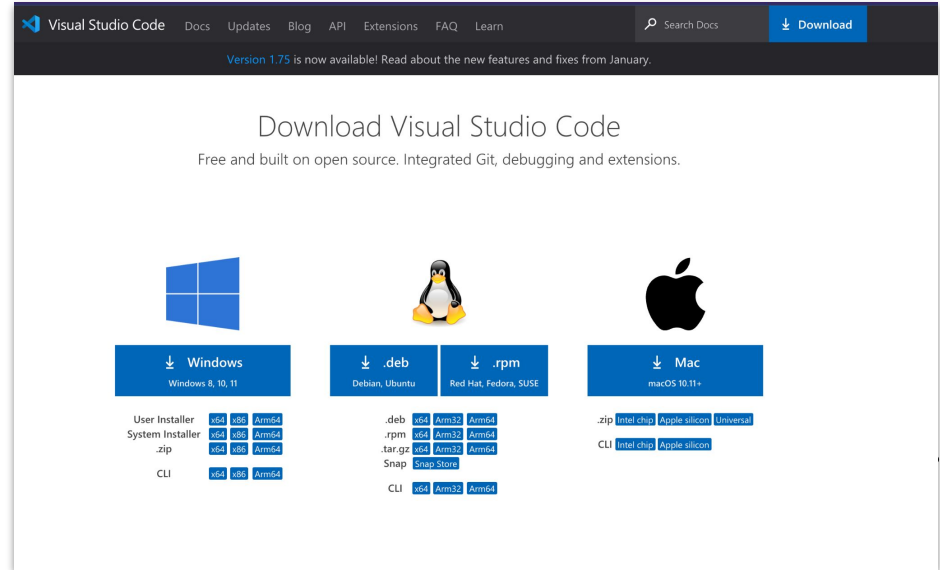
Install - VSCode



Open source text editor dari Microsoft.

Link Download:

<https://code.visualstudio.com/download>



(Alt.) Install - VSCode



Cara alternatif install VSCode



Via
chocolatey

```
choco install vscode
```



Via
snapstore

```
sudo snap install code --classic
```



Via
homebrew


```
brew install --cask visual-studio-code
```


Setup Python di VSCode



Download extension Python di VSCode





Python v2023.2.0

Microsoft | 76,633,348 | ★★★★★ (526)

IntelliSense (Pylance), Linting, Debugging (multi-threaded, remote), Jupyter Notebooks, code f...

Disable ▾

Uninstall ▾

Switch to Pre-Release Version

🔄

⚙️

This extension is enabled globally.



Setup VSCode Lainnya (Opt.)



Mempercantik VSCode dengan Theme? <https://www.makeuseof.com/visual-studio-top-themes/>

Mempermudah coding dengan Code Formatter?

<https://www.geeksforgeeks.org/python-code-formatting-using-black/>



02

Python 3

Ular bukan sembarang ular



Python



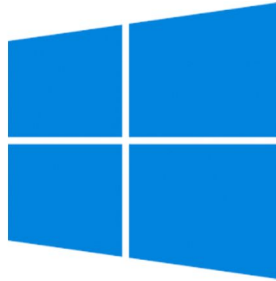
Sebuah bahasa pemrograman yang simpel dan mudah dipahami.

Lebih jauh tentang Python?

Website resmi: <https://www.python.org/>



Install - Python (**Warning**)



Menurut sebuah artikel Python di Microsoft Store broken bre..
Jadi tidak disarankan install dari sana.

Referensi:

<https://dev.to/naruaika/why-i-didn-t-install-python-from-the-microsoft-store-5cbd>



Real Install - Python



Step by step:



1. Download Python -> <https://www.python.org/downloads/>
2. Install

Nb: di Linux biasanya sudah terinstall untuk cek coba run di command `python3`



03

Hello World

Jalan ninja untuk menjadi seorang
programmer handal



Python 3 - Hello World



```
print("Hello, world")
```



04

Tipe Data

Data apa iniiiiii



Tipe Data

Tipe data adalah media atau memori yang ada pada komputer untuk menampung sebuah informasi.



