

Buku Panduan
Penggunaan PITON
(Python IntegraTed wOrkspace and visualizationN)



Tim Penyusun:
Elvina
Mewati Ayub
Oscar Karnalim
Maresha Caroline Wijanto

FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN MARANATHA
2018

KATA PENGANTAR

PITON (*Python IntegraTed wOrkspace and visualizatioN*) adalah aplikasi yang menyediakan lingkungan pembelajaran (*learning environment*) untuk pemrograman dasar dalam bahasa pemrograman Python yang dilengkapi fitur visualisasi dan fitur berbasis kebutuhan pengguna. Aplikasi ini disediakan untuk membantu pemula yang belajar pemrograman menggunakan bahasa Python, agar pemula dapat memahami jalannya eksekusi program dengan memakai fitur visualisasi. Fitur visualisasi dapat digunakan atau tidak digunakan sesuai kebutuhan pengguna.

Buku Panduan ini tersusun atas 3 bagian yaitu:

1. Cara Instalasi Perangkat Yang Diperlukan Aplikasi PITON

Untuk dapat menjalankan aplikasi ini, pada komputer yang digunakan harus sudah terinstalasi Sistem Operasi Windows 10, .NET Framework 3.5, serta kompailer Python versi 3.6.2.

2. Fitur Utama PITON

Dijelaskan fitur utama yang disediakan oleh PITON bagi pengguna.

3. Cara Penggunaan PITON

Dijelaskan cara penggunaan PITON dalam dua modus, yaitu cara penggunaan tanpa visualisasi dan cara penggunaan dengan visualisasi.

Harapan kami, buku panduan ini dapat membantu pengguna di dalam memanfaatkan aplikasi PITON untuk mempelajari pemrograman dengan bahasa Python.

Bandung, Mei 2018

Tim Penyusun

DAFTAR ISI

KATA PENGANTAR	2
DAFTAR ISI.....	3
I. CARA INSTALASI PERANGKAT YANG DIPERLUKAN APLIKASI PITON	4
II. FITUR UTAMA PITON	7
III. CARA PENGGUNAAN PITON	10
III.1 Penggunaan PITON tanpa fitur visualisasi	10
III.2 Penggunaan PITON dengan fitur visualisasi	23

I. CARA INSTALASI PERANGKAT YANG DIPERLUKAN APLIKASI PITON

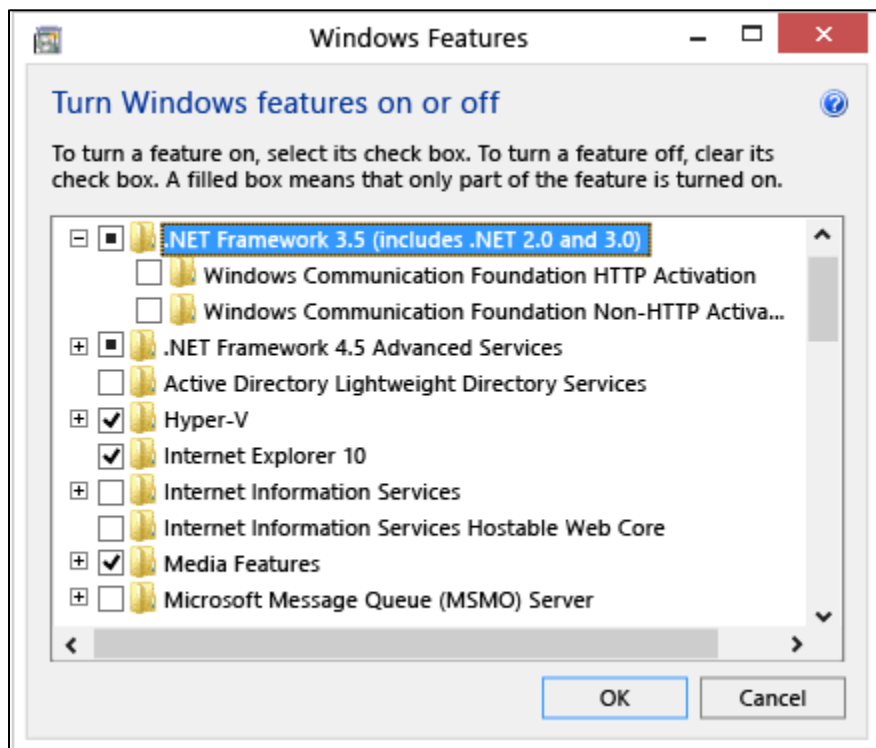
Perangkat lunak yang diperlukan untuk menggunakan PITON, adalah:

1. Operating System Windows 10

2. Aktivasi .NET Framework 3.5

Berikut adalah langkah-langkah untuk mengaktifkan **.NET Framework** pada Windows 10, seperti pada Gambar 1:

- Menekan tombol Windows  pada *keyboard*, kemudian ketik "Windows Features", dan tekan Enter. *Dialog box* "Turn Windows features on or off" akan muncul.
- Pilih ".NET Framework 3.5 (includes .NET 2.0 dan 3.0)" pada *checkbox*, tekan OK dan *reboot* komputer jika diminta.

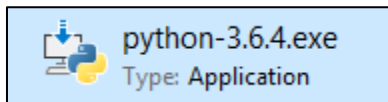


Gambar 1 Aktivasi .NET Framework

3. Instalasi Python Version 3.6.4

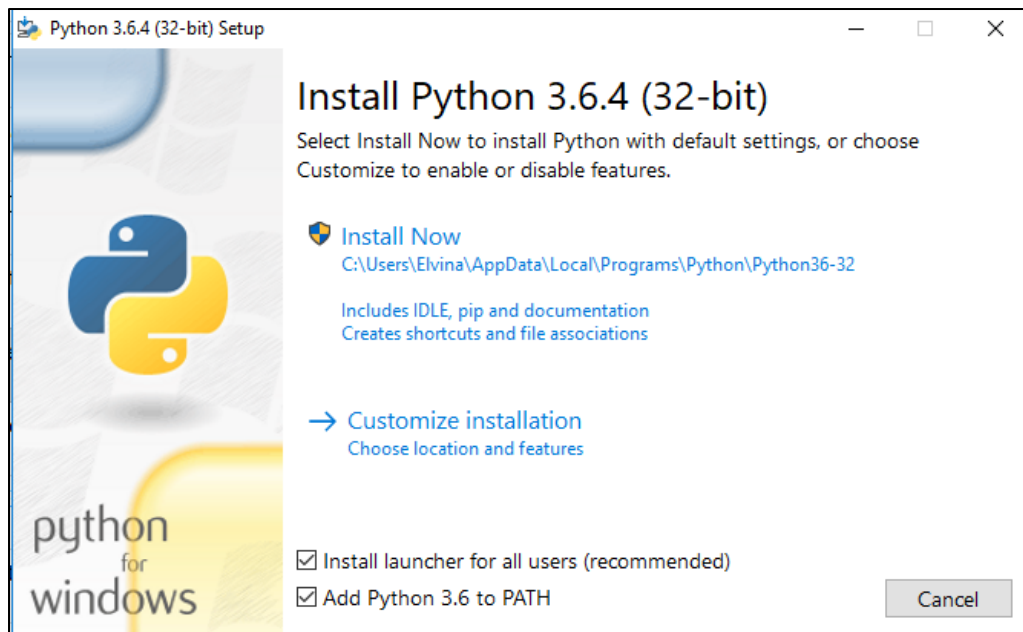
Berikut adalah langkah-langkah untuk mengaktifkan **Python Version 3.6.4** pada Windows 10, yaitu dengan menginstal Python dari CD program sebagai berikut :

- a. Salinlah *file* “python-3.6.4.exe” dari folder Python-364 dalam CD program seperti pada gambar 2.



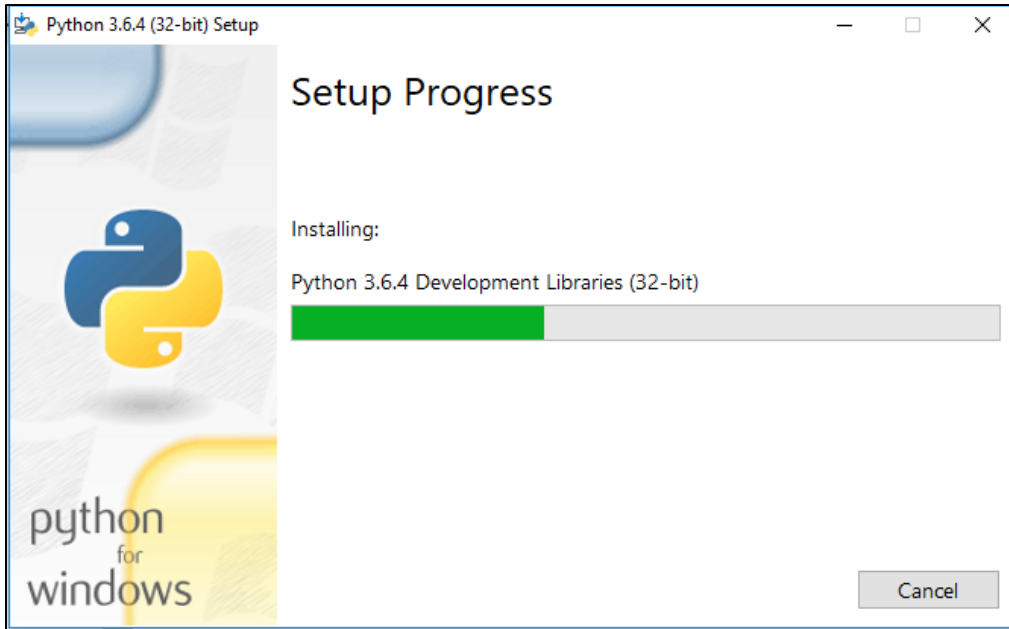
Gambar 2 File Python

- b. *Double* klik pada file “python-3.6.4.exe” untuk meng-*install* Python.
- c. Jendela untuk *setup* Python 3.6.4 akan muncul, berikan *checklist* pada “*Install launcher for all user (recommended)*” dan “*Add Python3.6 to PATH*” seperti pada gambar 3. Kemudian tekan “Install Now” untuk menginstalasi aplikasi Python. Jika komputer sudah pernah terinstalasi Python versi sebelumnya, maka “Install Now” akan berubah menjadi “Upgrade Now” seperti pada gambar 3.



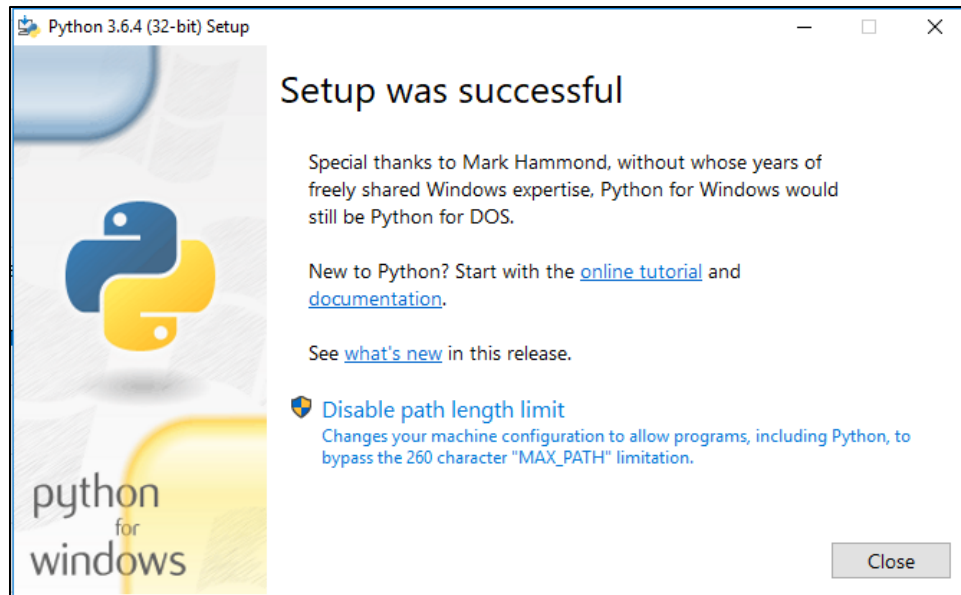
Gambar 3 Instalasi Python

- d. Instalasi Python akan berjalan seperti pada Gambar 4.



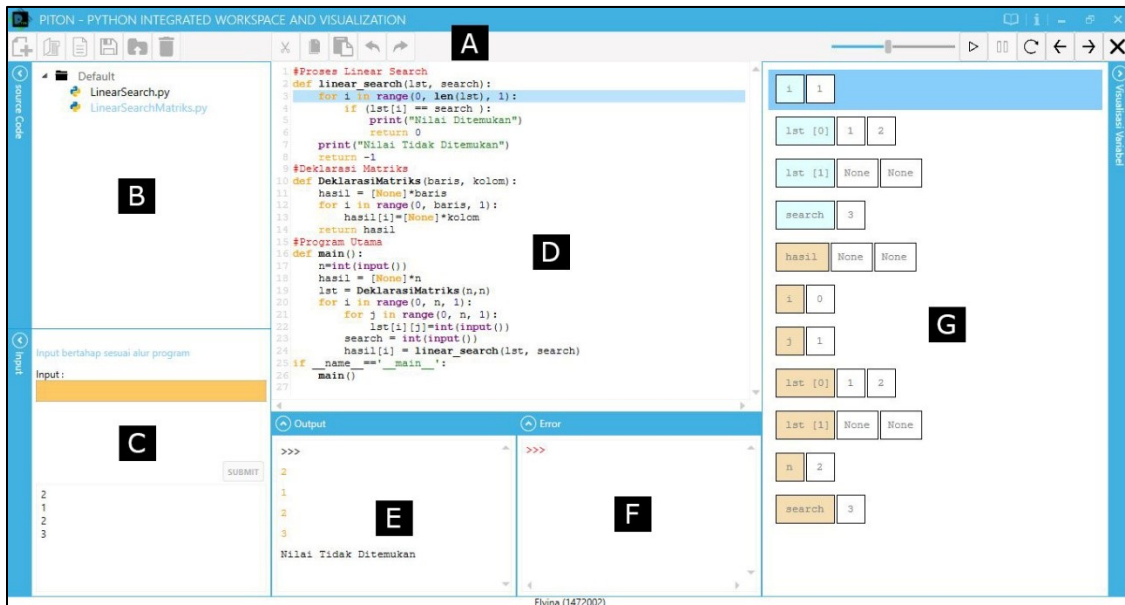
Gambar 4 Progres Instalasi

- e. Setelah selesai melakukan instalasi dan berhasil, maka akan muncul jendela *pop up* seperti gambar 5.



