

PRAKTIKUM 2

EKSPRESI FUNGSIONAL DASAR

Paradigma pemrograman fungsional adalah sebuah paradigma yang didasari oleh konsep pemetaan atau fungsi pada matematika. Fungsi dapat berbentuk fungsi dasar atau komposisi dari fungsi lainnya. Penyelesaian masalah dalam paradigma ini didasarkan atas aplikasi fungsi yang memetakan nilai awal (masukan) menjadi nilai akhir (keluaran) yang diharapkan.

Di dalam kuliah telah dipelajari paradigma pemrograman fungsional yang dituangkan dalam bentuk notasi fungsional. Notasi tersebut bebas dari bahasa pemrograman apapun dan tidak memiliki alat pemroses bahasa, sehingga belum bisa dieksekusi. Oleh karena itu, dalam praktikum ini notasi tersebut akan diterjemahkan ke dalam suatu bahasa pemrograman agar dapat dieksekusi dan diamati hasilnya. Adapun bahasa pemrograman yang digunakan untuk praktikum ini adalah bahasa Python.

Bahasa Python sendiri bukanlah bahasa pemrograman yang murni menganut paradigma fungsional. Akan tetapi, Python telah memiliki sejumlah fitur untuk mendukung implementasi pemrograman dalam paradigma fungsional.

1. Tujuan Praktikum

Pada praktikum kali ini mahasiswa akan berlatih menerjemahkan contoh kasus ekspresi dasar yang terdapat pada Diktat dari notasi fungsional ke bahasa pemrograman Python. Setelah praktikum ini, diharapkan mahasiswa mampu menyelesaikan persoalan yang mengaplikasikan fungsi dasar dalam bahasa pemrograman Python.

2. Tools

Tools yang dibutuhkan untuk melakukan praktikum ini adalah interpreter Python yang telah terinstal di komputer.

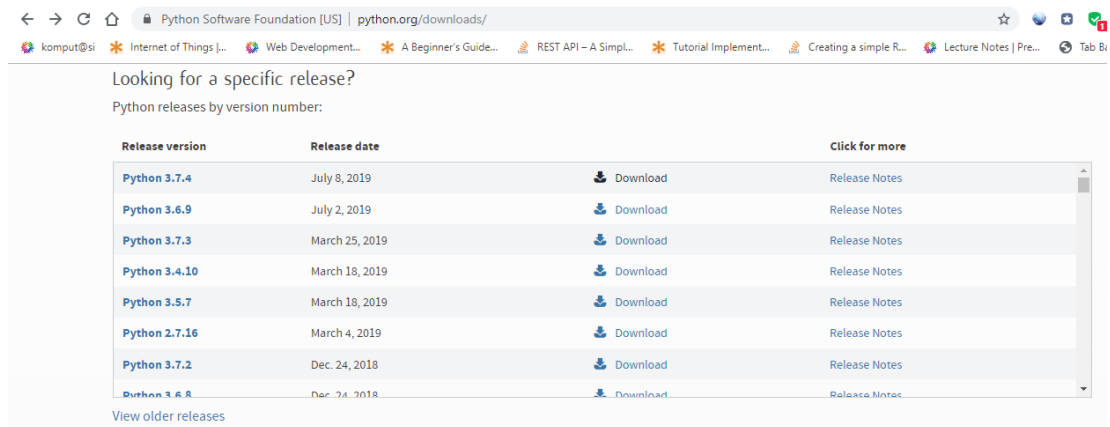
3. Materi Praktikum

Berikut ini dijelaskan langkah instalasi Python, cara menulis dan menjalankan program Python, serta contoh implementasi contoh kasus ke dalam bahasa Python. Contoh kasus dan latihan dalam praktikum ini mengacu pada Diktat Kuliah Dasar Pemrograman Bagian: Pemrograman Fungsional yang disusun oleh Inggriani Liem, Institut Teknologi Bandung.

