

▼ Pengenalan pemrograman Python

1. Apa itu Python?

- Python adalah bahasa pemrograman tingkat tinggi yang mudah dipelajari dan digunakan.
- Diciptakan oleh Guido van Rossum pada tahun 1989 dan pertama kali dirilis pada tahun 1991.

2. Kelebihan Python

- Mudah dibaca dan ditulis, sehingga cocok untuk pemula.
- Memiliki berbagai pustaka (libraries) yang kaya, seperti NumPy, pandas, dan Matplotlib.
- Portabilitas tinggi, dapat digunakan di berbagai platform.
- Populer di berbagai industri, termasuk pengembangan web, data science, dan pengembangan aplikasi desktop.

▼ 3. Instalasi Python

- Kunjungi situs resmi Python di python.org.
- Unduh installer Python yang sesuai dengan sistem operasi Anda.
- Ikuti panduan instalasi yang diberikan.

Instalasi di Windows 7, 8, 10

Untuk menginstall Python di OS Windows, berikut ini adalah tata caranya:

1. Masuk ke halaman resmi untuk [Download Python](https://python.org).
2. Download versi Python yang sudah stabil. Disini saya menggunakan versi 3.7.6.
3. Jalankan file *.exe yang sudah didownload.
4. Jangan lupa check tanda checklist pada opsi **Add python.exe to PATH** seperti gambar.
5. Checklist semua item pada **Optional Features** walaupun mungkin tidak dipakai.
6. Di menu **Advanced Options**, disarankan hanya mencentang opsi berikut: **Install for all users**, **Associate files with Python**, **Create shortcuts for installed applications**, dan **Add Python to environment variables**.



Instalasi di Linux

Instalasi Python di OS Linux cukup mudah. Jika anda menggunakan Fedora / Debian, cobalah untuk mengecek terlebih dahulu, apakah python sudah terinstall di PC anda dengan command berikut:

```
$ python -V
```

Jika sudah terinstall maka akan muncul informasi seperti ini

```
Python 3.10.11
```

Jika belum, maka dapat menginstal dengan mengetikkan command berikut:

```
# Untuk versi terbaru
$ sudo apt-get install python3

# Untuk versi spesifik
$ sudo apt-get install python3.8
# atau
$ sudo apt-get install python3.9
```

Membuat dan Menjalankan Skrip Python

- Buka terminal atau command prompt.
- Ketik `python` untuk masuk ke lingkungan Python interaktif atau buat file Python dengan ekstensi **.py** (misalnya, **hello.py**) dan ketik kode berikut:

```
print("Hello, World!")
```

- Simpan file dan jalankan dengan perintah `python hello.py`.
- Anda akan melihat output "Hello, World!".

▼ 4. Dasar-dasar Bahasa Pemrograman Python

a. Variabel dan Tipe Data

Variabel digunakan untuk menyimpan data. Tipe data dasar: int, float, string, boolean. Contoh:

```
1 angka = 10
2 nama = "John"
```

▼ b. Struktur Kontrol

Penggunaan kondisi (if-else) untuk percabangan. Penggunaan perulangan (for dan while) untuk mengulang tugas. Contoh:

```
1 if angka > 5:
2     print("Lebih besar dari 5")
3 else:
4     print("Kurang dari atau sama dengan 5")
```

```
Lebih besar dari 5
```

```
1 while angka < 12:
2     print("Angka masih kurang dari 12, angka saat ini: `angka` =", angka, "+ 1")
3     angka += 1
4 print("Angka saat ini:", angka)
```

```
Angka masih kurang dari 12, angka saat ini: `angka` = 10 + 1
Angka masih kurang dari 12, angka saat ini: `angka` = 11 + 1
Angka saat ini: 12
```

▼ c. List dan Dictionary

List digunakan untuk menyimpan kumpulan data. Dictionary adalah struktur data yang menghubungkan kunci dengan nilai. Contoh:

```
1 daftar_angka = [1, 2, 3, 4, 5]
2 kamus = {"nama": "John", "usia": 30}
```

▼ 5. Fungsi dan Modularitas

a. Fungsi

Fungsi adalah blok kode yang dapat dipanggil dan digunakan kembali. Anda dapat membuat fungsi sendiri. Contoh:

```
1 def sapa(nama):  
2     print(f"Halo, {nama}!")  
3  
4 sapa("Andi")  
  
    Halo, Andi!
```

b. Modularitas

Memisahkan kode ke dalam modul/modul untuk meningkatkan keterbacaan dan pemeliharaan. Menggunakan import untuk mengakses modul eksternal.

