Python merupakan bahasa pemrograman tingkat tinggi yang diracik oleh **[Guido van Rossum](https://id.wikipedia.org/wiki/Guido_van_Rossum).**

Untuk mengeceknya, silahkan ketik perintah **python --version** di terminal (LINUX).

$ python --version

Python 2.7.12

Cara membuka mode interaktif adalah dengan mengetik perintah python pada terminal.

Tanda >>>, artinya python siap menerima perintah.

Ada dua fungsi yang digunakan untuk mencari bantuan:

fungsi dir() untuk melihat fungsi apa saja yang tersedia pada sebuah modul;

fungsi help() untuk membuka dokumentasi suatu fungsi.

Pertama kita impor dulu modulnya ke mode interaktif:

>>> import math

Setelah itu kita bisa melihat-lihat, fungsi apa saja yang tersedia di modul tersebut.

>>> dir(math)['\_\_doc\_\_', '\_\_name\_\_', '\_\_package\_\_', 'acos', 'acosh', 'asin', 'asinh', 'atan', 'atan2', 'atanh', 'ceil', 'copysign', 'cos', 'cosh', 'degrees', 'e', 'erf', 'erfc', 'exp', 'expm1', 'fabs', 'factorial', 'floor', 'fmod', 'frexp', 'fsum', 'gamma', 'hypot', 'isinf', 'isnan', 'ldexp', 'lgamma', 'log', 'log10', 'log1p', 'modf', 'pi', 'pow', 'radians', 'sin', 'sinh', 'sqrt', 'tan', 'tanh', 'trunc']

Lalu, kita bisa cari tahu cara penggunaan fungsi-fungsi tersebut dengan help().

Misalkan kita ingin cari tahu cara penggunaan fungsi pow(), maka kita harus memberikan perintah help(math.pow).

Help on built-in function pow in module math:

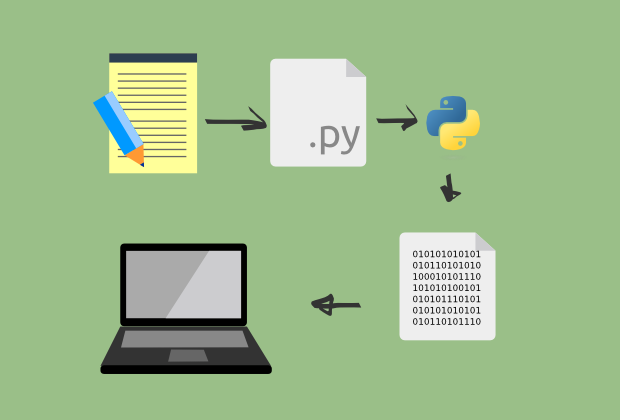
pow(...)

pow(x, y)

Return x\*\*y (x to the power of y).(END)

Keluar documentasi (q)

**Alur Kerja Pembuatan Program Python**



1. Membuat skrip python dengan teks editor.
2. Skrip python diterjemahkan ke dalam kode biner oleh (intepreter) python, sehingga komputer dapat mengerti arti perintah tersebut.
3. Komputer mengerjakan perintah tersebut.

**1. Penulisan Statement Python**

Statement adalah sebuah intruksi atau kalimat perintah yang akan dieksekusi oleh komputer.

Contoh:

print("Hello World!")

print("Belajar Python dari Nol")

nama = "petani kode"

Penulisan satu statement tidak diakhiri dengan tanda titik-koma.

Sedangkan, bila kita ingin menulis lebih dari satu statement dalam satu baris, maka kita harus memisahnya dengan titik-koma.

Contoh:

print("Hello"); print("World"); print("Tutorial Python untuk Pemula")

nama\_depan = "petani"; nama\_belakang = "kode"

Tapi…

Menurut beberapa style guide python, tidak dianjurkan menulis lebih dari satu statement dalam satu baris. Karena akan sulit dibaca.

**2. Penulisan String pada Python**

String adalah teks atau kumpulan dari karakter.

String dalam pemrograman biasanya ditulis dengan dibungkus menggunakan tanda petik.

Bisa menggunakan tanda petik tunggal maupun ganda.

**Contoh:**

judul = "Belajar Pemrograman Python sampai Bisa"

penulis = 'Petani Kode'

**Atau kita juga bisa menggunakan triple tanda petik.**

**Contoh:**

judul = """Belajar Python dengan Cepat"""

penulis = '''Petani Kode'''

**3. Penuilsan Case pada Python**

Sintak Python bersifat case sensitive, artinya teksini dengna TeksIni dibedakan.

