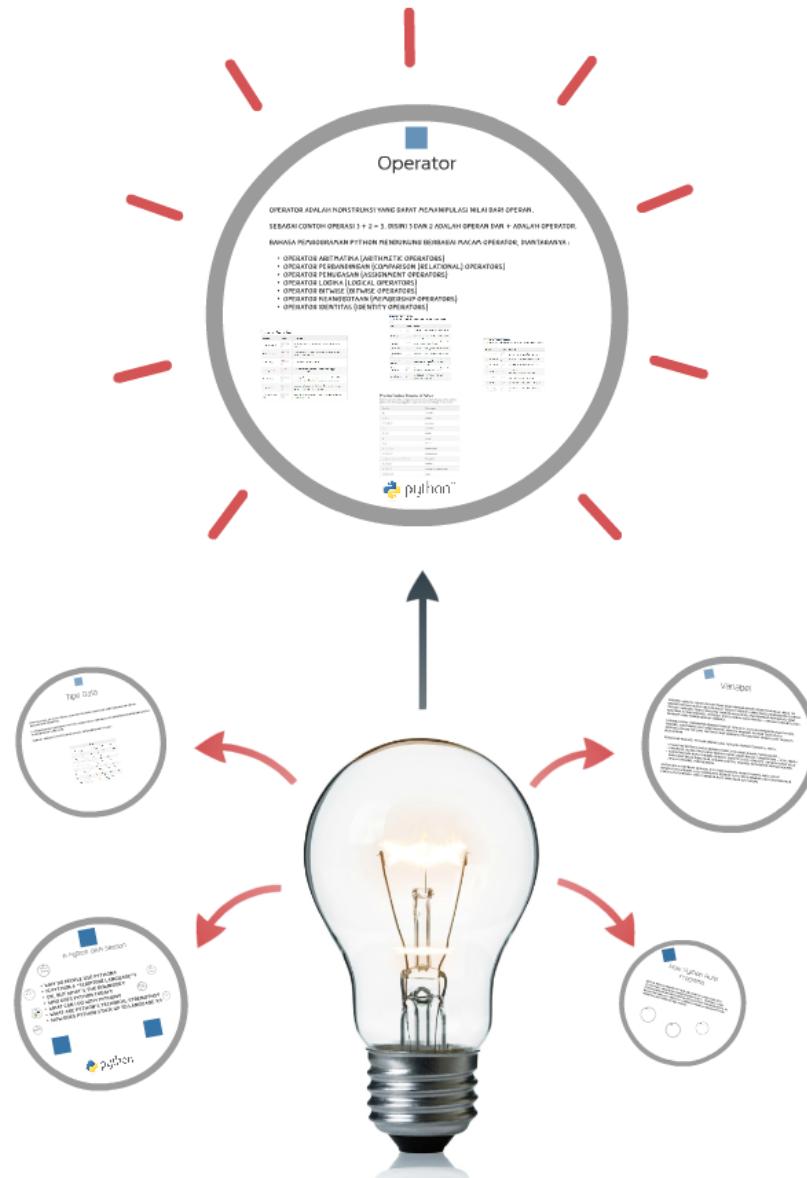


Pemrograman Berorientasi Objek

created by Aidil Saputra Kirsan



Pemrograman Berorientasi Objek

created by Aidil Saputra Kirsan



A Python Q&A Session

- WHY DO PEOPLE USE PYTHON?
- IS PYTHON A “SCRIPTING LANGUAGE”?
- OK, BUT WHAT’S THE DOWNSIDE?
- WHO USES PYTHON TODAY?
- WHAT CAN I DO WITH PYTHON?
- WHAT ARE PYTHON’S TECHNICAL STRENGTHS?
- HOW DOES PYTHON STACK UP TO LANGUAGE X?



 python™



Why Do People Use Python?

- Software Quality

SECARA DESAIN, PYTHON MENGIMPLEMENTASIKAN SINTAKS YANG SENGAJA DIBUAT SEDERHANA DAN MUDAH DIBACA SERTA MODEL PEMROGRAMAN YANG SANGAT KOHEREN. "FIT YOUR BRAIN"

- Developer Productivity

STAF PEMROGRAMAN SEKARANG SERING DIMINTA UNTUK MENYELESAIKAN TUGAS YANG SAMA DENGAN LEBIH SEDIKIT ORANG. DALAM SKENARIO INI, PYTHON TELAH BERSINAR SEBAGAI ALAT YANG MEMUNGKINKAN PEMROGRAM UNTUK MENYELESAIKAN LEBIH BANYAK DENGAN SEDIKIT USAHA





Is Python a “Scripting Language”?

KARENA ISTILAH "SCRIPTING LANGUAGE" MEMILIKI BEGITU BANYAK ARTI YANG BERBEDA UNTUK PENGAMAT YANG BERBEDA, BEBERAPA LEBIH SUKA BAHWA ITU TIDAK DITERAPKAN PADA PYTHON SAMA SEKALI. FAKTANYA, ORANG CENDERUNG MEMBUAT TIGA ASOSIASI YANG SANGAT BERBEDA, BEBERAPA DI ANTARANYA LEBIH BERGUNA DARIPADA YANG LAIN,

- SHELL TOOLS
- CONTROL LANGUAGE
- EASE OF USE

JADI, APAKAH PYTHON ADALAH BAHASA SCRIPTING ATAU BUKAN? ITU TERGANTUNG PADA SIAPA ANDA BERTANYA. SECARA UMUM, ISTILAH "SCRIPTING" MUNGKIN PALING BAIK DIGUNAKAN UNTUK MENGGAMBARKAN MODE PENGEMBANGAN YANG CEPAT DAN FLEKSIBEL YANG DIDUKUNG PYTHON, DARIPADA DOMAIN APLIKASI TERTENTU.