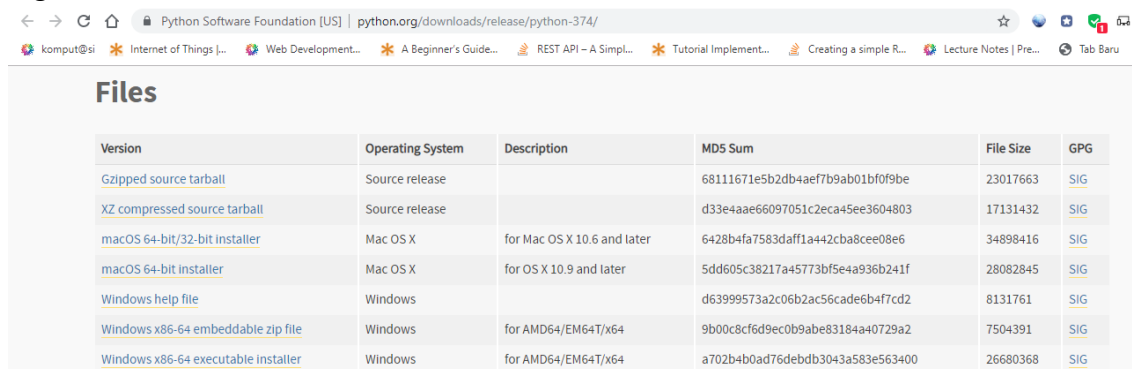
3.1. Instalasi Python

Instalasi Python dapat dilakukan dalam beberapa cara, salah satunya adalah melalui IDLE, langkah-langkahnya adalah sebagai berikut:

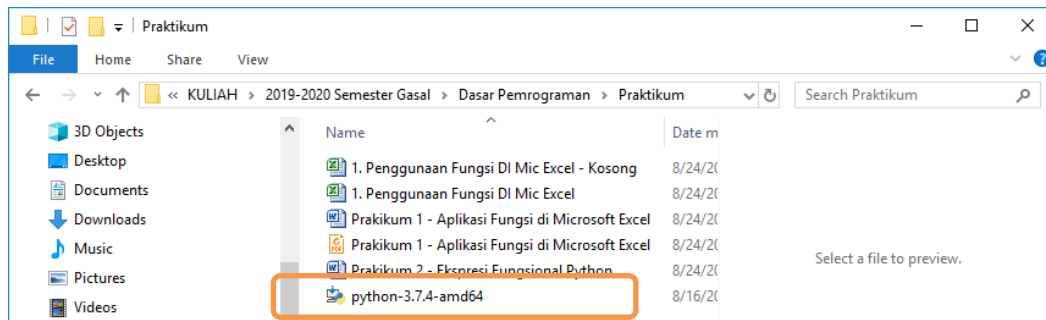
1. Unduh installer Python di <https://www.python.org/downloads/>. Pilih versi yang terbaru, yaitu Python 3.7.4.



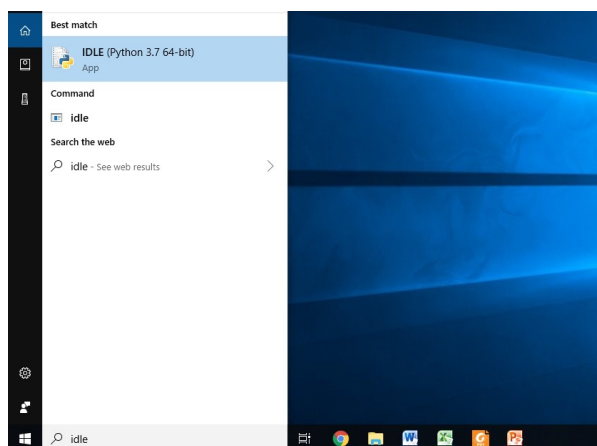
- Unduh versi installer yang sesuai dengan spesifikasi sistem operasi dan computer yang digunakan.



