

MODUL PRAKTIKUM

ALGORITMA DAN PEMROGRAMAN 2



Versi	3.1
Tahun Penyusunan	2012
Tim Penyusun	1. Ricky Agus T., ST., SSi., MM.
	2. Dharmayanti, ST., MMSI.
	3. Yuti Dewita Arimbi, ST., MMSI.
	4. Nuraini Purwandari, ST.
	5. Styra Putra Pratama, ST

Laboratorium Teknik Informatika

Teknik Informatika

Fakultas Teknologi Industri

UNIVERSITAS GUNADARMA

Pertemuan 1

Pengenalan Python

Objektif:

1. Mahasiswa dapat memahami sejarah perkembangan bahasa pemrograman Python.
2. Mahasiswa dapat mengetahui fitur-fitur penting yang terdapat pada Python.
3. Mahasiswa dapat mengetahui kelebihan dan kekurangan bahasa Python.
4. Mahasiswa dapat melakukan proses instalasi bahasa Python.
5. Mahasiswa mengerti konsep dan struktur bahasa pemrograman Python.
6. Mahasiswa mengerti konsep variabel, tipe data, dan operator pada Python.
7. Mahasiswa dapat membuat program sederhana menggunakan bahasa pemrograman Python.

P1.1 Teori

Sejarah Python



Python dikembangkan oleh Guido van Rossum pada tahun 1990 di CWI, Amsterdam sebagai kelanjutan dari bahasa pemrograman ABC. Versi terakhir yang dikeluarkan CWI adalah 1.2. Tahun 1995, Guido pindah ke CNRI sambil terus melanjutkan pengembangan Python. Versi terakhir yang dikeluarkan adalah 1.6. Tahun 2000, Guido dan para pengembang inti Python pindah ke BeOpen.com yang merupakan sebuah perusahaan komersial dan membentuk BeOpen PythonLabs. Python 2.0 dikeluarkan oleh BeOpen. Setelah mengeluarkan Python 2.0, Guido dan beberapa anggota tim PythonLabs pindah ke DigitalCreations.

Saat ini pengembangan Python terus dilakukan oleh sekumpulan pemrogram yang dikoordinir Guido dan Python Software Foundation. Python Software Foundation adalah sebuah organisasi non-profit yang dibentuk sebagai pemegang hak cipta intelektual Python sejak versi 2.1 dan dengan demikian mencegah Python *dimiliki* oleh perusahaan komersial. Saat ini distribusi Python sudah mencapai versi 2.6.1 dan versi 3.0.

Nama Python dipilih oleh Guido sebagai nama bahasa ciptaannya karena kecintaan Guido pada acara televisi Monty Python's Flying Circus. Oleh karena itu seringkali ungkapan-ungkapan khas dari acara tersebut seringkali muncul dalam korespondensi antar pengguna Python.

Pengenalan Python

Python merupakan bahasa pemrograman dinamis yang mendukung pemrograman berbasis objek. Python dapat digunakan untuk berbagai keperluan pengembangan perangkat lunak dan dapat berjalan di berbagai platform sistem operasi. Saat ini skrip python dapat dijalankan pada sistem berbasis : Windows, Linux / Unix, Mac OS X, OS/2, Amiga.

Python didistribusikan dengan beberapa lisensi yang berbeda dari beberapa versi. Lisensi Python tidak bertentangan baik menurut definisi Open Source maupun *General Public License (GPL)*. Interpreter Python dapat diperoleh diwebsite resminya di <http://www.python.org>

Mengapa Python?

Sisi utama yang membedakan Python dengan bahasa lain adalah dalam hal aturan penulisan kode program. Bagi para programmer di luar python siap-siap dibingungkan dengan aturan indentasi, tipe data, tuple, dan dictionary. Python memiliki kelebihan tersendiri dibandingkan dengan bahasa lain terutama dalam hal penanganan modul, ini yang membuat beberapa programmer menyukai python. Selain itu python merupakan salah satu produk yang opensource, free, dan multiplatform.

Beberapa **fitur** yang dimiliki Python adalah :

- memiliki kepustakaan yang luas; dalam distribusi Python telah disediakan modul-modul siap pakai untuk berbagai keperluan.
- memiliki tata bahasa yang jernih dan mudah dipelajari.
- memiliki aturan *layout* kode sumber yang memudahkan pengecekan, pembacaan kembali dan penulisan ulang kode sumber. berorientasi obyek.
- memiliki sistem pengelolaan memori otomatis (garbage collection, seperti java) modular, mudah dikembangkan dengan menciptakan modul-modul baru; modul-modul tersebut dapat dibangun dengan bahasa Python maupun C/C++.
- memiliki fasilitas pengumpulan sampah otomatis, seperti halnya pada bahasa pemrograman Java, python memiliki fasilitas pengaturan penggunaan ingatan komputer sehingga para pemrogram tidak perlu melakukan pengaturan ingatan komputer secara langsung.

Kelebihan dan Kekurangan Python

Beberapa kelebihan bahasa Python antara lain :

- Tidak ada tahapan kompilasi dan penyambungan (link) sehingga kecepatan perubahan pada masa pembuatan system aplikasi meningkat.
- Tidak ada deklarasi tipe sehingga program menjadi lebih sederhana, singkat, dan fleksible.
- Manajemen memori otomatis yaitu kumpulan sampah memori sehingga dapat menghindari pencatatan kode
- Tipe data dan operasi tingkat tinggi yaitu kecepatan pembuatan system aplikasi menggunakan tipe objek yang telah ada
- Pemrograman berorientasi objek
- Pelekatan dan perluasan dalam C

- Terdapat kelas, modul, eksepsi sehingga terdapat dukungan pemrograman skala besar secara modular
- Pemuatan dinamis modul C sehingga ekstensi menjadi sederhana dan berkas biner yang kecil
- Pemuatan kembali secara dinamis modul python seperti memodifikasi aplikasi tanpa menghentikannya
- Model objek universal kelas Satu
- Konstruksi pada saat aplikasi berjalan
- Interaktif, dinamis dan alamiah
- Akses hingga informasi interpreter
- Portabilitas secara luas seperti pemrograman antar platform tanpa ports
- Kompilasi untuk portable kode byte sehingga kecepatan eksekusi bertambah dan melindungi kode sumber
- Antarmuka terpasang untuk pelayanan keluar seperti perangkat Bantu system, GUI, persistence, database, dll

Beberapa kekurangan bahasa Python antara lain :

- Beberapa penugasan terdapat diluar dari jangkauan python, seperti bahasa pemrograman dinamis lainnya, python tidak secepat atau efisien sebagai statis, tidak seperti bahasa pemrograman kompilasi seperti bahasa C.
- Disebabkan python merupakan interpreter, python bukan merupakan perangkat bantu terbaik untuk pengantar komponen performa kritis.
- Python tidak dapat digunakan sebagai dasar bahasa pemrograman implementasi untuk beberapa komponen, tetapi dapat bekerja dengan baik sebagai bagian depan skrip antarmuka untuk mereka.
- Python memberikan efisiensi dan fleksibilitas tradeoff by dengan tidak memberikannya secara menyeluruh.
- Python menyediakan bahasa pemrograman optimasi untuk kegunaan, bersama dengan perangkat bantu yang dibutuhkan untuk diintegrasikan dengan bahasa pemrograman lainnya.