OK, but What's the Downside?

SATU-SATUNYA KELEMAHAN PYTHON YANG DITEMUKAN ADALAH,
SEPERTI YANG SAAT INI DITERAPKAN, KECEPATAN EKSEKUSINYA
MUNGKIN TIDAK SELALU SECEPAT BAHASA YANG DIKOMPILASI
SEPERTI C DAN C++.





Who Uses Python Today?

PYTHON MENIKMATI BASIS PENGGUNA YANG BESAR DAN KOMUNITAS PENGEMBANG YANG SANGAT AKTIF. KARENA PYTHON TELAH ADA SELAMA SEKITAR 30 TAHUN DAN TELAH DIGUNAKAN SECARA LUAS, PYTHON JUGA SANGAT STABIL DAN KUAT. SELAIN DIGUNAKAN OLEH PENGGUNA INDIVIDU, PYTHON JUGA DITERAPKAN DALAM PRODUK YANG MENGHASILKAN PENDAPATAN NYATA OLEH PERUSAHAAN NYATA. CONTOHNYA:

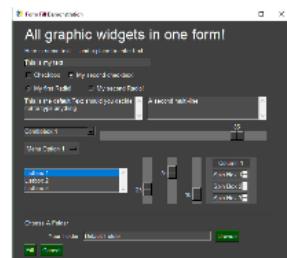
- GOOGLE MAKES EXTENSIVE USE OF PYTHON IN ITS WEB SEARCH SYSTEMS, AND EMPLOYS PYTHON'S CREATOR
 - THE YOUTUBE VIDEO SHARING SERVICE IS LARGELY WRITTEN IN PYTHON.
 - THE POPULAR BITTORRENT PEER-TO-PEER FILE SHARING SYSTEM IS A PYTHON PROGRAM.
 - GOOGLE'S POPULAR APP ENGINE WEB DEVELOPMENT FRAMEWORK USES PYTHON AS ITS APPLICATION LANGUAGE.
 - EVE ONLINE, A MASSIVELY MULTIPLAYER ONLINE GAME (MMOG), MAKES EXTENSIVE USE OF PYTHON
 - MAYA, A POWERFUL INTEGRATED 3D MODELING AND ANIMATION SYSTEM, PROVIDES A PYTHON SCRIPTING API
 - INTEL, CISCO, HEWLETT-PACKARD, SEAGATE, QUALCOMM, AND IBM USE PYTHON FOR HARDWARE TESTING
 - INDUSTRIAL LIGHT & MAGIC, PIXAR, AND OTHERS USE PYTHON IN THE PRODUCTION OF ANIMATED MOVIES
 - JPMORGAN CHASE, UBS, GETCO, AND CITADEL APPLY PYTHON FOR FINANCIAL MARKET FORECASTING
 - NASA, LOS ALAMOS, FERMILAB, JPL, AND OTHERS USE PYTHON FOR SCIENTIFIC PROGRAMMING TASKS
-
- iROBOT USES PYTHON TO DEVELOP COMMERCIAL ROBOTIC DEVICES.
 - ESRI USES PYTHON AS AN END-USER CUSTOMIZATION TOOL FOR ITS POPULAR GIS MAPPING PRODUCTS.
 - THE NSA USES PYTHON FOR CRYPTOGRAPHY AND INTELLIGENCE ANALYSIS.
 - THE IRONPORT EMAIL SERVER PRODUCT USES MORE THAN 1 MILLION LINES OF PYTHON CODE TO DO ITS JOB.
 - THE ONE LAPTOP PER CHILD (OLPC) PROJECT BUILDS ITS USER INTERFACE AND ACTIVITY MODEL IN PYTHON.





What Can I Do with Python?

- SYSTEMS PROGRAMMING
- GUI'S
- INTERNET SCRIPTING
- COMPONENT INTEGRATION
- DATABASE PROGRAMMING
- RAPID PROTOTYPING
- NUMERIC AND SCIENTIFIC PROGRAMMING
- GAMING, IMAGES, SERIAL PORTS, XML, ROBOTS, AND MORE





What Are Python's Technical Strengths?

JIKA ANDA BELUM MEMILIKI LATAR BELAKANG PEMROGRAMAN, BAHASA DI BEBERAPA BAGIAN BERIKUTNYA MUNGKIN AKAN SEDIKIT MEMBINGUNGKAN— JANGAN KHAWATIR, KITA AKAN MENJELAJAHI BEBERAPA FITUR TEKNIS PYTHON:

- IT'S OBJECT-ORIENTED
- IT'S FREE
- IT'S PORTABLE
- IT'S POWERFUL
- IT'S MIXABLE
- IT'S EASY TO USE
- IT'S EASY TO LEARN





How Does Python Stack Up to Language X?

