

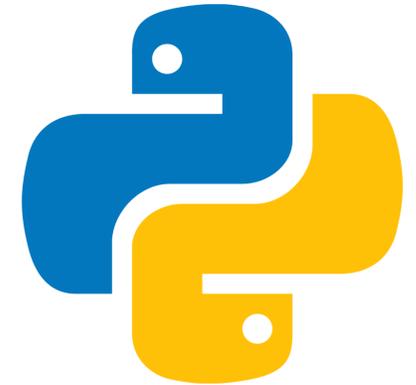
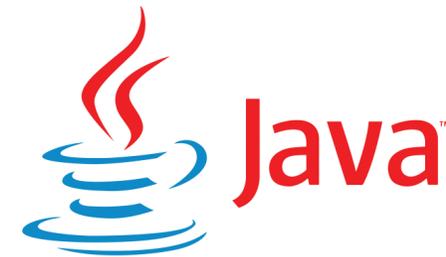
Eksplorasi Bahasa Pemrograman: Jenis, Perbedaan, dan Pilihan Terbaik

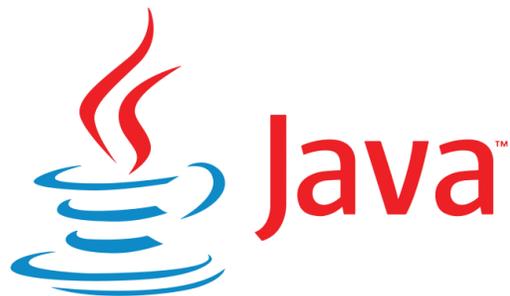
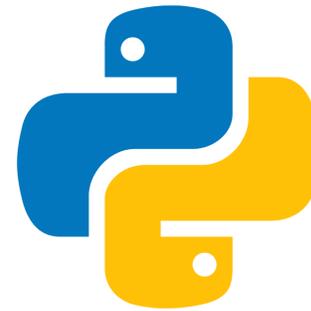
Hartono, S.Pd., M.T.I

Universitas Muhammadiyah Kotabumi

Bahasa Pemrograman

- Bahasa pemrograman adalah alat yang digunakan oleh pengembang perangkat lunak untuk berkomunikasi dengan komputer dan menginstruksikannya untuk melakukan tugas-tugas tertentu.
- Terdapat berbagai jenis bahasa pemrograman, masing-masing memiliki karakteristik dan kegunaan yang berbeda.
- Jenis-jenis bahasa pemrograman dapat dikelompokkan berdasarkan berbagai kriteria, seperti tingkat abstraksi, paradigma pemrograman, dan tujuan penggunaan.



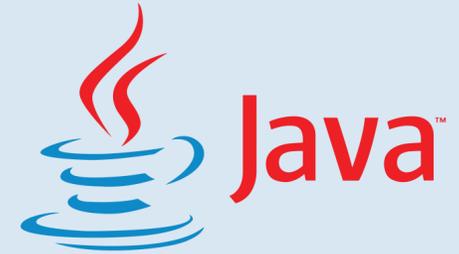
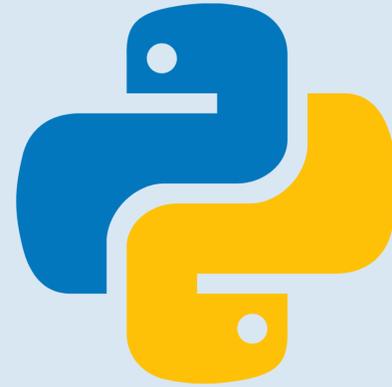


Tiga Tingkat (Level) Bahasa Pemrograman

- Bahasa Tingkat Rendah (*Low-Level Language*)
- Bahasa Tingkat Menengah (*Assembly Language*)
- Bahasa Tingkat Tinggi (*High-Level Language*)

Bahasa Tingkat Rendah (Low-Level Language)

- Bahasa mesin adalah bahasa pemrograman yang paling dekat dengan bahasa yang dimengerti oleh komputer itu sendiri.
- Bahasa ini terdiri dari serangkaian instruksi biner (0 dan 1) yang langsung dieksekusi oleh perangkat keras.
- Bahasa mesin sangat sulit dibaca dan ditulis oleh manusia karena kurangnya abstraksi dan struktur. Setiap jenis prosesor komputer memiliki bahasa mesin yang unik.
- Eksekusi bahasa mesin atau tingkat rendah sangat cepat, namun Bahasa ini sulit dipelajari oleh manusia.

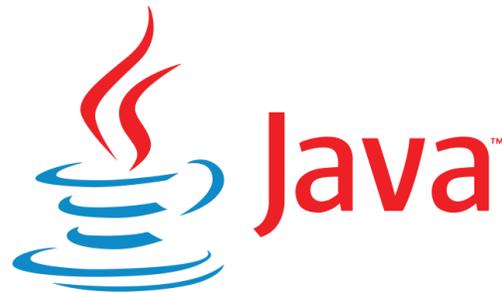
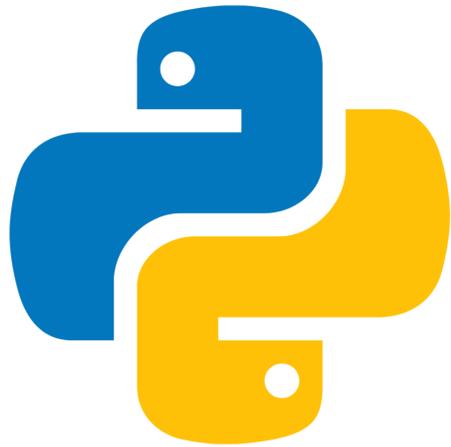


