


Nama: (M.Satria Pratama) NIM: (065002200017)	 Praktikum Algoritma & Pemrograman	MODUL 9 Nama Dosen: Anung B. Ariwibowo, M. Kom
Hari/Tanggal: Jum'at, 18/11 2022		Nama Asisten Labratorium: 1. Azhar Rizki Zulma 065001900001

File Handling

1. Teori Singkat

File Handling

File Handling merupakan sebuah istilah penanganan file dalam Bahasa Pemrograman. Dalam konsep file handling sendiri kita dapat membuka, menutup, membaca, menulis, menambahkan, serta mengcopy file. Python sendiri dalam memperlakukan beberapa file berbeda, entah itu dalam bentuk biner ataupun sebagai teks. Dalam mengimplementasikannya pada Bahasa pemrograman python sendiri, pemanggilanya, sintaksnya adalah: *file = open('nama file', 'mode')*. Ada tiga jenis mode, yang disediakan Python dan bagaimana file dapat dibuka:

"r", untuk membaca.

"w", untuk menulis.

"a", untuk menambahkan.

"r+", untuk membaca dan menulis

Dan jangan lupa jika kita telah selesai melakukan sesuatu pada file tersebut kita harus memanggil method untuk menutup file tersebut. Pemanggilan methodnya sendiri adalah dengan *file.close()*



Berikut Contoh programnya



```
# Write File
file = open("Test.txt", "w")
file.write("Nadiya Amanda Rizkania")
file.close()

# Read File
file = open("Test.txt", "r")
text = file.read()
print(text)
file.close()

# Add Text
file = open("Test.txt", "a")
file.write(" - 20 Tahun - Jakarta")
file.close()

# Read & Write File
file = open("Test.txt", "r+")
text = file.read()
print(text)
file.close()
```

2. Alat dan Bahan

Hardware : Laptop/PC

Software : Spyder (Anaconda Python)



3. Elemen Kompetensi

a. Latihan pertama

Buatlah sebuah Text File dengan nama Biodata.txt menggunakan implemementasi File Handling dengan output seperti dibawah ini yang diinputkan oleh user:

Nama: Nama Kalian

Umur: Umur Kalian

Alamat: Alamat Kalian

Email: Email Kalian

Dosen Wali: Dosen Wali Kalian

Masukkan kedua metode tulis dan metode baca kedalam fungsi agar program lebih terstruktur.

Source Code

```
def data(nama,Umur,alamat,email,dosenwali):  
  
    file = open('Biodata.txt','w')    file.write(f'Nama : {nama} \nUmur : {Umur} \nAlamat  
: {alamat} \nEmail : {email}  
\nDosen wali : {dosenwali}')  
    file.close()  
def  
baca():  
    file = open('Biodata.txt','r')  
    text = file.read()  
    print(text)    file.close()  
  
    nama = input('Masukkan Nama: ') Umur =  
    input('Masukkan Umur: ') alamat =  
    input('Masukkan alamat: ') email =  
    input('Masukkan email: ') dosenwali =  
    input('Masukkan dosen wali: ')  
    print('\nBerikut biodatamu')  
    data(nama,Umur,alamat,email,dosenwali)  
baca()
```

Output



The screenshot displays the Spyder Python IDE interface. The main editor window shows a Python script named 'untitled8.py' with the following code:

```
1  # -*- coding: utf-8 -*-
2  """
3  Created on Fri Nov 18 14:43:37 2022
4
5  @author: satria
6  """
7
8  def data(nama,Umur,alamat,email,dosenwali):
9
10     file = open('Biodata.txt','w')
11     file.write(f'Nama : {nama} \nUmur : {Umur} \n/
12     file.close()
13
14     def baca():
15         file = open('Biodata.txt','r')
16         text = file.read()
17         print(text)
18         file.close()
19
20     nama = input('Masukkan Nama: ')
21     Umur = input('Masukkan Umur: ')
22     alamat = input('Masukkan alamat: ')
23     email = input('Masukkan email: ')
24     dosenwali = input('Masukkan dosen wali: ')
25     print('\nBerikut biodatamu')
26     data(nama,Umur,alamat,email,dosenwali)
27     baca()
```

The right-hand side of the IDE contains two panels. The top panel, titled 'Terminal 1', shows the 'Usage' section of the Python help documentation. The bottom panel, titled 'Console 1/A', displays the output of the program execution:

```
In [5]: runfile('C:/Users/satria/.spyder-
py3/untitled8.py', wdir='C:/Users/
satria/.spyder-py3')
Masukkan Nama: M.Satria Pratama
Masukkan Umur: 18
Masukkan alamat: Bekasi
Masukkan email: 065002200017
Masukkan dosen wali: Teddy Siswanto

Berikut biodatamu
Nama : M.Satria Pratama
Umur : 18
Alamat : Bekasi
Email : 065002200017
Dosen wali : Teddy Siswanto

In [6]:
```