Proses Instalasi Python

Proses instalasi python pada Windows dapat dilakukan langkah-langkah sebagai berikut :

- 1) Pilih software Python yang diinginkan, contoh memakai python versi 2.6 yang dapat di download pada situs <http://www.python.org/ftp/python/2.6/python-2.6.msi>



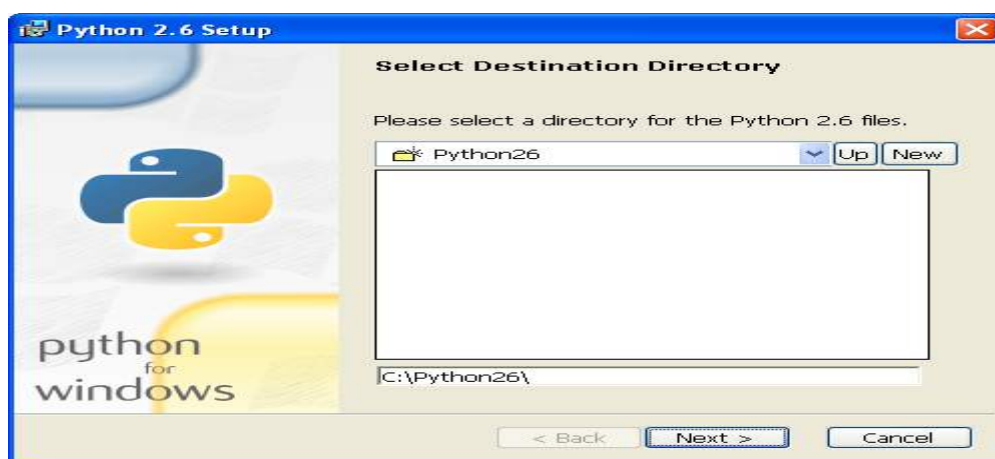
Gambar 1 Windows Intaller Package

- 2) Klik software python, lakukan peng-instalan pada computer dan ikuti langkah selanjutnya :



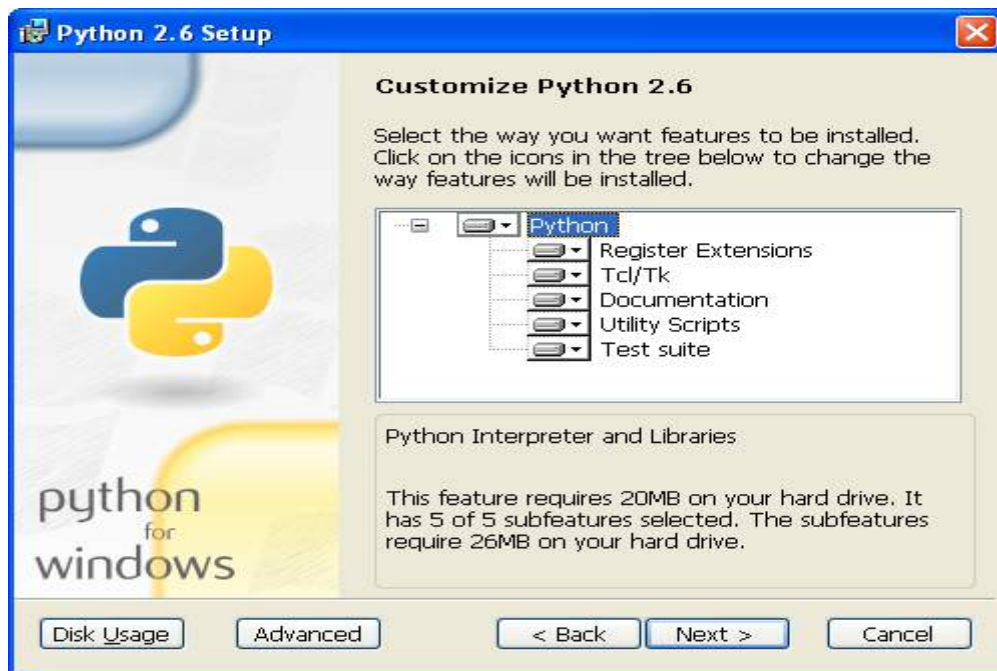
Gambar 2 Jendela Instalasi Python

- 3) Pilih instal untuk semua user, klik tombol next :



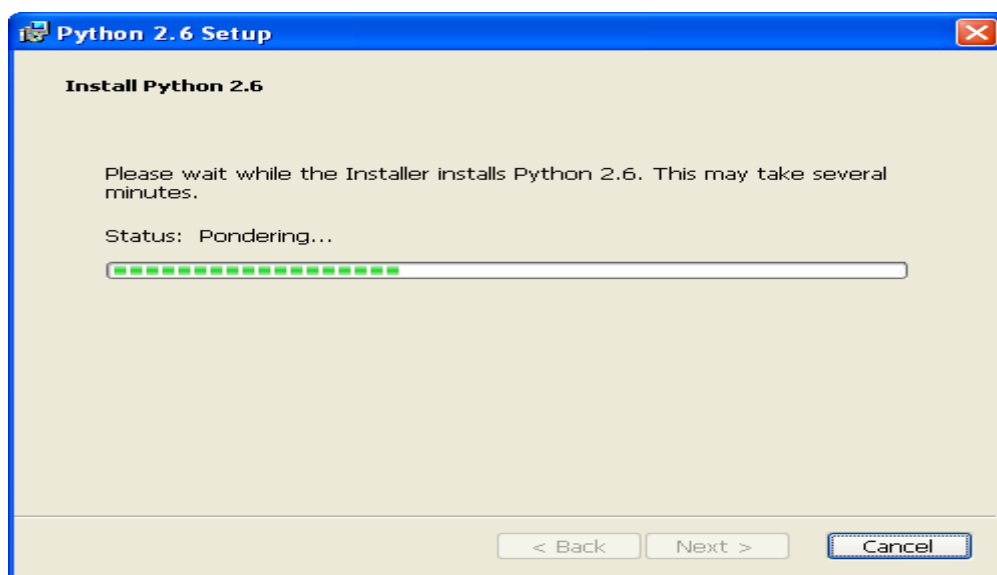
Gambar 3 Jendela untuk memilih direktori instalasi

