

# 1. Pengenalan Python

## Apa itu Python? 🐍

Halo, Sobat Codey! 🙌 Sudah pernah dengar tentang Python? Python itu ibarat sahabat pertama dalam dunia coding — gampang diajak kenalan, dan semakin lama kamu bareng Python, semakin seru petualangan yang bisa kamu jelajahi! 📈 Python adalah bahasa pemrograman yang super kece, simpel, dan terkenal di kalangan pemula maupun para pro di dunia IT. Dengan Python, kamu bisa bikin website, main-main sama data, bahkan bikin game keren. Pokoknya, Python ini all-rounder banget. Seru, kan? 😄

## Sejarah dan Kegunaan Python 📄

Bayangin aja, Python ini diciptakan sama seorang jenius bernama Guido van Rossum di tahun 1991! Dari situ, Python terus berkembang dan sekarang dipakai sama perusahaan-perusahaan raksasa kayak Google, Instagram, dan Spotify. Kenapa? Karena Python itu powerful banget dan bisa dipakai di berbagai bidang. Jadi, dengan belajar Python, kamu enggak cuma belajar bahasa pemrograman biasa; kamu juga membuka jalan ke berbagai peluang keren di masa depan! 🚀

## Kenapa Harus Belajar Python? 🏆

Penasaran kenapa Python selalu dapet spotlight di dunia coding? Ini nih beberapa alasannya:

- **Super Mudah Dipelajari:** Python itu kayak bahasa sehari-hari. Sintaksnya simpel, jadi gak bakal bikin kamu pusing tujuh keliling!
  - **Fleksibel Banget:** Kamu mau bikin aplikasi web, main-main sama data, atau bikin proyek machine learning? Python bisa bantu semuanya!
  - **Komunitas yang Solid:** Di balik Python, ada komunitas besar yang siap sedia buat bantu kamu. Lagi stuck? Tinggal tanya di forum atau grup Python, pasti banyak yang bantu.
  - **Pustaka Melimpah:** Python punya banyak pustaka alias library yang bisa bikin kerjaan kamu lebih cepat dan efektif. Misalnya, NumPy buat matematika, Pandas buat analisis data, dan masih banyak lagi!
- 

## 2. Instalasi Python 🚀

### Cara Instal Python di Windows, macOS, dan Linux

Sudah penasaran pengen mulai coding? Yuk, kita install Python dulu!

#### 1. Windows

1. Buka situs [python.org](https://python.org) dan download installer Python untuk Windows.
2. Saat menginstal, jangan lupa centang “Add Python to PATH” biar nanti kamu bisa akses Python dari Command Prompt tanpa repot. 🎉❤️

#### 2. macOS

1. Buka **Terminal** (search aja di Spotlight, pasti ketemu).
2. Kalau kamu belum punya Homebrew, install dulu dengan perintah ini:

```
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

3. Setelah Homebrew terpasang, ketik brew **install python** buat langsung install Python. Tinggal tunggu dan siap deh!

### 3. Linux

1. Buka **Terminal** (Sobat Linux pasti udah akrab sama yang satu ini, ya!).
2. Masukkan perintah di bawah ini sesuai dengan distro kamu:
  - **Debian/Ubuntu:** `sudo apt-get install python3`
  - **Fedora:** `sudo yum install python3`

Nah, sekarang Python sudah terpasang dan kamu tinggal mulai petualangan kodingmu, Sobat Codey!



---

---

## 3. Halo World Pertama Kamu! ✨

Nah, Sobat Codey, di dunia coding, biasanya kita mulai dari satu hal kecil yang bikin hati berbunga-bunga: ngeprint tulisan “Hello, World!”. Ini semacam tradisi buat kenalan sama bahasa baru. Yuk, kita coba bareng-bareng!

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

Cukup tulis perintah di atas di editor kamu atau di **Python Interactive Shell**. Terus tekan Enter. Boom! Hello, World! Sudah tampil. Kamu baru aja bikin program pertamamu. Mantap! 🙌

---

---

## 4. Tipe Data di Python

Di Python, ada berbagai tipe data yang bisa kita pakai. Yuk, kita kenalan satu-satu!

- **Integer (int):** Buat angka bulat kayak 7, 42, atau 2024.
- **Float:** Buat angka desimal kayak 3.14 atau 0.99.
- **String (str):** Buat teks, kayak "Hallo, Sobat Codey!".
- **Boolean (bool):** Cuma ada dua nilai: True atau False.

Python ini fleksibel banget, jadi kamu enggak perlu mendeklarasikan tipe data secara eksplisit. Python bakal otomatis ngerti dari data yang kamu masukkan. Enak, kan?

```
>>> umur = 20
>>> nama = "Codey"
>>> isStudent = True
>>>
>>> print(umur)
20
>>> print(nama)
Codey
>>> print(isStudent)
True
>>>
>>>
```

Coba bayangin, Sobat Codey, di dalam baris kode di atas kita sudah bikin variabel umur, nama, dan status mahasiswa dengan tipe data yang berbeda-beda. Simple tapi powerful!

---

## 5. Struktur Kontrol: Percabangan dan Perulangan

### Percabangan If-Else: Biar Programmu Makin Pintar

Oke, Sobat Codey, sekarang kita masuk ke percabangan. Jadi, kalau kamu pengen bikin program yang bisa mengambil keputusan sendiri, kamu perlu pake yang namanya if-else. Gampangnya, kalau if itu ibarat “kalau...”, dan else itu “kalau enggak...”. Yuk, langsung cobain!