Tipe data pada python berdasarkan kategorinya :

1. Text



- String

Menyatakan karakter/ kalimat berupa huruf, angka, dll. penulisanya diapit dengan tanda “” (double quotes) atau “”(single quotes).

Contoh: **“Aku cinta KSL”**

2. Angka

- Integer

Menyatakan bilangan bulat

Contoh : **25**



Tipe Data

- Float
Menyatakan bilangan real yang merupakan bilangan desimal / pecahan.
Contoh : 2.25
- Complex
Menyatakan nilai imajiner
Contoh : 9j
- Hexadecimal
Menyatakan bilangan dalam format hexadecimal (bilangan berbasis 16)
Contoh : 0x6a

3. Urutan

- List
Deretan data yang menyimpan berbagai tipe data yang isinya bisa diubah
Contoh : [1, 2, "tiga", "empat"]

Tipe Data

- Tuple
Deretan data yang menyimpan berbagai tipe data yang isinya tidak bisa diubah
Contoh: `(1, 2, "empat", "lima")`
- 4. Mapping
 - Dictionary
Deretan data yang menyimpan berbagai tipe data yang berupa pasangan key dan value
Contoh: `{"nama": "Maikel", "umur": 20}`
- 5. Set
 - Set
Menyatakan koleksi data yang bersifat unik (tidak dapat diduplikasi)
Contoh: `{"senin ", "selasa", "minggu", "minggu"}`

Tipe Data



- Frozenset

Menyatakan koleksi data yang bersifat unik, namun data tidak dapat diubah.

Contoh: `frozenset({"merah", "kuning", "hijau"})`

6. Boolean

Menyatakan benar **True** yang bernilai 1, atau salah **False** yang bernilai 0

Contoh: **True**

7. Binary

- Bytes

Mengkonversikan sebuah objek menjadi objek byte berdasarkan ukuran dan data yang diberi

Contoh: `b"Hallo"`

- Mengkonversikan objek berupa array menjadi objek byte

Contoh: `bytearray([1,2,3,4])`




Tipe Data



- Memoryview
Menghasilkan objek memory view dari objek diberikan.
Contoh : `memoryview(bytes(5))`

8. None
Mendefinisikan data `null`.
Contoh : `None`





05 Input Output

Keluar masuk
Masuk keluar

input/output



Input Output



Input

```
variabel = input()
```



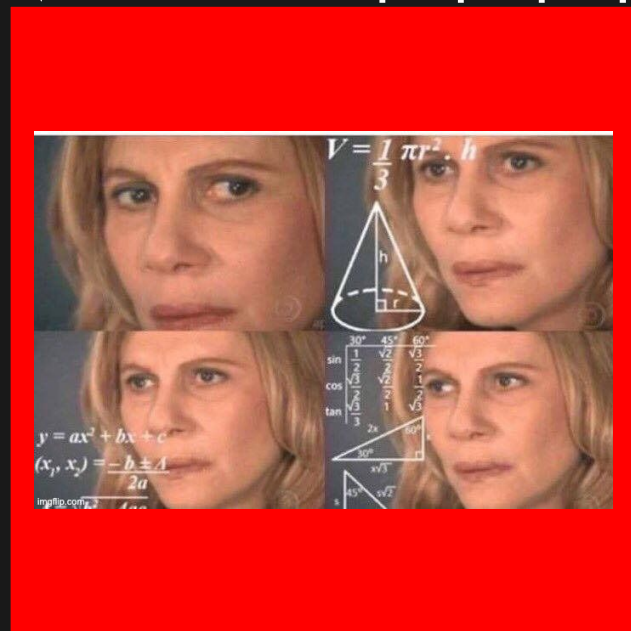
Output

```
print(variabel)
```



06 Aritmatika Sederhana

Matematika ilmu yang menyenangkan
+ - * / % ** //



Operator Aritmatika

Operator	Nama	Contoh
+	Penjumlahan	$x + y$
-	Pengurangan	$x - y$
*	Perkalian	$x * y$
/	Pembagian	x / y
%	Modulus	$x \% y$
**	Pangkat	$x ** y$
//	Pembagian bulat	$x // y$