SEMENTARA BAHASA LAIN JUGA MERUPAKAN ALAT YANG BERGUNA UNTUK DIKETAHUI DAN DIGUNAKAN, BANYAK ORANG MENEMUKN BAHWA PYTHON:

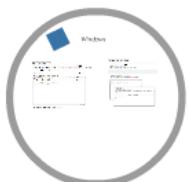
- LEBIH KUAT DARI TCL. DUKUNGAN PYTHON UNTUK "PEMROGRAMAN DALAM SKALA BESAR" MEMBUATNYA BERLAKU UNTUK PENGEMBANGAN SISTEM YANG LEBIH BESAR.
- MEMILIKI SINTAKS YANG LEBIH BERSIH DAN DESAIN YANG LEBIH SEDERHANA DARIPADA PERL, YANG MEMBUATNYA LEBIH MUDAH DIBACA DAN DIPELIHARA SERTA MEMBANTU MENGURANGI BUG PROGRAM.
- LEBIH SEDERHANA DAN LEBIH MUDAH DIGUNAKAN DARIPADA JAVA. PYTHON ADALAH BAHASA SCRIPTING, TETAPI JAVA MEWARISI BANYAK KOMPLEKSITAS DAN SINTAKS DARI BAHASA SISTEM SEPERTI C++.
- LEBIH SEDERHANA DAN LEBIH MUDAH DIGUNAKAN DARIPADA C++, TETAPI SERING KALI TIDAK BERSAING DENGAN C++; SEBAGAI BAHASA SCRIPTING, PYTHON BIASANYA MELAYANI PERAN YANG BERBEDA.
- LEBIH KUAT DAN LEBIH LINTAS PLATFORM DARIPADA VISUAL BASIC. SIFATNYA YANG OPEN SOURCE JUGA BERARTI TIDAK DIKENDALIKAN OLEH SATU PERUSAHAAN.
- LEBIH MUDAH DIBACA DAN BERTUJUAN UMUM DARIPADA PHP. PYTHON TERKADANG DIGUNAKAN UNTUK MEMBANGUN SITUS WEB, TETAPI JUGA BANYAK DIGUNAKAN DI HAMPIR SETIAP DOMAIN KOMPUTER LAINNYA, MULAI DARI ROBOTIKA HINGGA ANIMASI FILM.
- LEBIH MATANG DAN MEMILIKI SINTAKS YANG LEBIH MUDAH DIBACA DARIPADA RUBY. TIDAK SEPERTI RUBY DAN JAVA, OOP ADALAH OPSI DALAM PYTHON—PYTHON TIDAK MEMAKSAKAN OOP PADA PENGGUNA ATAU PROYEK YANG MUNGKIN TIDAK BERLAKU.
- MEMILIKI CITA RASA DINAMIS DARI BAHASA SEPERTI SMALLTALK DAN LISP, TETAPI JUGA MEMILIKI SINTAKS TRADISIONAL YANG SEDERHANA YANG DAPAT DIAKSES OLEH PENGEMBANG SERTA PENGGUNA AKHIR DARI SISTEM YANG DAPAT DISESUAIKAN.





How Python Runs Programs

UNTUK MENJALANKAN PYTHON ADA BANYAK CARA YANG BISA DILAKUKAN. ANDA BISA MENGGUNAKAN SHELL, TERMINAL ATAU MENGGUNAKAN IDE (INTEGRATED DEVELOPMENT ENVIRONMENT). DI BAWAH INI ADALAH LANGKAH-LANGKAH MENJALANKAN PYTHON DENGAN CARA YANG PALING MUDAH.





Linux

1. Buka terminal **CTRL + ALT + T**
2. Ketik python maka Anda akan masuk ke Python shell.
3. Tuliskan script Python Anda, contoh: `print("Selamat datang di Python")`. jika sudah tekan tombol **ENTER**, dan script Python akan dijalankan/eksekusi.
4. Untuk keluar dari Python shell ketik `exit()`

atau

1. Gunakan teks editor, misalnya gedit.
2. Buat file baru, dan ketikan script python Anda, contoh: `print("Selamat datang di Python")`.
3. Save As dengan ekstensi `.py` (contoh: `cetak.py`).
4. Jalankan file dengan menggunakan Terminal.
5. Buka terminal **CTRL + ALT + T**.
6. Masuk ke direktori dimana file Python Anda disimpan (contoh: `cd /Users/admin/Desktop/`).
7. Jalankan script Python dengan menggunakan python diikuti dengan nama file (contoh: `python cetak.py`).
8. Script Python Anda akan dieksekusi/dijalankan.



1. Buka terminal **CTRL + ALT + T**
2. Ketik python maka Anda akan masuk ke Python shell.
3. Tuliskan script Python Anda, contoh: `print("Selamat datang di Python")`. jika sudah tekan tombol **ENTER**, dan script Python akan dijalankan/eksekusi.
4. Untuk keluar dari Python shell ketik `exit()`

atau

1. Gunakan teks editor, misalnya gedit.
2. Buat file baru, dan ketikan script python Anda, contoh: `print("Selamat datang di Python")`.
3. Save As dengan ekstensi `.py` (contoh: `cetak.py`).
4. Jalankan file dengan menggunakan Terminal.
5. Buka terminal **CTRL + ALT + T**.
6. Masuk ke direktori dimana file Python Anda disimpan (contoh: `cd /Users/admin/Desktop/`).
7. Jalankan script Python dengan menggunakan `python` diikuti dengan nama file (contoh: `python cetak.py`).
8. Script Python Anda akan dieksekusi/dijalankan.