**Contoh:**

judul = "Belajar Dasa-dasar Python"

Judul = "Belajar Membuat Program Python"

**Antara variabel judul dengan Judul itu dibedakan…**

**Case Style**

**Menurut rekomendasi [style guide Google](https://google.github.io/styleguide/pyguide.html), berikut ini contoh penulisan case yang disarankan:**

## Snake Case digunakan pada:

module\_name, package\_name, method\_name, function\_name, , global\_var\_name, instance\_var\_name, function\_parameter\_name, local\_var\_name.

## CamelCase digunakan Pada:

ClassName, ExceptionName

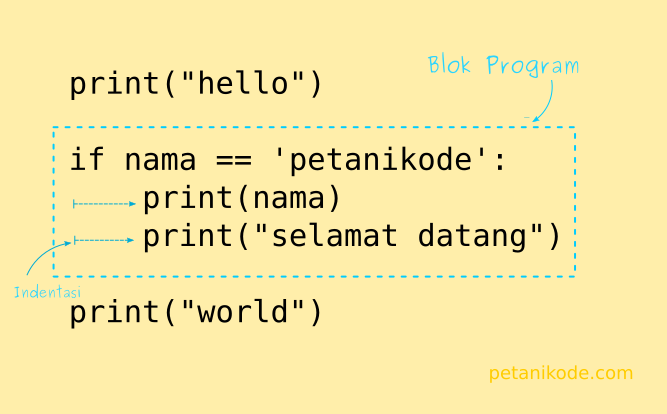
## ALL CAPS digunakan Pada:

GLOBAL\_CONSTANT\_NAME

**4. Penulisan Blok Program Python**

Blok program adalah kumpulan dari beberpaa statement yang digabungkan dalam satu blok.

Penulisan blok program harus ditambahkan indentasi (tab atau spasi 2x/4x).



**Ada beberapa macam blok program:**

1. [Blok Percabangan](https://www.petanikode.com/python-percabangan)
2. [Blok Perulangan](https://www.petanikode.com/python-perulangan)
3. [Blok Fungsi](https://www.petanikode.com/python-fungsi/)
4. Blok Class
5. Blok Exception
6. Blok With

**5. Cara Penulisan Komentar pada Python**

Komentar merupakan baris kode yang tidak akan dieksekusi.

Komentar digunakan untuk memberikan informasi tambahan dan untuk menonaktifkan kode.

Ada beberapa cara menulis komentar pada pemrograman Python.

**Menggunakan Tanda Pagar (#)**

**Menggunakan Tanda Petik**

Selain untuk mengapit teks (string), tanda petik juga dapat digunakan untuk membuat komentar.

Contoh:

"Ini adalah komentar dengan tanda petik ganda"'Ini juga komentar, tapi dengan tanda petik tunggal'

**Menggunakan Triple Tanda Petik**

**Membuat Variabel di Python**

Variabel di python dapat dibuat dengan format seperti ini:

nama\_variabel = <nilai>

**Contoh:**

variabel\_ku = "ini isi variabel"

variabel2 = 20

Kemudian untuk melihat isi variabel, kita dapat menggunakan fungsi print.

print variabel\_ku

print variabel2

**Aturan Penulisan Variabel**

1. Nama variabel boleh diawali menggunakan huruf atau garis bawah (\_), contoh: nama, \_nama, namaKu, nama\_variabel.
2. Karakter selanjutnya dapat berupa huruf, garis bawah (\_) atau angka, contoh: \_\_nama, n2, nilai1.
3. Karakter pada nama variabel bersifat sensitif (case-sensitif). Artinya huruf besar dan kecil dibedakan. Misalnya, variabel\_Ku dan variabel\_ku, keduanya adalah variabel yang berbeda.
4. Nama variabel tidak boleh menggunakan kata kunci yang sudah ada dalam python seperti if, while, for, dsb.

**Menghapus Variabel**

Ketika sebuah variabel tidak dibutuhkan lagi, maka kita bisa menghapusnya dengan **fungsi del()**

**Tipe data**

Cara mengisi nilai variabel ditentukan dengan jenis datanya, misalkan untuk tipe data **teks (string**) maka harus diapit dengan tanda petik ("..."). Sedangkan untuk **angka (integer)** dan **boolean** tidak perlu diapit dengan tanda petik.