Gambar 5 Instalasi berhasil

II. FITUR UTAMA PITON



Gambar 6 Antarmuka PITON

Gambar 6 merupakan tampilan antarmuka aplikasi PITON, setiap bagian dari antarmuka akan dijelaskan sebagai berikut:

A. **Command Toolbar** merupakan tempat fitur-fitur untuk memanipulasi kode program.

Gambar	Command	Fungsi
	<i>New File</i>	Membuat <i>file</i> baru
	<i>Get Folder</i>	Membuka <i>folder</i> berisi <i>file</i> kode program yang berekstensi (*.py)
	<i>Get File</i>	Membuka <i>file</i> kode program dengan ekstensi (*.py)
	<i>Save</i>	Menyimpan <i>file</i> kode program
	<i>Save As</i>	Menyimpan <i>file</i> kode program
	<i>Delete</i>	Menghapus <i>file</i> dari <i>Source Code Viewer</i>
	<i>Cut</i>	Memindahkan potongan kode program

Gambar	Command	Fungsi
	<i>Copy</i>	Menyalin atau menggandakan potongan kode program
	<i>Paste</i>	Menempelkan potongan kode program
	<i>Undo</i>	Membatalkan suatu perintah yang sudah dilakukan sebelumnya
	<i>Redo</i>	Mengulang kondisi yang telah dibatalkan sebelumnya
	<i>Speed Bar</i>	Mengatur kecepatan animasi visualisasi. Semakin ke kiri, semakin bertambah kecepatan visualisasi dan sebaliknya. Pengaturan <i>speed bar</i> hanya dapat dilakukan jika visualisasi berada pada kondisi tahap awal.
	<i>Play</i>	Menjalankan animasi untuk visualisasi
	<i>Pause</i>	Menunda visualisasi
	<i>Reset</i>	Mengembalikan kondisi visualisasi ketahap awal
	<i>Previous</i>	Memundurkan kondisi visualisasi ketahap sebelumnya
	<i>Next</i>	Memajukan kondisi visualisasi ketahap setelahnya
	<i>Stop</i>	Menghentikan visualisasi dan eksekusi
	<i>Run</i>	Menjalankan eksekusi
	<i>Stop</i>	Menghentikan eksekusi
☐ Visualisasi	<i>Checkbox Visualisasi</i>	Pilihan visualisasi

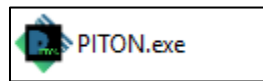
- B. *Source Code Viewer*** merupakan bagian dari PITON untuk menampilkan *folder* dan *file* yang dibuka.
- C. *Panel Input*** berfungsi untuk memasukkan entitas input baik yang satu ataupun banyak entitas input.
- D. *Source Code Editor*** berfungsi untuk memanipulasi kode program.
- E. *Panel Output*** berfungsi untuk menampilkan keluaran hasil eksekusi.
- F. *Panel Error*** berfungsi untuk menampilkan pesan kesalahan jika kode program tidak berhasil dieksekusi.
- G. *Panel Visualisasi*** berfungsi untuk menampilkan visualisasi variabel.

III. CARA PENGGUNAAN PITON

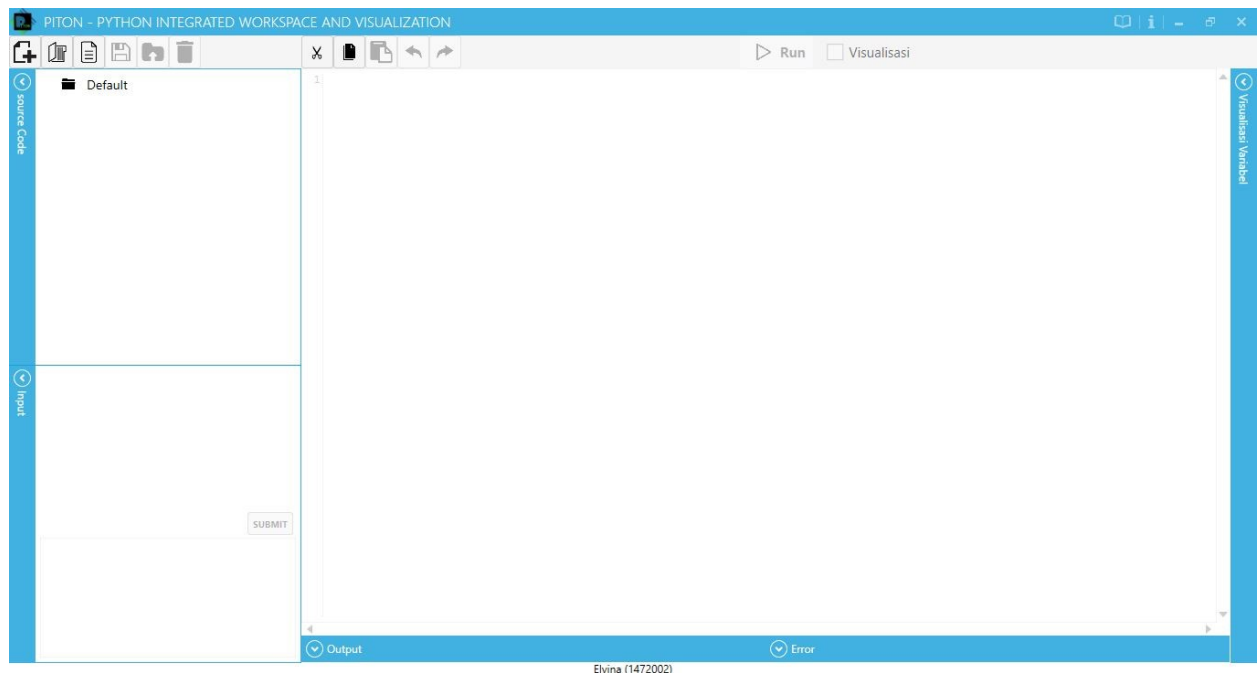
III.1 Penggunaan PITON tanpa fitur visualisasi

Tahapan penggunaan PITON tanpa fitur visualisasi, secara umum adalah sebagai berikut:

1. Salinlah folder PITON dari CD ke komputer.
2. Masuk ke folder PITON, kemudian *Double* klik pada file “PITON.exe” untuk menjalankan aplikasi PITON seperti pada gambar 7, sehingga muncul tampilan seperti gambar 8 di layar.



Gambar 7 File PITON



Gambar 8 Tampilan Awal PITON

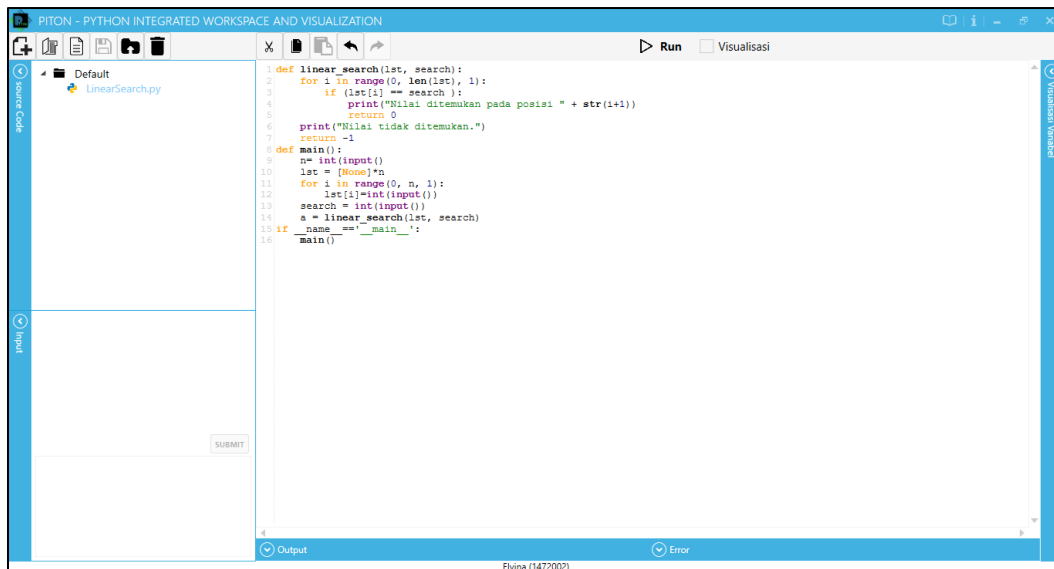
3. Salinlah kode program dari file “LinearSearch” pada folder Panduan dalam CD program ke bagian *source code editor* untuk mencoba aplikasi PITON, sehingga ditampilkan kode program berikut ini pada editor PITON, seperti pada Gambar 9.

```
def linear_search (lst, search):  
    for i in range (0, len(lst),1):  
        if (lst [i] == search):
```

```

        print ("Nilai ditemukan pada posisi " + str (i+ 1))
        return 0
    print ("Nilai tidak ditemukan.")
    return - 1
def main ():
    n= int (input())
    lst = [None] * n
    for i in range (0 ,n, 1) :
        lst [ i ] = int (input())
    search = int (input())
    a = linear_search (lst, search)
if __name__ == '__main__' :
    main ( )

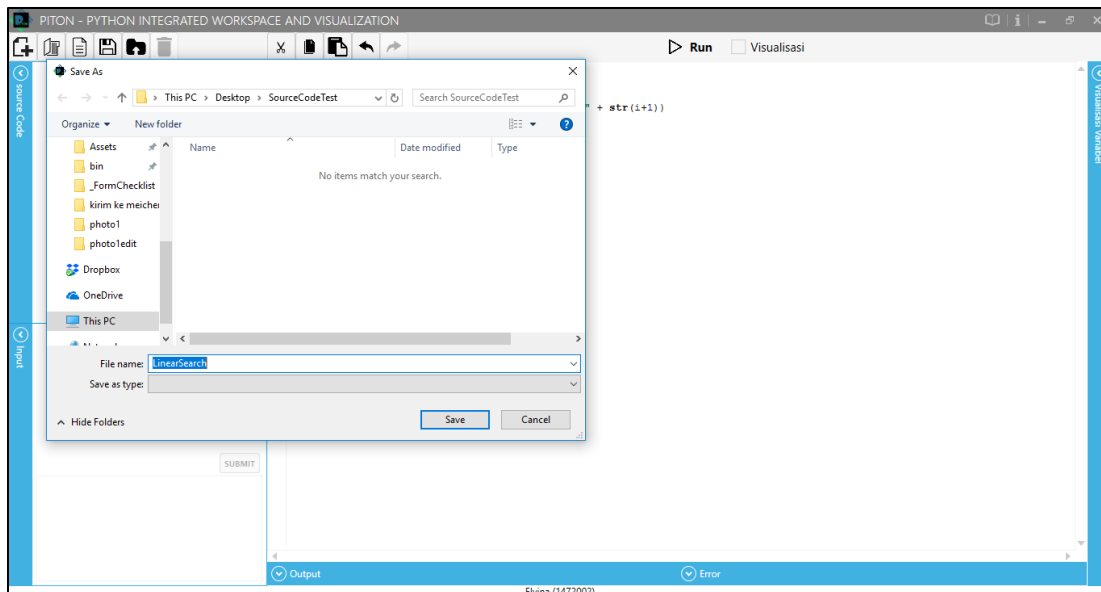
```



Gambar 9 Tampilan kode sumber program dalam PITON

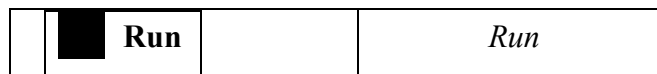
4. Menekan tombol *save/save as* pada *command toolbar* untuk menyimpan kode program. Simpanlah kode program dengan nama “LinearSearch” seperti pada Gambar 10.

	<i>Save</i>
	<i>Save As</i>

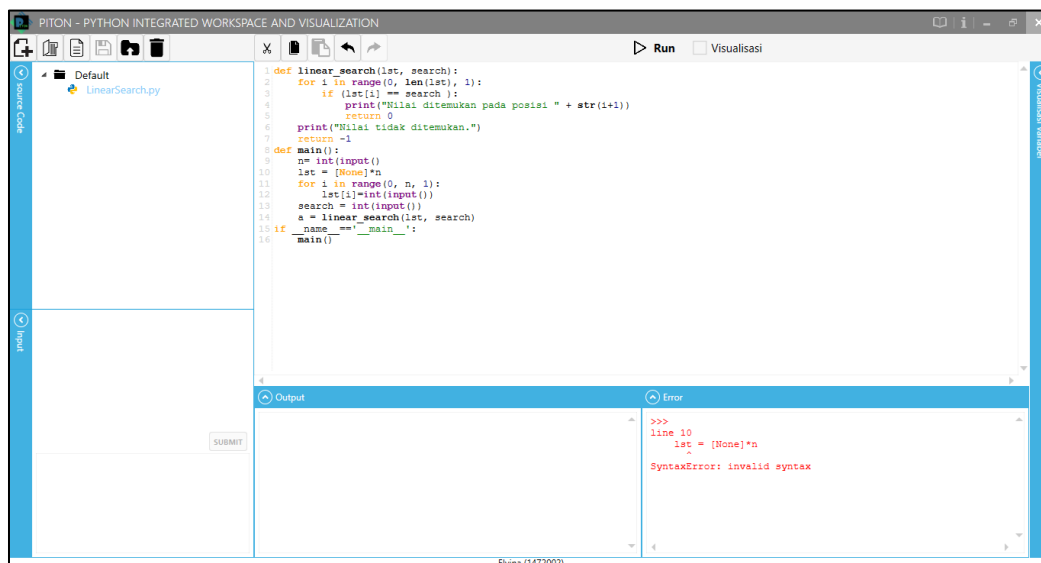


Gambar 10 Menyimpan file (SAVE)

- Kemudian jalankan kode program dengan menekan tombol “Run” pada *command toolbar*.