Windows

Menggunakan Shell

1. Buka IDLE (python shell di windows). Anda bisa mencarinya di tombol **START**.
2. Tuliskan script Python Anda contoh: `print("Selamat datang di Python")`, jika sudah tekan tombol **ENTER**, dan script Python akan dijalankan/eksekusi.

```
Python 3.7.6 Shell
File Edit Shell Debug Options Window Help
Python 3.7.6 (tags/v3.7.6:43167af, Dec 18 2018, 23:44:00) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> print("Selamat datang di Python")
Selamat datang di Python
>>>
```

Ln 6 Cell 4

1. Untuk keluar dari Python shell ketik `exit()`.

Menggunakan Script Editor

1. Untuk menjalankan script yang disimpan dalam file, buka IDLE (python shell di windows). Anda bisa mencarinya di tombol **START**.
2. Klik menu **File -> New File**
3. Tulis script Python pada window yang muncul, contoh:

```
print("Selamat Python")
print("di belajarpython.com")

1. Simpan script lewat menu File -> Save
2. Jalankan program dengan klik menu Run -> Run Module

Python 3.7.6 Shell
File Edit Shell Debug Options Window Help
Python 3.7.6 (tags/v3.7.6:43167af, Dec 18 2018, 23:44:00) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> python belajarpython.py
print("Selamat Python")
print("di belajarpython.com")

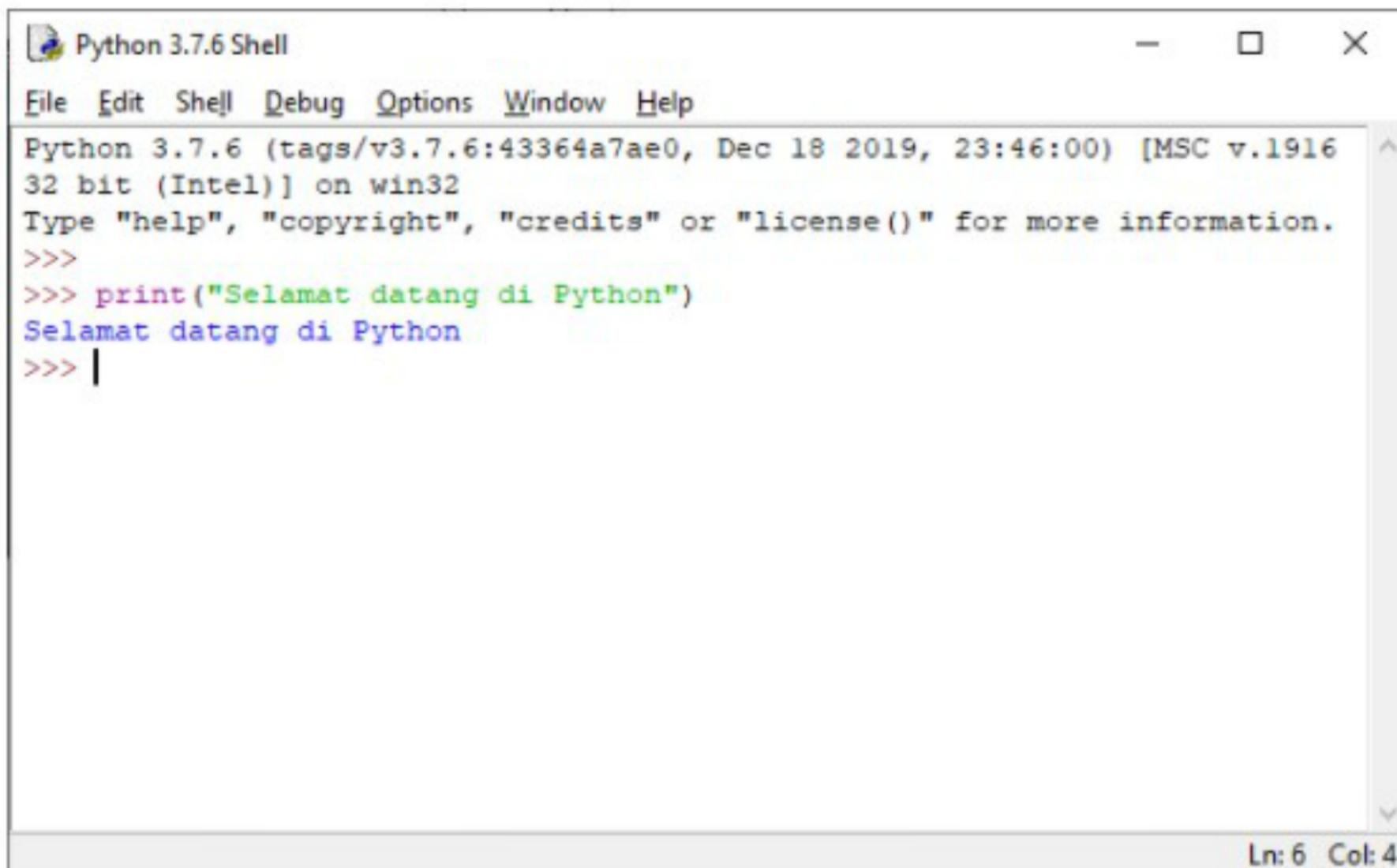
-----
Selamat Python
di belajarpython.com
>>>
```

Ln 7 Cell 4



Menggunakan Shell

1. Buka IDLE (python shell di windows), Anda bisa mencarinya di tombol **START**.
2. Tuliskan script Python Anda, contoh: `print("Selamat datang di Python")`. jika sudah tekan tombol **ENTER**, dan script Python akan dijalankan/eksekusi.