- 4) Pilih direktori tujuan untuk tempat menyimpan program python, lalu klik tombol next. Jika kita klik tombol Disk Usage maka muncul form yang berisi informasi berapa besar kapasitas disk yang dibutuhkan/digunakan untuk menginstal python(sekitar 49 MB) :



Gambar 4 Jendela Customize Python

- 5) Jika kita klik tombol advanced, pilih compile .py ke byte code setelah instalasi, jika tidak memilih juga tidak apa-apa. Tunggu beberapa menit selama proses instalasi berlangsung dan tekan finish, ikuti petunjuk selanjutnya :



Gambar 5 Jendela Proses Instalasi

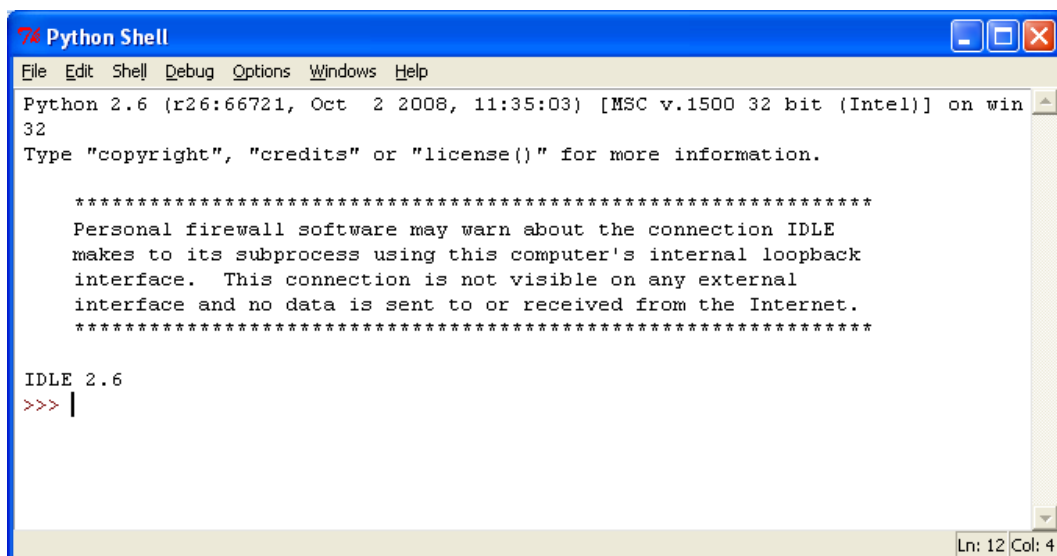


Gambar 6 Jendela Instalasi Python selesai dikerjakan

- 6) Selesai instalasi, komputer Anda telah ada Python, khususnya versi 2.6 pada folder C:\Python26.

Cara Menjalankan Program Python

- Klik tombol start Program Python26 IDLE(Python GUI), IDLE(GUI-Integrated Development Environment) dengan tampilan sebagai berikut :

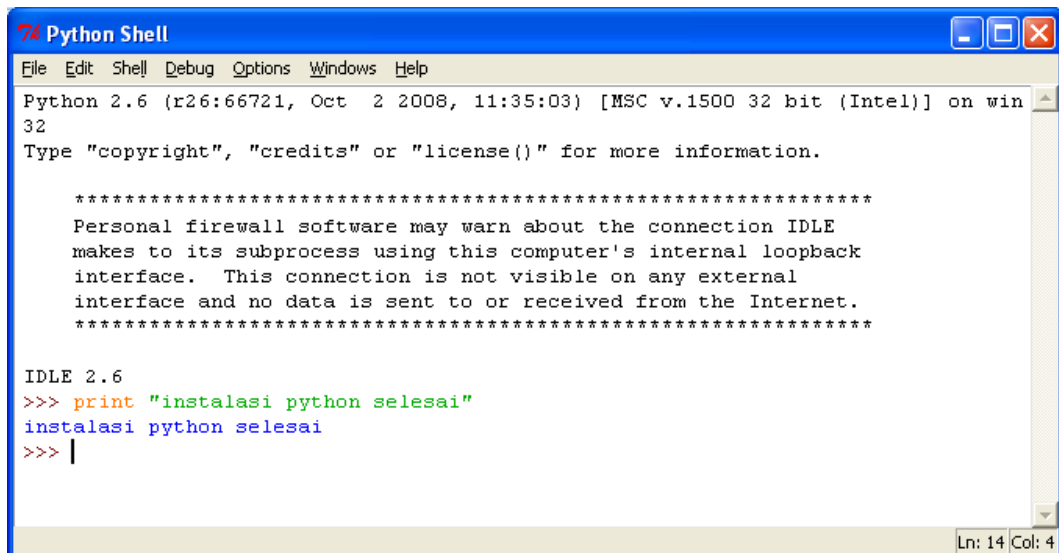


Gambar 7 Tampilan Awal Python

- Pada window diatas,,didalam prompt (>>>), tuliskan : print instalasi python selesai. Kemudian interpreter merespon dengan menampilkan pada layar :

instalasi python selesai

Outputnya adalah sebagai berikut :

A screenshot of a Python Shell window titled "Python Shell". The window has a menu bar with "File", "Edit", "Shell", "Debug", "Options", "Windows", and "Help". The main text area shows the following content: "Python 2.6 (r26:66721, Oct 2 2008, 11:35:03) [MSC v.1500 32 bit (Intel)] on win32", "Type 'copyright', 'credits' or 'license()' for more information.", a large block of text about a firewall warning, and "IDLE 2.6". Below this, the command ">>> print 'instalasi python selesai'" is entered, followed by the output "instalasi python selesai". The prompt ">>>|" is shown on the next line. The status bar at the bottom right indicates "Ln: 14 Col: 4".

```
Python 2.6 (r26:66721, Oct 2 2008, 11:35:03) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.

*****
Personal firewall software may warn about the connection IDLE
makes to its subprocess using this computer's internal loopback
interface. This connection is not visible on any external
interface and no data is sent to or received from the Internet.
*****

IDLE 2.6
>>> print "instalasi python selesai"
instalasi python selesai
>>> |
```

Gambar 8 Contoh menjalankan perintah sederhana pada python

Struktur Pemrograman Python

Tipe Data

Beberapa tipe data pada python, diantaranya :

1. Number

Tipe data Number merepresentasikan nilai-nilai berupa angka. Python menggolongkan beberapa tipe data umum seperti, Integer (bilangan bulat) dan Floating-point (bilangan desimal) ke dalam tipe data Number.