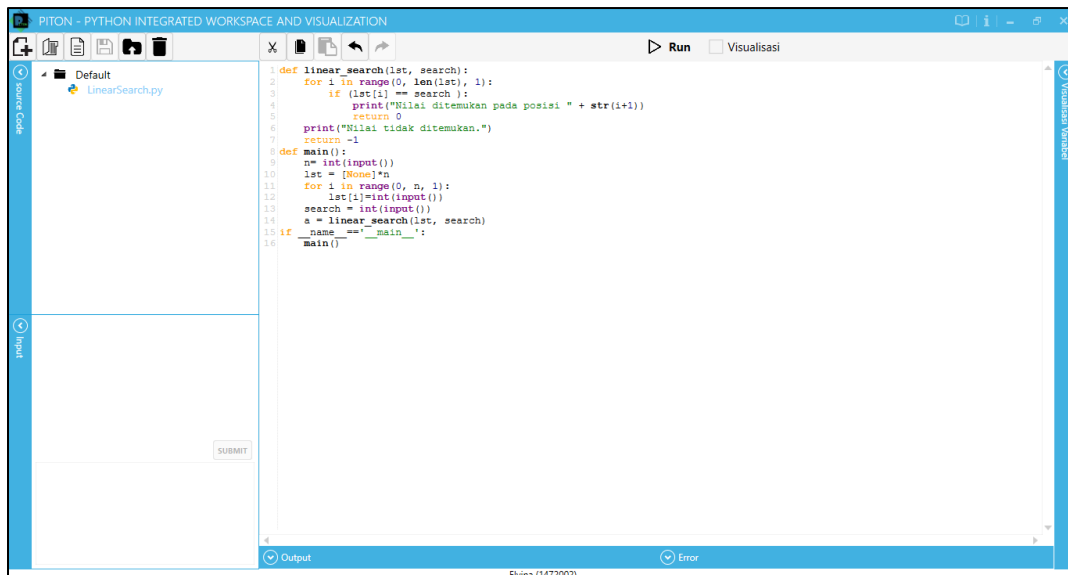
- PITON akan menampilkan pesan kesalahan karena kesalahan *syntax* pada kode program, seperti pada Gambar 11.



Gambar 11 Tampilan pesan kesalahan

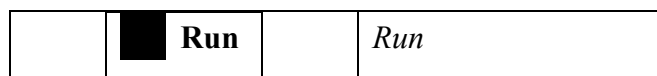
7. Perbaiki kode program dengan menambahkan “)” pada baris ke – 9, seperti pada Gambar 12.

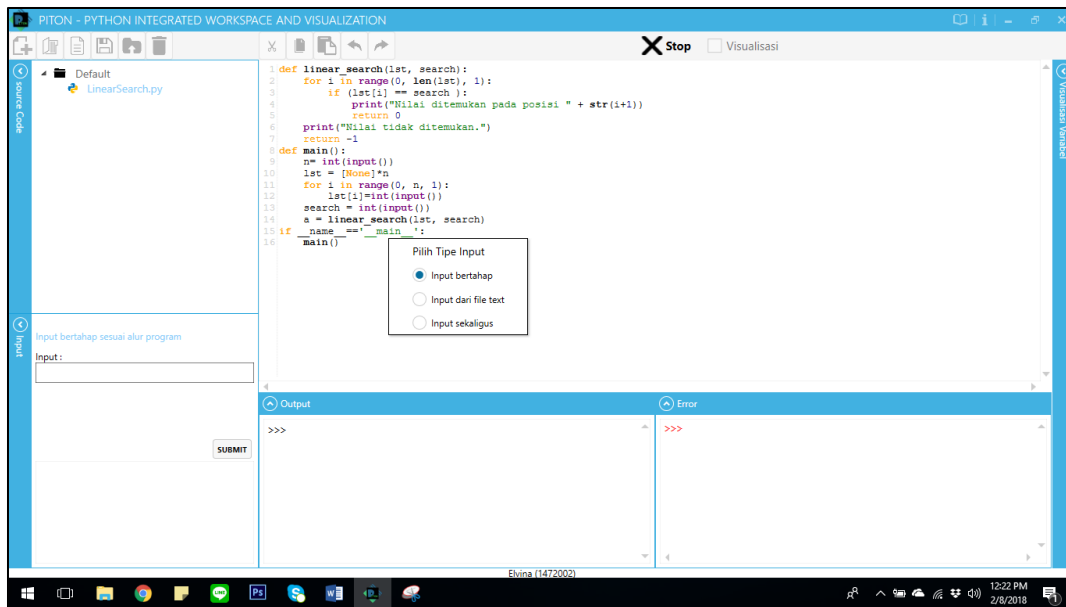
```
def linear_search (lst, search):
    for i in range (0, len (lst), 1):
        if (lst [ i] == search):
            print ("Nilai ditemukan pada posisi " + str (i+1))
            return 0
    print ("Nilai tidak ditemukan." )
    return - 1
def main ():
    n= int (input())                #baris ke-9
    lst = [ None ] * n
    for i in range ( 0 , n, 1 ) :
        lst [ i ] = int ( input ( ) )
    search = int ( input ( ) )
    a = linear_search ( lst, search )
if __name__ == '__main__' :
    main ( )
```



Gambar 12 Tampilan hasil perbaikan program

8. Kemudian jalankan kembali kode program dengan menekan tombol “Run”, sehingga muncul tampilan pada Gambar 13.

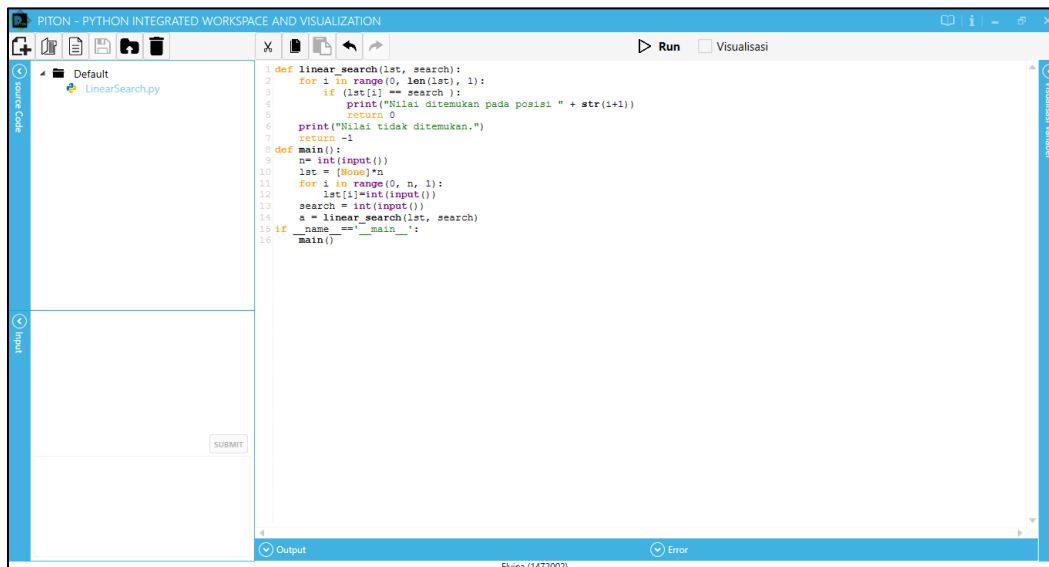




Gambar 13 Pilih Input bertahap

9. Pilihlah terlebih dahulu cara *input* yang diinginkan untuk memasukkan *input*. Terdapat 3 jenis cara *input* pada PITON, yaitu Input Bertahap, Input dari File Text, dan Input Sekaligus. Setiap cara akan dijelaskan secara rinci.

a. *Input Bertahap*



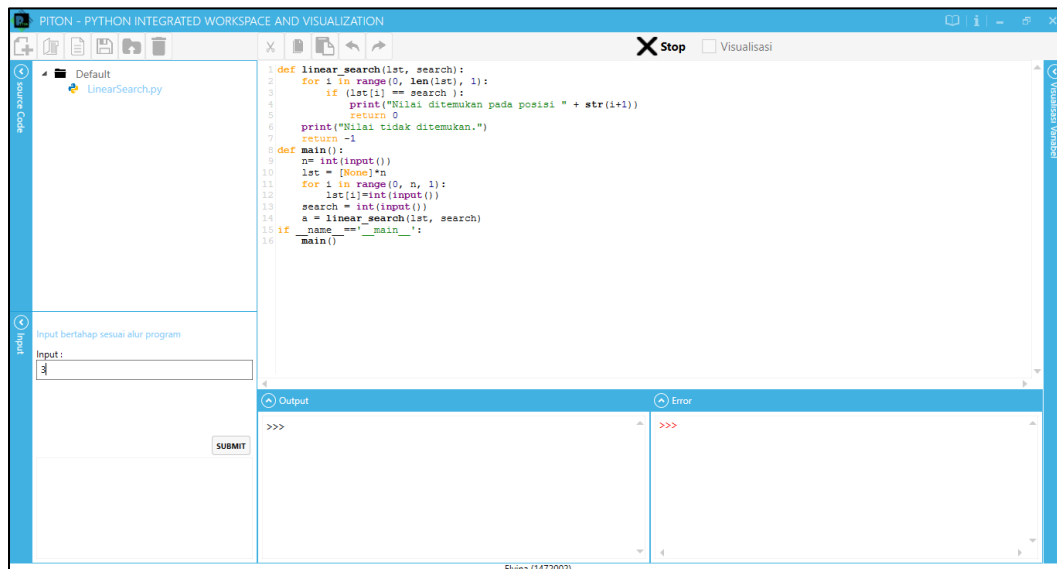
Gambar 14 Tampilan input bertahap

Memasukkan *input* untuk kode program secara bertahap. PITON akan meminta *input* kembali jika dibutuhkan setelah *input* yang dimasukkan sebelumnya berhasil diproses seperti pada Gambar 14 dan Gambar 15.

Gambar 15 Tampilan Input

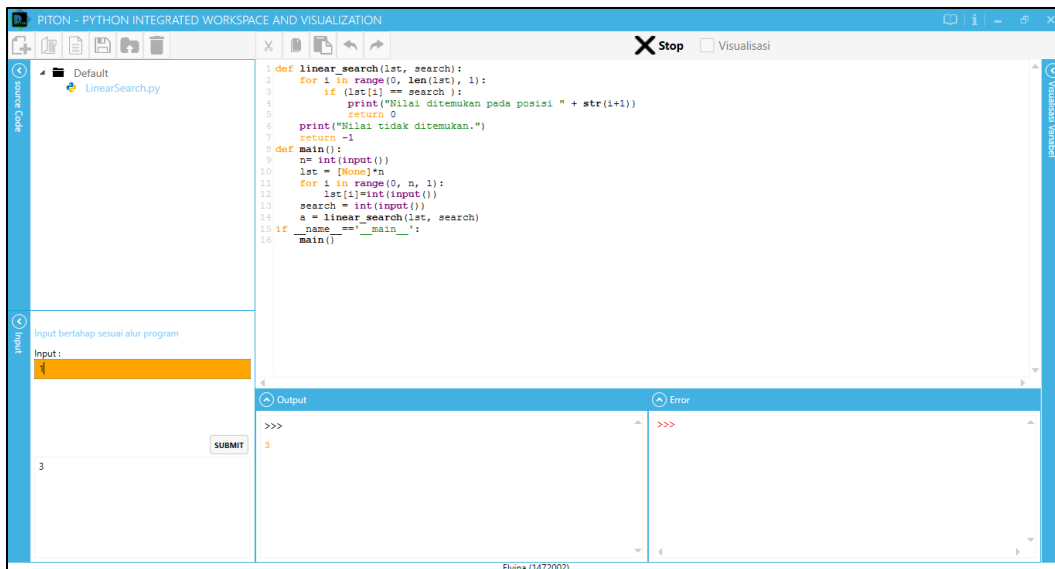
Tahapan penggunaan *Input* bertahap, adalah :

- i. Masukkan angka **3** kemudian tekan tombol “Submit” seperti pada Gambar 16.



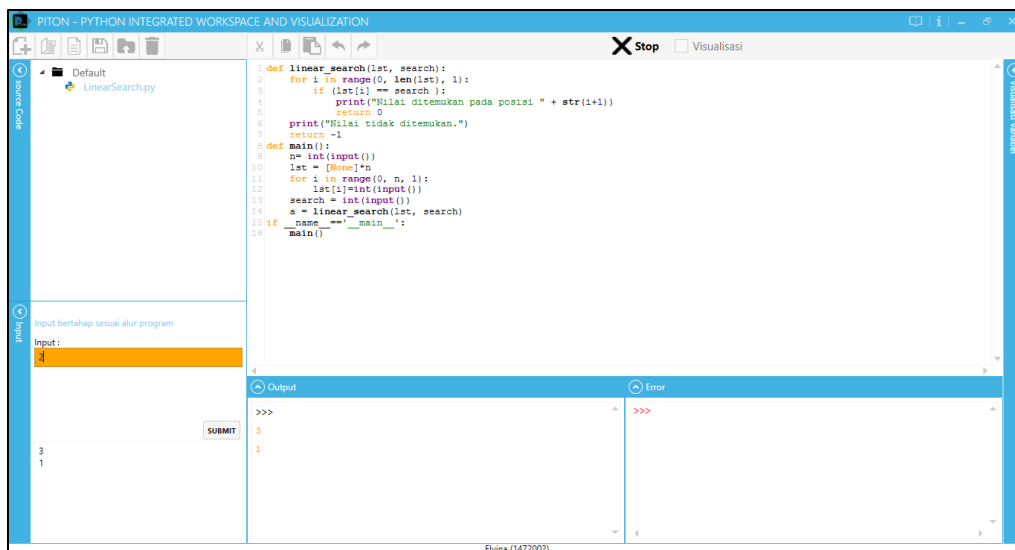
Gambar 16 Tampilan Input bertahap (1)

- ii. Masukkan kembali angka **1** kemudian tekan tombol “Submit” seperti pada Gambar 17.