6. Pustaka (Libraries) Python

Library diperlukan untuk mempermudah penggunaan Python untuk kebutuhan yang lebih rumit. Dalam hal Data Science, library yang sering dan banyak dipakai adalah sebagai berikut

a. NumPy

Untuk komputasi numerik, memungkinkan manipulasi array dan matriks.

b. Pandas

Untuk analisis dan manipulasi data, memudahkan pengolahan data dalam bentuk tabel.

c. Matplotlib

Untuk visualisasi data, menghasilkan grafik dan plot.

d. Scikit Learn

Scikit Learn menyediakan berbagai algoritma pembelajaran mesin dan alat-alat untuk tugas-tugas seperti klasifikasi, regresi, klastering, pengurangan dimensi, pemrosesan data, dan evaluasi model

7. Pengembangan Aplikasi dengan Python

a. Pengembangan Web

Pengembangan web dengan Python melibatkan penggunaan bahasa pemrograman Python dalam kerangka kerja seperti Django atau Flask untuk membuat aplikasi web. Framework-framework ini menyederhanakan proses pengembangan dengan menyediakan alat-alat dan struktur yang memungkinkan pengembang membangun situs web dan aplikasi web dengan cepat. Python digunakan untuk mengelola logika aplikasi, berinteraksi dengan basis data, dan mengelola permintaan HTTP. Selain itu, pengembang juga menggunakan teknologi web seperti HTML, CSS, dan JavaScript untuk mengatur tampilan dan interaksi dengan pengguna. Pengembangan web dengan Python memungkinkan pembuatan aplikasi web yang kuat dan skalabel dengan efisiensi.

b. Data Science

Data science dalam pengembangan aplikasi Python adalah penggunaan berbagai algoritma, teknik, dan pustaka khusus seperti NumPy, pandas, scikit-learn, dan TensorFlow untuk mengumpulkan, menganalisis, dan menginterpretasikan data. Ini memungkinkan pengembang untuk membuat aplikasi yang dapat memproses dan menghasilkan wawasan dari data, termasuk pemodelan prediktif, analisis statistik, visualisasi data, dan pemahaman mendalam tentang perilaku pengguna. Dengan Python, aplikasi dapat dikembangkan untuk mengotomatisasi tugas-tugas analitik, membuat rekomendasi, dan mengambil keputusan berdasarkan data, menjadikannya alat yang kuat dalam berbagai domain seperti bisnis, ilmu sosial, ilmu kesehatan, dan banyak lagi.

c. Aplikasi Desktop

Aplikasi desktop dalam pengembangan aplikasi Python adalah program komputer yang dirancang untuk dijalankan di lingkungan desktop atau lokal pada komputer pengguna. Python menyediakan berbagai alat dan pustaka, seperti Tkinter dan PyQt, yang memungkinkan pengembang untuk membuat antarmuka pengguna grafis (GUI) yang intuitif dan interaktif untuk aplikasi desktop. Dengan Python, pengembang dapat membuat aplikasi seperti editor teks, perangkat lunak pengolahan data, aplikasi bisnis, permainan sederhana, atau alat administratif yang dapat dijalankan secara mandiri di komputer pengguna tanpa perlu koneksi internet.

▼ Pengenalan Notebook Python

Cell (sel)

Notebook berisi dari beberapa sel. Sel berisi teks penjelasan atau kode yang dapat dieksekusi dan keluarannya. Klik sel untuk memilihnya.

Code cells (sel kode)

Di bawah ini adalah **code cell**. Selama di toolbar tertulis CONNECTED atau Kernelnya sudah terkoneksi, klik salah satu sel dan sel dapat dijalankan melalui beberapa metode:

- Klik **Tombol Play** di pojok kiri atas dari masing-masing sel;
- Tekan **Cmd/Ctrl+Enter** untuk menjalankan sel terseleksi;
- Tekan **Shift+Enter** untuk menjalankan sel terseleksi dan ubah fokus ke sel sebelumnya (akan menambah 1 sel jika tidak ada); atau
- Tekan **Alt+Enter** untuk menjalankan sel terseleksi dan menambah sel baru tepat dibawahnya.

Ada beberapa opsi dalam menjalankan beberapa atau semua sel di menu **Runtime** di atas.

```
1 a = 10
2 a

10
```

Text cells (sel text)

Ini adalah **text cell**. Anda dapat melakukan **klik 2x** untuk mengedit sel ini. Sel text menggunakan sintaks markdown. Untuk mengetahui lebih jelasnya dapat dilihat di halaman [Markdown Guide](#).

Anda juga dapat menambahkan rumus matematika bahkan rumus yang kompleks menggunakan [LaTeX](#) yang akan dirender oleh [MathJax](#). Cukup gunakan statement dengan pasangan tanda berikut $\$$. Contohnya $\sqrt{3x-1}+(1+x)^2$ akan menjadi

$$\sqrt{3x-1} + (1+x)^2.$$

Menambahkan dan Memindahkan Sel

Anda dapat menambahkan sel baru dengan menggunakan tombol **+ KODE** dan **+ TEKS** yang muncul saat Anda mengarahkan kursor di antara sel-sel. Tombol-tombol ini juga terdapat di toolbar di atas notebook, di mana Anda dapat menggunakannya untuk menambahkan sel di bawah sel yang saat ini dipilih.

