

A stylized, hand-drawn logo for 'ASPRO-20' in a bold, orange-yellow font with white outlines. The letters are thick and blocky, with a slightly irregular, artistic feel. The '20' is smaller and positioned below the 'PRO'.

MODUL 1

Fundamental Python

Tim Penyusun

NIM	NAMA	KODE ASISTEN
1202180127	ABDURRAHMAN AZIZ	ABZ
1202184189	AGRIVA DETTA GINTING	GIP
1202171049	AHMAD SHOHIBUS SULTHONI	TON
1202184143	ARDDHANA ZHAFRAN AMANULLAH	ZAF
1202184085	ARDY RIKARDO	RDY
1202184347	DANIEL ALEXANDER POLII	DNI
1202184135	DEWA MADE SURYA PERMANA MASTRA	DEM
1202184072	EKKY CHANDRA WIBOWO	EKY
1202183302	FAUZI ARIFIN ALGHIFARI	OZI
1202184126	GHUFRON FIKRIANTO	GHF
1202180229	IFEN FARIDIAN RAHMADAN	FEN
1202184159	ILMA NUR HIDAYATI	ILM
1202180092	JODY MARDIKA	JDM
1202184138	M. FAIZ TRIPUTRA	XAX
1202184310	MUHAMMAD DIFAGAMA IVANKA	PAA
1202184077	NANDA ARFAN HAKIM	NAN
1202174288	NUR AZIZAH HARUN	CHA
1202170250	RAHADIAN ALDI NUGROHO	ALD
1202184117	RAJA NANDA SATRYA	JAX
1202184062	RISKI ANANDA WIDIYA PUTRI	RAA
1202183322	RIZALRASYD DWISLIA RIDWANAH	ZAY
1202184019	SHAHNAZ KAMILAH	SAZ
1202184209	TITISARI RAMADHANE	TRD
1202184224	TODOAN JEHEZKIEL SANTANA MUARA S.	TDS
1202181065	ZAHWA ALIFAH AMMATULLAH	ZAW

Peraturan Praktikum

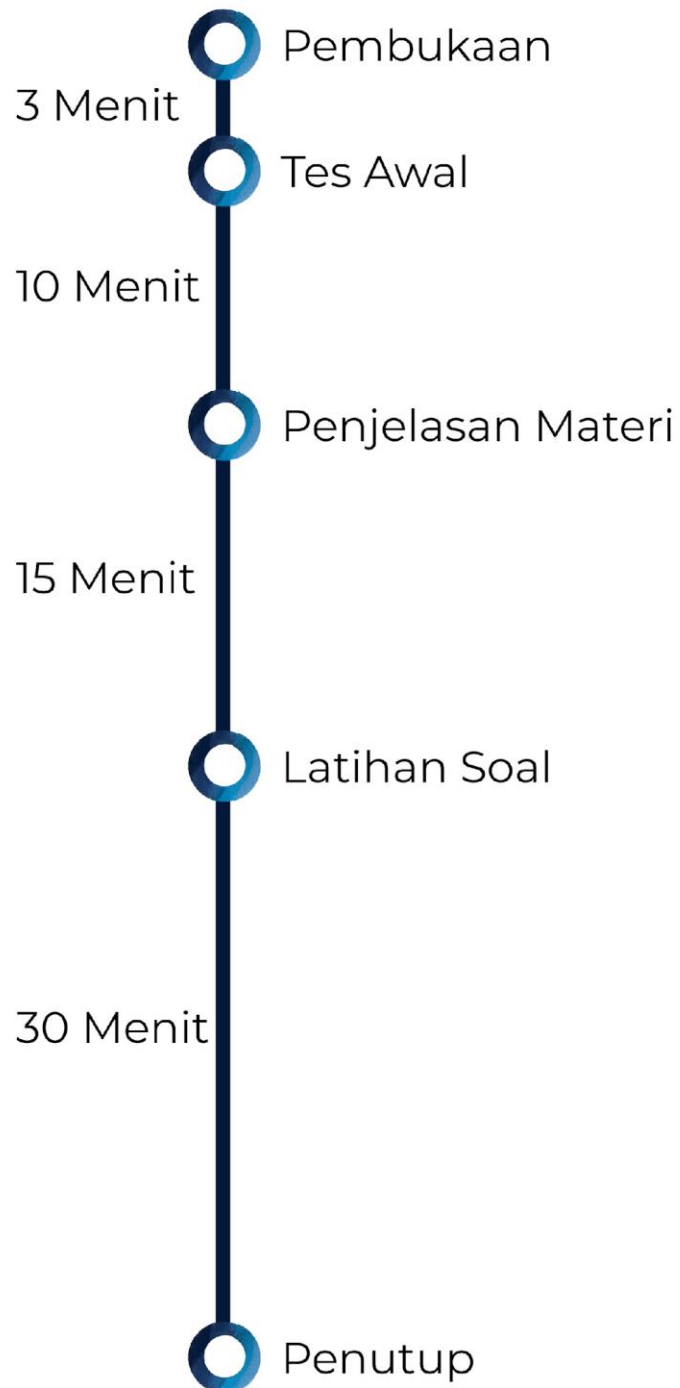
1. Tidak diperbolehkan menggunakan nilai praktikum tahun sebelumnya tanpa ada persetujuan koordinator dosen mata kuliah praktikum yang bersangkutan dengan kepala urusan laboratorium.
2. Jika praktikan mengajukan izin (sakit dan anggota keluarga meninggal), maka surat perizinan diberikan ke pihak Laboratorium maksimal 6 hari setelah jadwal praktikum masing-masing.
3. Izin lomba atau penugasan institusi tidak berlaku, kecuali sudah terdapat dispensasi di Igracias. NB: Dilampirkan screenshot dispensasi igracias.