Contoh:

nama\_ku = "Petani Kode"

umur = 20

tinggi = 183.22

Untuk memeriksa tipe data pada suatu variabel, kita bisa menggunakan **fungsi type().**

**Jenis-jenis Tipe Data**

Secara umum, tipe data primitif dalam python dibagi menjadi tiga jenis:

1. Tipe data angka
2. Tipe data teks
3. Tipe data boolan

**1. Tipe Data Angka**

Tipe data angka dibagi menjadi beberapa jenis lagi:

**int (Integer): bilangan bulat, contoh 32, 22, 12, 10, dsb.**

**float: bilangan pecahan, contoh 1.3, 4.2, 22.3, dsb.**

Contoh:

harga = 12000 #tipe int

berat = 23.12 #float

jarak = 3e3 #float 3000.0, huruf e artinya eksponen 10

**2. Tipe Data Teks**

Tipe data teks dibagi menjadi dua jenis lagi:

**Char: Karakter, contoh 'R'.**

**String: Kumpulan karakter, contoh "aku lagi makan".**

Penulisan tipe data teks harus diapit dengan tanda petik. Bisa menggunakan petik tunggal ('...'), ganda ("..."), dan tiga ('''...''' atau """...""").

Contoh:

nama = "Ivan"

jenis\_kelamin = 'L'

alamat = """

Jl. Suka Karya, No 32. RT Kode,

Kelurahan Mawar, Jakarta

"""

agama = 'islam'

**3. Tipe data boolean**

Tipe data boolean adalah tipe data yang hanya memiliki dua nilai yaitu True dan False atau 0 dan 1.

Penulisan True dan False, huruf pertamnya harus kapital dan tanpa tanda petik.

Contoh:

bergerak = True

nyala = 1 #sebenarnya tipenya int, tapi bisa juga menjadi bool

**Fungsi-fungsi untuk mengubah tipe data:**

* int() untuk mengubah menjadi integer;
* long() untuk mengubah menjadi integer panjang;
* float() untuk mengubah menjadi float;
* bool() untuk mengubah menjadi boolean;
* chr() untuk mengubah menjadi karakter;
* str() untuk mengubah menjadi string.
* bin() untuk mengubah menjadi bilangan Biner.
* hex() untuk mengubah menjadi bilangan Heksadesimal.
* oct() untuk mengubah menjadi bilangan okta.

**Cara Menampilkan Output**

Seperti yang kita sudah ketahui pada contoh-contoh sebelumnya.

Untuk menampilkan output teks, kita menggunakan fungsi print().

Contoh:

print "Hello World!"

print nama\_variabel

print "Gabung dengan", variabel

**Menampilkan Variabel dan Teks**

Pada contoh di atas kita menggunakan tanda koma (,) untuk menggabungkan teks dan variabel yang akan ditampilkan.

nama = "Petanikode"

print "Hello",nama

Hasil:

Hello Petanikode

Antara kata Hello dan Petanikode terdapat spasi sebagai pemisah, karena kita menggunakan tanda koma.

Hati-hati…

Jangan ditambahkan kurung seperti ini:

nama = "Petanikode"print("Hello",nama)

Karena akan dibaca sebagai Tuple yang akan mengahsilkan output seperti ini:

('Hello', 'Petanikode')

Sebaiknya jangan dikurung kalau menggunakan tanda koma.

Jika ingin menggunakan kurung, maka kita harus menggabungkan teks dan variabelnya dengan tanda plus (+).

Contoh:

nama = "Petanikode"print("Hello " + nama)

Hasilnya:

Hello Petanikode

Menggunakan Fungsi format()

Fungsi format() akan menggabungkan isi variabel dengan teks.

Contoh:

nama = raw\_input("Nama: ")print "Hello {} apa kabar?".format(nama)

Tanda {} akan otomatis diganti sesuai dengan nilai yang kita inputkan ke variabel nama.

Contoh lagi:

nama\_mu = raw\_input("Nama kamu: ")

nama\_dia = raw\_input("Nama dia: ")

print "{} dengan {} sepertinya pasangan yang serasi :)".format(nama\_mu, nama\_dia)

Menggunakan String Formatting Cara Lama

Penggabungan teks dan variabel cara lama menggunakan simbol persen (%).

Contoh:

nama = raw\_input("Inputkan nama: ")print "Selamat datang %s" % nama

Tanda %s akan otomatis diganti dengan nilai yang kita inputkan ke variabel nama.

