

**LAPORAN PRAKTIKUM ALGORITMA DAN PEMROGRAMAN
KOMPUTER
MODUL 2**

TIPE DATA DAN VARIABEL

Laporan ini disusun untuk memenuhi Tugas Mata Kuliah
Praktikum Basis Data



Disusun Oleh :

AHSAN MAULANA RIZQI

104062400071

S1 BD-04-B

**PROGRAM STUDI S1 BISNIS DIGITAL
TELKOM UNIVERSITY PURWOKERTO
2025**

1 Dasar Teori

1.1 Tipe Data

Tipe data adalah jenis data yang tersimpan dalam sebuah variabel. Berikut tipe data yang ada pada python:

Tipe Data	Penjelasan	
Number	int (Integer)	Bilangan bulat positif dan negative
	float	Bilangan pecahan atau desimal
	complex	Bilangan imajiner atau complex
String	string	Kumpulan karakter yang diapit oleh tanda petik satau atau dua
List	list	Kumpulan dari tipe data yang tersimpan dan sifat data tersebut dapat diubah
Tuple	tuple	Kumpulan dari tipe data yang tersimpan dan sifat data tersebut tidak dapat diubah
Dictionary	dict	Kumpulan dari tipe data yang tersimpan dan sifat data tersebut dapat diubah, didefinisikan menggunakan <i>key</i> dan <i>value</i>
Boolean	bool	Pernyataan <i>True</i> yang bernilai 1 atau <i>False</i> yang bernilai 0

1.2 Identifier

Identifier adalah pengenalan atau nama yang diberikan untuk mengidentifikasi sesuatu. Contoh identifier adalah variabel, fungsi, kelas dan lain sebagainya. Python memiliki aturan-aturan tertentu dalam penulisan identifier, aturan-aturan tersebut sebagai berikut:

- Identifier bersifat case sensitif artinya penulisan huruf besar dan huruf kecil dibedakan. Contoh penulisan variabel Nama dengan nama dianggap variabel yang berbeda.
- Karakter pertama pada identifier harus diawali huruf seperti (A-Z), (a-z) atau *underscore* (`_`). Contoh **nama_mahasiswa**, **Alamat_mahasiswa**.
- Identifier tidak boleh menggunakan karakter special atau simbol seperti (!, @, #, \$, %, dan lain sebagainya)
- Identifier tidak boleh menggunakan reserved keyword (kata kunci yang sudah dipakai) oleh python.
- Identifier tidak boleh dipisahkan menggunakan spasi contoh variabel nama mahasiswa.
- Pemberian nama identifier sebaiknya disesuaikan dengan kebutuhan atau nama identifier sebaiknya bisa menjelaskan maksudnya. Contoh variabel nama_mahasiswa menjelaskan variabel tersebut akan diisi dengan nilai nama mahasiswa.

1.3 Variabel

Variabel adalah wadah atau tempat yang digunakan untuk menampung suatu nilai. Python termasuk dynamically-typed language artinya kita tidak perlu mendefinisikan atau mendeklarasikan tipe data terlebih dahulu sebelum membuat variabel. Variabel bisa diisi dengan nilai apapun dan python secara otomatis akan mendeteksi tipe data dari variabel yang kita buat berdasarkan nilai yang diberikan.

2 Pembahasan Tugas Guided

2.1 Mengenal tipe data

Tulis kode program seperti pada gambar 2.1 menggunakan visual studio code dan jalankan programnya.

```
# variabel data
jenisTipeData = " ini adalah type data : "
dataList1 = 1,2,3,4
dataList2 = "satu","dua","tiga","empat"

# type data

# Integer
print (5,jenisTipeData,type(5))
# Float
print (3.14,jenisTipeData,type(3.14))
# Complex
print (1+2j,jenisTipeData,type(1+2j))

# List
print ([dataList1],jenisTipeData,type([dataList1]))
print ([dataList2],jenisTipeData,type([dataList2]))

# Tuple
print ((dataList1),jenisTipeData,type((dataList1)))
print ((dataList2),jenisTipeData,type((dataList2)))
# Dictionary
print ({'nama' : "budi","umur": 20},jenisTipeData,type({'nama' : "budi","umur": 20}))
```

Gambar 2.1 Jenis Tipe Data

Variabel data untuk menampung nilai yang digunakan secara berulang. Fungsi `type()` digunakan untuk mengetahui jenis dari tipe data yang digunakan. Contoh pada angka 5 yang merupakan tipe data number. Dengan menggunakan fungsi `type()` akan menghasilkan output `<class 'int'>` sebagai kelas atau objek dari tipe data integer.