Prioritas Operator

Operator	Nama
()	Tanda kurung
**	Pangkat
*, /, //, %	Perkalian, pembagian, pembagian bulat, modulus
+, -	Penjumlahan, pengurangan

07

Percabangan

Jika... maka...

Jika tidak... maka...



**SWITCH
STATEMENT**



**IF
ELIF
ELSE**

imgflip.com

If-Else

If-else digunakan untuk mengecek suatu kondisi, jika kondisi tersebut benar maka blok kode di dalam if akan dijalankan, jika tidak maka blok kode else yg dijalankan.

```
angka1 = 10
angka2 = 20

if angka1 < angka2:
    print(f"Benar {angka1} lebih kecil dari {angka2}")
else:
    print("Tet...! salah")
```

Indentation

Elif

Kita juga bisa mengecek beberapa kondisi sekaligus dengan menggunakan elif

```
bahasa = "India"

if bahasa == "Inggris":
    print("Hello world")

elif bahasa == "India":
    print("Namaste duniya")

elif bahasa == "Indonesia":
    print("Halo dunia")

else:
    print("Hah?")
```



08

Perulangan

Berulang-ulang

FOR LOOP...



WHILE LOOP...

For Loop

Perulangan for disebut counted loop (perulangan yang terhitung), biasanya digunakan untuk mengulangi kode yang sudah diketahui banyak perulangannya.

```
for i in range(10): # Menampilkan angka dari 0 sd 9(i-1)
    print(i)

for i in range(1, 11): # Menampilkan angka dari 1 sd 10(i-1)
    print(i)

for i in range(1, 11, 2): # Menampilkan angka dari 1 sd 10(i-1) dan +2
    print(i)
```

Indentation

While Loop

Perulangan while disebut uncounted loop (perulangan yg tak terhitung), biasanya digunakan untuk perulangan yg memiliki syarat dan tidak tentu berapa banyak perulangannya.

```
jawab = "y"  
ulang = 0
```



```
while(jawab == "y"):  
    [ ] ulang += 1  
    [ ] jawab = input("Ulangi lagi ?(y/n)")  
print(f"Total perulangan: {ulang}")
```

Indentation

09 List

Daftar nilai campuran

LISTS, LISTS, LISTS. I LOVE LISTS



List digunakan untuk menyimpan kumpulan nilai atau objek dalam satu variabel. List pada Python dapat dianggap sebagai daftar, atau kumpulan elemen yang tersusun secara teratur dan bisa berisi objek apa saja, termasuk angka, string, atau bahkan objek lain seperti list atau dictionary.