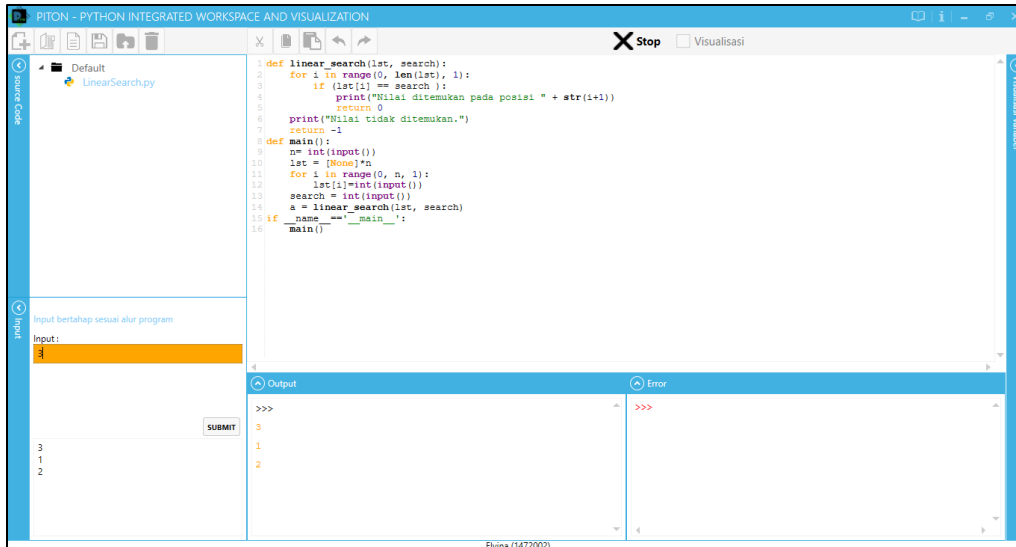
Gambar 17 Tampilan Input bertahap (2)

- iii. Masukkan kembali angka **2** kemudian tekan tombol “Submit” seperti pada Gambar 18.



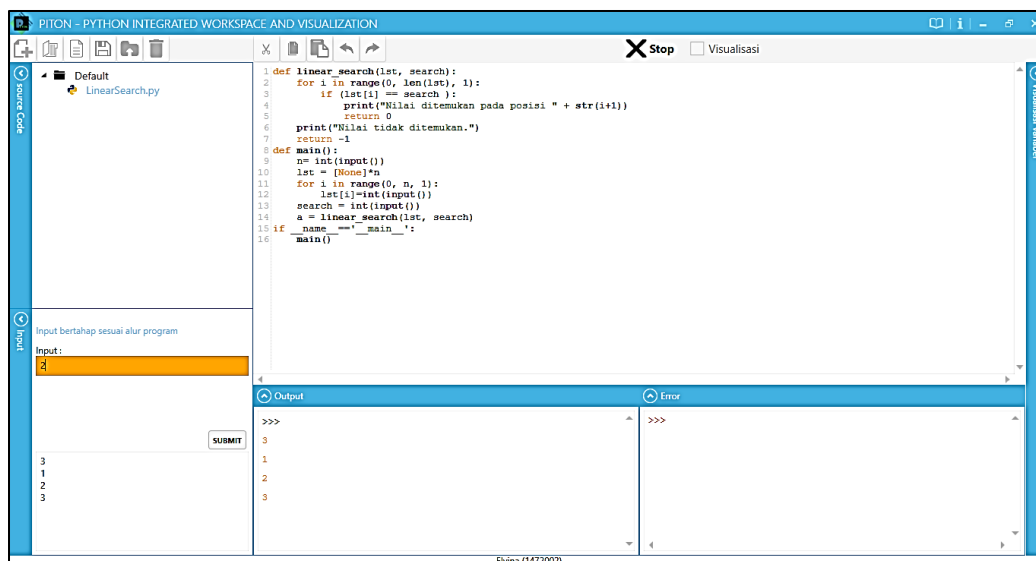
Gambar 18 Tampilan Input bertahap (3)

- iv. Masukkan kembali angka **3** kemudian tekan tombol “Submit” seperti pada Gambar 19.



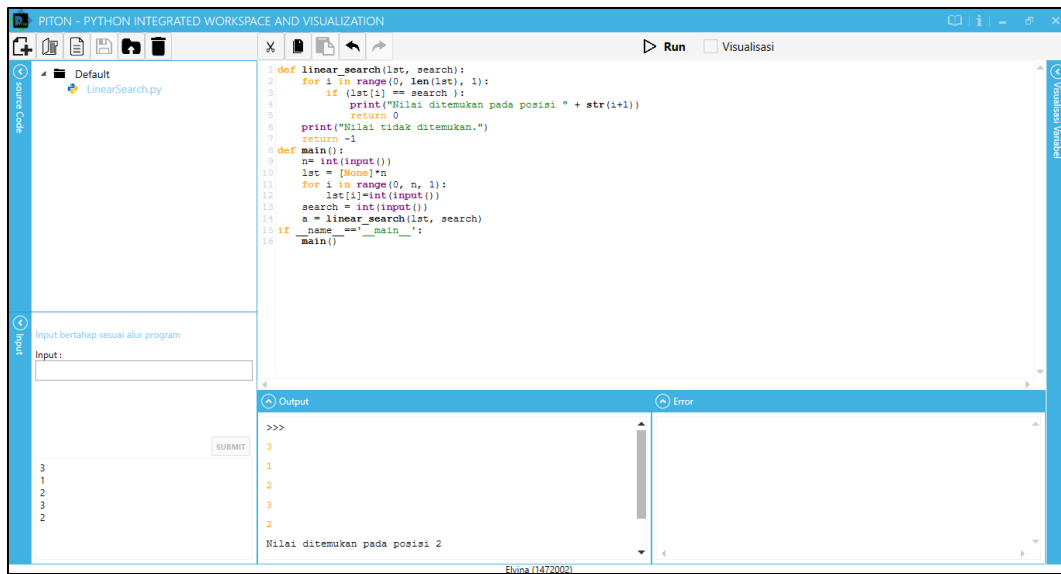
Gambar 19 Tampilan Input bertahap (4)

- v. Masukkan kembali angka 2 kemudian tekan tombol “Submit” seperti pada Gambar 20.



Gambar 20 Tampilan Input bertahap (5)

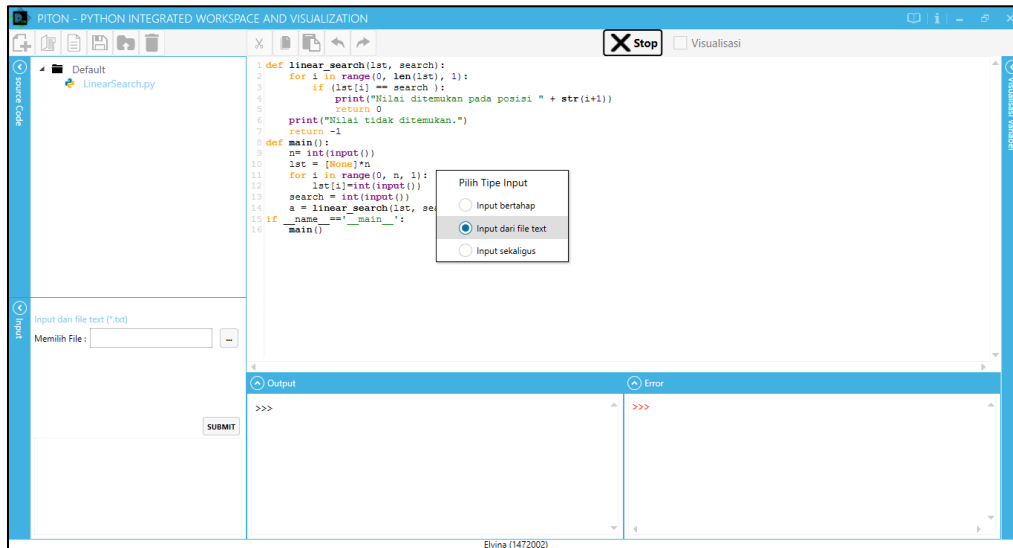
- vi. Aplikasi PTON akan mengeluarkan *output* hasil kode program pada kolom ekspansi output seperti pada Gambar 21.



Gambar 21 Tampilan Output

b. *Input dari File Text (*.txt)*

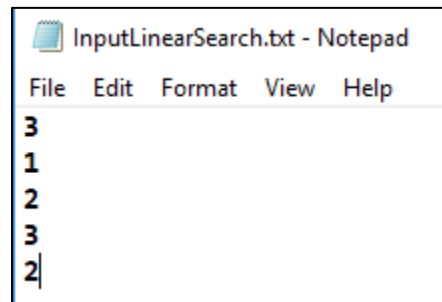
Memasukkan *input* untuk kode program melalui *file text* yang sudah dirancang seperti pada gambar 22. *File text* yang sama dapat digunakan berulang kali. Kelebihan “Enter” saat membuat *file text input*, akan membuat *input* untuk kode program dianggap tidak *valid*.



Gambar 22 Tampilan Input dari file text

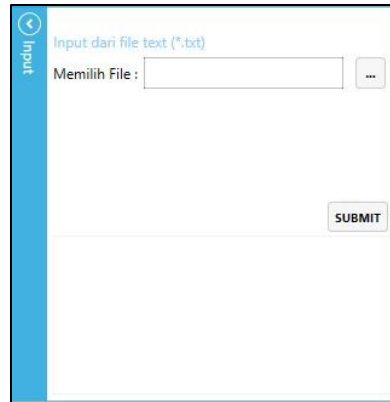
Tahapan penggunaan *input* dari *file text*, antara lain:

- i. Membuat *file text* (*.txt) dengan bantuan Notepad, berikan nama *file* tersebut dengan “InputLinearSearch”. File ini tersedia juga dalam folder Panduan pada CD program dan dapat disalin ke komputer. Isi file text dapat dilihat pada gambar 23.



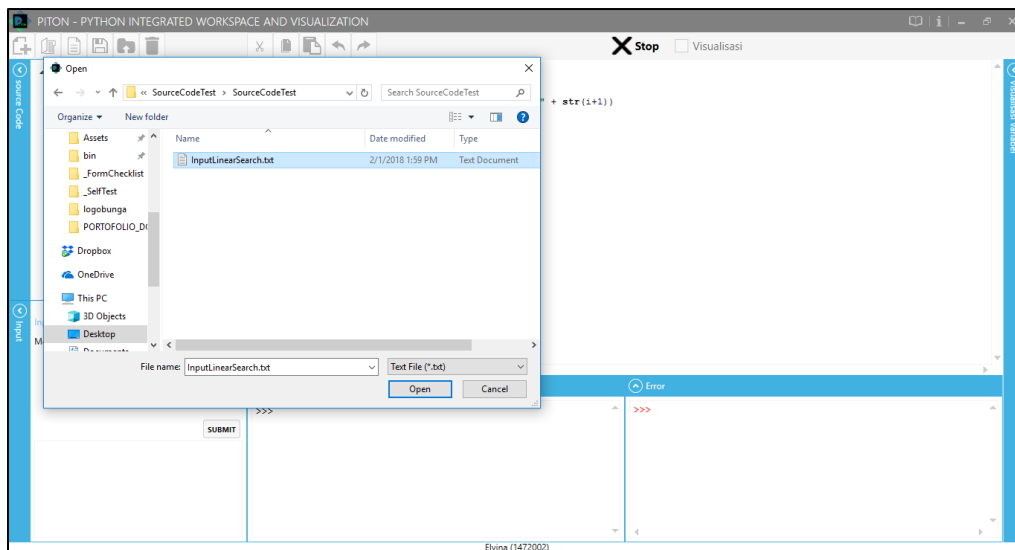
Gambar 23 Isi File text

- ii. Tekan tombol memilih “...” untuk memilih *file text* InputLinearSearch seperti pada gambar 24.



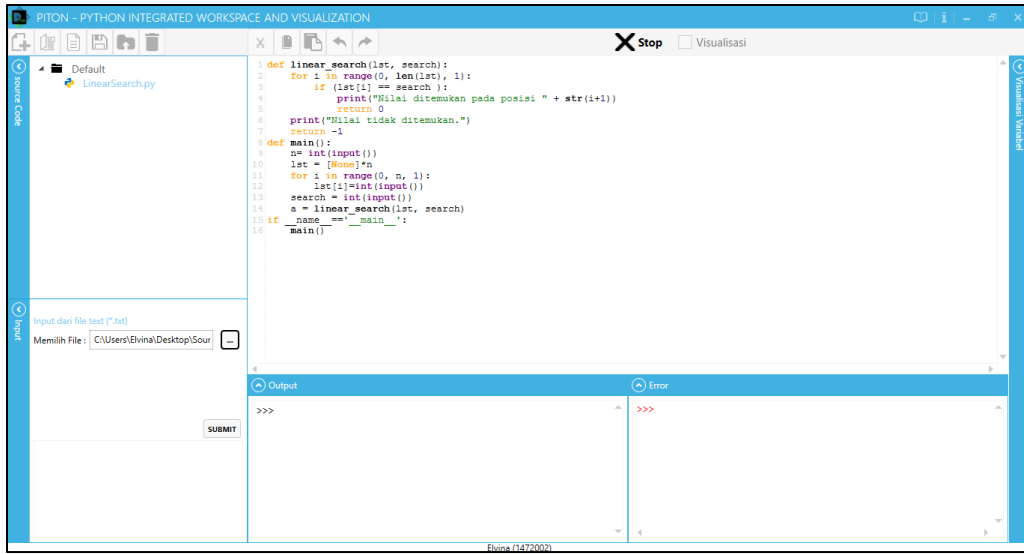
Gambar 24 Tampilan Pilih File text

- iii. Tekan *open* untuk memasukkan *file* InputLinearSearch seperti pada gambar 25.