- Jalankan installer Python yang telah diunduh. Pilih “Install Now” dan ikuti petunjuknya saat instalasi.



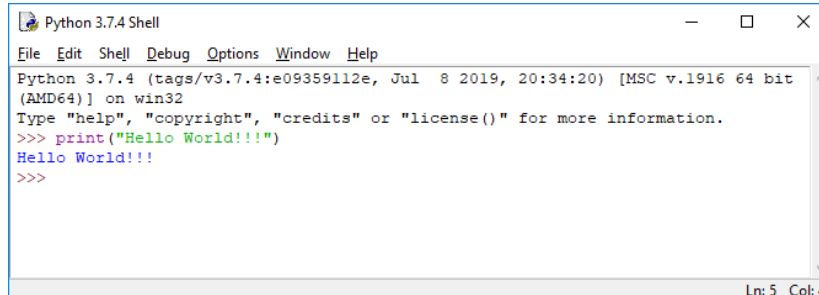
- Setelah diinstal maka aplikasi akan muncul pada Start Up menu.



3.2. Membuat dan menjalankan program Python melalui IDLE

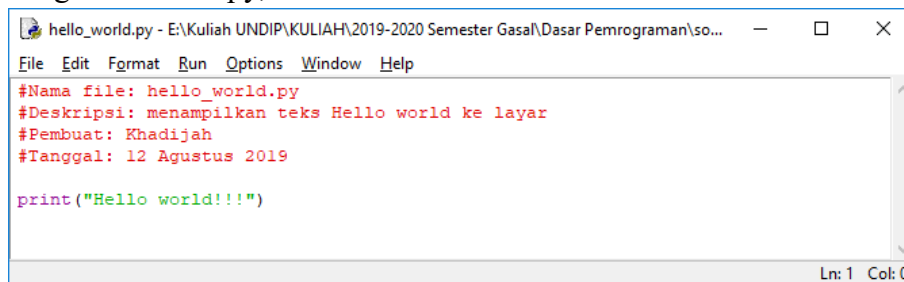
1. Buka aplikasi IDLE Python pada start up menu.
2. Jika berhasil akan muncul Python Shell seperti berikut. Anda dapat langsung menuliskan script pada Shell atau melalui script editor.

Contoh: ketikkan `print("Hello World!!!")` pada Shell, lalu tekan enter.



```
Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 20:34:20) [MSC v.1916 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> print("Hello World!!!")
Hello World!!!
>>>
```

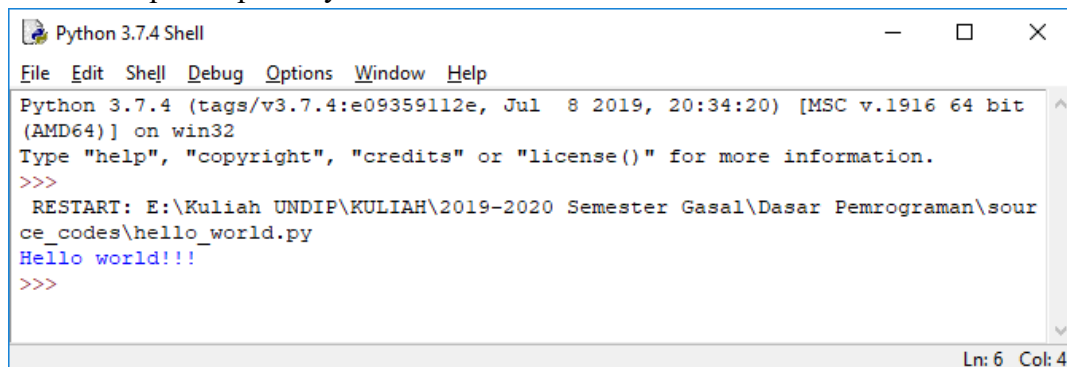
3. Untuk memulai menulis script melalui editor, buka menu File → New File. Tuliskan script-nya, lalu simpan dengan cara klik menu File → Save As, lalu tuliskan nama filenya dengan ekstensi .py, lalu klik OK.



```
hello_world.py - E:\KuliaH\UNDIP\KULIAH\2019-2020 Semester Gasal\Dasar Pemrograman\so...
File Edit Format Run Options Window Help
#Nama file: hello_world.py
#Deskripsi: menampilkan teks Hello world ke layar
#Pembuat: Khadijah
#Tanggal: 12 Agustus 2019

print("Hello world!!!")
```