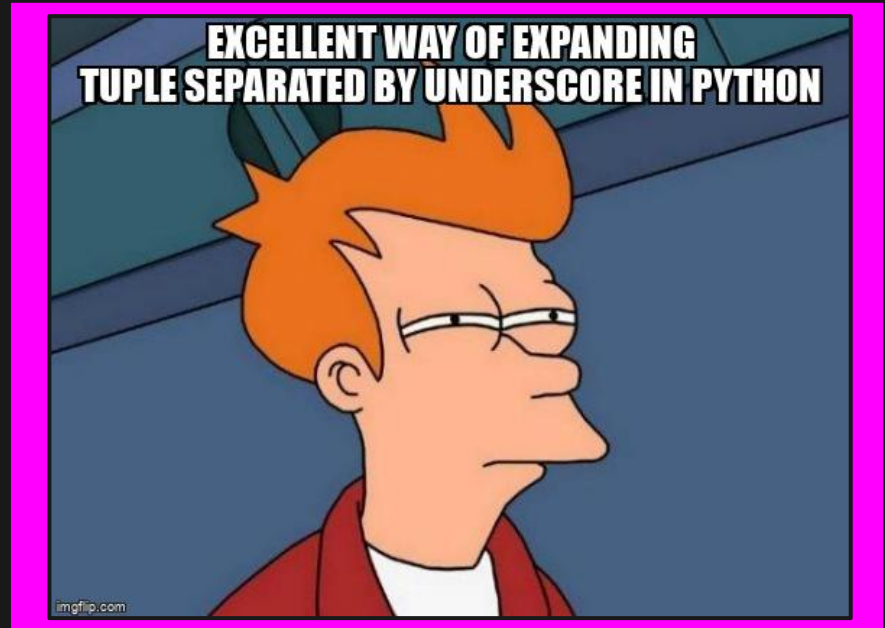
```
mix_list = [False, 1, 2, 3, "empat", "lima"]  
  
mix_list.append(['a','b','c']) # Memasukkan data di list terakhir  
mix_list.insert(1,True) # Memasukkan data di index ke-1  
mix_list[3] = 7 # Mengedit data di index ke-3  
mix_list.pop() # Menghapus data list index terakhir  
del mix_list[2] # Menghapus data di index ke-2
```



10

Tuple

Immutable object (tidak bisa diubah)



Tuple pada Python biasanya digunakan untuk menyimpan data yang tidak perlu diubah lagi. Contohnya koordinat pada titik tertentu, atau kumpulan data yang saling berhubungan namun tidak akan diubah, seperti informasi dalam sebuah record atau data pada database.



```
mix_tuple = (False, 1, 2, 3, "empat", "lima")  
  
mix_tuple += (6, "tujuh", 8) # Menambahkan data pada tuple  
  
print(mix_tuple[8])
```



11

Dictionary

Key : Value



Dictionary digunakan untuk menyimpan sepasang data, yaitu key dan value. Key pada dictionary harus unik dan tidak boleh diubah, sedangkan value dapat berupa tipe data apapun dan boleh diubah.

```
mahasiswa = {  
    "nama": "Krisna",  
    "nim": 210010062,  
    "prodi": "SK"  
}  
  
# Menampilkan seluruh pasangan key-value  
for key,value in mahasiswa.items():  
    print(f"{key}: {value}")
```

```
print(mahasiswa["nama"])  
  
# Menambahkan key-value baru  
mahasiswa["jenis_kelamin"] = True  
  
# Menghapus data key-value  
del mahasiswa["prodi"]
```



12

Set

Unique value (tidak ada duplikasi)



Set digunakan untuk menyimpan kumpulan elemen yang unik (tidak memiliki duplikat) dan tidak terurut.

```
set1 = {1, 2, 3, 4, 5, 5}
```

```
set2 = {4, 5, 6, 7, 8}
```

```
set1.add(10) # Menambahkan elemen ke set
```

```
set1.remove(2) # Menghapus elemen dari set
```

```
gabungan = set1.union(set2) # Menggabungkan set1 & set2
```

```
persamaan = set1.intersection(set2) # irisan set1 & set2
```