Contoh :

```
>>> 123 + 789
912
>>> 6 * 34
204
>>> 3 ** 2
9
>>> 3.245 * 3
9.7349999999999994
>>> print (5.21 + 6.234)
11.444
```

Untuk perhitungan aritmatika yang menghasilkan nilai desimal antara 0.0 – 0.9 maka akan dilakukan pembulatan ke bawah.

```
>>> 9 / 2
4
```

Operator penugasan (=) digunakan untuk memasukkan nilai kedalam variabel. Tidak ada hasil yang akan muncul sampai statemen selanjutnya.

```
>>> a = 8
>>> b = 3.5
>>> a * b
28.0
```

Nilai dapat di masukkan kedalam beberapa variabel secara simultan.

```
>>> x = y = z = 20
>>> x
20
>>> y
20
>>> z
20
```

2. String

Selain angka, python juga mampu melakukan manipulasi string, yang dapat di ekspresikan dengan beberapa cara. Penulisan nilai string pada python menggunakan tanda petik satu (') atau tanda petik dua ("). Contohnya,

```
>>> "Welcome to python"
'Welcome to python'
>>> 'welcome to python'
'welcome to python'
```

String literal juga dapat menggabungkan beberapa baris dalam berbagai cara. Dengan menggunakan operator (\n\) di akhir kalimat untuk menyambung kalimat selanjutnya yang berada di baris selanjutnya.

```
>>> text = "ini adalah contoh \n\
... penggunaan multiple line\n\
... di python"
>>> print text
ini adalah contoh
penggunaan multiple line
di python
```

Penulisan string untuk multiple line juga dapat dilakukan dengan menggunakan tanda petik dua atau satu sebanyak 3 kali, (“ “ “ atau ' ' ').

```
>>> print """
... my name is python
... i'm an object oriented programming language
... this is an example in using triple quotes
... """
my name is python
i'm an object oriented programming language
this is an example in using triple quotes
```

Untuk menggabungkan dua buah string atau lebih dapat dilakukan dengan dua cara. Pertama, dengan menulis langsung dua buah string yang diapit dengan tanda kutip atau dengan penggunaan operator tambah (+).

```
>>>"Belajar" "python"
'Belajarpython'
>>>'universitas' + 'gunadarma'
'universitasgunadarma'
```

Sebuah string, setiap karakternya dapat diindex, seperti pengindexan pada bahasa C. Karakter pertama pada sebuah string berindex 0, karakter ke-dua berindex 1 dan seterusnya.

```
>>> kata = "Gunadarma"
>>> kata[0]
'G'
>>> kata[5]
'a'
>>> kata[0:5]
'Gunad'
>>> kata[4:8]
'darm'
>>> kata[:7]
'Gunadar'
>>> kata[5:]
'arma'
```

3. List

List sering disebut juga array pada bahasa pemrograman lain. List adalah jenis data campuran yang bisa memiliki komponen penyusun yang berbeda-beda. Sebuah list dapat dibuat dengan menggunakan tanda kurung siku []. Anggota list didaftar dalam kurung siku tersebut dan masing-masing dipisahkan oleh tanda koma. Sifat-sifat list bisa didaftar seperti ini:

- Komponen penyusunnya bisa diganti.
- Komponen penyusunnya dapat dibaca dan dimanipulasi secara langsung.
- Komponen penyusunnya bisa ditambah.
- Komponen penyusunnya dapat diambil dengan menunjukkan indeksnya atau dengan notasi slice.
- Komponen penyusun sebuah list dapat juga berupa list yang lain.

Contohnya,

```
>>> a = ['lab', 'TI', 2010, 'J1']
>>> a
['lab', 'TI', 2010, 'J1']
>>> a[3]
'J1'
>>> a[2]
2010
>>> a[1:-1]
['TI', 2010]
>>> a[:3] + ['gunadarma', 'ATA', 2009, 2010]
['lab', 'TI', 2010, 'gunadarma', 'ATA', 2009, 2010]
>>> len(a)
4
```

4. Dictionary

Berbeda dengan list yang memakai indeks angka untuk merujuk pada isi variabel, dictionary memakai *key* untuk merujuk pada isi variabelnya. Sifat kedua jenis data ini hanya berbeda dalam beberapa hal saja. Untuk mendeklarasikan sebuah dictionary, Python memakai tanda { }.

```

>>> D = { 'food' : 'spam', 'quality' : 4, 'color' : 'blue'}
>>> D
{'food': 'spam', 'color': 'blue', 'quality': 4}
>>> D['color']
'blue'

>>> status = {}
>>> status['nama']='python'
>>> status['desc']='programming language'
>>> status['age']=6
>>> status
{'nama': 'python', 'age': 6, 'desc': 'programming language'}

```

Variabel

Variabel merupakan simbol yang mewakili nilai tertentu. Pembuatan variabel dalam python sangat sederhana. Berikut adalah ketentuan mengenai variabel dalam python,

- Variabel tidak perlu dideklarasikan mempunyai tipe data tertentu
- Jenis data dalam variabel dapat berubah-ubah
- Penulisan variabel harus diawali dengan huruf, dan untuk karakter selanjutnya bisa berupa huruf atau angka
- Penulisan variabel tidak boleh dipisah oleh <spasi>
- Untuk variabel yang terdiri dari 2 suku kata, dapat dipisah dengan simbol underscore (_)

Statemen yang tidak boleh dijadikan nama variabel :

and	continue	else	for	import	not	raise
assert	def	except	from	in	or	return
break	del	exec	global	is	pass	try
class	elif	finally	if	lambda	print	while

Operator

Operator dalam Python dibagi menjadi 3 bagian, yaitu :

- operator aritmatika : +, -, *, /, %
- operator perbandingan : >=, <=, !=, >, <, ==
- operator penugasan : *=, /=, %=, +=, -=

1. Operator Aritmatika

Tabel 1.1 Operator Aritmatika

Operator	Deskripsi	Contoh	Hasil
*	Perkalian	7 * 3	21
/	Pembagian	7 / 3	2
%	Modulus	7 % 3	1
+	Penjumlahan	7 + 3	10
-	Pengurangan	7 - 3	4

2. Operator Perbandingan

Tabel 1.2 Operator Perbandingan

Operator	Deskripsi	Contoh	Hasil
>=	Lebih besar atau sama dengan	7 >= 9	FALSE
<=	Lebih kecil atau sama dengan	3 <= 8	TRUE
!=	Tidak sama dengan	1 != 10	TRUE
<	Lebih kecil	14 < 6	FALSE
>	Lebih besar	5 > 3	TRUE
==	Sama dengan	4 == 4	TRUE