Contoh lagi:

nama = raw\_input("Inputkan nama: ")

umur = input("Inputkan umur: ")

tinggi = input("Inputkan tinggi badan: ")

print "Hello %s, saat ini usiamu %d tahun dan tinggi badanmu %f cm" % (nama, umur, tinggi)

Tanda %s untuk tipe data teks, %d untuk angka (desimal), dan %f untuk bilangan pecahan.

**. Operator Aritmatika**

Opeartor aritmatika termasuk dalam operator yang paling sering digunakan dalam pemrograman.

Opeartor aritmatika terdiri dari:

| Operator | Simbol |
| --- | --- |
| Penjumlahan | + |
| Pengurangan | - |
| Perkalian | \* |
| Pembagian | / |
| Sisa Bagi | % |
| Pemangkatan | \*\* |

**2. Operator Penugasan**

| Operator | Simbol |
| --- | --- |
| Pengisian | = |
| Penjumlahan | += |
| Pengurangan | -= |
| Perkalian | \*= |
| Pembagian | /= |
| Sisa Bagi | %= |
| Pemangkatan | \*\*= |

**3. Operator Pembanding**

Operator ini digunakan untuk membandingkan dua buah nilai. Operator ini juga dikenal dengan operator relasi dan sering digunakan untuk membuat sebuah logika atau kondisi.

Opertor ini terdiri dari:

| Operator | Simbol |
| --- | --- |
| Lebih Besar | > |
| Lebih Kecil | < |
| Sama Dengan | == |
| Tidak Sama dengan | != |
| Lebih Besar Sama dengan | >= |
| Lebih Kecil Sama dengan | <= |

**4. Operator Logika**

Operator logika digunakan untuk membuat operasi logika, seperti logika AND, OR, dan NOT.

Operator logika terdiri dari:

| Nama | Simbol di Python |
| --- | --- |
| Logika AND | and |
| Logika OR | or |
| Negasi/kebalikan | not |

**5. Operator Bitwise**

Operator Bitwise adalah operator untuk melakukan operasi berdasarkan bit/biner.

Operator ini terdiri dari:

| Nama | Simbol |
| --- | --- |
| AND | & |
| OR | | |
| XOR | ^ |
| Negasi/kebalikan | ~ |
| Left Shift | << |
| Right Shift | >> |

**6. Operator Ternary**

Operator ternary juga dikenal dengan operator kondisi, karena digunakan untuk membuat sebuah ekspresi kondisi seperti [percabgan IF/ELSE](https://www.petanikode.com/python-percabangan).

Operator ternary sebenarnya tidak ada dalam Python, tapi python punya cara lain untuk menggantikan operator ini.

Pada bahasa pemrograman lain operator ternary menggunakan tanda tanya (?) dan titik dua (:).

kondisi ? <nilai true> : <nilai false>

Contoh:

aku = (umur < 10) ? "bocah" : "dewasa"

Dalam Python bentuknya berbeda, yaitu menggunakann IF/ELSE dalam satu baris.