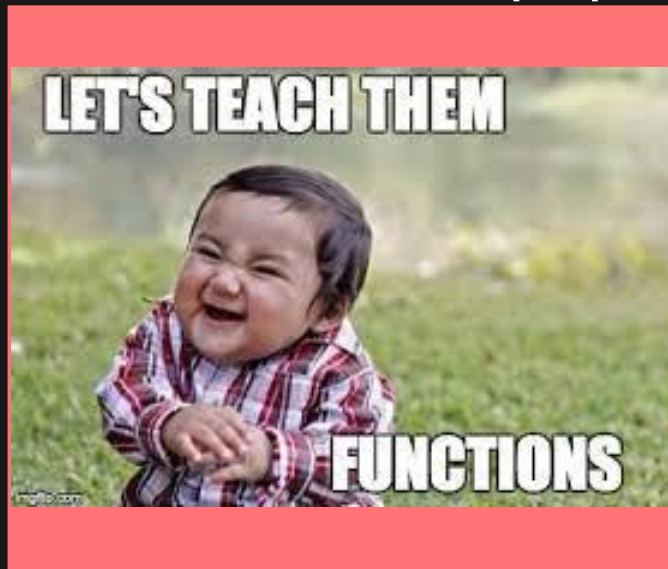
```
perbedaan = set1.difference(set2) # selisih set1 & set2
```

```
print(gabungan)
```

13

Function

Digunakan ulang



Function berfungsi untuk mengelompokkan baris kode tertentu sehingga dapat digunakan ulang. Function di python dapat me-return nilai, ataupun tidak me-return nilai.

```
# Function tanpa return nilai
def halo_dunia():
    print("Hello World!")

# Function dengan parameter dan return nilai
def tambah(a, b):
    return a + b


halo_dunia()
print(tambah(5,10))
```

14

Module

Kumpulan kode python





Module adalah sebuah file yang berisi kumpulan kode Python yang dapat digunakan oleh program Python lainnya. Ada 3 kategori modul yaitu Modul bawaan (built-in module), Modul pihak ketiga (third-party module), dan Modul buat sendiri (custom module). Ada tiga cara untuk memanggil module di python :

1. Menggunakan perintah 'import' untuk memuat modul di python ✨

```
import math # Untuk memuat modul 'math'
print(math.pi) # menghasilkan nilai konstanta pi yg terdapat di modul 'math'
```

2. Menggunakan 'alias' agar lebih mudah dipanggil di dalam program. ✨

```
import math as m
print(m.pi)
```

3. Menggunakan perintah 'from' untuk memuat fungsi atau variabel tertentu dari sebuah modul, tanpa harus memuat seluruh modul tersebut.

```
# Agar dapat menggunakannya langsung tanpa harus menambahkan awalan nama modul.
from math import sqrt
print(sqrt(25)) # menghasilkan nilai akar kuadrat dari 25.
```

Modul bawaan (built-in module) : Modul yang telah tersedia secara default di dalam Python dan dapat digunakan tanpa perlu menginstal atau mengimpor modul tambahan. Contoh modul bawaan di Python antara lain math, random, datetime, dan sebagainya. List lengkapnya bisa dilihat di dokumentasi python berikut : <https://docs.python.org/3/library/index.html>



```
from random import randint as rd
from datetime import datetime as dt

# randint untuk menghasilkan angka random dari 0 sampai 1000
print(rd(0, 1000))

# datetime.now() untuk mendapatkan tanggal & waktu sekarang
print(dt.now())
```

Modul pihak ketiga (third-party module): Modul yang dikembangkan oleh pihak ketiga dan tidak termasuk dalam instalasi Python standar. Untuk menggunakannya, kita perlu menginstal modul tersebut terlebih dahulu menggunakan pip atau alat manajemen paket lainnya. Contoh modul pihak ketiga di Python antara lain NumPy, Pwntools, Pandas, Matplotlib, dan sebagainya.

