

Nama: Muh Zaky Maulana Haryanto NIM: 065002500026	 UNIVERSITAS TRISAKTI	MODUL 7 (V.1.0)
Hari/Tanggal: Hari Rabu 5 November 2025	Praktikum Algoritma & Pemrograman	Nama Dosen: Ratna Shofiaty, S.Kom, M.Kom

Fungsi pada Pemrograman Python - Latihan

1. Teori Singkat

Fungsi merupakan suatu bagian dari program yang dimaksudkan untuk mengerjakan suatu tugas tertentu dan letaknya terpisah dari program yang memanggilnya. Pada pembuatan program yang kompleks dan memiliki banyak fitur, kita diharuskan menggunakan fungsi. Fungsi diperlukan agar mempermudah kita dalam membaca sebuah kode program dan mempermudah untuk merawatnya. Dengan adanya sebuah fungsi kita tidak perlu menulis kode sepanjang gerbang kereta api di program utama dan kita dapat memecah atau membaginya lalu tinggal memanggil fungsinya saja didalam program utama. Dan itu juga akan mempersingkat penulisan dari program utama tersebut nantinya

Fungsi memiliki parameter yang dapat dipergunakan untuk memasukkan atau menampung variabel kedalam sebuah fungsi. Dalam pendeklarasian parameter pada sebuah fungsi kita juga dapat memasukkan *default argument/parameter*.

Fungsi juga dapat mengembalikan nilai dengan cara menggunakan keyword *return* yang di taruh didalam blok program fungsi itu sendiri untuk mengembalikan nilai yang ingin kita kembalikan kedalam program utama.



Membuat Fungsi pada Python

Fungsi pada Python, dibuat dengan kata kunci *def* kemudian diikuti dengan nama fungsinya.

```
# Pendeklarasian sebuah fungsi
def nama_fungsi():
    print("Hello Ini Fungsi")

# Pemanggilan sebuah fungsi
nama_fungsi()
```

Fungsi dengan parameter pada Python

Memasukkan parameter kedalam sebuah fungsi

```
# pendeklarasian fungsi
def luas_persegi(sisi):
    luas = sisi * sisi
    return luas

# pemanggilan fungsi
print("Luas persegi: ", luas_persegi(6))
```

2. Alat dan Bahan

Hardware : Laptop/PC

Software : Spyder (Anaconda Python)



3. Elemen Kompetensi

Elemen Kompetensi I : Mahasiswa dapat menggunakan struktur percabangan berdasarkan data yang diberikan oleh user.

a. Latihan pertama

Buatlah sebuah program yang bisa menentukan apakah itu bilangan prima atau bukan. Gunakan 2 fungsi beserta implementasikan parameternya.

Contoh Output

```
Masukkan angka: 3
3 adalah bilangan Prima
> |
```

```
Masukkan angka: 4
4 bukanlah bilangan Prima
|
```

Source Code

```
def bilangan_prima():

    def is_prime(n):
        """
        ...
        ...
        if n < 2:
            return False

        for i in range(2, int(n**0.5) + 1):
            if n % i == 0:
                return False

        return True

    try:
        angka = int(input("Masukkan angka: "))

        if is_prime(angka):
            print(f"{angka} adalah bilangan Prima")
        else:
```



```

print(f"{{angka}} bukanlah bilangan Prima")

except ValueError:
    print("Input tidak valid. Harap masukkan angka.")

bilangan_prima()

```

Output

The screenshot shows the Spyder IDE interface. On the left, the code editor displays a Python script named 'def fungsi bersarang.py'. The code defines a function 'bilangan_prima' that checks if a number is prime. The code includes a docstring and a check for numbers less than 2. It then iterates from 2 to the square root of n, checking for divisibility. If any divisor is found, it returns False; otherwise, it returns True. The right side of the interface shows the 'Console 1/A' tab where the script is run. The output shows three test runs: 1, 0, and 3, with their respective results ('bukanlah bilangan Prima', '0 bukanlah bilangan Prima', and '3 adalah bilangan Prima').

b. Latihan Kedua

Buatlah sebuah program yang bisa menampilkan output seperti berikut ini:

```

>>1 //Input
>>1st //output
>>2 //input
>>2nd //output.
>>13 //input
>>13rd //output

```

Dst. Menggunakan implementasi fungsi dengan parameter.