The status bar at the bottom indicates the interpreter is 'internal (Python 3.8.10)', with various settings like 'Completions: internal', 'LSP: Python', 'Line 26, Col 7', 'UTF-8', 'CRLF', 'RW', and 'Mem 44%'.

b. Latihan Kedua

Buatlah sebuah program dimana program tersebut dapat membuat file, membaca file dan menambahkan text ke dalam file yang dimana, nama file didapat dari hasil inputan user dan juga data yang ditambahkan kedalam file didapat dari inputan user, implementasikan program kedalam fungsi dan juga implementasikan percabangan serta perulangan pada



program seperti yang diajarkan pada materi sebelumnya dimana jika pilihan menu close tidak dipilih (diinputkan oleh user) maka program akan terus berjalan.

Source Code

```
def make_file():
    file = open("Praktikum_9_Latihan_2.txt", "w")
    nama = str(input("Masukan nama anda: "))
    nim = str(input("Masukan NIM anda: "))
    angkatan = str(input("Masukan tahun angkatan anda: "))
    file.write("Nama: " + nama)
    file.write("\nNIM: " + nim)
    file.write("\nAngkatan: " + angkatan)
    file.close()

def read_file():
    file = open("Praktikum_9_Latihan_2.txt", "r")
    text = file.read()
    print(text)
    file.close()

def add_file():
    file = open("Praktikum_9_Latihan_2.txt", "a")
    friend = str(input("Masukan nama sahabat anda: "))
    note = input("Masukan catatan anda: ")
    file.write("\nSahabat: " + friend)
    file.write("\nCatatan: " + note)
    file.close()

def panggil_file():
    ulang = True
    while(ulang):
        print("### Program File Handling ###")
        print("1. Membuat & Menulis File")
        print("2. Membaca File")
        print("3. Menambahkan Isi File")
        print("4. Keluar dari Program")
        choose = int(input("Masukan Nomor Pilihan Anda:"))
```

Fff



```
if (choose == 1):
    print("\n")
    make_file()
    print("\n")

if (choose == 2):
    print("\n")
    read_file()
    print("\n")
if (choose == 3):
    print("\n")
    add_file()
    print("\n")
if (choose == 4):
    print("Terimakasih telah menggunakan program")
    ulang = False

panggil_file()
```

```
Masukan Nomor Pilihan Anda: 1

Masukan nama anda: M.Satria Pratama
Masukan Nim anda: 065002200017
Masukan tahun angkatan anda: 2022

### Program File Handling ###
1. Membuat & Menulis Fle
2. Membaca File
3. Menambahkan Isi File
4. Keluar dari Program
Masukan Nomor Pilihan Anda: 2

nama anda: M.Satria Pratama
Nim anda: 065002200017
tahun angkatan anda: 2022

### Program File Handling ###
1. Membuat & Menulis Fle
2. Membaca File
3. Menambahkan Isi File
4. Keluar dari Program
Masukan Nomor Pilihan Anda: 2

nama anda: M.Satria Pratama
Nim anda: 065002200017
```



4. File Praktikum

Github Repository:

```
print("Taruh link github repository praktikum kalian disini yaa...")
```

5. Soal Latihan Soal:

1. Apa saja kegunaan file handling dalam sebuah bahasa pemrograman dan kenapa file handling diperlukan dalam sebuah Bahasa pemrograman?
2. Deskripsikan serta narasikan jalannya alur source code program yang sebelumnya telah kalian buat pada Elemen Kompetensi Latihan Kedua!

Jawaban:

1. Kegunaan file handling dalam bahasan pemrograman adalah untuk user dapat membuat data, membaca data, menambahkan data, atau menghapus data yang dibuat user. Dengan adanya file handling dapat memudahkan seseorang untuk membuat file
2. Pertama membuat fungsi untuk membuat file .txt, lalu difungsi ke 2 kita membuat code untuk membaca file. Lalu di fungsi ke 3 membuat fungsi untuk menambahkan di file yang dibuat. Selanjutnya di fungsi ke 4 membuat fungsi untuk memanggil file yang dibuat. Selanjutnya dengan if kita dapat membuat user untuk memilih fungsi” yang sebelumnya di buat6.

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 bahwa dengan python kita dapat menggunakan fungsi file untuk membuat data, menambahkan data, membaca data, atau menghapus data

7. Cek List ✓

No	Elemen Kompetensi	Penyelesaian	
		Selesai	Tidak Selesai
1.	Latihan Pertama	✓	
2.	Latihan Kedua	✓	

8. Formulir Umpan Balik

No	Elemen Kompetensi	Waktu Pengerjaan	Kriteria
----	-------------------	------------------	----------



1.	Latihan Pertama	20Menit	Baik
2.	Latihan Kedua	30 Menit	Baik

Keterangan: 1.

Menarik

2. Baik

3. Cukup

4. Kurang