4. Seragam praktikum

- a. Bagi mahasiswa tidak diperkenankan menggunakan celana berbahan jeans/ chino (tidak stretch) saat praktikum.
- b. Bagi mahasiswi diwajibkan menggunakan rok panjang tidak ketat saat praktikum
- c. Jika ditemukan praktikan menggunakan hal-hal terkait, maka diperkenankan untuk mengganti dengan dresscode yang sesuai dengan peraturan Universitas Telkom dan tidak ada penambahan waktu.

NB: Untuk dresscode hari Senin dapat menggunakan kemeja merah telkom atau kemeja putih polos, dresscode hari Selasa sampai Rabu menggunakan kemeja putih. Untuk dresscode hari Kamis s/d Sabtu praktikan tetap diwajibkan menggunakan kemeja formal berkancing depan dari atas sampai bawah dan berkerah (bukan kerah shanghai dan bukan polo) dan celana atau rok bahan berwarna hitam/biru gelap.

Rundown Praktikum



Daftar Isi

Tim Penyusun	1
Peraturan Praktikum	2
Rundown Praktikum	3
Daftar Isi	4
1. Syntax Dasar	5
1.1 Print	5
1.2 Variabel	5
1.3 User Input	7
2. Aturan Syntax Dasar	8
2.1 Penulisan Statement	8
2.2 Penulisan String	8
2.3 Case Sensitive	9
2.4 Komentar	10
3. Tipe Data	11
3.1 String	11
3.2 Bilangan(Angka)	14
3.3 Boolean	15
4. Operator	16
4.1 Operator Aritmatika	16
4.2 Operator Perbandingan	17
4.3 Operator penugasan	18
4.4 Operator logika	19
4.5 Operator keanggotaan	20
4.6 Operator identitas	21
4.7 Operator bitwise	22
Referensi	23

1. Syntax Dasar

Beberapa syntax dasar dalam python adalah sebagai berikut:

1.1 Print

Fungsi print untuk mencetak atau menampilkan teks atau objek ke terminal dalam bentuk teks.