4. Untuk menjalankan script, klik Run → Run Module atau tekan F5 dari keyboard. Hasilnya akan ditampilkan pada Python Shell.



```
Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 20:34:20) [MSC v.1916 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
RESTART: E:\KuliaH\UNDIP\KULIAH\2019-2020 Semester Gasal\Dasar Pemrograman\sour
ce_codes\hello_world.py
Hello world!!!
>>>
```

3.3. Ekspresi

Program fungsional dituliskan dalam bentuk ekspresi. Ekspresi adalah sebuah teks yang terdiri dari: nama, simbol, operator, fungsi, (), yang dapat menghasilkan suatu nilai berkat evaluasi dari ekspresi. Secara sederhana ekspresi terdiri atas operator dan operan.

Notasi Fungsional	Notasi Python
<u>Ekspresi</u>	
<operator> (<operan>, <operan>)	<operator> (<operan>, <operan>)

3.4. Tipe Data

Suatu operan ataupun nilai yang dihasilkan dari evaluasi ekspresi memiliki tipe data tertentu. Berikut adalah perbandingan antara tipe data yang dikenal dalam notasi fungsional dengan tipe data di Python.

Tipe Data dalam Notasi Fungsional	Tipe Data di Python	Keterangan
<u>integer</u>	integer	Bilangan bulat rasional, contoh: -32, 0, 100
<u>real</u>	float	Bilangan real rasional, contoh: -3.12, 0.00, 1.7
<u>character</u>	str	Teks berisi 1 karakter, contoh: "A", "1"
<u>string</u>		Teks berisi lebih dari 1 karakter, contoh: "Python"
<u>boolean</u>	bool	Nilai kebenaran, terdiri atas: True atau False

Dalam bahasa pemrograman Python, tipe data yang digunakan pada operan atau hasil evaluasi ekspresi tidak perlu didefinisikan secara eksplisit.

3.5. Operator

Dalam pemrograman fungsional operator disebut juga sebagai fungsi dan operan disebut juga sebagai parameter atau argumen. Parameter yang digunakan dalam definisi fungsi disebut sebagai parameter formal. Saat fungsi tersebut diaplikasikan, maka parameter formal akan digantikan dengan sebuah nilai yang bersesuaian tipe datanya. Nilai tersebut disebut sebagai parameter aktual.

Dalam Python tersedia berbagai jenis atau operator dasar yang dapat langsung digunakan, seperti operator aritmatika, operator relasional dan operator Boolean. Operator tersebut disebut juga sebagai fungsi dasar atau primitif. Sebuah ekspresi yang menggunakan operator berupa fungsi dasar yang telah disediakan oleh sistem, disebut sebagai ekspresi dasar. Berikut ini adalah beberapa operator dasar yang ada di Python.

Operator aritmatika

Operator aritmatika melibatkan operan bertipe numeric dan hasilnya juga bertipe numeric.

Notasi Fungsional	Notasi Python	Keterangan	Contoh di Python
+	+	Menjumlahkan nilai dari dua buah operan	>> 5 + 7
-	-	Mengurangi nilai operan pertama dengan nilai operan kedua	>> 8 - 3
*	*	Mengalikan nilai dari dua buah operan	>> 6 * 6
/	/	Membagi nilai operan pertama dengan nilai operan kedua	>> 4 / 5

div	//	Membagi nilai operan pertama dengan nilai operan kedua, hasilnya bilangan bulat.	>> 4 // 5
mod	%	Mengembalikan sisa pembagian operator pertama dengan operator kedua	>> 5 % 4

Operator relasional

Operator relasional melibatkan operan bertipe numeric atau teks, dan hasilnya bertipe Boolean.