3. Operator Penugasan

Tabel 1.3 Operator Penugasan

Operator	Contoh	Sama dengan
*=	x *= 100	x = x * 100
/=	x /= 100	x = x / 100
%=	x %= 100	x = x % 100
+=	x += 100	x = x + 100
-=	x -= 100	x = x - 100

1. Contoh Program Sederhana Menggunakan Python

Modus penulisan kode python dapat dilakukan dengan dua cara, yaitu

1. Menggunakan mode interaktif
2. Menggunakan Skrip.

1. Menggunakan Mode Interaktif

Menggunakan mode interaktif berarti kita bekerja menggunakan Prompt interpreter dari python. Penulisan kode python dilakukan per-statement, contohnya,

```
>>> print "Hello Python"
Hello Python
>>>
```

Statement “Hello Python” adalah hasil eksekusi dari perintah “print “Hello Python””. Dengan menggunakan prompt interpreter python, kita dapat melakukan uji coba setiap statement-statement atau fungsi-fungsi yang ada pada Python.

2. Menggunakan Skrip

Menggunakan skrip berarti menyusun statement-statement menjadi sebuah satu kesatuan file python. Dengan membuat skrip berarti kita melakukan kompilasi file python dengan bantuan Interpreter dari Python lewat Command Prompt (Windows) atau Terminal (Linux/Unix). Contohnya,

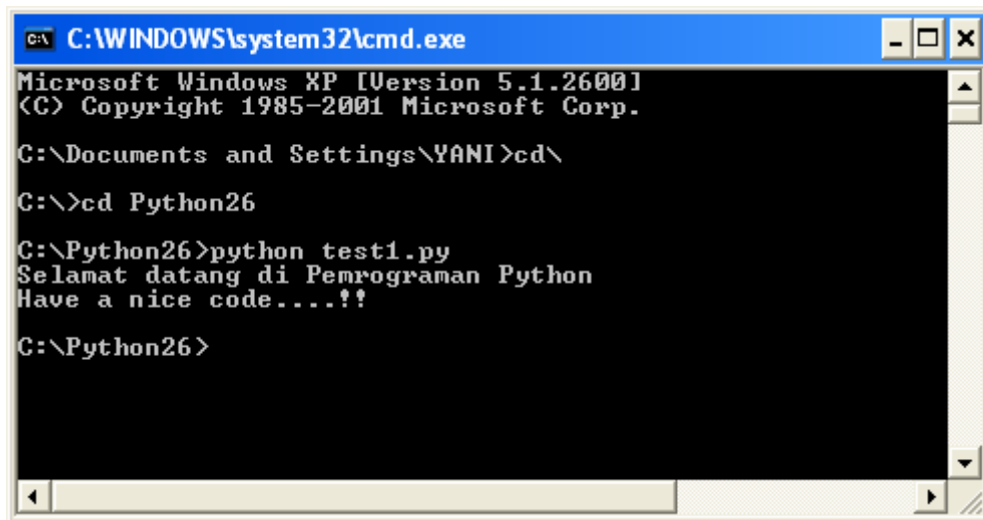
program 1. *test1.py*

```
# Program python 1
print ("Selamat datang di Pemrograman Python")
print ("Have a nice code....!!")
```

Tulis skrip diatas pada Text Editor, kemudian lakukan kompilasi,

```
C:\Python26> python test1.py
```

Output :



Gambar 9 Tampilan Output Program test1.py

P1.2 Contoh Kasus

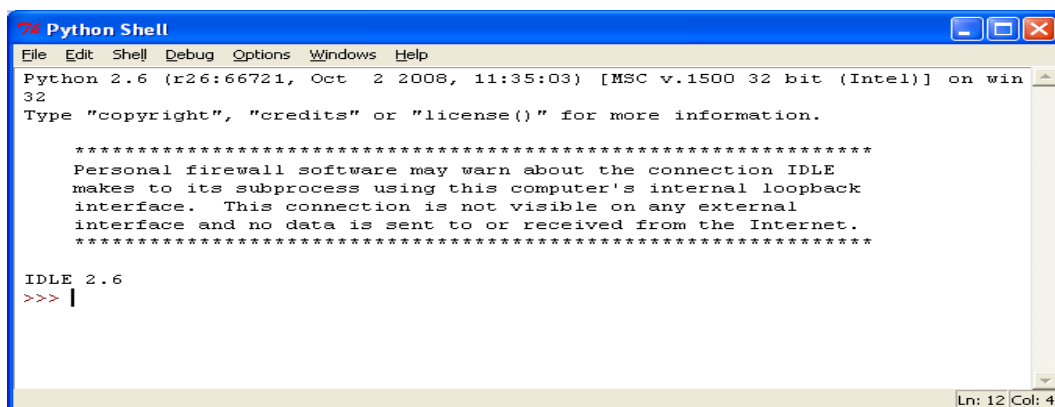
Contoh Kasus 1 :

Pada contoh kasus yang pertama yaitu membuat program menampilkan hasil penjumlahan dua buah bilangan menggunakan tipe data dan operator pada python. Output yang akan ditampilkan adalah sebagai berikut :

```
IDLE 2.6
>>> ===== RESTART =====
>>>
>>> Contoh Kasus Pertama Praktikum AP2
>>> Ini adalah program yang ditulis dengan bahasa Python
>>> Berikut ini contoh program penjumlahan
>>> hasil = a + b
>>> hasil = 10 + 20
>>> hasil = 30
>>> |
```

Langkah-langkah pengerjaan adalah sebagai berikut :

1. Klik tombol start Program Python26 IDLE(Python GUI), IDLE(GUI-Integrated Development Environment) dengan tampilan sebagai berikut :