The screenshot shows the Python 3.7.6 Shell window. The title bar reads "Python 3.7.6 Shell". The menu bar includes File, Edit, Shell, Debug, Options, Window, and Help. The main window displays the Python interpreter's welcome message and a command-line session. The user has typed `>>> print("Selamat datang di Python")` and pressed Enter, resulting in the output "Selamat datang di Python". The status bar at the bottom right indicates "Ln: 6 Col: 4".

```
Python 3.7.6 (tags/v3.7.6:43364a7ae0, Dec 18 2019, 23:46:00) [MSC v.1916  
32 bit (Intel)] on win32  
Type "help", "copyright", "credits" or "license()" for more information.  
>>>  
>>> print("Selamat datang di Python")  
Selamat datang di Python  
>>> |
```



1. Untuk keluar dari Python shell ketik `exit()`

Menggunakan Script Editor

1. Untuk menjalankan script yang disimpan dalam file, buka IDLE (python shell di windows), Anda bisa mencarinya di tombol **START**.
2. Klik menu **File - New File**
3. Tulis script Python pada window yang muncul, contoh:

```
print("Belajar Python")
print("di belajarpython.com")
```

1. Simpan script lewat menu **File - Save**
2. Jalankan program dengan klik menu **Run - Run Module**

The screenshot shows two windows side-by-side. The top window is titled 'coba1.py - D:\coba1.py (3.7.6)' and contains the Python code:

```
print("Belajar Python")
print("di belajarpython.com")
```

The bottom window is titled 'Python 3.7.6 Shell' and shows the output of running the script:

```
Python 3.7.6 (tags/v3.7.6:43364a7ae0, Dec 18 2019, 23:46:00) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: D:\coba1.py =====
Belajar Python
di belajarpython.com
>>> |
```

The status bar at the bottom right of the shell window indicates 'Ln: 7 Col: 4'.



Mac OS

1. Buka terminal.
2. Ketik `python` maka Anda akan masuk ke Python shell.
3. Tuliskan script Python Anda, contoh: `print("Selamat datang di Python")`, jika sudah tekan tombol `ENTER`, dan script Python akan dijalankan/eksekusi.
4. Untuk keluar dari Python shell ketik `exit()`

atau

1. Gunakan teks editor.
2. Buat file baru, dan ketikan script python Anda, contoh: `print("Selamat datang di Python")`.
3. Save As dengan ekstensi `.py` (contoh: `cetak.py`).
4. Jalankan file dengan menggunakan Terminal.
5. Buka terminal `CTRL + ALT + T`
6. Masuk ke direktori dimana file Python Anda disimpan (contoh: `cd /Users/admin/Desktop/`).
7. Jalankan script Python dengan menggunakan `python` diikuti dengan nama file (contoh: `python cetak.py`).
8. Script Python Anda akan dieksekusi/dijalankan.



1. Buka terminal.
2. Ketik `python` maka Anda akan masuk ke Python shell.
3. Tuliskan script Python Anda, contoh: `print("Selamat datang di Python")`, jika sudah tekan tombol **ENTER**, dan script Python akan dijalankan/eksekusi.
4. Untuk keluar dari Python shell ketik `exit()`

atau

1. Gunakan teks editor.
2. Buat file baru, dan ketikan script python Anda, contoh: `print("Selamat datang di Python")`.
3. Save As dengan ekstensi `.py` (contoh: `cetak.py`).
4. Jalankan file dengan menggunakan Terminal.
5. Buka terminal **CTRL** + **ALT** + **T**
6. Masuk ke direktori dimana file Python Anda disimpan (contoh: `cd /Users/admin/Desktop/`).
7. Jalankan script Python dengan menggunakan `python` diikuti dengan nama file (contoh: `python cetak.py`).
8. Script Python Anda akan dieksekusi/dijalankan.



Tipe Data

TIPE DATA ADALAH SUATU MEDIA ATAU MEMORI PADA KOMPUTER YANG DIGUNAKAN UNTUK MENAMPUNG INFORMASI.

PYTHON SENDIRI MEMPUNYAI TIPE DATA YANG CUKUP UNIK BILA KITA BANDINGKAN DENGAN BAHASA PEMROGRAMAN YANG LAIN.

BERIKUT ADALAH TIPE DATA DARI BAHASA PEMROGRAMAN PYTHON :

Tipe Data	Contoh	Penjelasan
Boolean	<code>True</code> atau <code>False</code>	Menyatakan benar <code>True</code> yang bernilai <code>1</code> , atau salah <code>False</code> yang bernilai <code>0</code>
String	<code>"ayo belajar Python"</code>	Menyatakan karakter/kalimat bisa berupa huruf angka, dll (dipisahkan " atau '')
Integer	<code>25</code> atau <code>1289</code>	Menyatakan bilangan bulat
Float	<code>3.14</code> atau <code>0.99</code>	Menyatakan bilangan yang mempunyai koma
Hexadecimal	<code>9a</code> atau <code>1d3</code>	Menyatakan bilangan dalam format heksa (bilangan berbasis 16)
Complex	<code>1 + 5j</code>	Menyatakan pasangan angka real dan imajiner
List	<code>['xyz', 786, 2.23]</code>	Data antalan yang menyimpan berbagai tipe data dan isinya bisa diubah-ubah
Tuple	<code>('xyz', 786, 2.23)</code>	Data antalan yang menyimpan berbagai tipe data tapi isinya tidak bisa diubah
Dictionary	<code>{'name': 'adi', 'id': 12}</code>	Data antalan yang menyimpan berbagai tipe data berupa pasangan pernyataan dan nilai