Notasi Fungsional	Notasi Python	Keterangan	Contoh di Python
<	<	True, jika nilai operan pertama kurang dari nilai operan kedua, False jika sebaliknya.	>> 7 < 5
>	>	True, jika nilai operan pertama lebih dari nilai operan kedua, False jika sebaliknya.	>> 10 > 2
=	==	True, jika nilai operan pertama sama dengan nilai operan kedua, False jika sebaliknya.	>> "A" == "a"
≤	<=	True, jika nilai operan pertama kurang dari atau sama dengan nilai operan kedua, False jika sebaliknya.	>> 12 <= 2
≥	>=	True, jika nilai operan pertama lebih dari atau sama dengan nilai operan kedua, False jika sebaliknya.	>> 12 >= 3
≠	!=	True, jika nilai operan pertama tidak sama dengan nilai operan kedua, False jika sebaliknya.	>> "1" != 1

Operator Boolean

Operator Boolean melibatkan operan bertipe Boolean dan asilnya adalah sebuah Boolean.

Notasi Fungsional	Notasi Python	Keterangan	Contoh di Python
<u>and</u>	and	True, jika operan pertama dan kedua sama-sama bernilai True.	>> (7 < 5) and ("A" == "a")
<u>or</u>	or	True, jika minimal salah satu operan bernilai True.	>> False or True

3.6. Pendefinisian Fungsi

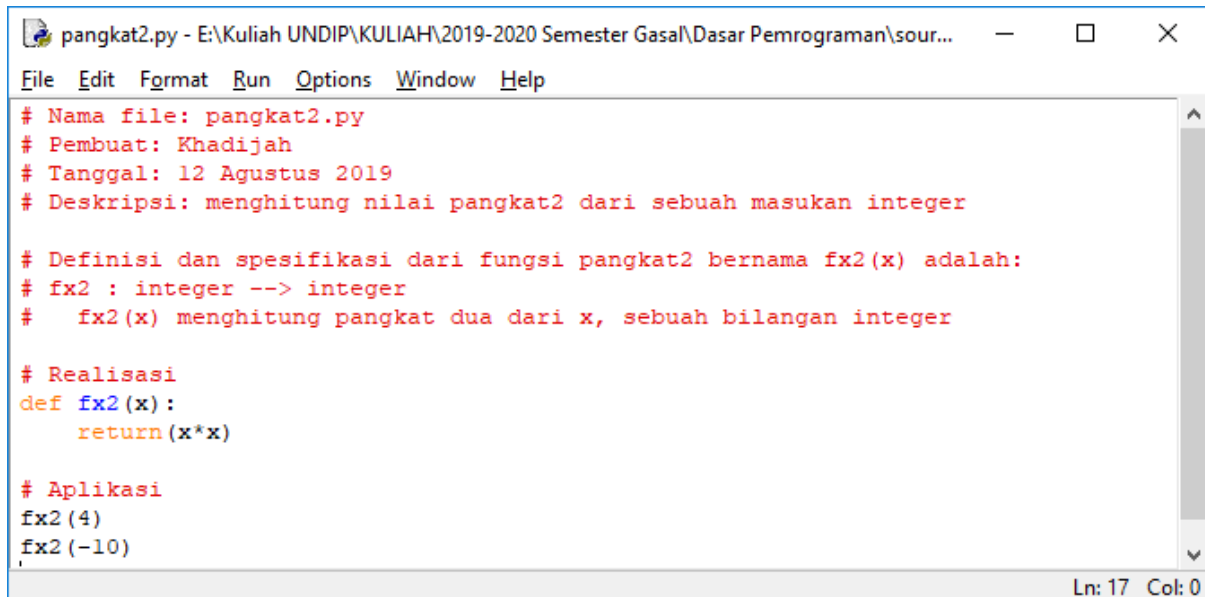
Fungsi yang digunakan dalam pemrograman fungsional dapat berupa fungsi dasar yang telah tersedia pada system, atau berupa fungsi yang didefinisikan sendiri yang didalamnya menggunakan fungsi dasar. Berikut ini adalah perbandingan antara pendefinisian fungsi dalam notasi fungsional dengan pendefinisian fungsi dalam Python.