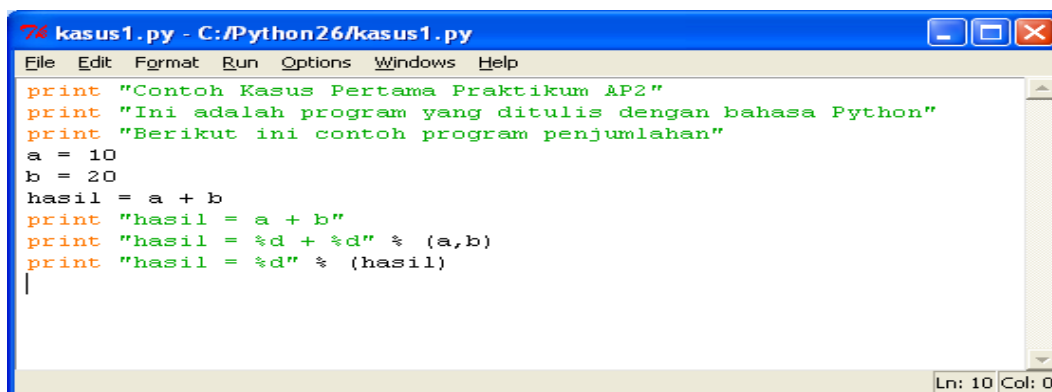


Gambar 10 Tampilan Python GUI

2. Klik Menu File -> New Window lalu ketikkan listing program sebagai berikut.

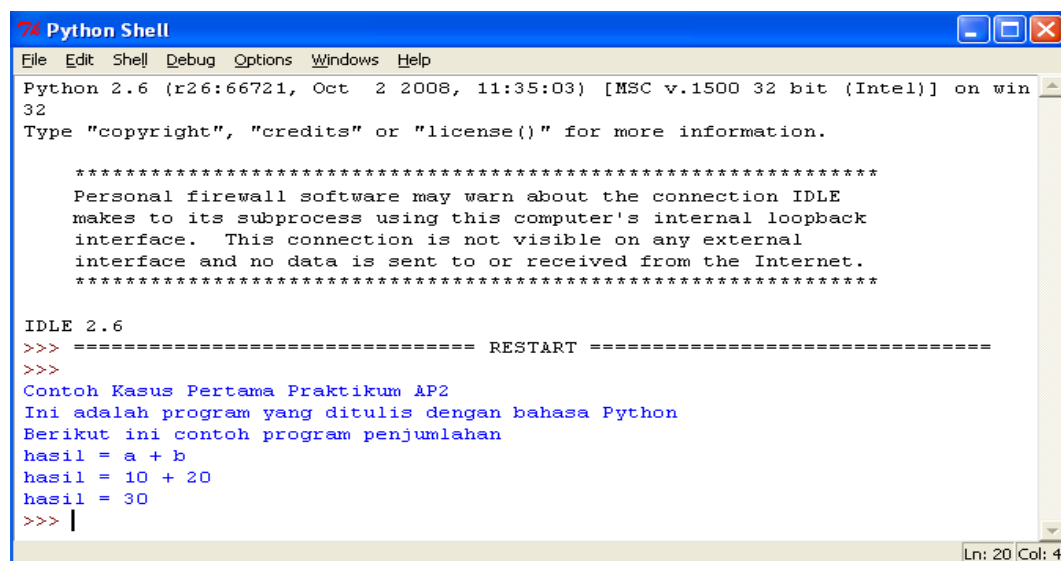
```
print "Contoh Kasus Pertama Praktikum AP2"
print "Ini adalah program yang ditulis dengan bahasa Python"
print "Berikut ini contoh program penjumlahan"
a = 10
b = 20
hasil = a + b
print "hasil = a + b"
print "hasil = %d + %d" % (a,b)
print "hasil = %d" % (hasil)
```

3. Setelah selesai mengetikkan code, langkah selanjutnya menyimpan file tersebut dengan cara klik menu File -> Save As. Masukkan nama file dengan nama kasus1.py



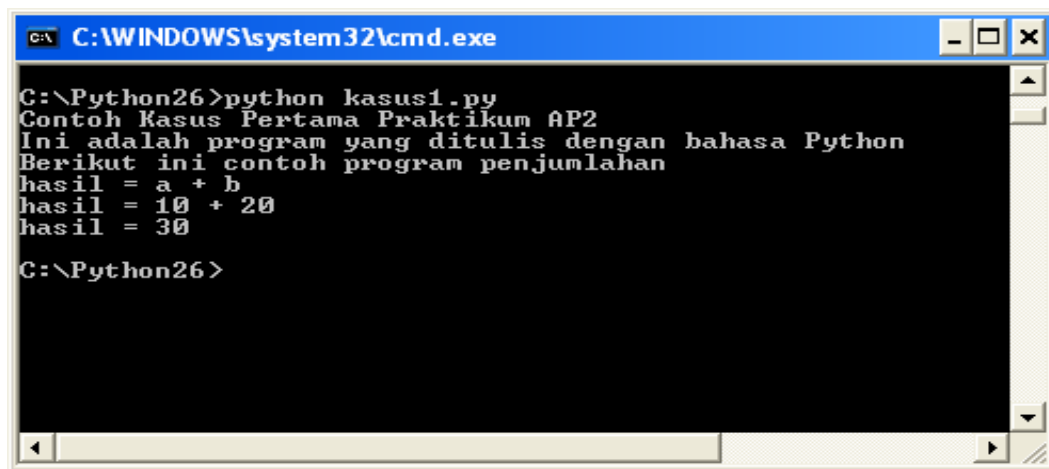
Gambar 11 Tampilan Pengetikan Program Python

4. Setelah itu menjalankan program dengan cara klik menu Run -> Run Module atau dengan menekan tombol F5.



Gambar 12 Tampilan Hasil Output

5. Selain itu kita juga dapat menjalankan program pada command prompt dengan cara mengetikkan `C:\Python26> python kasus1.py`



```
C:\WINDOWS\system32\cmd.exe

C:\Python26>python kasus1.py
Contoh Kasus Pertama Praktikum AP2
Ini adalah program yang ditulis dengan bahasa Python
Berikut ini contoh program penjumlahan
hasil = a + b
hasil = 10 + 20
hasil = 30

C:\Python26>
```

Gambar 13 Tampilan menjalankan python pada command prompt

6. Apabila tidak ada error maka program yang kita compile berhasil.

Contoh Kasus 2

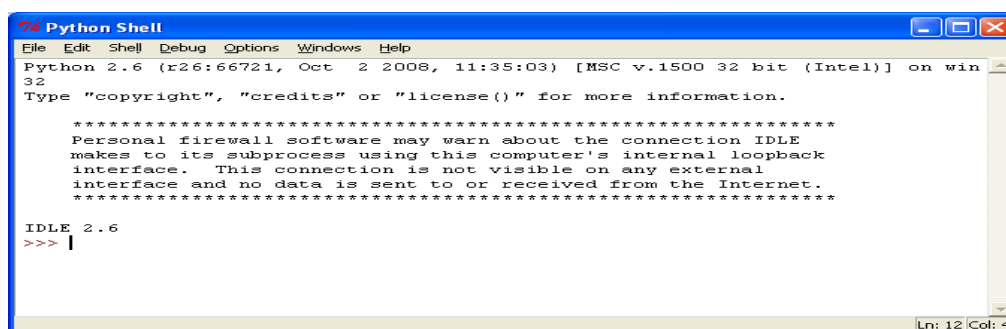
Pada contoh kasus yang kedua yaitu membuat program menginput dan menampilkan kalimat menggunakan bahasa python. Output yang akan ditampilkan adalah sebagai berikut :

```
IDLE 2.6
>>> ===== RESTART =====
>>>
Masukkan nama Anda : Nuraini
Masukkan npm Anda : 54902348

Praktikum Algoritma Pemrograman 2
>>> |
```