Contoh:

```
print("ALPRO MUDAH")
```

Output:

```
ALPRO MUDAH
```

1.2 Variabel

Variabel merupakan tempat menyimpan data. Variabel bersifat *mutable*, artinya nilainya bisa berubah-ubah. Variabel di python dapat dibuat dengan format ini.

a. Membuat Variabel

Variabel di python dibuat dengan format berikut:

```
nama_variabel = <nilai>
```

Contoh:

```
ini_modul = "Algoritma Pemrograman"  
print(ini_modul)
```

Output :

```
Algoritma Pemrograman
```

b. 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-sensitive*). Artinya huruf besar dan kecil dibedakan. Contoh: `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.
- 5) Nama variabel tidak boleh menggunakan angka. Contoh: `123`
- 6) Jenis penamaan variabel yang dikenal umum ada 3 yaitu Camel Case, Snake Case, dan All Caps
 - a. Camel Case merupakan gaya penulisan yang memiliki ciri-ciri setiap huruf depan pada nama variabel menggunakan huruf kapital.
Contoh : `sakatonikAbc`, `HappyCoding`
 - b. Snake Case dicirikan dengan penulisannya yang menggunakan garis bawah/underscore (`_`).
Contoh : `raja_ganteng`, `_jumlahnilai`
 - c. All Caps merupakan gaya penulisan yang memiliki ciri-ciri dimana semua huruf kapital. Biasanya digunakan sebagai penamaan konstanta.
Contoh : `PI`, `GRAVITY`, `URL_STRING`

c. Menghapus Variabel

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

Contoh:

```
judul = "Algoritma Pemrograman Modul 1"
print(judul)
del(judul)
print(judul)
```

Output

```
Algoritma Pemrograman Modul 1
Traceback (most recent call last):
  File "d:/belajar pyhton/modul.py", line 4, in <module>
    print(judul)
NameError: name 'judul' is not defined
```

Output di baris paling atas memunculkan isi variabel, tetapi setelah menggunakan fungsi `del()`, dan diprint kembali, maka mendapatkan **NameError**, yang artinya variabel sudah dihapus.

1.3 User Input

Input adalah masukan yang diberikan ke program, dan program akan memprosesnya.

Python sudah menyediakan *Syntax* **`input()`** untuk mengambil inputan dari keyboard.

```
nama_variabel = input("Masukkan Inputan : ")
```

Artinya, teks yang kita inputkan akan disimpan ke dalam **`nama_variable`**.

Contoh :

```
nama = input("Masukkan Nama : ")
print("Halo",nama,",kamu adalah calon programmer hebat!")
```

Maka akan menghasilkan :

```
Masukkan Nama : King
Halo King ,kamu adalah calon programmer hebat!
```

Dimana "King" merupakan inputan dari user, dan program akan mengeluarkan output dari inputan user tersebut.

2. Aturan Syntax Dasar

2.1 Penulisan Statement

Statement adalah sebuah intruksi atau kalimat perintah yang akan dieksekusi oleh komputer. Bila kita ingin menulis lebih dari satu statement dalam satu baris, maka harus memisahkannya dengan titik-koma. Tetapi tidak dianjurkan, karena akan sulit dibaca.

```
print("Algoritma Pemrograman"); print("mudah banget")
```

Dianjurkan

```
print("Algoritma Pemrograman")  
print("mudah banget")
```

2.2 Penulisan String

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

```
print("Algoritma Pemrograman 2020")  
print('menggunakan python')
```

Bisa juga menggunakan tanda petik tiga. Tanda kutip tiga digunakan untuk string multibaris.

Contoh

```
print('''
    Algoritma Pemrograman 2020
    menggunakan Python
''')
```

Output:

```
Algoritma Pemrograman 2020
menggunakan python
```

2.3 Case Sensitive

Syntax di python bersifat *case sensitive*, python menganggap huruf besar dan kecil memiliki perbedaan. Contoh, jika kita menulis `print()` maka akan berjalan. Jika kita mengetik `PRINT()` atau `Print()`, maka akan terjadi **error**.

Contoh:

```
judul = "Algoritma Pemrograman 2020"
Judul = "Modul 1"
print(Judul)
print(judul)
```

Output :

```
Modul 1
Algoritma Pemrograman 2020
```

2.4 Komentar

Komentar merupakan baris kode yang tidak akan dieksekusi. Komentar digunakan untuk menandai atau memberikan keterangan dan untuk menonaktifkan kode.

Ada beberapa cara penulisan komentar pada Python:

1. Tanda pagar(#)

Cara ini paling sering digunakan, karena mudah membedakannya dengan kode program.