**Bahasa
Tingkat
Rendah
(Low-Level
Language)**

Binary Code

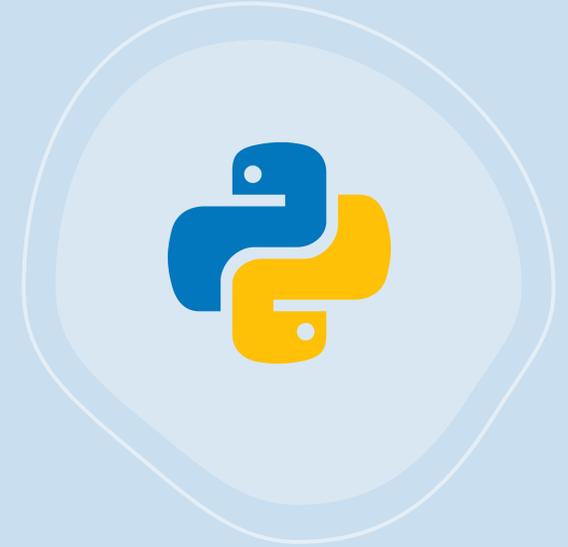
A	100 0001	H	100 1000	O	100 1111	V	101 0110
B	100 0010	I	100 1001	P	101 0000	W	101 0111
C	100 0011	J	100 1010	Q	101 0001	X	101 1000
D	100 0100	K	100 1011	R	101 1010	Y	101 1001
E	100 0101	L	100 1100	S	101 0011	Z	101 1010
F	100 0110	M	100 1101	T	101 0100	a	110 0001
G	100 0111	N	100 1110	U	101 0101	b	110 0010





Bahasa Tingkat Menengah (Assembly Language)

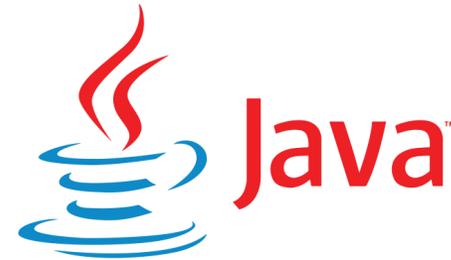
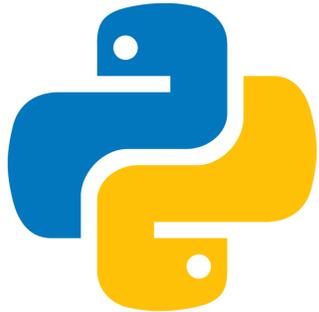
- Bahasa rakitan (assembly language) adalah langkah lebih tinggi dari bahasa mesin.
- Bahasa ini menggunakan simbol-simbol atau mnemonik untuk mewakili instruksi-instruksi tertentu dalam bahasa mesin.
- Meskipun lebih mudah dibaca daripada bahasa mesin, bahasa rakitan masih berhubungan erat dengan arsitektur perangkat keras dan biasanya spesifik untuk satu jenis prosesor.
- Bahasa rakitan memungkinkan programmer untuk bekerja dengan representasi yang lebih manusiawi daripada kode biner langsung.



Contoh Bahasa Tingkat Menengah (Assembly Language)

- x86 Assembly Language (Intel/AMD)
- ARM Assembly Language
- MIPS Assembly Language

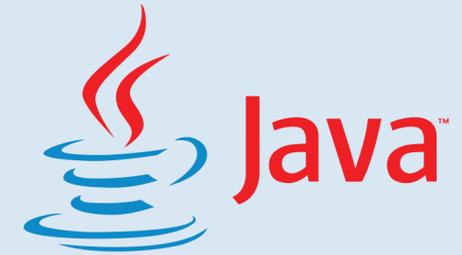
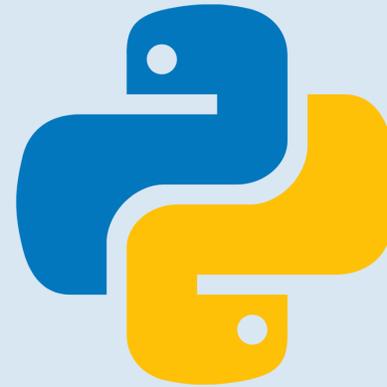
Bahasa Tingkat Tinggi (High Level Language)

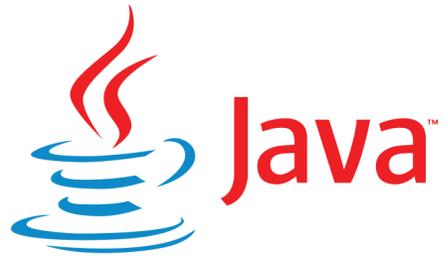
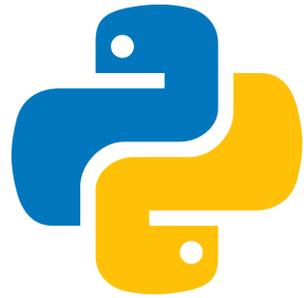


- Bahasa tingkat tinggi (high-level language) adalah jenis bahasa pemrograman yang dirancang untuk memudahkan programmer dalam mengekspresikan logika pemrograman dan algoritma dengan cara yang lebih manusiawi.
- Bahasa ini memiliki tingkat abstraksi yang lebih tinggi daripada bahasa rakitan atau bahasa mesin, sehingga memungkinkan programmer untuk lebih fokus pada konsep-konsep pemrograman daripada detail teknis yang kompleks.

Ciri-Ciri Bahasa Tingkat Tinggi

1. Sintaks yang Mudah Dibaca: Bahasa tingkat tinggi menggunakan sintaks yang lebih dekat dengan bahasa manusia, sehingga lebih mudah dipahami dan dibaca oleh programmer. Ini membuat kode menjadi lebih jelas dan intuitif.
2. Portabilitas: Kode yang ditulis dalam bahasa tingkat tinggi lebih portabel, artinya kode tersebut dapat dijalankan di berbagai platform dan sistem operasi tanpa perlu modifikasi besar.
3. Abstraksi: Bahasa tingkat tinggi menawarkan abstraksi yang lebih tinggi daripada bahasa rakitan atau bahasa mesin. Programmer dapat menggunakan konsep seperti fungsi, kelas, dan objek untuk memodelkan dunia nyata atau mengorganisir kode dengan lebih baik.