Langkah-langkah pengerjaannya adalah sebagai berikut :

- 1) Klik tombol start Program Python26 IDLE(Python GUI), IDLE(GUI-Integrated Development Environment) dengan tampilan sebagai berikut :

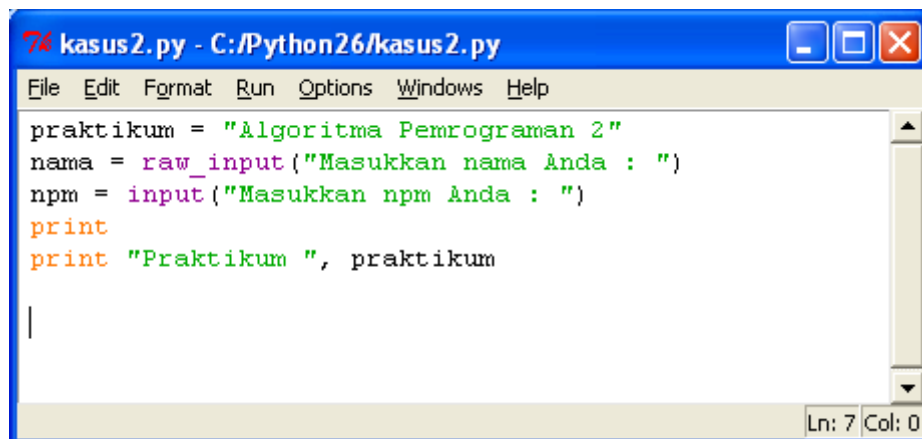


Gambar 14 Tampilan Python GUI

- 2) Klik Menu File -> New Window lalu ketikkan listing program sebagai berikut.

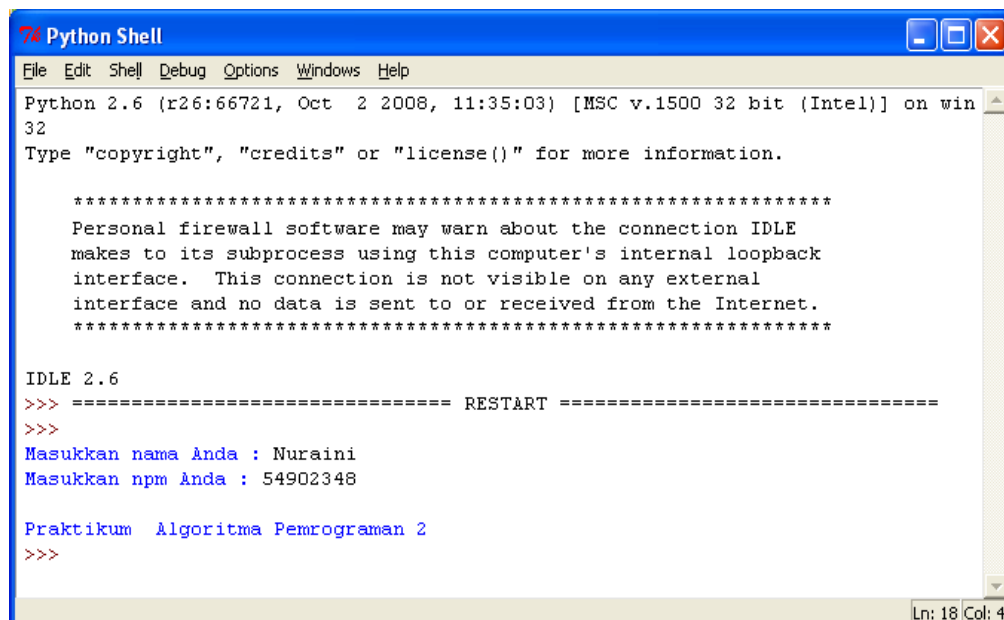
```
praktikum = "Algoritma Pemrograman 2"
nama = raw_input("Masukkan nama Anda : ")
npm = input("Masukkan npm Anda : ")
print
print "Praktikum ", praktikum
```

- 3) Setelah selesai mengetikkan code, langkah selanjutnya menyimpan file tersebut dengan cara klik menu File -> Save As. Masukkan nama file dengan nama kasus2.py



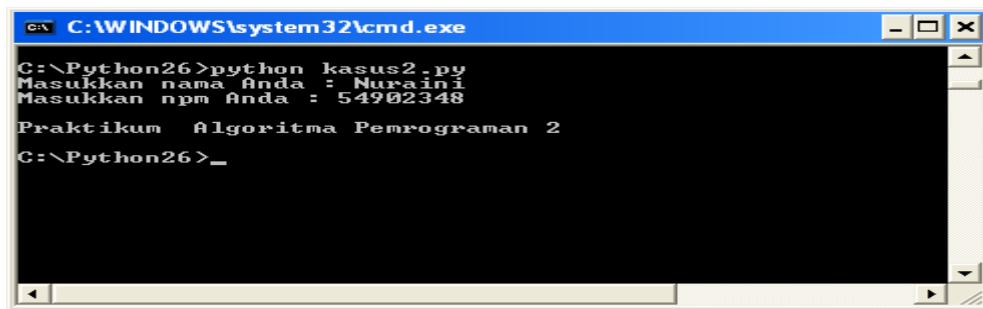
Gambar 15 Tampilan pengetikan program sederhana kasus 2

- 4) Setelah itu menjalankan program dengan cara klik menu Run -> Run Module atau dengan menekan tombol F5.



Gambar 16 Tampilan Output program sederhana kasus 2

- 5) Selain itu kita juga dapat menjalankan program pada command prompt dengan cara mengetikkan `C:\Python26> python kasus2.py`



```
C:\WINDOWS\system32\cmd.exe
C:\Python26>python kasus2.py
Masukkan nama Anda : Nuraini
Masukkan npm Anda : 54902348
Praktikum Algoritma Pemrograman 2
C:\Python26>
```

Gambar 17 Tampilan output pada command prompt

- 6) Apabila tidak ada error maka program yang kita compile berhasil.

P1.3 Latihan

Latihan 1

Berikut ini merupakan program yang menggunakan bahasa python versi 2.6 untuk mencari hasil penjumlahan dua bilangan serta menginput dan menampilkan nama, npm, mata praktikum. Pada code editor di python 2.6 ketikkan program berikut.