Jika program python dijalankan akan menghasilkan output seperti pada gambar 2.2.

```

kamize@server:~/web/alpro$ /bin/python3 "/home/kamize/web/alpro/Week 2/guided1.py"
5 ini adalah type data : <class 'int'>
3.14 ini adalah type data : <class 'float'>
(1+2j) ini adalah type data : <class 'complex'>
[(1, 2, 3, 4)] ini adalah type data : <class 'list'>
[('satu', 'dua', 'tiga', 'empat')] ini adalah type data : <class 'list'>
(1, 2, 3, 4) ini adalah type data : <class 'tuple'>
('satu', 'dua', 'tiga', 'empat') ini adalah type data : <class 'tuple'>
{'nama': 'budi', 'umur': 20} ini adalah type data : <class 'dict'>

```

Gambar 2.2 Hasil Output Jenis Tipe Data

2.2 Membuat variable

Tulis kode program seperti pada gambar 2.3 menggunakan visual studio code dan jalankan programnya.

Variable Nama, Alamat, Pekerj44n, Kampus pada gambar 2.3 digunakan untuk menampung string. Fungsi print () digunakan untuk menghasilkan output seperti pada gambar 2.4.

```

Nama = "arif"
Alamat = "Purwokerto, Jawa Tengan"
Pekerj44n = "Mahasiswa"
Kampus = "TEL-U Purwokerto"

print (Nama)
print (Alamat)
print (Pekerj44n)
print (Kampus)

```

Gambar 2.3 Variabel

```

kamize@server:~/web/alpro$ /bin/python3 "/home/kamize/web/alpro/Week 2/guided2.py"
arif
Purwokerto, Jawa Tengan
Mahasiswa
TEL-U Purwokerto

```

Gambar 2.4 Hasil Output Variabel

Variable banyak_baris pada gambar 2.5 digunakan untuk menampung string dengan *multiline* menggunakan tanda petik tiga. Fungsi print () digunakan untuk menghasilkan output seperti pada gambar 2.6.

```
# multiple line

Banyak_baris = """
saya
berkuliah
di
TEL-U Puewokerto
"""

print(Banyak_baris)
```

Gambar 2.5 String Dengan Multiple-line

```
kamize@server:~/web/alpro$ /bin/python3 "/home/kamize/web/alpro/Week 2/guided3.py"

saya
berkuliah
di
TEL-U Puewokerto
```

Gambar 2.6 Hasil Output Multiple-line

2.3 Memahami indentasi pada python

Tulis kode program seperti pada gambar 2. menggunakan visual studio code dan jalankan programnya.

Indentasi merupakan bagian untuk memberitahu bahwa print ("makan", i+1) termasuk bagian yang akan dijalankan oleh *for loop* ditandai dengan titik dua dan spasi. Pada gambar 2.7 bagian atas akan menghasilkan output gambar 2.8 bagian atas dan seterusnya.

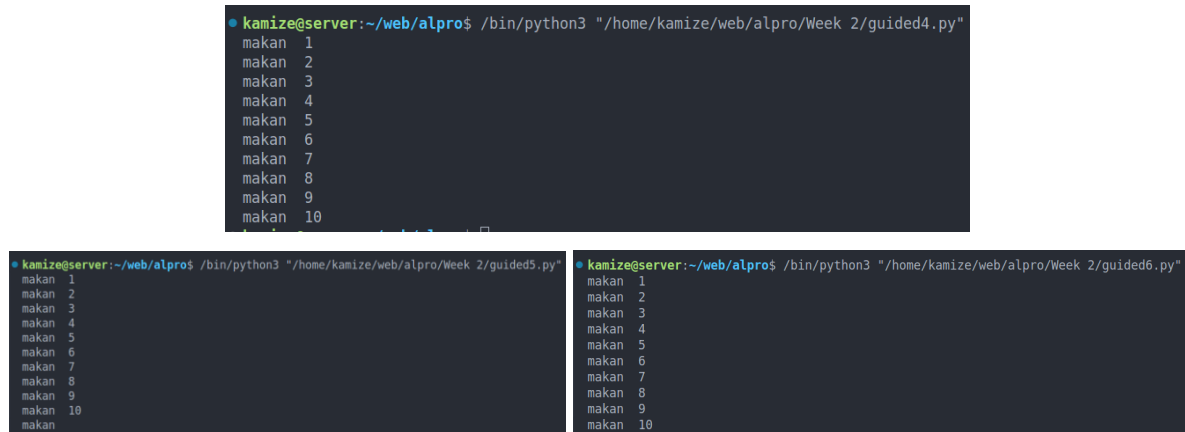
```
for i in range (10):
    print("makan ",i+1)
```

```
for i in range (10):
    print("makan ",i+1)
print("makan")
```

```
for i in range (10):
    # print("makan")
    print("makan ",i+1)
```