Contoh Output



```
Ordinal Number
ketik 0 untuk mengentikan program
masukkan angka: 1
(1, 'st')
masukkan angka: 2
(2, 'nd')
masukkan angka: 3
(3, 'rd')
masukkan angka: 0
(0, 'th')
terima kasih telah menggunakan program saya
```

Source Code

```
@author: Muhammad Zaky Maulana Haryanto
"""
def run_ordinal_program():
    """
    Fungsi utama yang membungkus seluruh logika program.
    """

    def get_ordinal_tuple(number):
        """
        Fungsi ini "bersarang" di dalam run_ordinal_program.
        Fungsi ini hanya bisa diakses dari dalam fungsi luarnya.
        """

        if number % 100 in [11, 12, 13]:
            suffix = 'th'
        else:
            last_digit = number % 10
            if last_digit == 1:
                suffix = 'st'
            elif last_digit == 2:
                suffix = 'nd'
            elif last_digit == 3:
                suffix = 'rd'
```



```
else:  
    suffix = 'th'  
  
return (number, suffix)  
print("Ordinal Number")  
print("ketik 0 untuk mengentikan program")  
  
while True:  
    try:  
        angka_input = int(input("masukkan angka: "))  
  
        hasil_tuple = get_ordinal_tuple(angka_input)  
  
        print(hasil_tuple)  
  
        if angka_input == 0:  
            print("terima kasih telah menggunakan program saya")  
            break  
  
    except ValueError:  
        print("Input tidak valid. Silakan masukkan angka integer.")  
  
run_ordinal_program()
```

Output



The screenshot shows the Spyder IDE interface. On the left, there is a code editor with several tabs open, including 'untitled1.py*', which contains the following Python code:

```
35 while True:
36     try:
37         angka_input = int(input("masukkan angka: "))
38
39         hasil_tuple = get_ordinal_tuple(angka_input)
40
41         print(hasil_tuple)
42
43         if angka_input == 0:
44             print("terima kasih telah menggunakan program saya")
45             break
46
47     except ValueError:
48         print("Input tidak valid. Silakan masukkan angka integer")
49
50 run_ordinal_program()
```

To the right of the code editor is a 'Console 1/A' window showing the output of the program. The user has run the file and entered several integers, receiving the corresponding ordinal number strings as output:

```
Type "copyright", "credits" or "license" for more information.

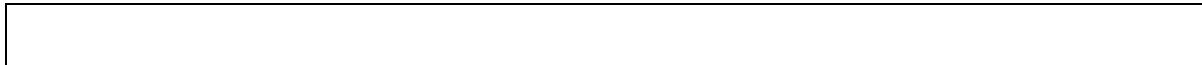
IPython 8.37.0 -- An enhanced Interactive Python. Type '?` for help.

In [1]: %runfile C:/Users/ASUS/Downloads/untitled1.py --wdir
Ordinal Number
ketik 0 untuk mengentikan program
masukkan angka: 1
(1, 'st')
masukkan angka: 2
(2, 'nd')
masukkan angka: 3
(3, 'rd')
masukkan angka: 0
(0, 'th')
terima kasih telah menggunakan program saya

In [2]:
```

4. File Praktikum

Github Repository:



5. Soal Latihan

Soal:

1. Jika sebuah deklarasi fungsi memiliki parameter a “`def nama_fungsi(a):`”, tetapi ketika dipanggil pada program utama kita mengisi fungsi tersebut dengan nama variabel b “`nama_fungsi(b)`” berbeda dengan nama parameter yang sebelumnya dideklarasikan, apakah program tetap akan berjalan dengan sesuai? dan jika sesuai mengapa demikian dan jika tidak mengapa demikian?
2. Deskripsikan serta narasikan jalannya alur source code program yang sebelumnya telah kalian buat pada Elemen Kompetensi Latihan Kedua!

Jawaban:

1. (Isi Dengan Jawaban Kalian)
2. (Isi Dengan Jawaban Kalian)

6. Kesimpulan

- a. Dalam pengerjaan program dengan bahasa pemrograman Python, kita harus benar-benar teliti dalam menginputkan suatu fungsi untuk menampilkan suatu keluaran pada layar dengan sesuai.
- b. Kita dapat mengetahui... (Tolong Isi lebih dari dua baris!)

7. Cek List (✓)

No	Elemen Kompetensi	Penyelesaian	
		Selesai	Tidak Selesai
1.	Menggunakan struktur percabangan berdasarkan data yang diberikan oleh user.		

8. Formulir Umpam Balik

No	Elemen Kompetensi	Waktu Pengerjaan	Kriteria
1.	Menggunakan struktur percabangan berdasarkan data yang diberikan oleh user.	... Menit	...



Keterangan:

1. Menarik
2. Baik
3. Cukup
4. Kurang

Menyetujui
Jakarta, 1 Juli 2024

Penanggung Jawab Praktikum



(Anung B. Ariwibowo, M.Kom)

Kepala Laboratorium
Pemrograman



(Anung B. Ariwibowo, M.Kom)



Program Studi S1 Informatika | Program Studi S1 Sistem Informasi
Jurusan Teknik Informatika - Fakultas Teknologi Industri