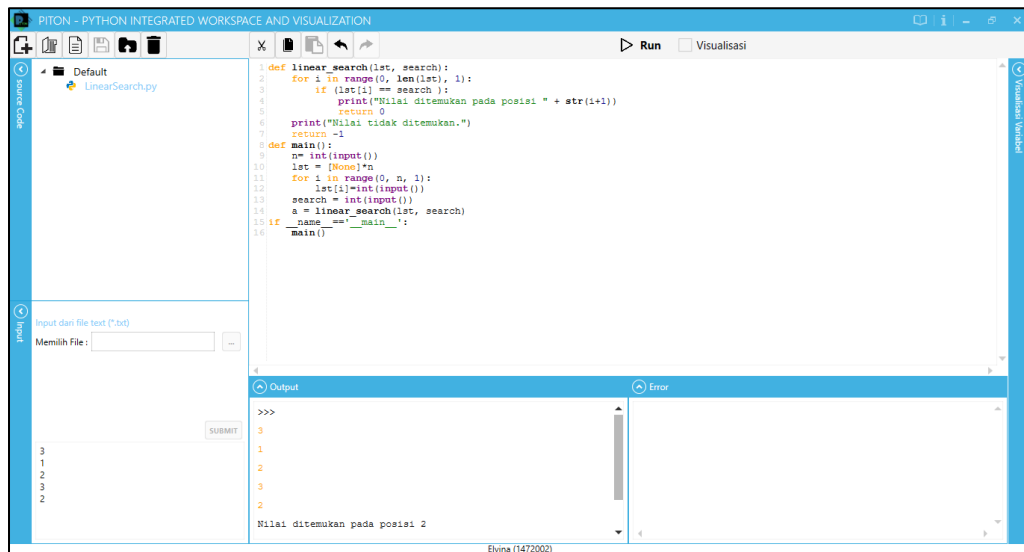


Gambar 25 Tampilan Open File text

- iv. Kemudian tekan tombol “submit” untuk melanjutkan kode program seperti pada gambar 26. Aplikasi PITON akan mengeluarkan *output* hasil kode program pada kolom ekspansi *output* seperti pada gambar 27.

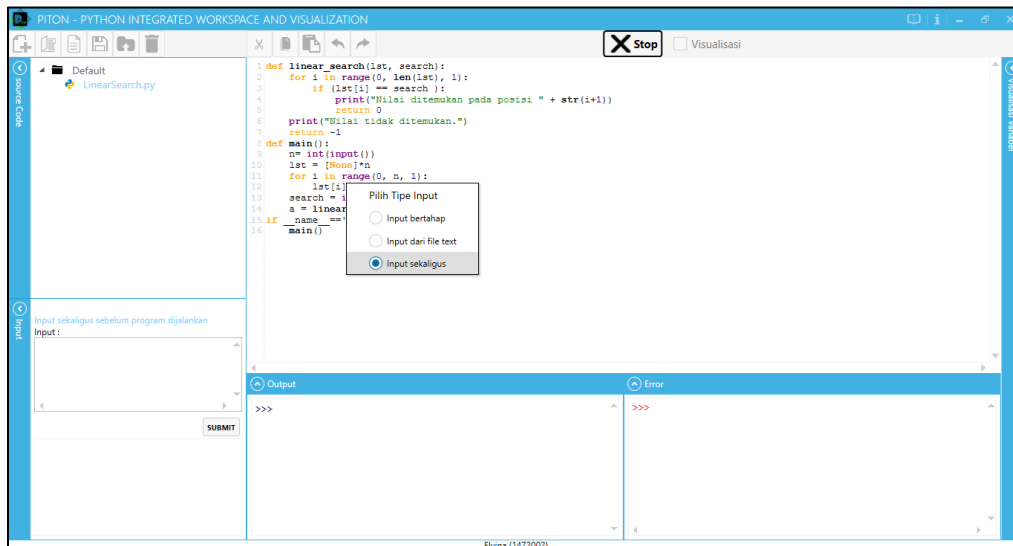


Gambar 26 Tampilan hasil Open File text



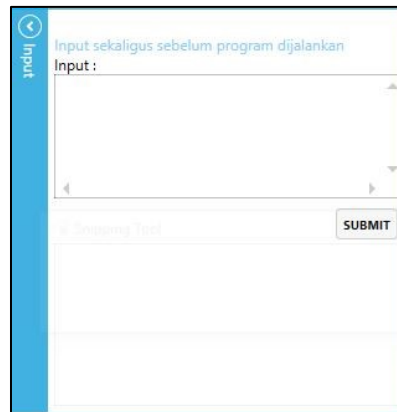
Gambar 27 Tampilan hasil output

c. *Input* sekaligus



Gambar 28 Tampilan Input sekaligus

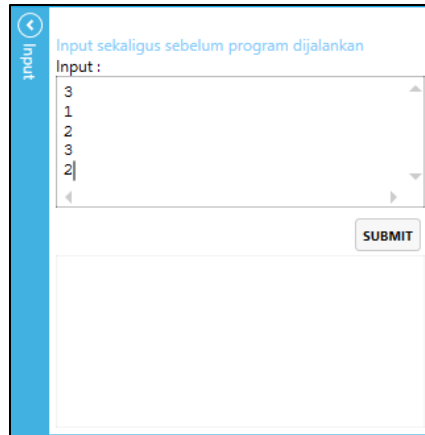
Memasukkan *input* untuk kode program dengan memasukkan seluruh *input* yang dibutuhkan kode program seperti pada gambar 28. *Input* dimasukkan dalam *text box* seperti pada gambar 29. Kelebihan “Enter” saat memasukkan *input* untuk kode program, membuat input untuk kode program dianggap tidak *valid*.



Gambar 29 Tampilan Input sekaligus

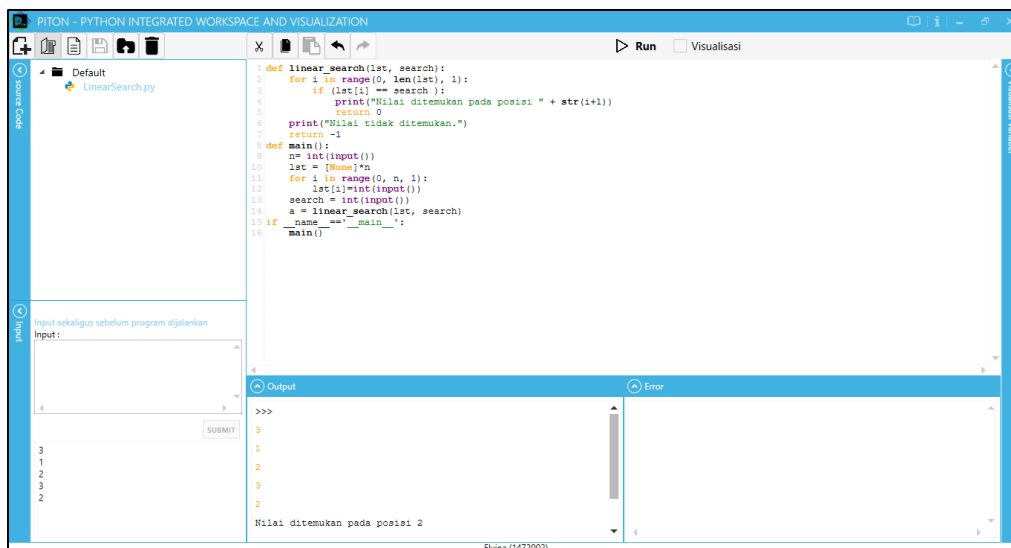
Tahapan penggunaan *input* sekaligus, sebagai berikut:

- i. Masukkan *input* seperti pada gambar 30.



Gambar 30 Tampilan setelah mengetik input sekaligus

- ii. Kemudian tekan tombol “submit” untuk melanjutkan kode program. Aplikasi PITON akan mengeluarkan *output* hasil kode program pada kolom ekspansi *output* seperti pada gambar 31.



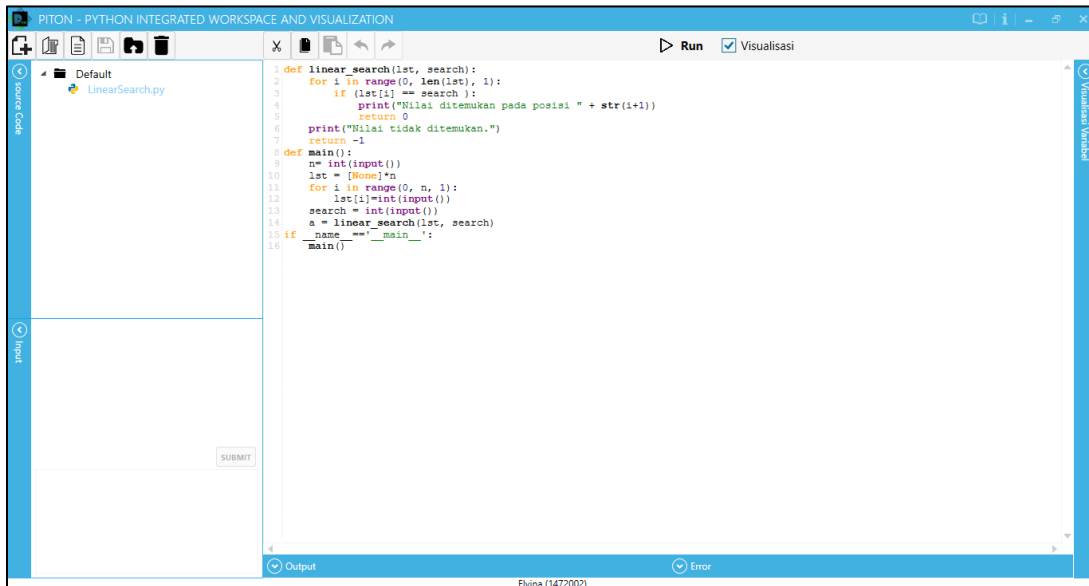
Gambar 31 Tampilan hasil output

III.2 Penggunaan PITON dengan fitur visualisasi

Penggunaan PITON dengan fitur visualisasi akan menampilkan panel visualisasi di sebelah kanan source code editor. Pada panel visualisasi akan ditunjukkan variabel yang dipakai oleh program dan juga perubahan nilai selama program berjalan. Fitur visualisasi juga akan menunjukkan posisi instruksi/perintah dalam program yang sedang dijalankan. Tahapan penggunaan fitur visualisasi adalah :

1. Berikan *checkbox* pada *checkbox* visualisasi, kemudian jalankan kode program dengan menekan tombol “Run” seperti pada gambar 32.

☐	Visualisasi	Checkbox Visualisasi
☒	Run	Run

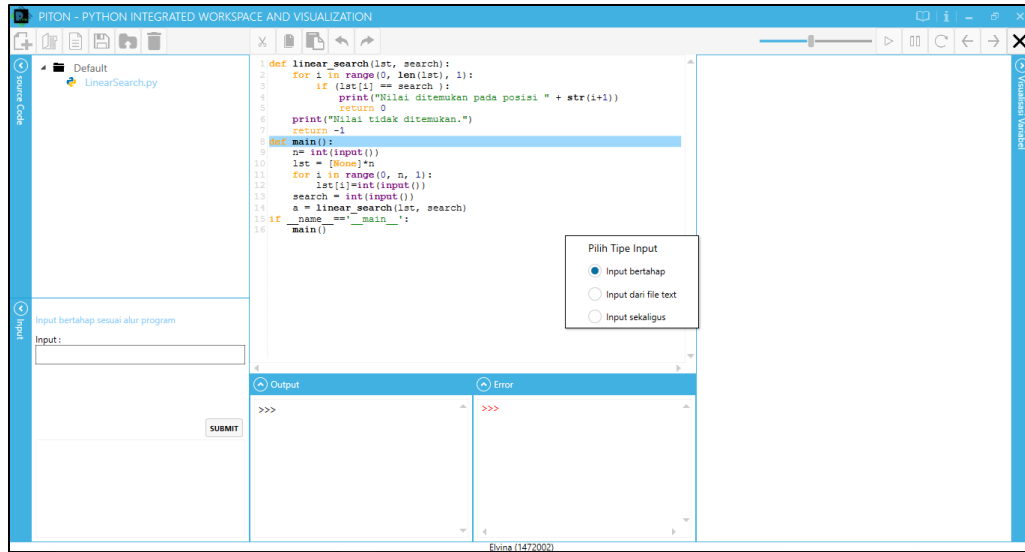


Gambar 32 Tampilan dengan fitur Visualisasi

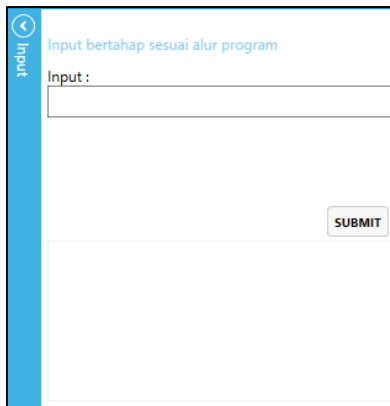
2. Pilihlah terlebih dahulu cara *input* yang diinginkan untuk memasukkan *input*. Terdapat 3 jenis cara *input* pada PTON, yaitu:

a. *Input Bertahap*

Memasukkan input untuk kode program secara bertahap seperti pada gambar 33. PTON akan meminta input kembali jika dibutuhkan setelah input yang dimasukkan sebelumnya berhasil diproses seperti pada gambar 34.