Gambar 2.7 Indentasi Menggunakan for loop

Fungsi for digunakan untuk mengulang langkah sebanyak fungsi range(10) yang menghasilkan total 10 angka (dimulai dari angka 0 sampai 9). Variable i digunakan untuk menampung hasil dari iterasi. i+1 digunakan untuk menambahkan 1 dari setiap nilai agar dimulai dari angka 1. Fungsi print () untuk menghasilkan output seperti pada gambar 2.8.



```
kamize@server:~/web/alpro$ /bin/python3 "/home/kamize/web/alpro/Week 2/guided4.py"
makan 1
makan 2
makan 3
makan 4
makan 5
makan 6
makan 7
makan 8
makan 9
makan 10

kamize@server:~/web/alpro$ /bin/python3 "/home/kamize/web/alpro/Week 2/guided5.py"
makan 1
makan 2
makan 3
makan 4
makan 5
makan 6
makan 7
makan 8
makan 9
makan 10

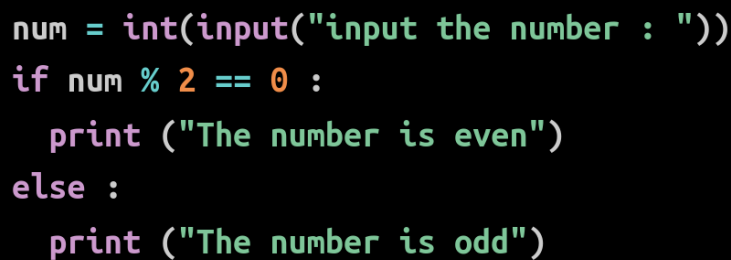
kamize@server:~/web/alpro$ /bin/python3 "/home/kamize/web/alpro/Week 2/guided6.py"
makan 1
makan 2
makan 3
makan 4
makan 5
makan 6
makan 7
makan 8
makan 9
makan 10
```

Gambar 2.8 Hasil Output for loop

2.4 Memahami indentasi pada python

Tulis kode program seperti pada gambar 2.9 menggunakan visual studio code dan jalankan programnya.

Indentasi pada *if* dan *else* ditandai dengan titik dua setelah pengkondisian dan baris setelahnya memiliki 4 spasi.



```
num = int(input("input the number : "))
if num % 2 == 0 :
    print ("The number is even")
else :
    print ("The number is odd")
```

Gambar 2.9 Indentasi Fungsi if dan else

Variable num untuk menyimpan nilai input dari user menggunakan fungsi input() yang sudah dikonversi menjadi tipe data integer menggunakan fungsi int(). Kemudian divalidasi menggunakan fungsi *if dan else* dengan kondisi jika hasil dari variable num dimoduluskan dengan 2 sama dengan 0 maka hasil outputnya *The number is even* jika tidak maka hasil outputnya *The number is odd* seperti pada gambar 2.10.

```

• kamize@server:~/web/alpro$ /bin/python3 "/home/kamize/web/alpro/Week 2/guided7.py"
input the number : 1
The number is odd
• kamize@server:~/web/alpro$ /bin/python3 "/home/kamize/web/alpro/Week 2/guided7.py"
input the number : 2
The number is even

```

Gambar 2.10 Hasil Output Indentasi if dan else

2.5 Memahami kembali konsep pembuatan variable dengan kasus menukar isi gelas.

Tulis kode program seperti pada gambar 2.11 menggunakan visual studio code dan jalankan programnya.

```

# variabel
glass_a = "susu"
glass_b = "jus"

# sebelum
print ("Sebelum ditukar isi gelas : ")
print (f"gelas berisi : {glass_a}")
print (f"gelas berisi : {glass_b}")

# variabel yang ditukar
glass_a,glass_b = glass_b,glass_a

# setelah
print ("Setelah ditukar isi gelas : ")
print (f"gelas berisi : {glass_a}")
print (f"gelas berisi : {glass_b}")

```

Gambar 2.11 Case Menukar Isi Gelas

Variable *glass_a* dan *glass_b* untuk menyimpan nilai susu dan jus. Variable tersebut ditukar dengan cara *glass_a = glass_b* dan *glass_b = glass_a*. maka akan menghasilkan output seperti pada gambar 2.12.

```

• kamize@server:~/web/alpro$ /bin/python3 "/home/kamize/web/alpro/Week 2/guided8.py"
Sebelum ditukar isi gelas
gelas berisi : susu
gelas berisi : jus
Setelah ditukar isi gelas
gelas berisi : jus
gelas berisi : susu