Tipe Data	Contoh	Penjelasan
Boolean	<code>True</code> atau <code>False</code>	Menyatakan benar <code>True</code> yang bernilai <code>1</code> , atau salah <code>False</code> yang bernilai <code>0</code>
String	<code>"Ayo belajar Python"</code>	Menyatakan karakter/kalimat bisa berupa huruf angka, dll (diapit tanda <code>"</code> atau <code>'</code>)
Integer	<code>25</code> atau <code>1209</code>	Menyatakan bilangan bulat
Float	<code>3.14</code> atau <code>0.99</code>	Menyatakan bilangan yang mempunyai koma
Hexadecimal	<code>9a</code> atau <code>1d3</code>	Menyatakan bilangan dalam format heksa (bilangan berbasis 16)
Complex	<code>1 + 5j</code>	Menyatakan pasangan angka real dan imajiner
List	<code>['xyz' , 786 , 2.23]</code>	Data untaian yang menyimpan berbagai tipe data dan isinya bisa diubah-ubah
Tuple	<code>('xyz' , 768 , 2.23)</code>	Data untaian yang menyimpan berbagai tipe data tapi isinya tidak bisa diubah
Dictionary	<code>{ 'nama' : 'adi' , 'id' : 2 }</code>	Data untaian yang menyimpan berbagai tipe data berupa pasangan penunjuk dan nilai





Variabel

VARIABEL ADALAH LOKASI MEMORI YANG DICADANGKAN UNTUK MENYIMPAN NILAI–NILAI. INI BERARTI BAHWA KETIKA ANDA MEMBUAT SEBUAH VARIABEL ANDA MEMESAN BEBERAPA RUANG DI MEMORI. VARIABEL MENYIMPAN DATA YANG DILAKUKAN SELAMA PROGRAM DIEKSEKUSI, YANG NANTINYA ISI DARI VARIABEL TERSEBUT DAPAT DIUBAH OLEH OPERASI – OPERASI TERTENTU PADA PROGRAM YANG MENGGUNAKAN VARIABEL.

VARIABEL DAPAT MENYIMPAN BERBAGAI MACAM TIPE DATA. DI DALAM PEMROGRAMAN PYTHON, VARIABEL MEMPUNYAI SIFAT YANG DINAMIS, ARTINYA VARIABEL PYTHON TIDAK PERLU DIDEKLARASIKAN TIPE DATA TERTENTU DAN VARIABEL PYTHON DAPAT DIUBAH SAAT PROGRAM DIJALANKAN.

PENULISAN VARIABEL PYTHON SENDIRI JUGA MEMILIKI ATURAN TERTENTU, YAITU :

- KARAKTER PERTAMA HARUS BERUPA HURUF ATAU GARIS BAWAH/UNDERSCORE _
- KARAKTER SELANJUTNYA DAPAT BERUPA HURUF, GARIS BAWAH/UNDERSCORE _ ATAU ANGKA
- KARAKTER PADA NAMA VARIABEL BERSIFAT SENSITIF (CASE-SENSITIF). ARTINYA HURUF KECIL DAN HURUF BESAR DIBEDAKAN. SEBAGAI CONTOH, VARIABEL NAMADEPAN DAN NAMADEPAN ADALAH VARIABEL YANG BERBEDA.

UNTUK MULAI MEMBUAT VARIABEL DI PYTHON CARANYA SANGAT MUDAH, ANDA CUKUP MENULISKAN VARIABEL LALU MENGISINYA DENGAN SUATU NILAI DENGAN CARA MENAMBAHKAN TANDA SAMA DENGAN = DIIKUTI DENGAN NILAI YANG INGIN DIMASUKAN.





Operator

OPERATOR ADALAH KONSTRUKSI YANG DAPAT MEMANIPULASI NILAI DARI OPERAN.

SEBAGAI CONTOH OPERASI $3 + 2 = 5$. DISINI 3 DAN 2 ADALAH OPERAN DAN + ADALAH OPERATOR.

BAHASA PEMROGRAMAN PYTHON MENDUKUNG BERBAGAI MACAM OPERATOR, DIANTARANYA :

- OPERATOR ARITMATIKA (ARITHMETIC OPERATORS)
- OPERATOR PERBANDINGAN (COMPARISON (RELATIONAL) OPERATORS)
- OPERATOR PENUGASAN (ASSIGNMENT OPERATORS)
- OPERATOR LOGIKA (LOGICAL OPERATORS)
- OPERATOR BITWISE (BITWISE OPERATORS)
- OPERATOR KEANGGOTAAN (MEMBERSHIP OPERATORS)
- OPERATOR IDENTITAS (IDENTITY OPERATORS)

Operator Aritmatika		
Operator	Contoh	Penjelasan
Kali (*)	$3 * 2$	Mengalikan nilai dari kiri ke kanan operan atau bilangan
Pembagian (/)	$3 / 2$	Mengbagi nilai operan kiri dengan nilai operan di sebelahnya
Potongan (%)	$3 \% 2$	Mengambil sisa pembagian
Pangkat (**)	$3 ** 2$	Berfungsi untuk mengalihkan operan di sebelahnya
Sisa (%)	$3 \% 2$	Mengambil sisa pembagian dari operan di sebelahnya
Pecahan (/)	$3 // 2$	Mengambil operan depan dan operan di sebelahnya
Pengambilan ([])	$[1, 2, 3][0]$	Berfungsi untuk mengambil operan di sebelahnya