Anda dapat memindahkan sel dengan memilihnya dan mengklik **Sel Atas** atau **Sel Bawah** di toolbar atas. Anda juga dapat menarik dan memindahkan teks dengan menggunakan cara **drag & drop**

▼ Bekerja dengan Python

Berikut adalah beberapa contoh fungsi kenyamanan yang disediakan.

Proses Python yang berjalan dalam waktu lama dapat diinterupsi. Jalankan sel berikut dan **pilih Runtime -> Interrupt execution** untuk menghentikan eksekusi.

```
1 import time
2 print("Sleeping")
3 time.sleep(30) # sleep for a while; interrupt me!
4 print("Done Sleeping")

Sleeping
-----
KeyboardInterrupt                                Traceback (most recent call last)
<ipython-input-7-fb6d98dc021a> in <cell line: 3>()
      1 import time
      2 print("Sleeping")
----> 3 time.sleep(30) # sleep for a while; interrupt me!
      4 print("Done Sleeping")

KeyboardInterrupt:
```

TELUSURI STACK OVERFLOW

▼ Sel 'Sihir' Notebook

Notebook merupakan turunan lebih lanjut dari IPython, sedangkan IPython sendiri memiliki sistem perintah yang kami sebut 'sihir' yang menyediakan bahasa *command-line* mini yang efektif dan bersifat independen terhadap sintaks Python, serta dapat diperluas oleh pengguna dengan perintah-perintah baru. Sihir dimaksudkan untuk diketik secara interaktif, sehingga mereka menggunakan pola *command-line*, seperti menggunakan spasi untuk memisahkan argumen, tanda hubung untuk pilihan, dan konvensi lain yang khas dari bawaan *command-line*.

Sihir terdiri dari dua jenis:

Sihir baris: perintah yang diawali dengan satu karakter % dan argumennya hanya mencakup hingga akhir baris saat ini. Sihir sel: menggunakan dua karakter persen sebagai penanda (%%), dan mereka menerima sebagai argumen baik baris saat ini di mana mereka dinyatakan maupun seluruh isi sel. Perlu diperhatikan bahwa sihir sel hanya dapat digunakan sebagai baris pertama dalam sel, dan sebagai prinsip umum, mereka tidak dapat 'ditumpuk' (yaitu, Anda hanya dapat menggunakan satu sihir sel per sel). Sihir %lsmagic digunakan untuk menampilkan semua sihir yang tersedia, dan akan menampilkan baik sihir baris maupun sihir sel yang saat ini ditentukan:

```
1 %lsmagic

Available line magics:
%alias %alias_magic %autoawait %autocall %automagic %autosave %bookmark %cat %cd %clear %colors %conda %config %connect_info
%killbgscripts %l_dir %less %lf %lk %ll %load %load_ext %loadpy %logoff %logon %logstart %logstate %logstop %ls %lsmagic %l
%pdef %pdot %pfile %pinfo %pinfo2 %pip %popd %pprint %precision %prun %psearch %psource %pushd %pwd %pycat %pylab %qtconso
%rmdir %run %save %sc %set_env %shell %store %sx %system %tb %tensorflow_version %time %timeit %unalias %unload_ext %who %

Available cell magics:
%%! %%HTML %%SVG %%bash %%bigquery %%capture %%debug %%file %%html %%javascript %%js %%latex %%markdown %%perl %%prun %%pyp
%%time %%timeit %%writefile

Automagic is ON, % prefix IS NOT needed for line magics.
```

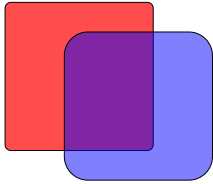
Salah satu contoh magic yang sangat menarik adalah kode HTML yang dapat langsung ditampilkan disini. Berikut contohnya:

```
1 %%html
2 <marquee style='width: 30%; color: blue;'><b>Halo!</b></marquee>
```

```

1 %%html
2 <svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 450 400" width="200" height="200">
3   <rect x="80" y="60" width="250" height="250" rx="10" style="fill:red; stroke:black; fill-opacity:0.7" />
4   <rect x="180" y="110" width="250" height="250" rx="40" style="fill:blue; stroke:black; fill-opacity:0.5;" />
5 </svg>

```



▼ *Output* yang kaya dan interaktif

Selama ini semua *output* yang dihasilkan telah berupa teks, tetapi *output* ini juga dapat menjadi lebih menarik, seperti grafik di bawah ini. Anda bahkan dapat berinteraksi dengan grafik seperti memperbesar, mengubah range, melihat nilai ketika melakukan *hover*, bahkan memilih grafik yang ditampilkan.

```

1 import numpy as np
2 import pandas as pd
3 import plotly.express as px
4
5 Data = np.random.normal(loc=0, scale=0.5, size=[100, 3])
6 Data = pd.DataFrame(Data, columns=['Sumbu X', 'Sumbu Y', 'Sumbu Z'])
7
8 fig = px.scatter_3d(Data, x='Sumbu X', y='Sumbu Y', z='Sumbu Z', color='Sumbu Z')
9 fig.update_layout(height=600)
10 fig.show()

```