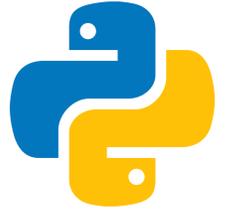
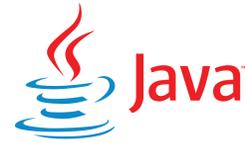




Ciri-Ciri Bahasa Tingkat Tinggi

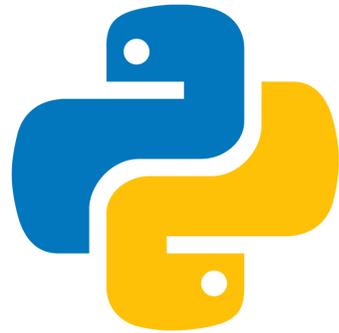
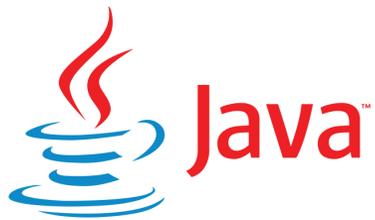
4. Dukungan untuk Struktur Kontrol: Bahasa tingkat tinggi mendukung struktur kontrol seperti pernyataan kondisional (if-else), perulangan (loop), dan pemanggilan fungsi. Ini memungkinkan pengembangan algoritma yang lebih kompleks.
5. Banyak Pustaka dan Modul: Bahasa tingkat tinggi umumnya dilengkapi dengan banyak pustaka dan modul yang memudahkan programmer dalam mengakses fungsionalitas yang sudah ada tanpa perlu mengimplementasikan semuanya dari awal.

Jenis-Jenis Bahasa Pemrograman Tingkat Tinggi



1. Bahasa Pemrograman Prosedural
2. Bahasa Pemrograman Berorientasi Objek (OOP - Object-Oriented Programming)
3. Bahasa Pemrograman Fungsional
4. Bahasa Pemrograman Skrip (Scripting Language)
5. Bahasa Pemrograman Fungsional dan Objek (FP & OOP)
6. Bahasa Pemrograman Logika

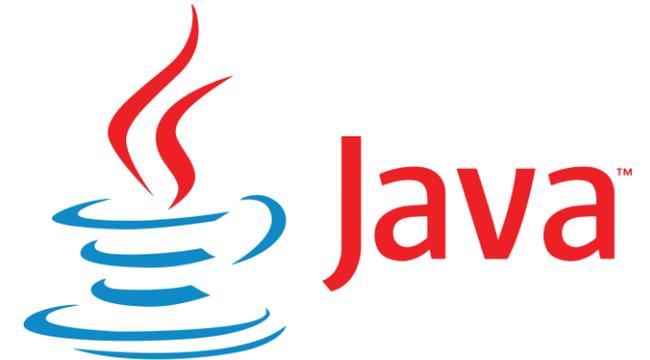
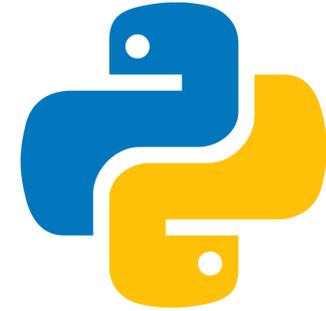
Bahasa Pemrograman Prosedural

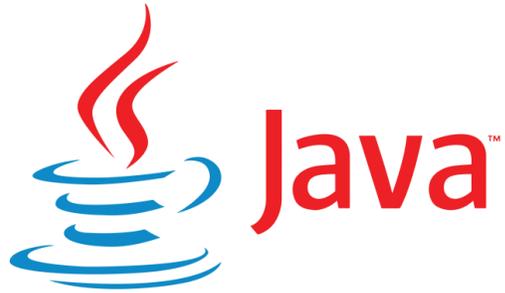
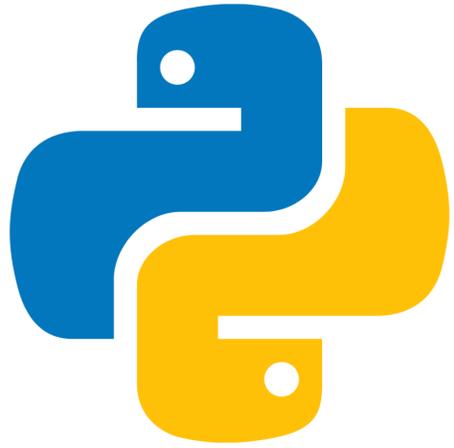


- Bahasa Pemrograman Prosedural adalah jenis bahasa pemrograman yang berfokus pada pemrograman berbasis prosedur atau fungsi.
- Dalam paradigma pemrograman ini, program diorganisir menjadi serangkaian prosedur atau blok kode yang dapat dipanggil untuk menjalankan tugas tertentu.
- Bahasa pemrograman prosedural menekankan pemecahan masalah dengan membagi tugas menjadi langkah-langkah logis yang dapat dieksekusi secara berurutan.

Ciri-Ciri Utama Bahasa Pemrograman Prosedural

1. **Prosedur atau Fungsi:** Kode dibagi menjadi blok-blok prosedur atau fungsi. Setiap prosedur adalah kumpulan instruksi yang dilakukan dalam urutan tertentu untuk mencapai tujuan tertentu.
2. **Struktur Kontrol:** Bahasa pemrograman prosedural mendukung struktur kontrol seperti pernyataan kondisional (if-else), perulangan (loop), dan pengulangan berdasarkan iterasi.
3. **Variabel dan Pengaturan Data:** Variabel digunakan untuk menyimpan data, dan tipe data serta variabel dapat didefinisikan dalam bahasa pemrograman prosedural.
4. **Modularitas:** Program dipecah menjadi modul-modul yang dapat dikelola dan dimengerti secara terpisah. Ini membantu dalam pengembangan tim dan pemeliharaan kode.
5. **Reusabilitas:** Kode yang ditulis dalam bahasa pemrograman prosedural cenderung lebih mudah untuk digunakan kembali, karena prosedur atau fungsi dapat dipanggil dari berbagai bagian program.
6. **Penggunaan Prosedur Berulang-ulang:** Pemrograman prosedural memungkinkan prosedur yang sama digunakan berulang kali dalam berbagai bagian program.