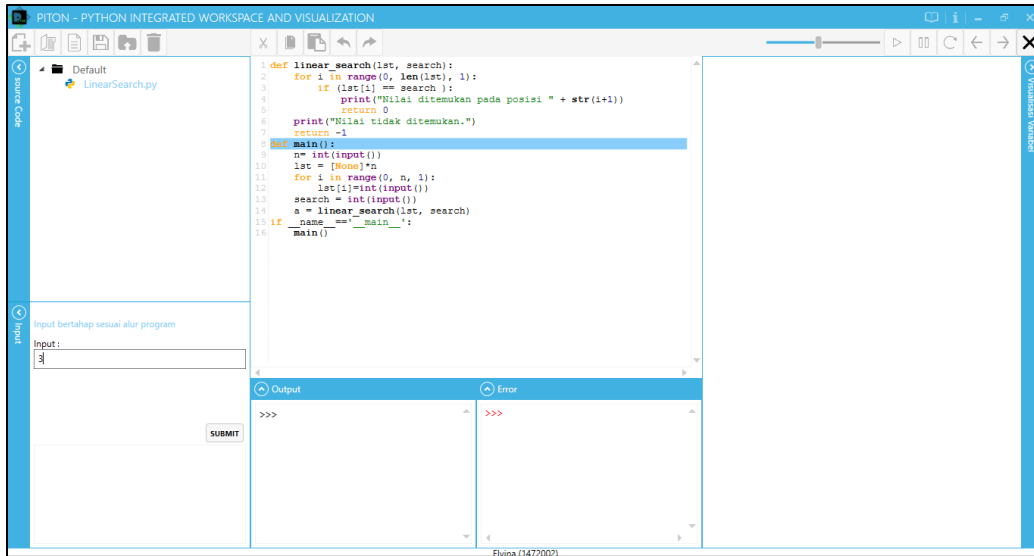
Gambar 33 Visualisasi dengan Input bertahap



Gambar 34 Tampilan input bertahap

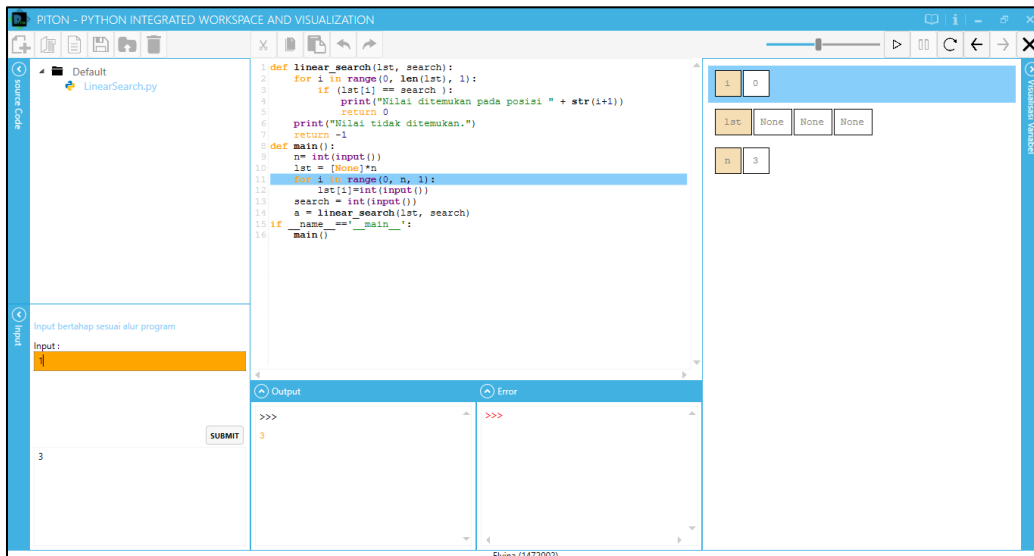
Tahapan penggunaan *input* bertahap, antara lain:

- i. Masukkan angka **3** kemudian tekan tombol “Submit” seperti pada Gambar 35.



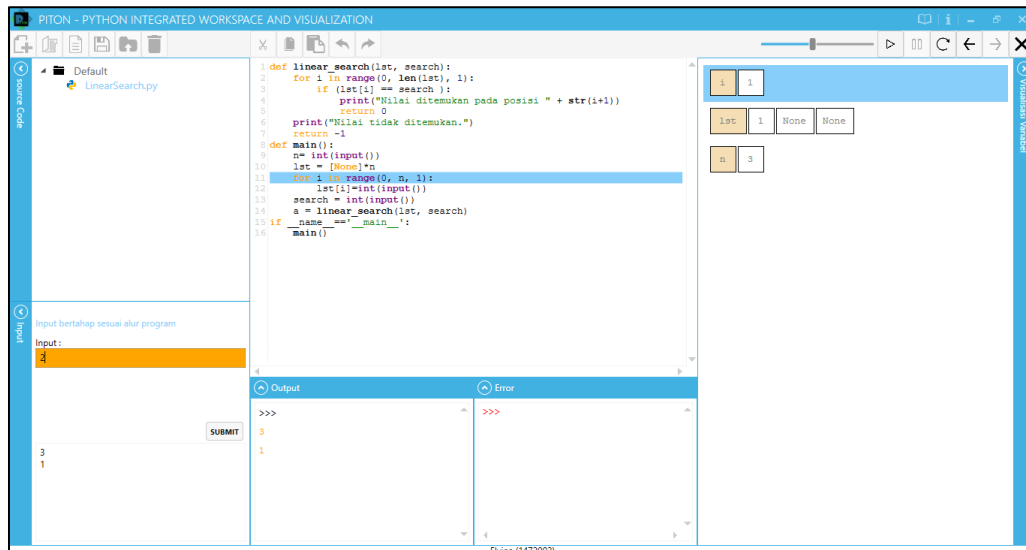
Gambar 35 Tampilan input bertahap (1)

- ii. Masukkan angka **1** kemudian tekan tombol “Submit” seperti pada Gambar 36.



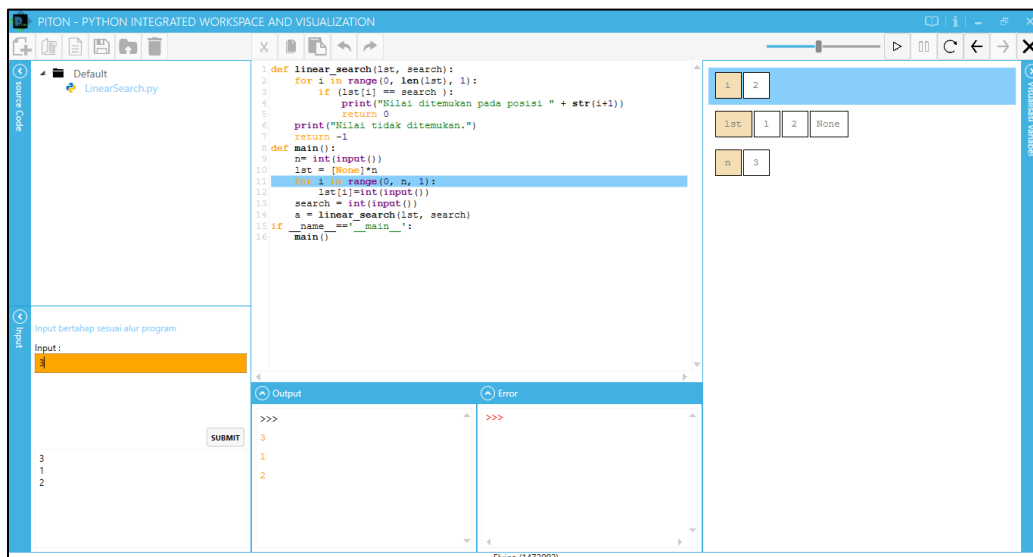
Gambar 36 Tampilan input bertahap (2)

- iii. Masukkan angka 2 kemudian tekan tombol “Submit” seperti pada Gambar 37.



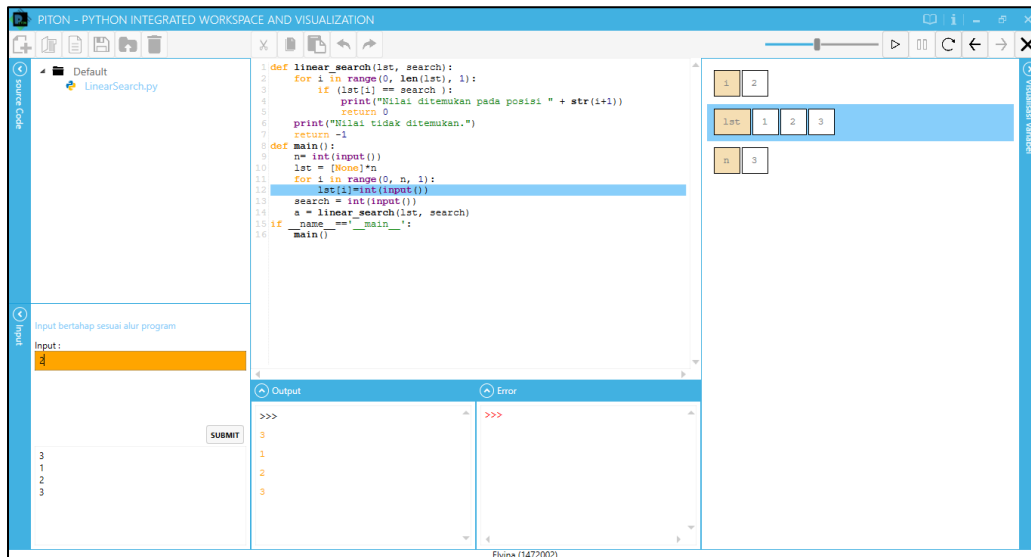
Gambar 37 Tampilan input bertahap (3)

- iv. Masukkan angka 3 kemudian tekan tombol “Submit” seperti pada Gambar 38.



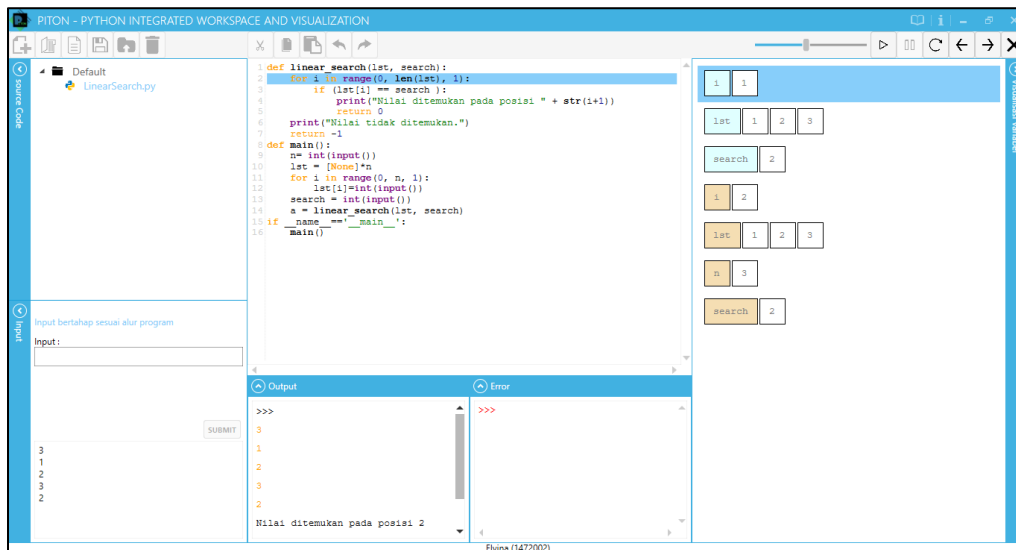
Gambar 38 Tampilan input bertahap (4)

- v. Masukkan angka **2** kemudian tekan tombol “Submit” seperti pada Gambar 39.



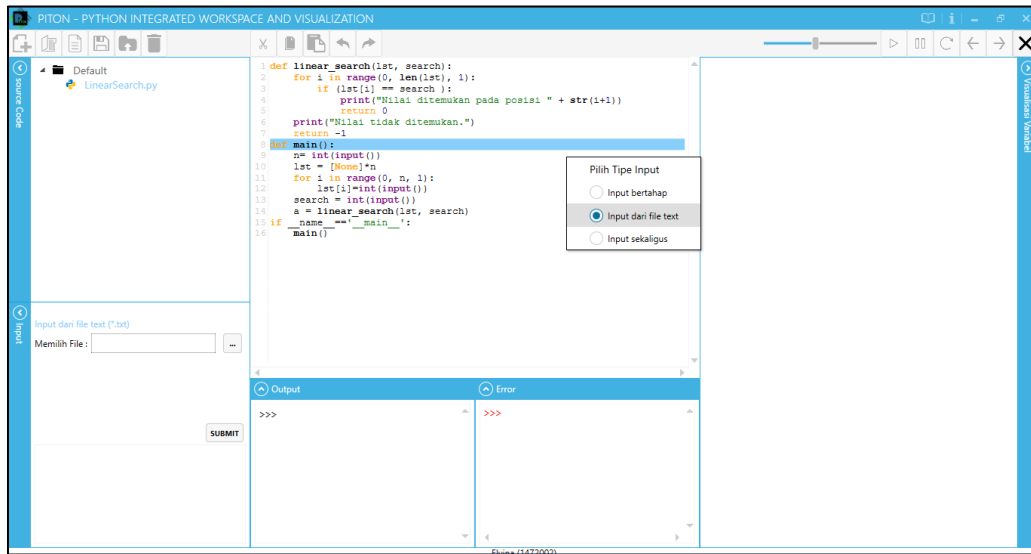
Gambar 39 Tampilan input bertahap (5)

- vi. Aplikasi PITON akan mengeluarkan *output* hasil kode program pada kolom ekspansi output dan menjalankan visualisasi pada panel visualisasi variabel seperti pada gambar 40.