```
>>> umur = 17
>>>
>>> if umur >= 18:
...     print("Selamat, kamu sudah dewasa!")
... else:
...     print("Kamu masih di bawah umur, nih!")
...
Kamu masih di bawah umur, nih!
>>>
```

Kalau kamu pengen menambah kondisi lain, bisa pakai elif (kependekan dari else if). Ini buat nambahin pilihan yang lebih detail:

```
>>> nilai = 85
>>>
>>> if nilai >= 90:
...     print("Kamu dapat nilai A, keren!")
... elif nilai >= 80:
...     print("Kamu dapat nilai B, tetap semangat!")
... else:
...     print("Ayo belajar lebih giat lagi!")
...
Kamu dapat nilai B, tetap semangat!
>>>
```

Wohooo, kamu keren banget sihh 🙌

**Perulangan for dan while: Looping sampai puas!** 

Nah, di sini kita mulai main sama **looping**. Perulangan bikin kita bisa menjalankan kode berkali-kali tanpa harus ngetik berulang-ulang. Ada dua perulangan utama: for dan while.

### Contoh perulangan for

```
>>> for i in range(1, 6):  
...     print("Angka ke-", i)  
...  
Angka ke- 1  
Angka ke- 2  
Angka ke- 3  
Angka ke- 4  
Angka ke- 5  
>>> █
```

Looping ini bakal mencetak angka dari 1 sampai 5, satu per satu. Cuma butuh beberapa baris buat melakukan tugas yang berulang-ulang! Seru kan?

### Perulangan while

while itu cocok buat kondisi yang belum pasti batasnya. Jadi, perulangan terus jalan selama kondisi tertentu masih True.

```
>>> count = 1  
>>> while count <= 5:  
...     print("Hitungan ke-", count)  
...     count += 1  
...  
Hitungan ke- 1  
Hitungan ke- 2  
Hitungan ke- 3  
Hitungan ke- 4  
Hitungan ke- 5  
>>> █
```

Loop ini bakal terus berjalan selama count kurang atau sama dengan 5. Praktis banget kalau lagi butuh perulangan yang dinamis.

Yeahh, keren banget kan, *while loop* kaya cintaku padamu— tak pernah berhenti, terus berulang tanpa batas.

---

## 6. Fungsi: Bikin Program Jadi Lebih Rapi

Fungsi itu kayak kotak ajaib yang bisa kamu panggil kapanpun kamu butuh. Ini bikin programmu rapi dan gampang dimodifikasi. Yuk kita coba buat satu fungsi sederhana!

```
>>> def sapa(nama):
...     print("Halo,", nama, "! Selamat datang di Python.")
...
>>> sapa("Sobat Codey")
Halo, Sobat Codey ! Selamat datang di Python.
>>>
```

Di sini, sapa adalah nama fungsi kita, dan nama itu parameternya. Jadi, setiap kali kamu panggil sapa("Sobat Codey"), Python bakal nge-print “Halo, Sobat Codey! Selamat datang di Python.”

## Fungsi dengan Nilai Balik (Return)

Kamu bisa bikin fungsi yang ngembaliin nilai, biar bisa dipakai buat perhitungan atau data lain. Contohnya:

```
>>> def luas_persegi(sisi):
...     return sisi * sisi
...
>>> print("Luas persegi adalah", luas_persegi(4))
Luas persegi adalah 16
>>>
```

Nah, return ini ngebalikin hasil perhitungan yang bisa kamu pakai di luar fungsi. Jadi, fungsi kita sekarang makin powerful!

## 7. Pemrograman Berorientasi Objek (Object-Oriented Programming)

Nah, Python juga punya fitur keren buat bikin kode jadi lebih terstruktur, namanya **Pemrograman Berorientasi Objek** atau Object-Oriented Programming (OOP). Ini bikin kita bisa bikin objek dengan atribut dan perilakunya sendiri. Bayangin kayak bikin blueprint buat karakter atau objek di game!

```
class Kucing:
    def __init__(self, nama, umur):
        self.nama = nama
        self.umur = umur

    def meong(self):
        print(f"{self.nama} berumur {self.umur} tahun, meong!")

kucing1 = Kucing("Luna", 2)
kucing1.meong()

# output Luna berumur 2 tahun, meong!
```

Dengan `class`, kita bisa mendefinisikan bagaimana objek kita berperilaku dan apa saja atribut yang dimiliki. Dalam contoh ini, kita bisa melihat bahwa setiap objek Kucing dapat memiliki nama yang berbeda dan dapat meong() dengan suara unik mereka. Keren, kan? 😊

Jadi, ingatlah bahwa dengan memahami konsep kelas dan objek, kamu bisa mulai membangun program-program yang lebih kompleks dan menarik! Teruslah berlatih dan eksplorasi, Sobat Codey! Sampai jumpa di petualangan coding berikutnya! 🚀 💻

---

Nah, Sobat Codey, itu dia petualangan dasar di Python! Mulai dari print sederhana sampai bikin objek sendiri, sekarang kamu punya bekal buat bikin banyak hal seru. Semangat terus dan selalu eksplor kemampuanmu di Python! 🚀