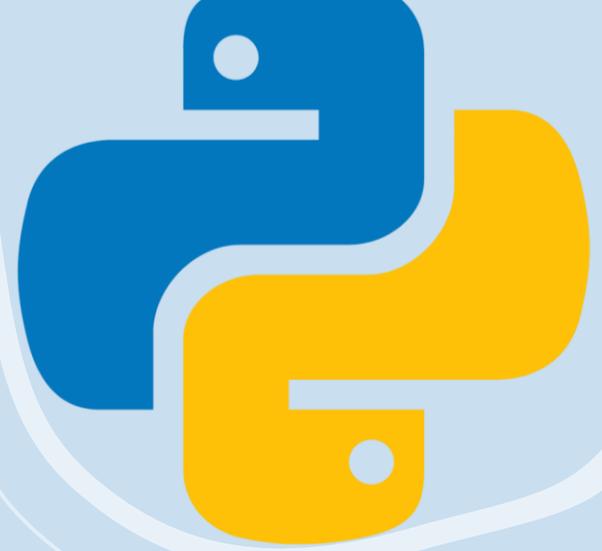


Contoh Bahasa Pemrograman Prosedural

1. C
2. Pascal
3. Fortran
4. COBOL

Bahasa C

- C adalah salah satu bahasa pemrograman yang sangat populer dan kuat. Dikembangkan pada awal tahun 1970-an, C memiliki sintaksis yang relatif sederhana dan efisiensi dalam pemrosesan data. C sering digunakan dalam pengembangan sistem operasi, perangkat lunak berkinerja tinggi, dan pemrograman perangkat terbatas.
- C memiliki dukungan yang kuat untuk operasi pada level bit, sehingga cocok untuk pengembangan perangkat keras atau aplikasi yang memerlukan manipulasi bit.
- Sintaksis C terdiri dari banyak simbol dan operator yang kuat, membuatnya lebih mendekati tingkat mesin.
- C memiliki dukungan untuk alokasi memori dinamis, yang memungkinkan pengguna mengelola alokasi dan dealokasi memori secara manual.
- Contoh penggunaan C termasuk pengembangan sistem operasi, perangkat lunak tertanam, perangkat lunak berkinerja tinggi, dan pengembangan game.

The PHP logo, featuring the letters 'php' in a bold, black, lowercase font with a white outline, set against a purple oval background.

php

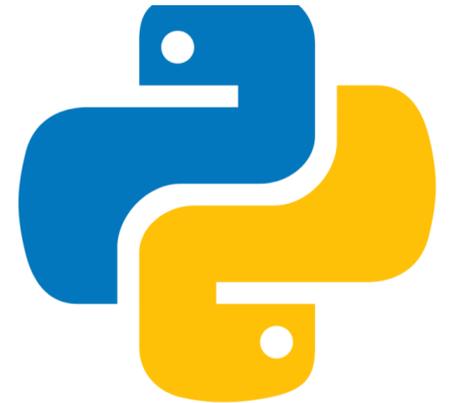
Bahasa Pascal

- Pascal adalah bahasa pemrograman yang dirancang untuk pengajaran dan pembelajaran. Dikembangkan pada tahun 1970-an oleh Niklaus Wirth, Pascal memiliki sintaksis yang jelas dan sederhana, sehingga sangat cocok untuk pemula.
- Pascal sangat populer di kalangan pendidikan karena sintaksisnya yang mudah dipahami dan pendekatan yang jelas terhadap konsep pemrograman.
- Pascal umumnya digunakan untuk pengembangan aplikasi yang lebih sederhana dan pendidikan.
- Bahasa Pascal telah berkembang menjadi variasi lain seperti Free Pascal dan Lazarus, yang mendukung fitur-fitur modern.



Bahasa Fotran

- Fortran (Formula Translation) adalah salah satu bahasa pemrograman tertua dan telah melalui beberapa iterasi. Awalnya dikembangkan untuk perhitungan ilmiah dan teknik, Fortran adalah bahasa pemrograman yang kuat untuk komputasi numerik.
- Fortran terkenal karena kecepatan dan efisiensinya dalam perhitungan numerik dan ilmiah.
- Fortran memiliki dukungan yang baik untuk array dan operasi matriks, menjadikannya pilihan yang bagus untuk pengolahan data ilmiah dan teknis.
- Meskipun Fortran umumnya digunakan dalam konteks komputasi ilmiah, beberapa varian terbaru (seperti Fortran 90 dan seterusnya) telah meningkatkan dukungan untuk pemrograman umum.

The PHP logo is displayed in a stylized, lowercase font within a dark blue oval. The oval is centered within a larger, light blue circular graphic element that has a thick, layered border.