```

Gambar 2.12 Hasil Output Case Menukar Isi Gelas

3 Pembahasan Tugas Unguided

3.1 Buatlah beberapa variabel yang mendeskripsikan diri kalian dan tampilkan menggunakan fungsi `print()`.

```
# data diri
nama = "Ahsan Maulana Rizqi"
nim = 104062400071
alamat = "Sumpiuh, Banyumas, Jawa Tengah"
tanggal_lahir = "15 april 2003"
jenisKelamin = "Laki Laki"
umur = 21
noHP = "+62 895384238116"

print(f"nama      : {nama}")
print(f"nim       : {nim}")
print(f"alamat     : {alamat}")
print(f"tanggal_lahir : {tanggal_lahir}")
print(f"jenisKelamin : {jenisKelamin}")
print(f"umur       : {umur}")
print(f"noHP      : {noHP}")
```

Gambar 3.1 Variabel Data Diri

Variabel variabel data diri seperti pada gambar 3.1 dibuat untuk menyimpan nilai. Fungsi print (f "...") digunakan untuk menyisipkan nilai variabel langsung ke dalam string. Akan menghasilkan output seperti pada gambar 3.2.

```
● kamize@server:~/web/alpro$ /bin/python3 "/home/kamize/web/alpro/Week 2/unguided1.py"
nama      : Ahsan Maulana Rizqi
nim       : 104062400071
alamat    : Sumpiuh, Banyumas, Jawa Tengah
tanggal_lahir : 15 april 2003
jenisKelamin : Laki Laki
umur      : 21
noHP     : +62 895384238116
```

Gambar 3.2 Hasil Output Variabel Data Diri

- 3.2 Buatlah variabel yang berisi multiple line dengan isi menceritakan impian kalian setelah lulus dari IT Telkom Purwokerto, kemudian panggil variabel tersebut dengan fungsi print()

```
# multiple line

impian = """
Setelah lulus dari Telkom University Purwokerto.
Saya Bermimpi menjadi technopreneurship di bidang refraksi optisi.
Dengan tujuan untuk membantu para pengusaha dalam mengembangkan bisnisnya.
"""

print (impian)
```

Gambar 3.3 Multiple-line string

Variabel impian yang menyimpan nilai *multiple-line string* akan menghasilkan output sesuai dengan yang dituliskan tanpa menggunakan \n untuk baris baru. Hasil outputnya seperti pada gambar 3.4


```
● kamize@server:~/web/alpro$ /bin/python3 "/home/kamize/web/alpro/Week 2/unguided2.py"  
  
Setelah lulus dari Telkom University Purwokerto.  
Saya Bermimpi menjadi technopreneurship di bidang refraksi optisi.  
Dengan tujuan untuk membantu para pengusaha dalam mengembangkan bisnisnya.
```

Gambar 3.4 Hasil Output Multiple-line String

4 Ringkasan

Laporan ini membahas tentang tipe data, variabel, dan indentasi dalam pemrograman Python. Tipe data dalam Python mencakup beberapa kategori utama, seperti number (integer, float, complex), string, list, tuple, dictionary, dan boolean. Setiap tipe data memiliki karakteristik dan penggunaan yang berbeda dalam menyimpan serta memanipulasi nilai.

Identifier dalam Python adalah nama yang digunakan untuk variabel, fungsi, atau elemen lainnya. Python memiliki aturan khusus dalam penamaan identifier, seperti bersifat case-sensitive, tidak boleh menggunakan karakter khusus, dan tidak boleh menggunakan kata kunci yang telah digunakan dalam Python. Pemberian nama identifier sebaiknya sesuai dengan fungsi dan mudah dipahami.

Variabel dalam Python berfungsi sebagai wadah penyimpanan data yang dapat berubah sesuai kebutuhan program. Python adalah bahasa yang menggunakan dynamic typing, yang berarti tidak perlu mendeklarasikan tipe data saat membuat variabel, karena Python secara otomatis akan mengenali tipe data berdasarkan nilai yang diberikan.

Indentasi dalam Python sangat penting karena menentukan blok kode yang akan dieksekusi. Dalam looping seperti for loop, indentasi digunakan untuk menandakan bahwa perintah di dalamnya adalah bagian dari perulangan. Demikian juga pada pengkondisian seperti if-else, indentasi membantu menentukan kode yang dijalankan berdasarkan kondisi tertentu.

Tugas dalam laporan ini mencakup pemahaman berbagai konsep pemrograman dasar, termasuk pembuatan variabel data diri, penggunaan multiline string, dan penerapan konsep swapping nilai antar variabel. Dengan memahami dasar-dasar ini, kita dapat menulis kode Python yang lebih rapi, efisien, dan mudah dipahami.