Operator Perbandingan		
Operator	Contoh	Penjelasan
Lebih besar (>)	$3 > 2$	Menunjukkan bahwa operan kiri lebih besar dari operan di sebelahnya
Lebih kecil (<)	$3 < 2$	Menunjukkan bahwa operan kiri lebih kecil dari operan di sebelahnya
Tidak sama dengan (!=)	$3 != 2$	Menunjukkan bahwa operan kiri tidak sama dengan operan di sebelahnya
Sama dengan (==)	$3 == 2$	Menunjukkan bahwa operan kiri sama dengan operan di sebelahnya
Lebih besar atau sama dengan (>=)	$3 >= 2$	Menunjukkan bahwa operan kiri lebih besar atau sama dengan operan di sebelahnya
Lebih kecil atau sama dengan (<=)	$3 <= 2$	Menunjukkan bahwa operan kiri lebih kecil atau sama dengan operan di sebelahnya

Operator Pendekatan		
Operator	Contoh	Penjelasan
Isinstance (isinstance())	<code>isinstance(1, int)</code>	Menunjukkan bahwa operan kiri merupakan objek tipe data yang diminta
Is (is)	<code>a is b</code>	Menunjukkan bahwa operan kiri dan operan di sebelahnya adalah referensi ke alamat yang sama
Is not (is not)	<code>a is not b</code>	Menunjukkan bahwa operan kiri dan operan di sebelahnya bukan referensi ke alamat yang sama
Isinstance not (isinstance not)	<code>isinstance(1, str)</code>	Menunjukkan bahwa operan kiri bukan merupakan objek tipe data yang diminta
Is not same (is not same)	<code>a is not same b</code>	Menunjukkan bahwa operan kiri dan operan di sebelahnya bukan referensi ke alamat yang sama

Prioritas Eksekusi Operator di Python		
Operator	Penjelasan	Prioritas
()	Awal	1
**	Bilangan	2
*, /, %, //	Awal	3
+, -	Awal	4
*, /, %, //	Bilangan	5
+, -	Bilangan	6
*, /, %, //	Perbandingan	7
==, !=, >, <, >=, <=	Perbandingan	8
is, is not	Pendekatan	9
in, not in	Pendekatan	10
and, or, not	Logika	11



Operator Aritmatika

Operator	Contoh	Penjelasan
Penjumlahan $+$	$1 + 3 =$ 4	Menjumlahkan nilai dari masing-masing operan atau bilangan
Pengurangan $-$	$4 - 1 =$ 3	Mengurangi nilai operan di sebelah kiri menggunakan operan di sebelah kanan
Perkalian $*$	$2 * 4 =$ 8	Mengalikan operan/bilangan
Pembagian $/$	$10 / 5 =$ 2	Untuk membagi operan di sebelah kiri menggunakan operan di sebelah kanan
Sisa Bagi $\%$	$11 \% 2 =$ 1	Mendapatkan sisa pembagian dari operan di sebelah kiri operator ketika dibagi oleh operan di sebelah kanan
Pangkat $**$	$8 ** 2 =$ 64	Memangkatkan operan disebelah kiri operator dengan operan di sebelah kanan operator
Pembagian Bulat	$10 // 3$ = 3	Sama seperti pembagian. Hanya saja angka dibelakang koma dihilangkan

Operator Penugasan

Operator penugasan digunakan untuk memberikan atau memodifikasi nilai ke dalam sebuah variabel.

Operator	Contoh	Penjelasan
Sama dengan <code>=</code>	<code>a = 1</code>	Memberikan nilai di kanan ke dalam variabel yang berada di sebelah kiri.
Tambah sama dengan <code>+=</code>	<code>a += 2</code>	Memberikan nilai variabel dengan nilai variabel itu sendiri ditambah dengan nilai di sebelah kanan.
Kurang sama dengan <code>-=</code>	<code>a -= 2</code>	Memberikan nilai variabel dengan nilai variabel itu sendiri dikurangi dengan nilai di sebelah kanan.
Kali sama dengan <code>*=</code>	<code>a *= 2</code>	Memberikan nilai variabel dengan nilai variabel itu sendiri dikali dengan nilai di sebelah kanan.
Bagi sama dengan <code>/=</code>	<code>a /= 4</code>	Memberikan nilai variabel dengan nilai variabel itu sendiri dibagi dengan nilai di sebelah kanan.
Sisa bagi sama dengan <code>%=</code>	<code>a %= 3</code>	Memberikan nilai variabel dengan nilai variabel itu sendiri dibagi dengan nilai di sebelah kanan. Yang diambil nantinya adalah sisa baginya.
Pangkat sama dengan <code>**=</code>	<code>a **= 3</code>	Memberikan nilai variabel dengan nilai variabel itu sendiri dipangkatkan dengan nilai di sebelah kanan.
Pembagian bulat sama dengan <code>//=</code>	<code>a // 3</code>	Membagi bulat operan sebelah kiri operator dengan operan sebelah kanan operator kemudian hasilnya diisikan ke operan sebelah kiri.