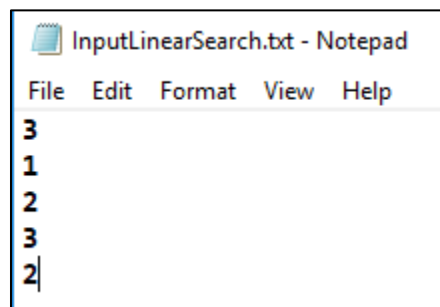
Gambar 40 Tampilan output (visualisasi)

b. *Input dari File Text (*.txt)*



Gambar 41 Visualisasi dengan Input dari file text

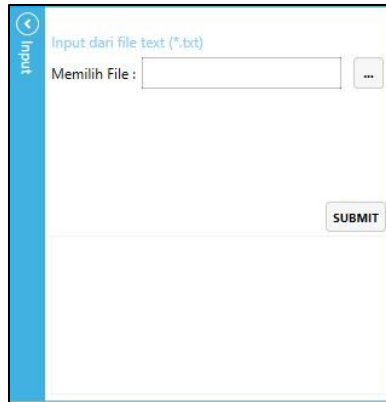
Memasukkan *input* untuk kode program melalui *file text* yang sudah dirancang seperti pada gambar 41. Untuk itu akan digunakan *file text* “InputLinearSearch” yang sama seperti pada III.1, seperti ditampilkan pada Gambar 42.



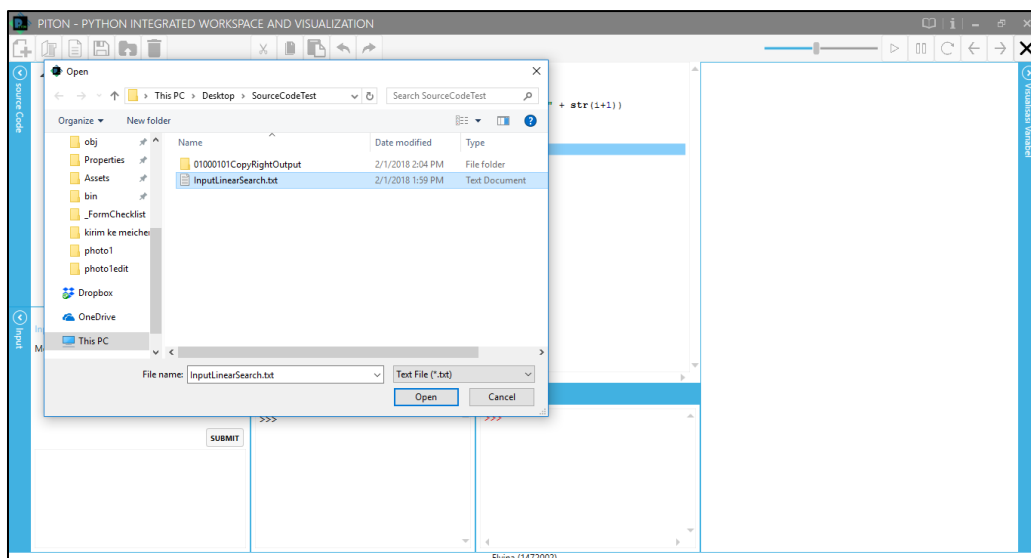
Gambar 42 Isi file text

Tahapan penggunaan *input* dari *file text*, antara lain:

- Setelah file “InputLinearSearch” disiapkan, tekan tombol memilih “...” seperti pada gambar 43 untuk memilih *file text* InputLinearSearch seperti pada gambar 44.

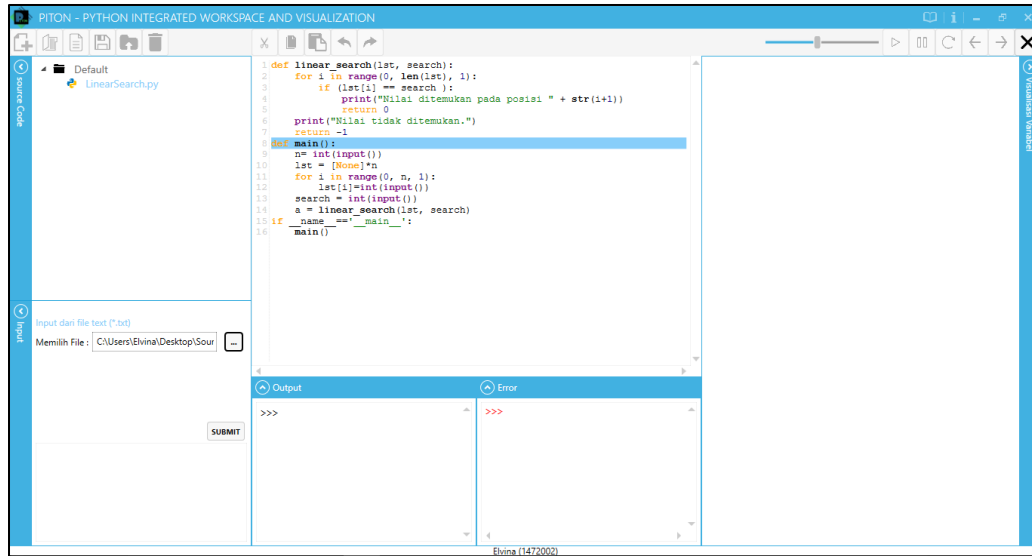


Gambar 43 Memilih file text



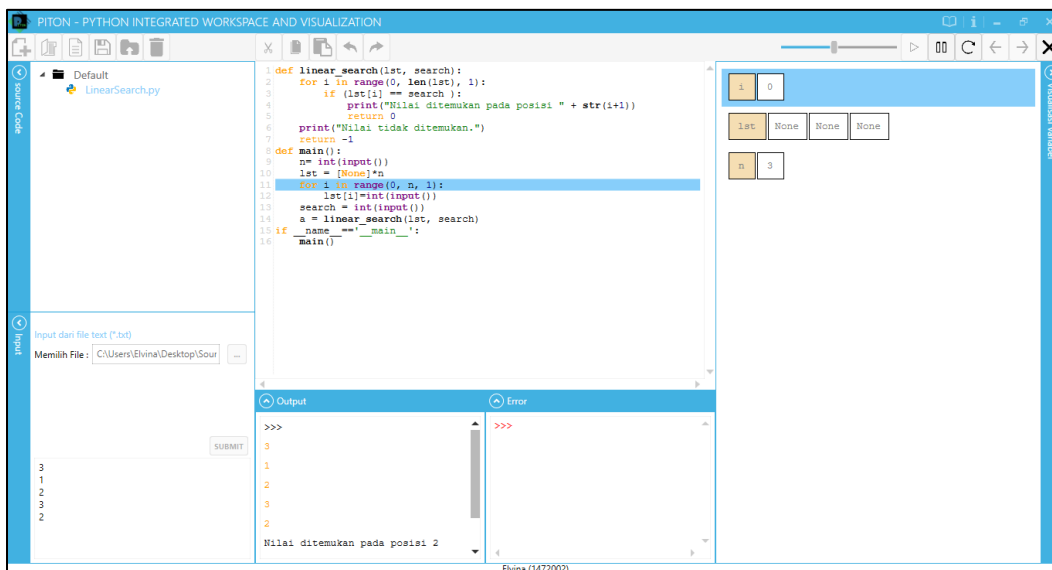
Gambar 44 Tampilan Open file text

- ii. Tekan *open* untuk memasukkan *file* InputLinearSearch, sehingga ditampilkan seperti pada gambar 45.



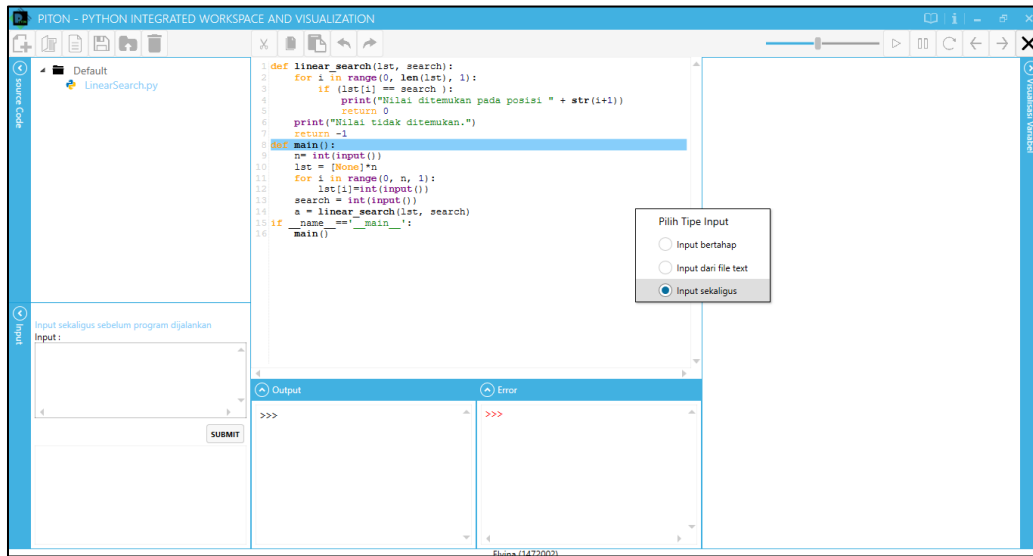
Gambar 45 Tampilan hasil memilih file text

- iii. Kemudian tekan tombol “submit” untuk melanjutkan kode program. Aplikasi PITON akan mengeluarkan *output* hasil kode program pada kolom ekspansi *output* dan menjalankan visualisasi pada kolom ekspansi visualisasi variabel seperti pada gambar 46.



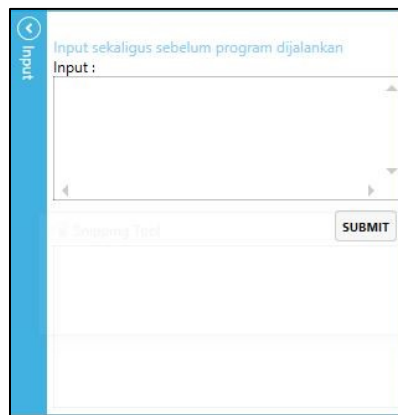
Gambar 46 Tampilan hasil output (visualisasi)

c. *Input* sekaligus



Gambar 47 Visualisasi dengan Input sekaligus

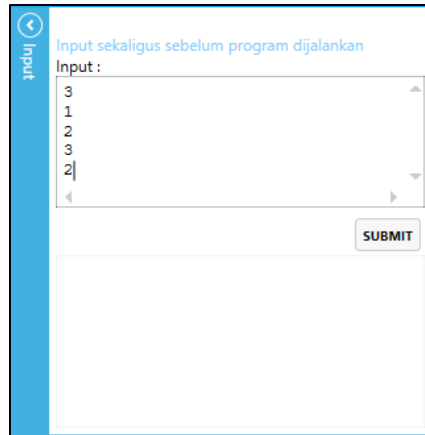
Memasukkan *input* untuk kode program dengan memasukkan seluruh *input* yang dibutuhkan kode program seperti pada gambar 47. Kelebihan “*Enter*” saat memasukkan *input* untuk kode program, membuat input untuk kode program dianggap tidak *valid*. Input sekaligus dimasukkan melalui bagian seperti pada gambar 48.



Gambar 48 Tampilan input sekaligus

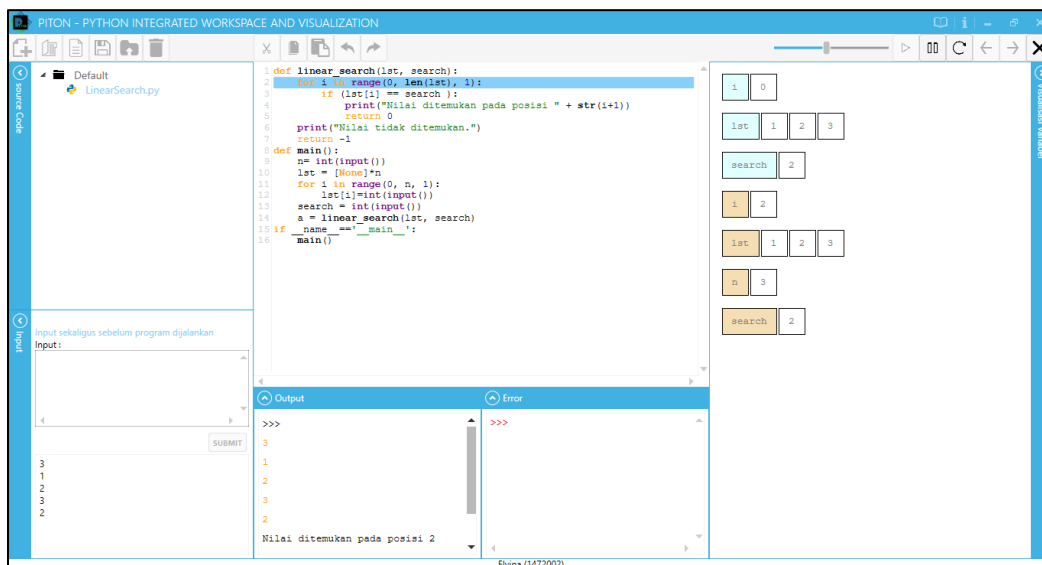
Tahapan penggunaan *input* sekaligus, antara lain:

- Masukkan *input* sebagai berikut seperti pada gambar 49:



Gambar 49 Tampilan setelah input sekaligus diketik

- ii. Kemudian tekan tombol “submit” untuk melanjutkan kode program. Aplikasi PITON akan mengeluarkan *output* hasil kode program pada kolom ekspansi *output* dan menjalankan visualisasi pada kolom ekspansi visualisasi variabel seperti pada gambar 50.