<Nilai True> if Kondisi else <Nilai False>

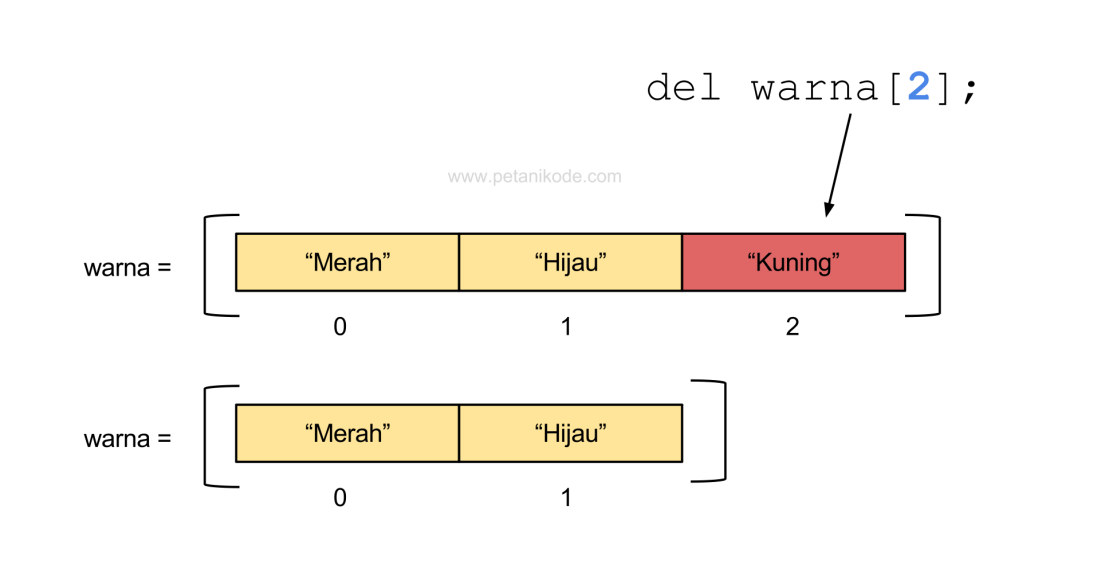
**Menambahkan Item List**

Tedapat Tiga metode (method) atau fungsi yang bisa digunakan untuk menambahkan isi atau item ke List:

* prepend(item) menambahkan item dari depan;
* append(item) menambahkan item dari belakang.
* insert(index, item) menambahkan item dari indeks tertentu

Menghapus Item di List

Untuk menghapus salah satu isi dari List, kita bisa menggunakan perintah del.



Perintah del akan menghapus sebuah variabel dari memori.

Contoh:

# Membuat List

todo\_list = [

"Balajar Python",

"Belajar Django",

"Belajar MongoDB",

"Belajar Sulap",

"Belajar Flask"]

# Misalkan kita ingin menghapus "Belajar Sulap"# yang berada di indeks ke-3del todo\_list[3]

print todo\_list

Hasilnya, "Belajar Sulap" akan dihapus:

['Balajar Python', 'Belajar Django', 'Belajar MongoDB', 'Belajar Flask']

Selain menggunakan perintah del, kita juga bisa menggunakan method remove() dengan paramter item yang akan dihapus.

Contoh:

# mula-mula kita punya list

a = ["a", "b", "c", "d"]# kemudian kita hapus b

a.remove("b")

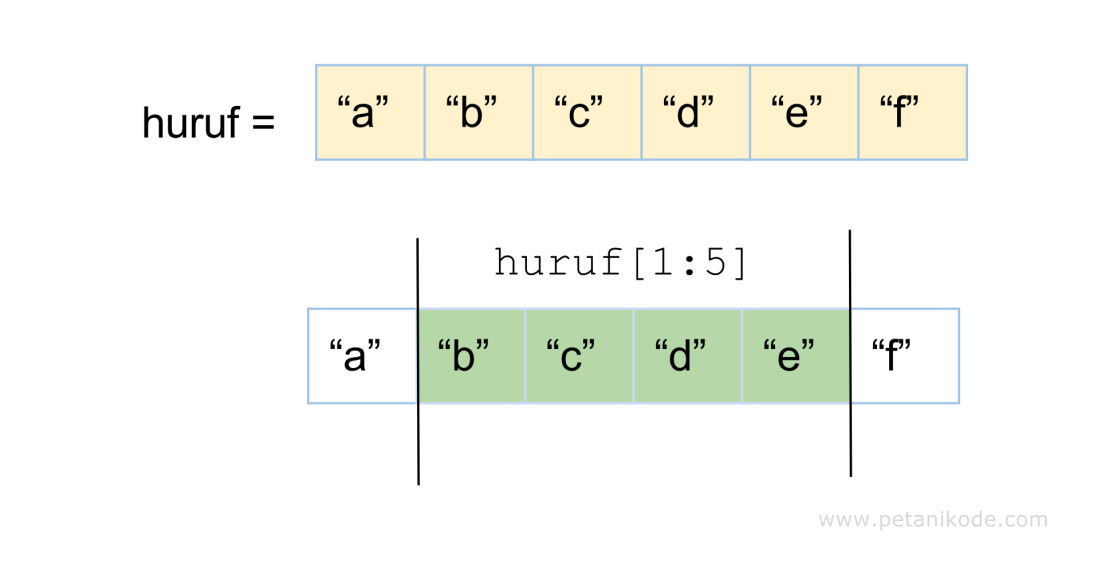
print a

Hasilnya:

["a", "c", "d"]

Memotong list

Seperti string, list juga dapat dipotong-potong.



Contoh:

# Kita punya list warna

warna = ["merah", "hijau", "kuning", "biru", "pink", "ungu"]

# Kita potong dari indeks ke-1 sampai ke-5print warna[1:5]

Hasilnya:

['hijau', 'kuning', 'biru', 'pink']