2. Tanda Petik(' ') / (" ")

Penulisan komentar dengan tanda petik jarang digunakan, karena akan sulit membedakan komentar dengan kode program.

3. Triple Tanda Petik ("'' ''') / (""" """)

Penulisan komentar yang sama dengan tanda petik tunggal/ganda.

```
praktikum = "ALPRO 2020" # ini variabel
print(praktikum)

'ini komentar dalam satu baris'

"""
Happy Coding!!
-Use Your CODE for POSITIVE Things-
by DasproLaboraotry
"""
```

3. Tipe Data

Tipe data adalah jenis data yang tersimpan dalam variabel. Tipe data di python bersifat primitif, yaitu tipe data angka, teks dan boolean.

3.1 String

Seperti pada aturan syntax dasar, penulisan tipe data String harus diapit dengan tanda petik. Bisa menggunakan petik tunggal ('.....'), ganda ("....."), dan tiga ("'.....'" atau ""'.....'").

Contoh :

```
nama = 'Raja Nanda'

jeniskelamin = "Pria"

alamat = """
    Jl. Sukabirus C/55 RT XX
    Dayeuh Kolot, Bandung
    """

motohidup = '''
    HAPPY CODING!
    '''

print(nama)
print(jeniskelamin)
print(alamat)
print(motohidup)
print(type(nama))
print(type(jeniskelamin))
print(type(alamat))
print(type(motohidup))
```

maka ketika semua variable di print secara berurutan akan mengeluarkan

Output :

```
Raja Nanda
Pria

    Jl. Sukabirus C/55 RT XX
    Dayeuh Kolot, Bandung

    HAPPY CODING!

<class 'str'>
<class 'str'>
<class 'str'>
<class 'str'>
```


Mengakses String

Di dalam tipe data String kita dapat mengakses nilainya per karakter dengan bebas sesuai yang kita inginkan, dimana penghitungan karakter pertama dimulai dari 0. dan menggunakan tanda kurung siku [...] dan dibatasi dengan titik dua untuk mengambil karakternya.

```
kata = "happy coding!"
```

```
print(kata[0:4])
```

Contoh :

```
kata = "happy coding!"  
print(kata[0:4])
```

dimana huruf 'h' adalah indeks ke-0 dan 'y' merupakan indeks ke-4

maka outputnya akan seperti ini :

The image shows the output of the Python code, which is the string "happ". The characters are displayed in a light blue color on a dark background.

Penjelasan : mengapa yang keluar hanya kata "happ" ? itu dikarenakan program akan membaca dari indeks ke-0 dan karakter sebelum indeks ke-4 sehingga yang keluar hanya kata "happ".

Selain dapat mengakses nilai String kita juga dapat mempengaruhi besar kecilnya tulisan menggunakan beberapa *Syntax* khusus :

.title() = huruf awal setiap kata menjadi kapital.

.upper() = semua huruf kapital.

.lower() = semua huruf kecil.

Contoh :

```
kata = "happy coding!"

print("Kondisi awal kata : ")
print(kata)
print("")
print("Menggunakan title()")
print(kata.title())
print("")
print("Menggunakan upper()")
print(kata.upper())
print("")
word = "DASPRO"
print("Kondisi awal kata :")
print(word)
print("")
print("Menggunakan lower()")
print(word.lower())
```

maka outputnya akan seperti ini :

```
Kondisi awal kata :
happy coding!

Menggunakan title()
Happy Coding!

Menggunakan upper()
HAPPY CODING!

Kondisi awal kata :
DASPRO

Menggunakan lower()
daspro
```

3.2 Bilangan(Angka)

Tipe data angka dibagi menjadi dua jenis, integer dan Float. Untuk mengetahui tipe data dalam variabel, dapat menggunakan `type()`.

a) Integer(int)

Int adalah bilangan bulat. Tipe data int bisa digabungkan dengan teks.

Contoh

```
harga = 25000
print(type(harga))
```

Output:

```
<class 'int'>
```

b) Float

Angka desimal yang menggunakan titik(.) bukan koma(.). Float bisa diubah menjadi String atau Integer. Ketika Float diubah menjadi tipe data Integer, maka angka dibelakang titik akan dihilangkan.