Notasi Fungsional	Notasi Python
<u>Pendefinisian fungsi</u> <code><nama-fungsi> (<parameter-formal>):</code> <code><badan-fungsi></code>	<code>def <nama-fungsi> (<parameter-formal>):</code> <code>return <badan-fungsi></code>

3.7. Aturan Standar Penulisan File Program

Setiap file yang dibuat diberikan keterangan yang mencakup nama file, pembuat, tanggal pembuatan, deskripsi. Keempat hal tersebut dituliskan sebagai komentar. Dalam bahasa pemrograman komentar tidak akan dieksekusi. Komentar dalam Python ditulis dengan diawali tanda #.

Setiap program fungsional keterangan sesuai aturan dalam notasi fungsional yang mencakup: header, definisi dan spesifikasi, realisasi dan aplikasi.



```

# Nama file: pangkat2.py
# Pembuat: Khadijah
# Tanggal: 12 Agustus 2019
# Deskripsi: menghitung nilai pangkat2 dari sebuah masukan integer

# Definisi dan spesifikasi dari fungsi pangkat2 bernama fx2(x) adalah:
# fx2 : integer --> integer
#   fx2(x) menghitung pangkat dua dari x, sebuah bilangan integer

# Realisasi
def fx2(x):
    return (x*x)

# Aplikasi
fx2(4)
fx2(-10)

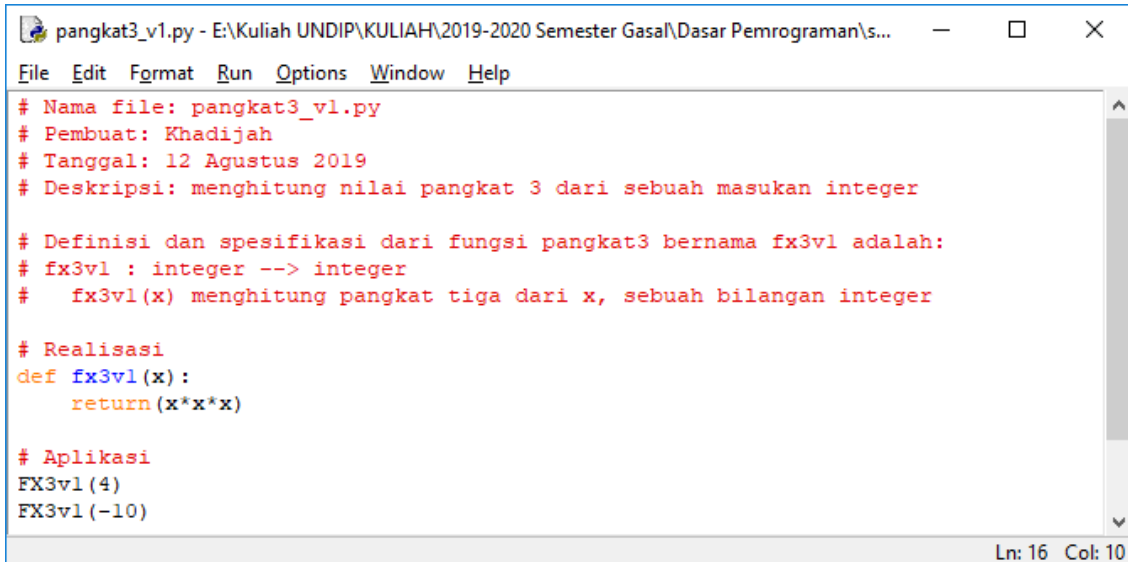
```

3.8. Contoh 1: Ekspresi Numerik PANGKAT2

Realisasi ekspresi numeric PANGKAT2 pada Diktat dapat dilihat pada contoh sebelumnya.