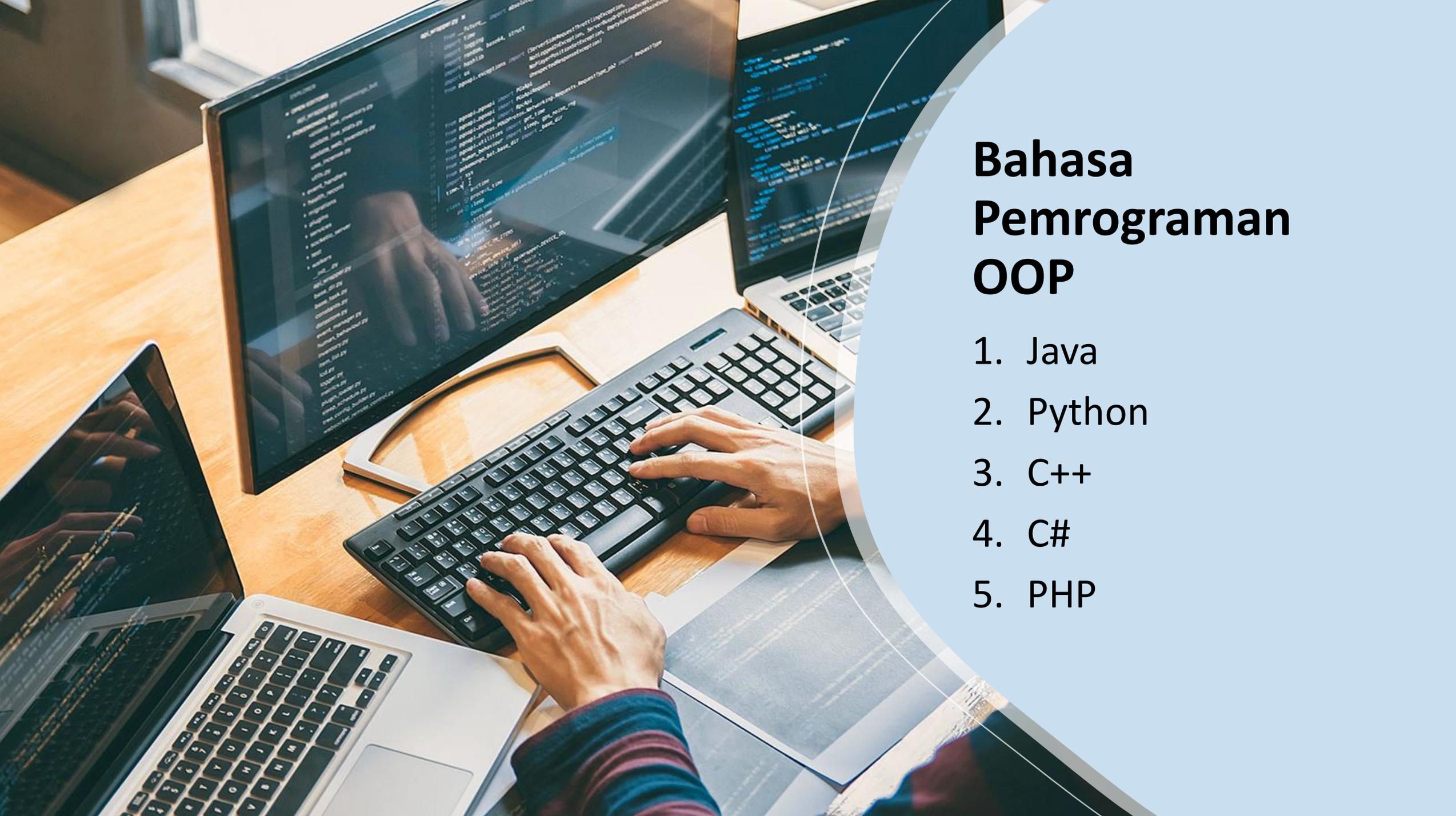
Bahasa COBOL

- COBOL (Common Business-Oriented Language) adalah bahasa pemrograman yang dirancang untuk pemrograman bisnis dan aplikasi perusahaan. COBOL dirancang untuk mengelola data besar dan kompleks.
- COBOL umumnya digunakan dalam lingkungan bisnis dan aplikasi perusahaan, terutama di sektor keuangan dan administrasi.
- Bahasa ini memiliki dukungan yang kuat untuk pemrosesan file besar dan kompleks serta pengelolaan data berstruktur.
- COBOL umumnya digunakan untuk aplikasi yang berfokus pada pemrosesan transaksi dan administrasi bisnis.

Bahasa Pemrograman Berorientasi Objek (OOP - Object-Oriented Programming)

- Bahasa Pemrograman Berorientasi Objek (OOP - Object-Oriented Programming) adalah paradigma pemrograman yang berfokus pada konsep "objek".
- Konsep ini memungkinkan pemodelan dunia nyata ke dalam kode program, di mana data dan fungsi terkait digabungkan dalam entitas yang disebut objek.
- OOP memungkinkan pengorganisasian kode yang lebih terstruktur, memfasilitasi pengembangan perangkat lunak yang lebih mudah dipahami, diubah, dan diperluas.