(Lengkapi kode program berikut dengan mengisi titik-titik yang berwarna merah)

```
praktikum = "Algoritma Pemrograman 2" // pendeklarasian variable praktikum
print "Berikut ini contoh program penjumlahan"
```

```
.....
```

```
npm = input("Masukkan npm Anda : ") // pendeklarasian variable input npm
```

```
.....
```

```
print "Nama : ", nama //mencetak variable nama yang sudah diinput
```

```
.....
```

```
a = 50 // pendeklarasian variable bilangan 1
```

```
b = 50 // pendeklarasian variable bilangan 2
```

```
.....
```

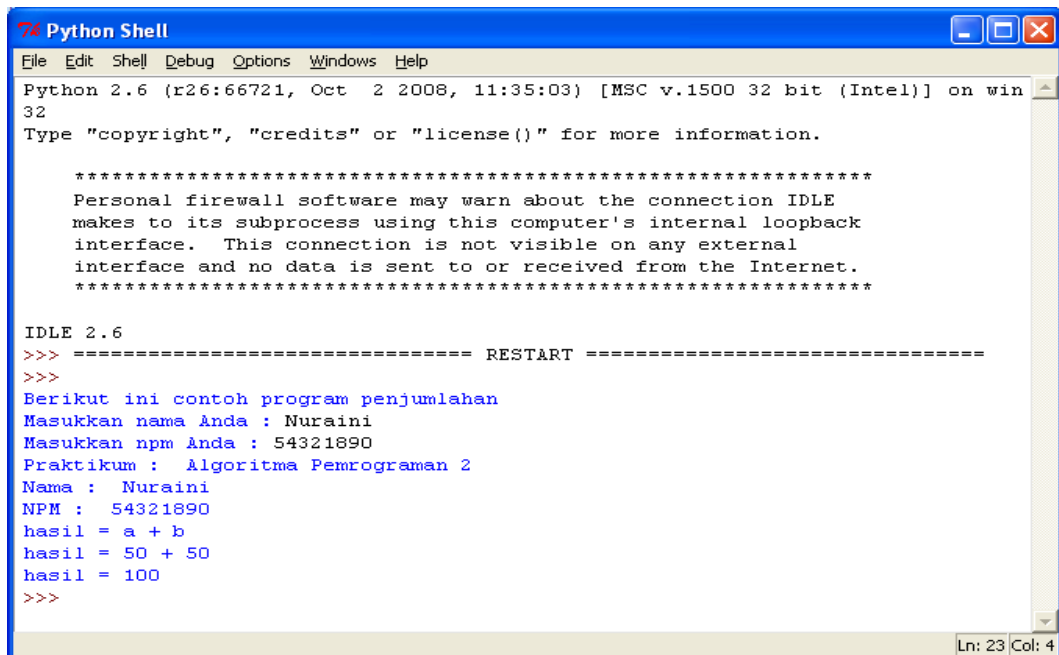
```
print "hasil = a + b"
```

```
.....
```

```
print "hasil = %d" % (hasil) // mencetak variable hasil
```

(Save program diatas dengan nama file `latihan.py`)

TAMPILAN OUTPUT PROGRAM LATIHAN

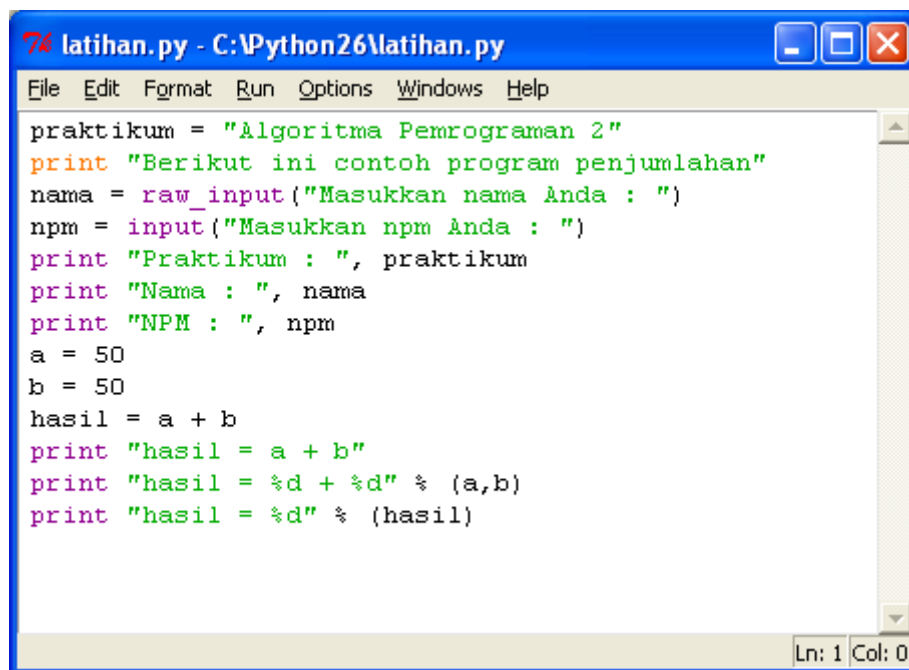


```
Python 2.6 (r26:66721, Oct 2 2008, 11:35:03) [MSC v.1500 32 bit (Intel)] on win
32
Type "copyright", "credits" or "license()" for more information.

*****
Personal firewall software may warn about the connection IDLE
makes to its subprocess using this computer's internal loopback
interface. This connection is not visible on any external
interface and no data is sent to or received from the Internet.
*****

IDLE 2.6
>>> ===== RESTART =====
>>>
Berikut ini contoh program penjumlahan
Masukkan nama Anda : Nuraini
Masukkan npm Anda : 54321890
Praktikum : Algoritma Pemrograman 2
Nama : Nuraini
NPM : 54321890
hasil = a + b
hasil = 50 + 50
hasil = 100
>>>
```

KOREKSI LATIHAN PROGRAM



```
latihan.py - C:\Python26\latihan.py
File Edit Format Run Options Windows Help

praktikum = "Algoritma Pemrograman 2"
print "Berikut ini contoh program penjumlahan"
nama = raw_input("Masukkan nama Anda : ")
npm = input("Masukkan npm Anda : ")
print "Praktikum : ", praktikum
print "Nama : ", nama
print "NPM : ", npm
a = 50
b = 50
hasil = a + b
print "hasil = a + b"
print "hasil = %d + %d" % (a,b)
print "hasil = %d" % (hasil)
```

P1.4 Daftar Pustaka

- [1] Kadir, Abdul. *Dasar Pemrograman Python*. 2005. Yogyakarta. Andi Publisher
- [2] www.python.org