Perhatikan bahwa nama fungsi saat implementasi dalam bahasa pemrograman dapat berbeda dengan nama fungsi pada saat perancangan dalam notasi fungsional. Penamaan fungsi dapat menggunakan rekomendasi case pada bahasa pemrograman yang digunakan. Pada bahasa Python, *case style* untuk penamaan fungsi adalah *snaked_case*.

3.9. Contoh 2: Ekspresi Numerik PANGKAT3v1



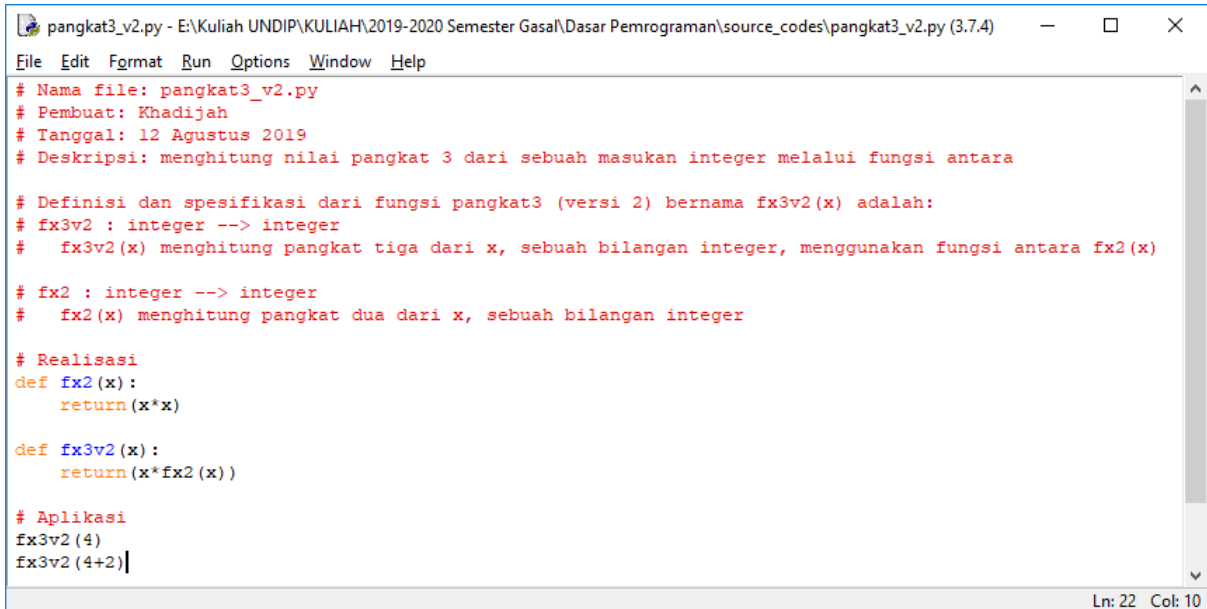
```
pangkat3_v1.py - E:\Kuliah UNIP\KULIAH\2019-2020 Semester Gasal\Dasar Pemrograman\s...
File Edit Format Run Options Window Help
# Nama file: pangkat3_v1.py
# Pembuat: Khadijah
# Tanggal: 12 Agustus 2019
# Deskripsi: menghitung nilai pangkat 3 dari sebuah masukan integer

# Definisi dan spesifikasi dari fungsi pangkat3 bernama fx3v1 adalah:
# fx3v1 : integer --> integer
#   fx3v1(x) menghitung pangkat tiga dari x, sebuah bilangan integer

# Realisasi
def fx3v1(x):
    return(x*x*x)

# Aplikasi
FX3v1(4)
FX3v1(-10)
Ln: 16 Col: 10
```