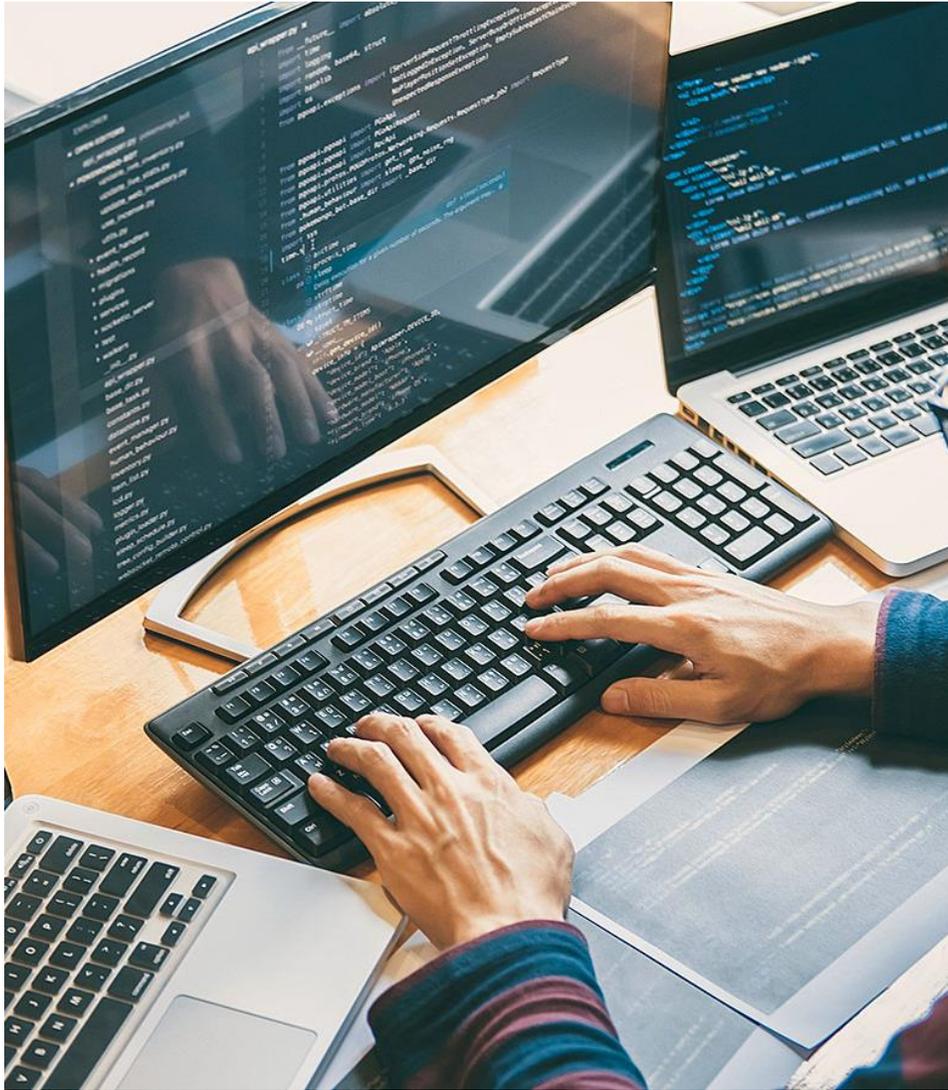
Bahasa Pemrograman OOP

1. Java
2. Python
3. C++
4. C#
5. PHP

Java

- **Kelebihan:** Java memiliki portabilitas yang tinggi, artinya kode yang ditulis dalam Java dapat dijalankan di berbagai platform tanpa modifikasi. Java juga memiliki dukungan yang kuat untuk pengembangan aplikasi yang besar dan kompleks. Lingkungan yang aman dan sistem manajemen memori yang otomatis membuatnya cocok untuk aplikasi yang lebih aman dan handal.
- **Kelemahan:** Meskipun Java cukup efisien, performa dalam beberapa kasus bisa lebih lambat dibandingkan bahasa yang lebih dekat dengan mesin. Beberapa pengembang juga merasa sintaksis Java terkadang terlalu verbose (panjang).
- **Platform:** Java Virtual Machine (JVM), menjadikannya portabel dan dapat dijalankan di berbagai sistem operasi.





Python

- Kelebihan: Python memiliki sintaksis yang bersahabat, membuatnya mudah dipelajari dan mudah digunakan. Python memiliki beragam pustaka dan modul yang mempercepat pengembangan. Python juga sangat cocok untuk pengembangan prototipe dan aplikasi berorientasi data.
- Kelemahan: Python bisa lebih lambat dalam hal performa dibandingkan bahasa yang dikompilasi. Ini juga tidak sesuai untuk aplikasi yang sangat berkinerja tinggi seperti game atau aplikasi yang memerlukan optimasi yang tinggi.
- Platform: Python dapat dijalankan di berbagai platform, termasuk Windows, macOS, dan Linux.

C++

- **Kelebihan:** C++ menggabungkan fitur-fitur C dengan paradigma OOP, sehingga memungkinkan pengembangan aplikasi yang efisien dan berorientasi objek. C++ dapat digunakan untuk pengembangan perangkat lunak yang berkinerja tinggi, seperti game atau aplikasi yang membutuhkan optimasi.
- **Kelemahan:** Sintaksis C++ bisa kompleks dan lebih rumit dibandingkan bahasa pemrograman lainnya. Memahami dan mengelola memory management juga bisa menjadi tugas yang kompleks.
- **Platform:** C++ adalah bahasa yang bersifat lintas platform, dan kode C++ dapat dijalankan di berbagai sistem operasi.