Contoh

```
jarak = 8.5
print("Jarak : ", jarak)
print("Tipe data :", type(jarak))
print("Jarak di casting menjadi Integer :", int(jarak))
```

Ouput:

```
Jarak : 8.5
Tipe data : <class 'float'>
Jarak di casting menjadi Integer : 8
```

3.3 Boolean

Tipe data *boolean* adalah tipe data yang hanya memiliki dua nilai yaitu **True** atau **1** dan **False** atau **0**. Ketika mencetak angka 1 atau 0 akan mengeluarkan angka itu sendiri, bukan boolean. Untuk mengubah menjadi boolean, menggunakan **bool()**.

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

Contoh

```
bergerak = True
jalan = 1
print(bergerak)
print(jalan)
print(bool(jalan))
```

Output:

```
True
1
True
```

Ketika **print(jalan)** maka yg ditampilkan adalah angka. Untuk membuat angka menjadi boolean, harus menggunakan **bool()**. Karena, ketika print sebuah variabel yang berisi angka, maka masih dianggap integer.

4. Operator

Operator dalam Python merupakan sebuah simbol yang digunakan untuk melakukan sebuah operasi terhadap satu atau lebih operan. Operan yang dimaksud dapat berupa *variable* atau nilai yang digunakan untuk melakukan operasi.

4.1 Operator Aritmatika

Operator Aritmatika merupakan operator yang digunakan untuk dasar operasi matematika.

Operator Aritmatika	Nama	Contoh	Hasil
+	Penjumlahan	9 + 10	19
-	Pengurangan	7.5 – 4	3.5
*	Perkalian	8 * 10	80
/	Pembagian	90 / 12	7.5
%	Modulus atau sisa hasil bagi	7 % 3	1
**	Eksponensial	3 ** 2	9
//	Pembagian bulat	19 // 3	6

Contoh studi kasus:

Untuk memperingati ulang tahun ke-33 Matahari Mall, sedang diadakan diskon besar-besaran untuk beberapa *item* tertentu. Bantulah Yaya yang merupakan seorang kasir baru di Matahari Mall, dengan membuat program untuk menghitung total harga belanjaan seorang pembeli dengan masing-masing diskon senilai 60%:

- Kaos Greenlight = Rp250.000
- Sepatu Nike = Rp920.000

Solusi:

```
print('====PROGRAM KASIR MATAHARI MALL====')
kaos = float(input('Harga kaos = '))
sepatu = float(input('Harga sepatu = '))
total = (kaos-(kaos*0.6)) + (sepatu-(sepatu*0.6))
print('Total harga = Rp' + str(total))
```

Output:

```
====PROGRAM KASIR MATAHARI MALL====
Harga kaos = 250000
Harga sepatu = 920000
Total harga = Rp468000.0
```

4.2 Operator Perbandingan

Operator Perbandingan melakukan perbandingan terhadap dua operan, seperti halnya lebih dari, kurang dari, atau sama dengan.

Operat or	Nama	Contoh	Hasil
<	Kurang dari	0 < 8	True
>	Lebih dari	8 > 9	False
<=	Kurang dari atau sama dengan	7 <= 8	True
>=	Lebih dari atau sama dengan	9 >= 9	True
==	Sama dengan	0 == 0	True
!=	Tidak sama dengan	8 != 8	False

4.3 Operator penugasan

Operator penugasan memberikan nilai terhadap suatu *variable*. Pada tabel di bawah ini, misalkan $x = 9$.

Operator	Contoh	Pengertian	Hasil
=	$x = 9$	$x = 9$	9
+=	$x += 10$	$x = x + 10$	19
-=	$x -= 2$	$x = x - 2$	7
*=	$x *= 2$	$x = x * 2$	18
/=	$x /= 3$	$x = x / 3$	3
%=	$x \% = 3$	$x = x \% 2$	1
**=	$x ** = 2$	$x = x ** 2$	81
//=	$x //= 2$	$x = x // 2$	4

4.4 Operator logika

Operator logika merupakan konjungsi yang digunakan untuk menggabungkan suatu pernyataan bersyarat.