Operator Perbandingan

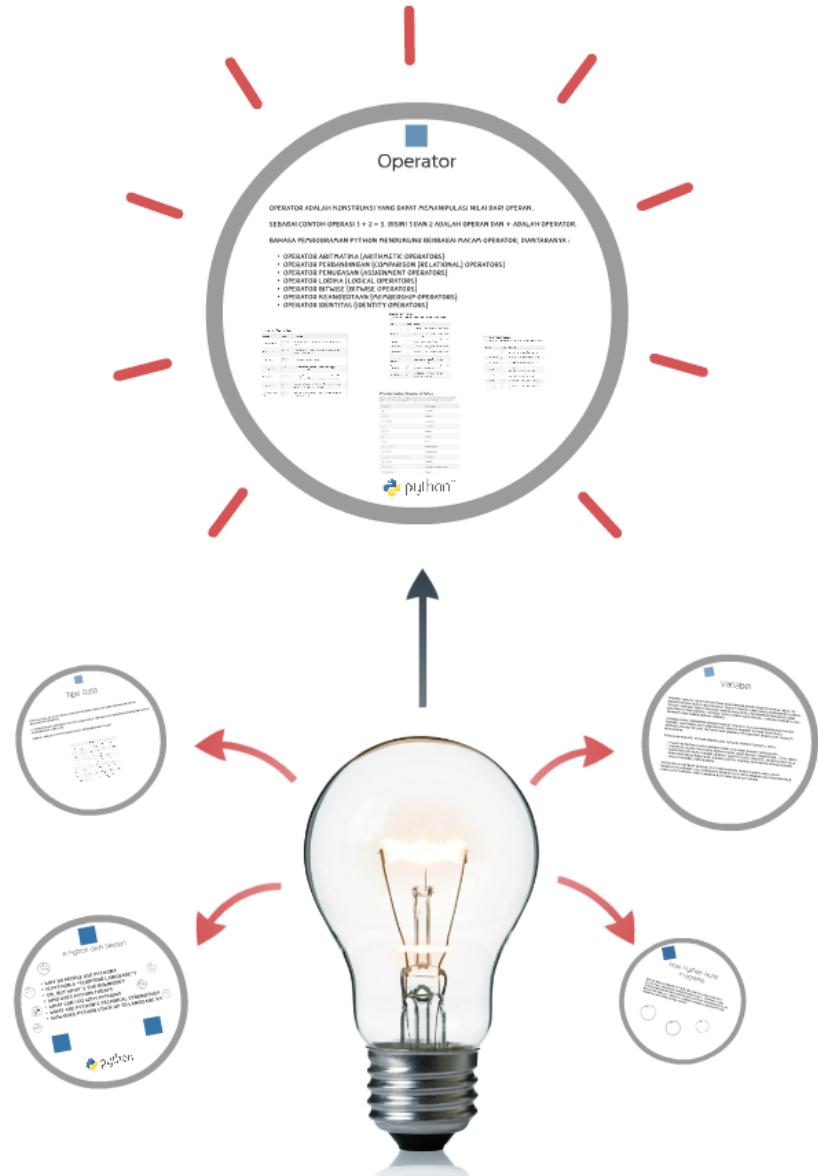
Operator perbandingan (comparison operators) digunakan untuk membandingkan suatu nilai dari masing-masing operan.

Operator	Contoh	Penjelasan
Sama dengan <code>==</code>	<code>1 == 1</code>	bernilai True Jika masing-masing operan memiliki nilai yang sama, maka kondisi bernilai benar atau True.
Tidak sama dengan <code>!=</code>	<code>2 != 2</code>	bernilai False Akan menghasilkan nilai kebalikan dari kondisi sebenarnya.
Tidak sama dengan <code><></code>	<code>2 <> 2</code>	bernilai False Akan menghasilkan nilai kebalikan dari kondisi sebenarnya.
Lebih besar dari <code>></code>	<code>5 > 3</code>	bernilai True Jika nilai operan kiri lebih besar dari nilai operan kanan, maka kondisi menjadi benar.
Lebih kecil dari <code><</code>	<code>5 < 3</code>	bernilai True Jika nilai operan kiri lebih kecil dari nilai operan kanan, maka kondisi menjadi benar.
Lebih besar atau sama dengan <code>>=</code>	<code>5 >= 3</code>	bernilai True Jika nilai operan kiri lebih besar dari nilai operan kanan, atau sama, maka kondisi menjadi benar.
Lebih kecil atau sama dengan <code><=</code>	<code>5 <= 3</code>	bernilai True Jika nilai operan kiri lebih kecil dari nilai operan kanan, atau sama, maka kondisi menjadi benar.

Prioritas Eksekusi Operator di Python

Dari semua operator diatas, masing-masing mempunyai urutan prioritas yang nantinya prioritas pertama akan dilakukan paling pertama, begitu seterusnya sampai dengan prioritas terakhir.

Operator	Keterangan
**	Aritmatika
~, +, -	Bitwise
*, /, %, //	Aritmatika
+, -	Aritmatika
>>, <<	Bitwise
&	Bitwise
^,	Bitwise
<=, <, >, >=	Perbandingan
<>, ==, !=	Perbandingan
=, %=, /=, //=, -=, +=, *=, **=	Penugasan
is, is not	Identitas
in, not in	Membership (Keanggotaan)
not, or, and	Logika



Pemrograman Berorientasi Objek

created by Aidil Saputra Kirsan