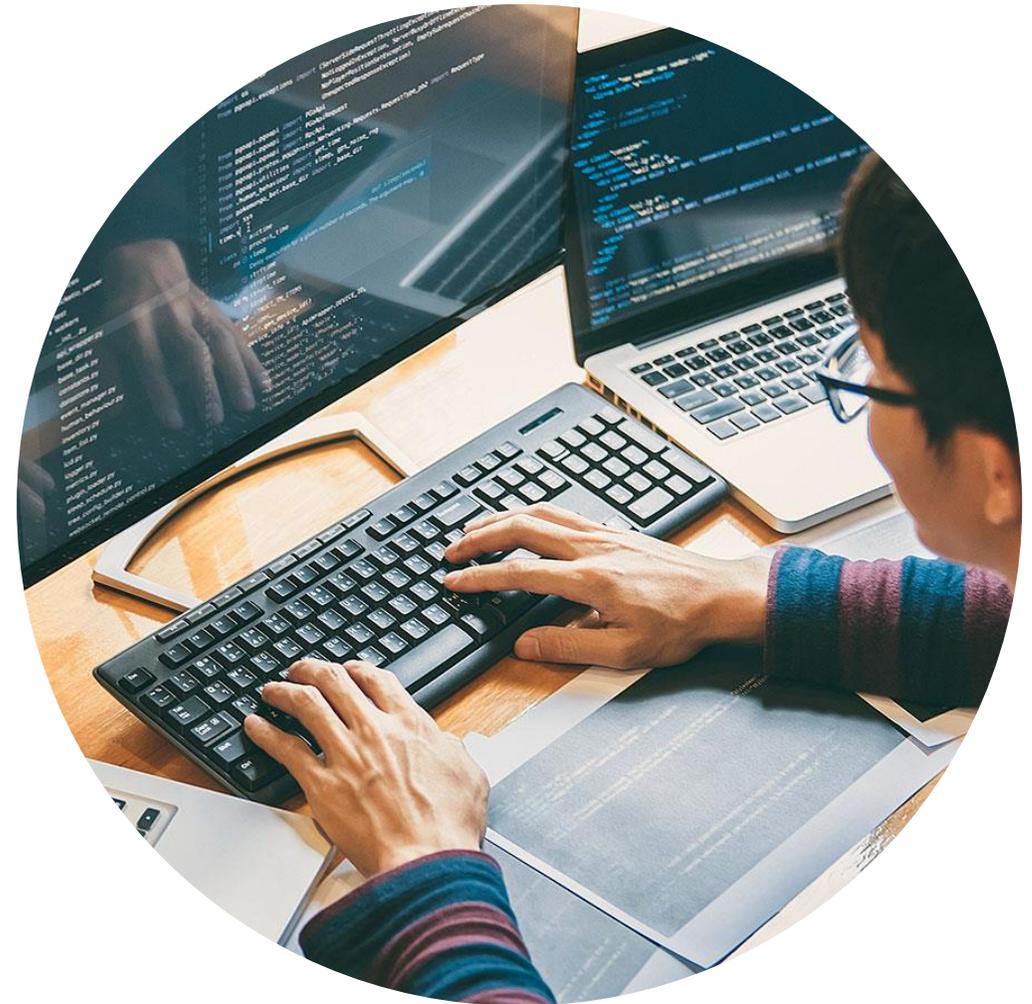
PHP

- **Kelebihan:**

- PHP adalah bahasa yang banyak digunakan untuk pengembangan web, khususnya pada sisi server. Itu memiliki dukungan yang kuat untuk interaksi dengan database dan pemrosesan formulir.
- Cocok untuk pengembangan situs web dinamis, berbasis database, dan aplikasi web berorientasi data.
- PHP memiliki banyak pustaka dan kerangka kerja (framework) yang dapat membantu dalam pengembangan aplikasi web yang cepat dan efisien.

- **Kelemahan:**

- PHP umumnya lebih lambat dalam hal performa dibandingkan dengan bahasa pemrograman lain seperti C++ atau Java.
- Sintaksis PHP terkadang dianggap tidak konsisten dan kurang rapi dibandingkan dengan bahasa lain.



C#

- **Kelebihan:** C# (C Sharp) adalah bahasa yang dirancang oleh Microsoft dan banyak digunakan untuk pengembangan aplikasi Windows dan aplikasi berbasis .NET. C# memiliki integrasi yang baik dengan platform Microsoft dan menggabungkan fitur-fitur dari C++ dan Java.
- **Kelemahan:** Ketergantungan pada platform Microsoft bisa menjadi kendala jika ingin menjalankan aplikasi di luar lingkungan Windows. Pengembangan aplikasi berbasis C# juga lebih terkait dengan ekosistem .NET.
- **Platform:** C# umumnya digunakan di lingkungan Windows dengan dukungan .NET Framework atau .NET Core.





Javascript

▪ Kelebihan:

- JavaScript adalah bahasa yang digunakan terutama untuk pengembangan web di sisi klien (front-end). Ini memungkinkan pembuatan antarmuka interaktif dan dinamis di dalam browser.
- JavaScript sangat penting untuk pengembangan aplikasi web modern, termasuk single-page applications (SPA).
- Dengan framework seperti React, Angular, dan Vue.js, JavaScript dapat digunakan dalam pengembangan aplikasi berorientasi objek yang lebih kompleks.

▪ Kelemahan:

- JavaScript terutama digunakan dalam pengembangan sisi klien, yang berarti kode dieksekusi di browser pengguna. Ini dapat mempengaruhi performa, terutama pada perangkat dengan sumber daya terbatas.
- Meskipun JavaScript dapat digunakan untuk pengembangan di sisi server melalui teknologi seperti Node.js, tidak sepopuler bahasa server-side seperti PHP.

Bahasa Pemrograman Populer

Aug 2023	Aug 2022	Change	Programming Language	Ratings	Change
1	1		 Python	13.33%	-2.30%
2	2		 C	11.41%	-3.35%
3	4	▲	 C++	10.63%	+0.49%
4	3	▼	 Java	10.33%	-2.14%
5	5		 C#	7.04%	+1.64%
6	8	▲	 JavaScript	3.29%	+0.89%
7	6	▼	 Visual Basic	2.63%	-2.26%
8	9	▲	 SQL	1.53%	-0.14%
9	7	▼	 Assembly language	1.34%	-1.41%
10	10		 PHP	1.27%	-0.09%
11	21	▲▲	 Scratch	1.22%	+0.63%
12	15	▲	 Go	1.16%	+0.20%
13	17	▲▲	 MATLAB	1.05%	+0.17%
14	18	▲▲	 Fortran	1.03%	+0.24%
15	31	▲▲	 COBOL	0.96%	+0.59%
16	16		 R	0.92%	+0.01%
17	19	▲	 Ruby	0.91%	+0.18%
18	11	▼▼	 Swift	0.90%	-0.35%
19	22	▲	 Rust	0.89%	+0.32%
20	28	▲▲	 Julia	0.85%	+0.41%

Bahasa Pemrograman Populer

Programming Language		Ratings
	Python	13.33%
	C	11.41%
	C++	10.63%
	Java	10.33%
	C#	7.04%

Kelebihan Python

1. Sintaksis yang Mudah Dibaca dan Dipahami
2. Dukungan untuk Berbagai Bidang
3. Banyak Pustaka dan Modul
4. Portabilitas dan Keterbacaan Kode
5. Pengembangan Cepat
6. Komunitas yang Aktif
7. Pemrosesan Data dan Ilmu Data
8. Kecerdasan Buatan dan Pembelajaran Mesin
9. Dokumentasi yang Kaya dan Mudah Dibaca





Instalasi Python di Windows

- 1. Unduh Installer:**

Kunjungi situs resmi Python di <https://www.python.org/downloads/> dan pilih versi terbaru dari Python yang sesuai dengan sistem operasi Anda (Windows, macOS, atau Linux).
- 2. Pilih Opsi Instalasi:** Saat mengunduh, Anda mungkin akan memiliki beberapa pilihan instalasi. Jika Anda pemula, pilih versi installer yang mencakup Python Interpreter dan pip (Python Package Installer).

Instalasi Python di Windows

1. **Windows:** Buka berkas installer yang telah diunduh dan ikuti panduan wizard instalasi. Pastikan Anda memeriksa opsi "Add Python X.Y to PATH" agar Python dapat diakses melalui command prompt.
2. **macOS:** Buka berkas installer yang telah diunduh dan ikuti panduan instalasi. Biasanya, Python sudah terpasang di macOS secara default, tetapi menginstal versi terbaru juga merupakan pilihan.
3. **Linux:** Buka terminal dan gunakan perintah `sudo apt-get install python3` (pada distribusi berbasis Debian/Ubuntu) atau `sudo yum install python3` (pada distribusi berbasis Red Hat/Fedora).