Operator	Deskripsi	Contoh
and	Mengembalikan nilai True apabila kedua pernyataan bernilai <i>true</i> .	$x < 6$ and $x < 8$
or	Mengembalikan nilai True apabila salah satu pernyataan bernilai <i>true</i> .	$x > 2$ or $x < 0$
not	Membalik hasil dan mengembalikan nilai False apabila hasilnya bernilai <i>true</i> .	not($x < 6$ and $x < 8$)

Contoh studi kasus:

Untuk mendapatkan indeks A, seorang mahasiswa harus bisa mendapatkan nilai lebih dari sama dengan 80 untuk mata kuliah Kalkulus dan Matvek. Buatlah program yang dapat mengetahui apakah mahasiswa dapat mendapatkan indeks A.

Solusi:

```
print('====PROGRAM INPUT NILAI====')
kalkulus = int(input('Nilai Kalkulus = '))
matdis = int(input('Nilai Matdis = '))
nilai = kalkulus >= 80 and matdis >= 80
print('Indeks nilai A : ' + str(nilai))
```

Output:

```
====PROGRAM INPUT NILAI====
Nilai Kalkulus = 80
Nilai Matdis = 77
Indeks nilai A : False
```


4.5 Operator keanggotaan

Operator keanggotaan digunakan untuk mengetes keanggotaan suatu nilai terhadap sebuah *sequence*.

Operator	Deskripsi	Contoh	Output
In	Mengembalikan nilai True apabila sebuah nilai terdapat pada <i>sequence</i>	<pre>nama = 'Ilma Nur Hidayati' print('nu' in nama) print('Nu' in nama)</pre>	False True
not in	Mengembalikan nilai True apabila sebuah nilai tidak terdapat pada <i>sequence</i>	<pre>nama = 'Ilma Nur Hidayati' print('hi' not in nama) print('Hi' not in nama)</pre>	True False

4.6 Operator identitas

Operator identitas digunakan untuk membandingkan suatu objek terhadap kesamaannya dan lokasi memorinya.

Operator	Deskripsi	Contoh	Output
is	Mengembalikan nilai True apabila kedua <i>variable</i> adalah objek yang sama.	<pre>x = 15 y = 10 z = 15 print(x is y) print(x is z)</pre>	False True
is not	Mengembalikan nilai True apabila kedua <i>variable</i> bukanlah objek yang sama.	<pre>x = 15 y = 10 z = 15 print(x is not y) print(x is not z) print(x is not x)</pre>	True False False

4.7 Operator bitwise

Operator bitwise merupakan operator yang beroperasi bit per bit dan digunakan untuk membandingkan bilangan biner. Contohnya, angka 2 ditulis 10 dalam notasi biner dan angka 7 ditulis 111. Pada tabel di bawah ini, misalkan **x = 10 (1010 dalam biner)** dan **y = 8 (1000 dalam biner)**.

Operat or Bitwise	Nama	Deskripsi	Contoh
&	AND	Bernilai 1 apabila kedua bit adalah 1	$x \& y = 8$ (1000)
	OR	Bernilai 1 apabila salah satu dari dua bit adalah 1	$x y = 10$ (1010)
^	XOR	Bernilai 1 apabila hanya satu dari dua bit yang merupakan 1	$x \wedge y = 2$ (0010)
~	NOT	Membalikkan nilai suatu bit, sehingga akan bernilai 1 apabila suatu bit adalah 0 dan sebaliknya.	$\sim x = 5$ (0101)
<<	Left shift	Menggeser ke kiri setiap bit dan menghilangkan bit paling kiri.	$x \ll 2 = 8$ (1000)
>>	Right shift	Menggeser ke kanan setiap bit dan menghilangkan bit paling kanan.	$x \gg 2 = 2$ (0010)



Referensi

<https://www.petanikode.com/python-sintaks/>

<https://data-flair.training/blogs/python-operator/>

https://www.w3schools.com/python/python_operators.asp

[PPT Alpro Dosen](#)

<https://www.petanikode.com/python-variabel-dan-tipe-data/>

<https://www.petanikode.com/python-input-output/>

<https://www.petanikode.com/case-kode-program/>