3.10. Contoh 2: Ekspresi Numerik PANGKAT3 v2



```
pangkat3_v2.py - E:\Kuliah UNIP\KULIAH\2019-2020 Semester Gasal\Dasar Pemrograman\source_codes\pangkat3_v2.py (3.7.4)
File Edit Format Run Options Window Help
# Nama file: pangkat3_v2.py
# Pembuat: Khadijah
# Tanggal: 12 Agustus 2019
# Deskripsi: menghitung nilai pangkat 3 dari sebuah masukan integer melalui fungsi antara

# Definisi dan spesifikasi dari fungsi pangkat3 (versi 2) bernama fx3v2(x) adalah:
# fx3v2 : integer --> integer
#   fx3v2(x) menghitung pangkat tiga dari x, sebuah bilangan integer, menggunakan fungsi antara fx2(x)

# fx2 : integer --> integer
#   fx2(x) menghitung pangkat dua dari x, sebuah bilangan integer

# Realisasi
def fx2(x):
    return(x*x)

def fx3v2(x):
    return(x*fx2(x))

# Aplikasi
fx3v2(4)
fx3v2(4+2)
Ln: 22 Col: 10
```

3.11. Contoh 3: Ekspresi dengan analisa bottom up: realisasi MAX3, jika dipunyai MAX2

```
max3.py - E:/Kuliah UNDIP/KULIAH/2019-2020 Semester Gasal/Dasar Pemrograman/source_codes/max3.py (3.7.4)
File Edit Format Run Options Window Help
# Nama file: max3.py
# Pembuat: Khadijah
# Tanggal: 12 Agustus 2019
# Deskripsi: menentukan nilai maksimum dari 3 bilangan integer

# Definisi dan spesifikasi dari fungsi max3 bernama max3(a,b,c) adalah:
# max3 : 3 integer --> integer
#   max3(a,b,c) menentukan nilai maksimum dari 3 bilangan integer a, b dan c, menggunakan fungsi antara max2(a,b)

# max2 : 2 integer --> integer
#   max2(a,b) menentukan nilai maksimum dari 2 bilangan integer a dan b

# Realisasi
def max2(a,b):
    return ((a+b) + abs(a-b)) // 2

def max3(a,b,c):
    return max2(max2(a,b),c)

# Aplikasi
max3(4,9,-10)
max3(100,-20,300)
|
```

3.12. Contoh 7 Ekspresi boolean: APAKAH ORIGIN

```
is_orogin.py - E:/Kuliah UNDIP/KULIAH/2019-2020 Semester Gasal/Dasar Pemrograman/source_codes/is_orogin.py (3.7.4)
File Edit Format Run Options Window Help
# Nama file: is_orogin.py
# Pembuat: Khadijah
# Tanggal: 12 Agustus 2019
# Deskripsi: mengecek apakah sebuah dua buah nilai integer (x,y) mewakili titik origin (0,0)

# Definisi dan spesifikasi dari fungsi apakah origin bernama is_origin(x,y) adalah:
# is_origin : 2 real --> boolean
#   is_origin(x,y) benar jika (x,y) adalah dua nilai yang mewakili titik origin (0,0)

# Realisasi
def is_origin(x,y):
    return (x == 0) and (y == 0)

# Aplikasi
is_origin(0,0)
is_origin(4.2,0.0)
```

4. Latihan

Terjemakanlah contoh notasi fungsional pada Diktat berikut ke dalam bahasa pemrograman Python (perhatikan aturan standar penulisan program):

1. Contoh-4 Ekspresi numeric: Mean Olympique (MO)
2. Contoh-5 Ekspresi Boolean: POSITIF
3. Contoh-6 Ekspresi Boolean: APAKAH HURUF A
4. Contoh-7 Ekspresi Boolean: APAKAH VALID
5. Ekspresi numeric: Least Square (Jarak 2 Titik)

===== Selamat Mengerjakan =====