Python IDLE

1. Python IDLE adalah lingkungan pengembangan interaktif bawaan yang disediakan bersamaan dengan instalasi Python. Ini memungkinkan Anda untuk menulis, menjalankan, dan menguji kode Python secara interaktif.
2. Pada Windows, Anda dapat mencarinya dengan mengetik "IDLE" di menu Start atau melalui folder instalasi Python. Pada macOS dan Linux, Anda dapat membuka terminal dan mengetik `idle` atau `python3 -m idle`.
3. Setelah Python IDLE terbuka, Anda akan melihat jendela Shell yang interaktif. Anda dapat mulai menulis kode Python langsung di Idle.
4. Anda bisa menulis kode langsung di Shell dan menekan Enter untuk menjalankannya. Juga, Anda dapat membuat dan menjalankan file Python (.py) melalui menu "File" > "New File". Setelah menulis kode, simpan file dan jalankan dengan mengklik "Run" > "Run Module" atau menggunakan tombol pintas `Ctrl+F5` (pada Windows) atau `Command+Shift+Enter` (pada macOS).

Command Prompt (CMD)

1. **Buka CMD:** Pada Windows, Anda bisa mencari "CMD" di menu Start dan membukanya. Jika Anda menggunakan macOS atau Linux, Anda bisa membuka terminal.
2. **Navigasi ke Direktori Kode:** Menggunakan perintah `cd`, navigasilah ke direktori di mana file Python Anda disimpan. Misalnya, jika file Anda ada di "C:\Users\NamaUser\Documents", Anda akan mengetik `cd Documents` setelah membuka CMD.
3. **Jalankan Kode:** Setelah Anda berada di direktori yang benar, ketik `python namafile.py` (ganti "namafile.py" dengan nama file Anda) dan tekan Enter. Ini akan menjalankan kode Python yang ada dalam file tersebut.





ts

cene.objects]

```
[].obj".format(item.name))
path, use_selection=True,
f.axis_forward_setting,
is_up_setting,
self.use_animation_setting,
ers=self.use_mesh_modifiers_setting,
use_edges_setting,
s=self.use_smooth_groups_setting,
s_bitflags=self.use_smooth_groups_bitflags_setting,
use_normals_setting,
s_setting,
self.use_materials_setting,
```

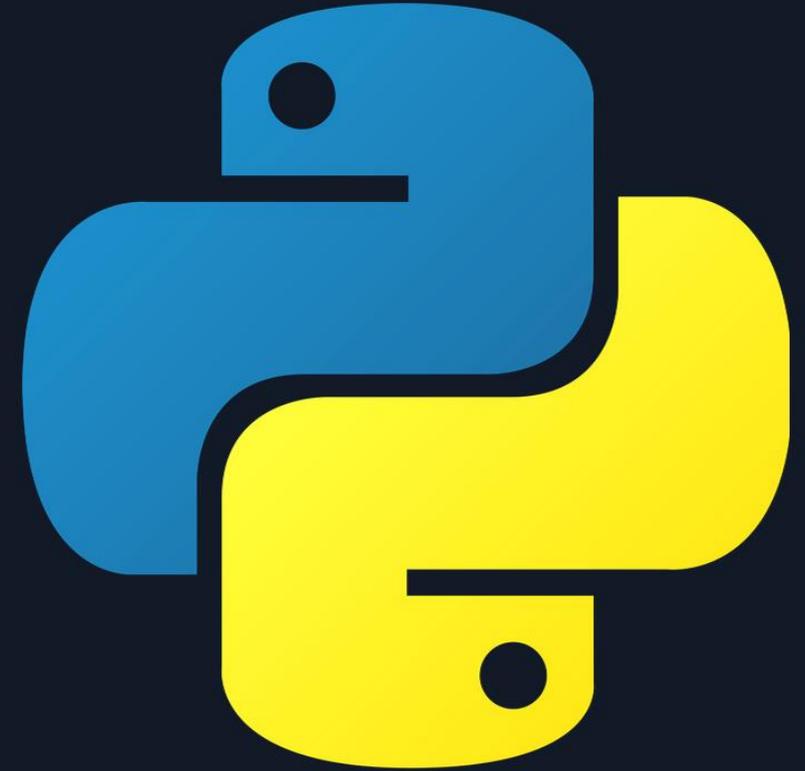
Latihan

- 1) **Soal 1: Python IDLE:** Tuliskan program sederhana menggunakan Python IDLE yang mencetak pesan "Halo, dunia!" ke layar.
- 2) **Soal 2: Menjalankan File Python melalui CMD**
Anda memiliki file "hitung.py" yang berisi fungsi untuk menghitung jumlah dari dua bilangan. Bagaimana Anda menjalankan file tersebut melalui Command Prompt?
- 3) **Soal 3: Python IDLE dan Kalkulator** Gunakan Python IDLE untuk membuat kalkulator sederhana. Buat program yang meminta pengguna memasukkan dua bilangan dan operasi matematika (+, -, *, /), lalu tampilkan hasilnya.

Latihan

- 4) **Soal 4: Menggunakan Python IDLE dan CMD**
Anda telah menulis program sederhana dalam Python dan menyimpannya sebagai "program.py". Bagaimana Anda menjalankan program ini melalui Python IDLE dan CMD?

- 5) **Soal 5: Menggunakan CMD untuk Menjalankan Program** Anda ingin menjalankan skrip Python yang berada dalam direktori "C:\Users\NamaUser\Documents". Bagaimana Anda akan menggunakan Command Prompt untuk menjalankan skrip tersebut?



```
ne.objects]
```

```
.obj".format(item.name))  
th, use_selection=True,  
.axis_forward_setting,  
_up_setting,  
f.use_animation_setting,  
s=self.use_mesh_modifiers_setting,  
e_edges_setting,  
=self.use_smooth_groups_setting,  
_bitflags=self.use_smooth_groups_bitflags_setting,  
use_normals_setting,  
vs_setting,  
f.use_materials_setting,
```