Gambar 50 Tampilan output (visualisasi)

3. Pengaturan Visualisasi kode program

Jika *checkbox* visualisasi diaktifkan sebelum menjalankan kode program, maka PITON akan menampilkan fitur tambahan pada *command toolbar* untuk pengaturan visualisasi seperti pada gambar 51. Fitur tambahan dapat digunakan untuk mengatur animasi pada saat visualisasi.



Gambar 51 Fitur tambahan untuk Visualisasi pada *Command toolbar*

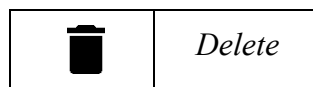
Cobalah fitur visualisasi dengan memilih simbol sesuai urutan berikut:

Simbol	Nama Simbol	Aksi
	<i>Pause</i>	Menunda visualisasi
	<i>Previous</i>	Memundurkan kondisi visualisasi ketahap sebelumnya
	<i>Next</i>	Memajukan kondisi visualisasi ketahap setelahnya
	<i>Reset</i>	Mengembalikan kondisi visualisasi ketahap awal
	<i>Speed Bar</i>	Mengatur kecepatan animasi visualisasi. Semakin ke kiri, semakin bertambah kecepatan visualisasi dan sebaliknya. Pengaturan <i>speed bar</i> hanya dapat dilakukan jika visualisasi berada pada kondisi tahap awal.
	<i>Play</i>	Menjalankan animasi untuk visualisasi
	<i>Stop</i>	Menghentikan visualisasi dan eksekusi

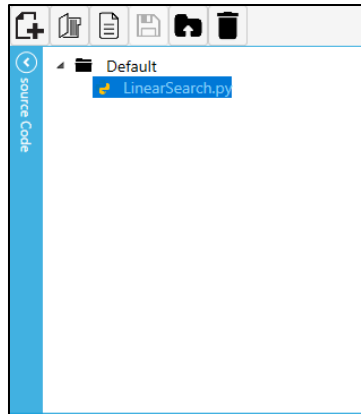
Beberapa fitur tambahan yang dapat digunakan, antara lain:

1. Menghapus *file* dari aplikasi PITON

Menghapus *file* dapat dilakukan dengan menggunakan fitur “delete”

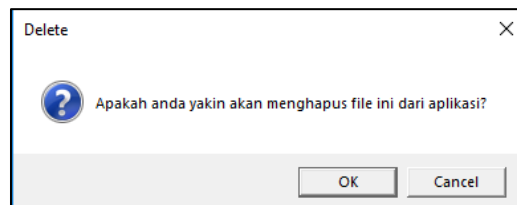


- a. Pilih *file* yang akan dihapus dari aplikasi PITON seperti pada Gambar 52.



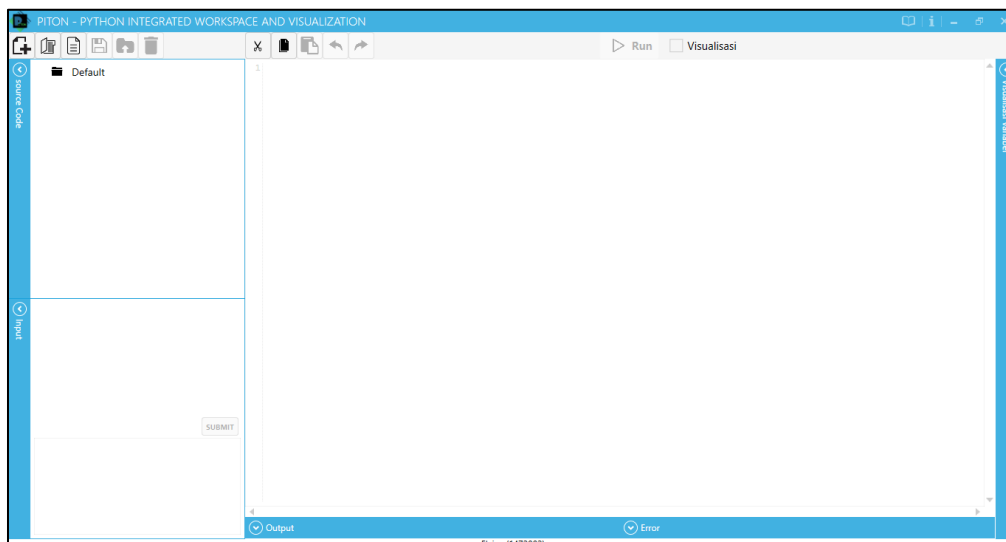
Gambar 52 Memilih file

- b. Tekan tombol “delete”, aplikasi akan memunculkan jendela *pop up* berupa pertanyaan seperti pada gambar 53. Tekan tombol “OK” untuk menghapus *file*. *File* tidak akan terhapus dari memori komputer, melainkan *file* hanya terhapus dari aplikasi PITON.



Gambar 53 Konfirmasi untuk menghapus

- c. *File* akan hilang dari tampilan aplikasi PITON seperti pada gambar 54.

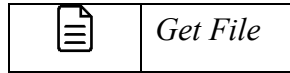


Gambar 54 Tampilan setelah file dihapus

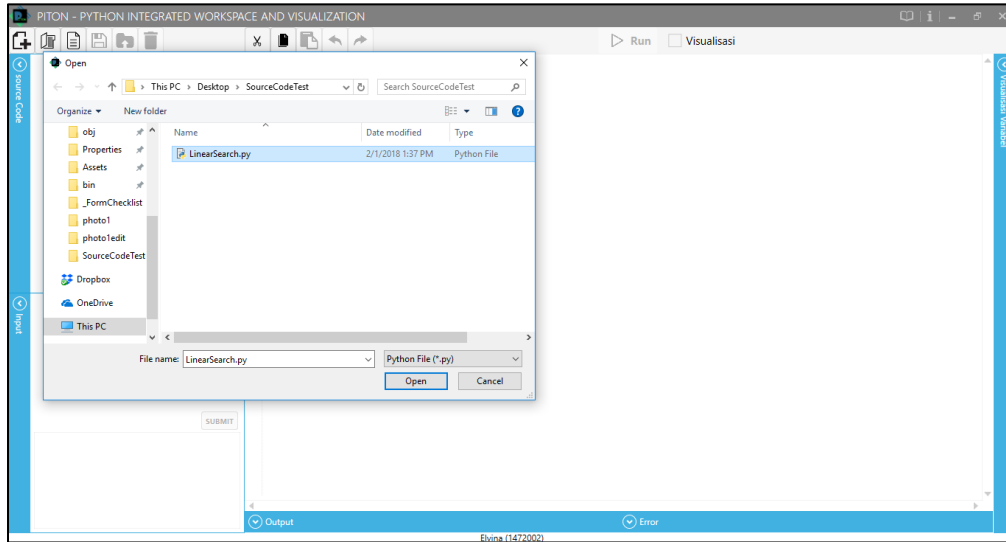
2. Membuka kembali kode program (*.py)

Membuka kode program dapat dilakukan dengan menggunakan fitur *Get File*.

- a. Tekan tombol “Get File”

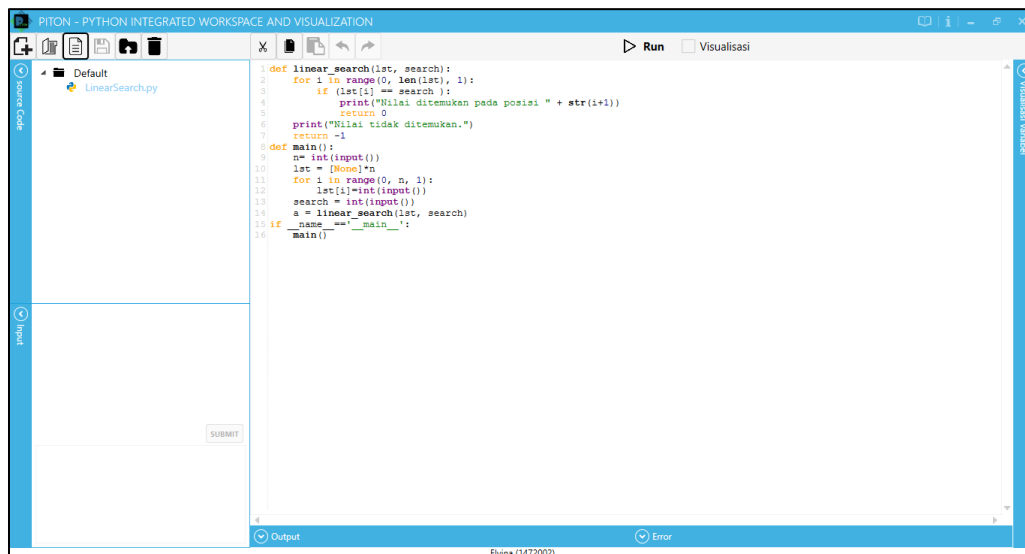


- b. Pilih *file* dengan ekstensi *.py yang akan dieksekusi maupun dimanipulasi.
Pada kasus ini pilihlah *file* “LinearSearch.py” seperti pada gambar 55.



Gambar 55 Tampilan memilih File untuk dibuka

- c. Tekan tombol “Open”, maka *file* LinearSearch akan muncul pada ekspansi *Source Code* seperti gambar 56.

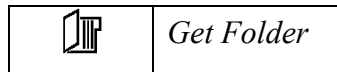


Gambar 56 Tampilan setelah Open file

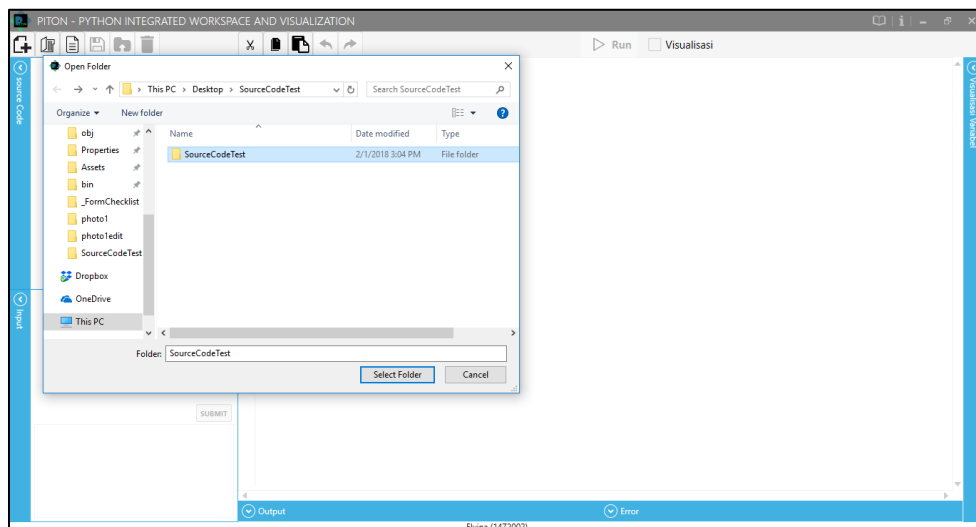
3. Membuka *folder* berisi beberapa kode program (*.py)

Membuka *folder* berisi beberapa kode program dapat dilakukan dengan menggunakan fitur *get folder*.

a. Tekan tombol “get folder”

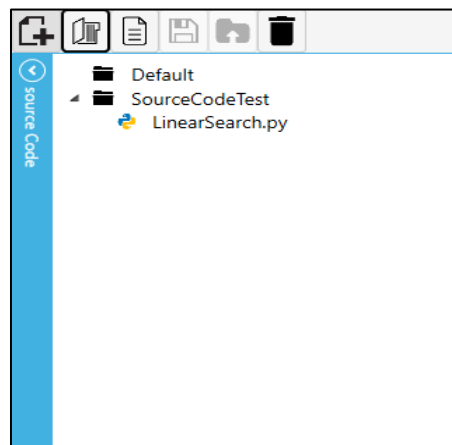


b. Pilih *folder* yang akan ditampilkan isi *file*-nya. Pilihlah *folder* yang berisi LinearSearch.py seperti pada gambar 57. Tekan “Select Folder” untuk menampilkan semua *file* dengan ekstensi (*.py) pada *folder* tersebut.



Gambar 57 Tampilan memilih folder

c. *File* akan ditampilkan sebagai *file* dari direktori *folder* yang dipilih seperti pada gambar 58.



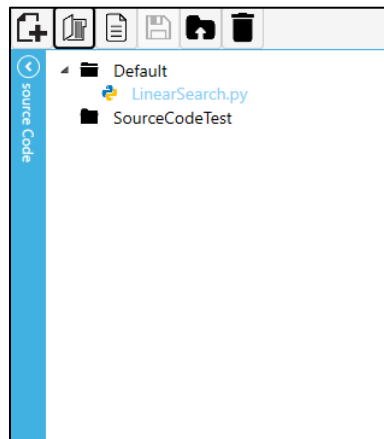
Gambar 58 Tampilan setelah membuka folder

- d. Jika *file* yang terdapat pada *folder* sudah ada pada aplikasi PITON, maka akan muncul jendela *pop up* yang menyatakan bahwa *file* tersebut sudah ada seperti pada gambar 59.



Gambar 59 Konfirmasi file sudah ada

- e. *Folder* yang dipilih tetap akan ditampilkan, namun *file* yang sudah ada tidak akan ditampilkan lagi seperti pada gambar 60.



Gambar 60 Tampilan Semula