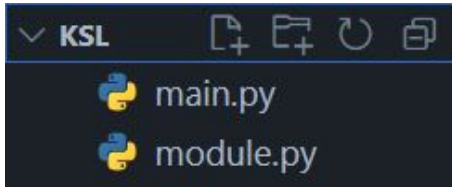
```
from pwn import xor

flag = "KSL2023{t3s_d0ank}"
# Melakukan operasi XOR pada setiap byte dari flag dengan integer 7
print(xor(flag, 7))
```

```
import numpy as np


a = np.array([1, 2, 3, 4, 5]) # Membuat array 1D
print(np.max(a)) # Menghasilkan nilai maksimum dari array
print(a * 5) # Menghasilkan setiap nilai array yg dikali 5
```


Modul buat sendiri (custom module): Modul yang dibuat oleh pengguna sendiri sesuai dengan kebutuhan mereka. Modul ini dapat digunakan oleh program Python yang sama, atau oleh program Python lainnya yang membutuhkan fungsionalitas yang sama. Untuk menggunakan modul ini, kita perlu menyimpannya di dalam direktori yang sama dengan program Python yang memanggil modul tersebut.




```
# module.py
def tambah(a, b):
    return a + b

def kurang(a, b):
    return a - b
```



```
# main.py
import module as md

print(md.tambah(1,2))
```



15

Read & Write File

Membaca & menulis isi file

"The best way to learn a language is to
speak to natives."

The guy learning python:




Di Python, membaca (read) dan menulis (write) file dapat dilakukan dengan mudah menggunakan fungsi bawaan yang disediakan oleh bahasa Python.

Untuk membaca file, kita dapat menggunakan fungsi open() dan read(). Fungsi open() digunakan untuk membuka file dan menetapkan mode bacaan, sedangkan fungsi read() digunakan untuk membaca isi file.



```
# mode r untuk read file sebagai text
read_file = open("contoh.txt", "r")
print(read_file.read())
read_file.close() # menutup file
```


```
# mode rb untuk read file sebagai bytes
read_file = open("contoh.txt", "rb")
print(read_file.read())
read_file.close()
```



Untuk menulis file, kita dapat menggunakan fungsi open() dan write(). Fungsi open() juga digunakan untuk membuka file dan menetapkan mode penulisan, sedangkan fungsi write() digunakan untuk menulis teks ke dalam file.

```
# mode w untuk write file sebagai text
file = open("contoh.txt", "w")
print(file.write("Teks yang akan ditulis ke dalam file"))
file.close() # menutup file
```

```
# mode wb untuk write file sebagai bytes
file = open("contoh.txt", "wb")
print(file.write(b"Teks yang akan ditulis ke dalam file"))
file.close()
```



Pada Python, kita dapat menggunakan with statement untuk membuka file. with statement secara otomatis menutup file ketika blok kode di dalamnya selesai dieksekusi, sehingga kita tidak perlu lagi memanggil fungsi close() secara eksplisit.

```
# read file
with open("contoh.txt", "r") as file:
    print(file.read())

# write file
with open("contoh.txt", "w") as file:
    file.write("Teks yang akan ditulis ke dalam file")
```



Tamat

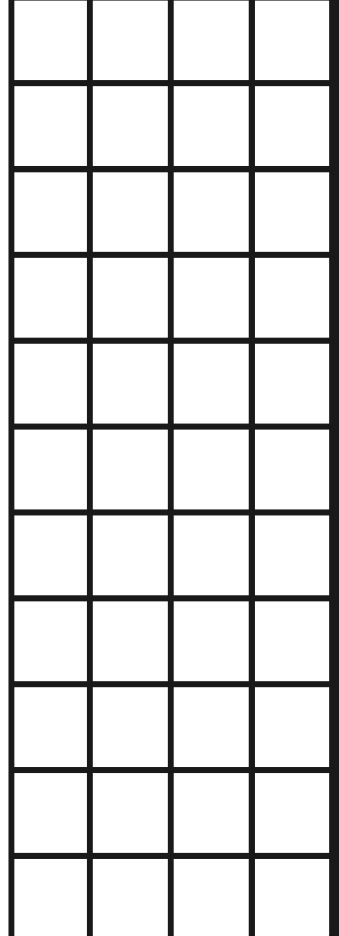
#Python_Basics



✧ ✧

Ada Pertanyaan?

✧



Latihan!

