

LAPORAN PRAKTIKUM
MODUL I
VARIABEL, TIPE DATA DAN OPERATOR



Disusun Oleh :

Raihan Iqbal Pasya Hsb

NIM : 19104030

Dosen

Ariq Cahya Wardhana, S. Kom.,M.Kom.

PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK
FAKULTAS INFORMATIKA
INSTITUT TEKNOLOGI TELKOM PURWOKERTO
2020

BAB I

TUJUAN PRAKTIKUM

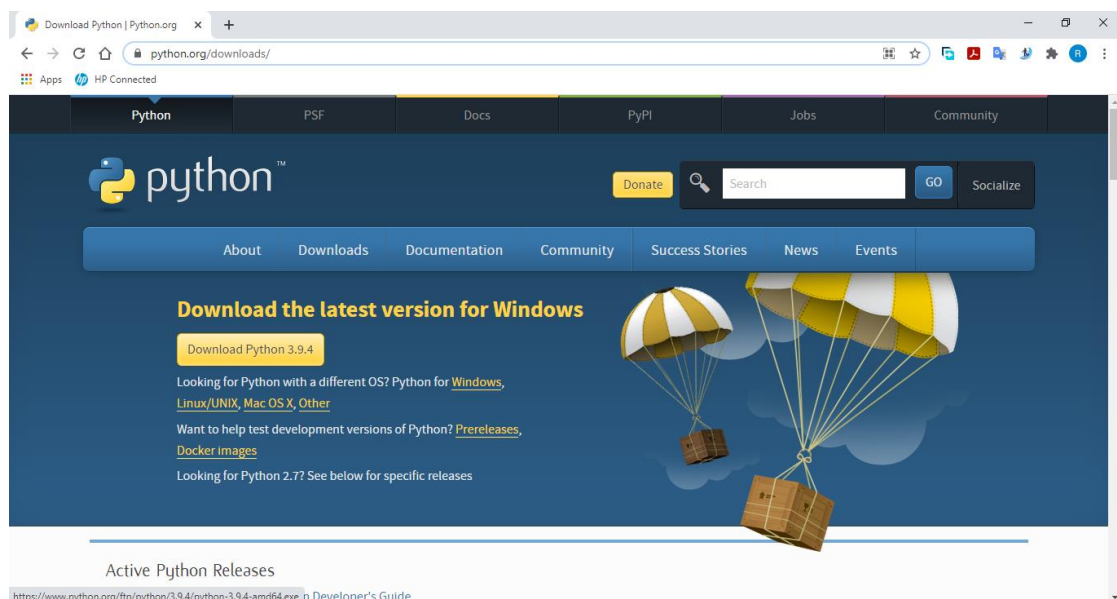
1. Mahasiswa mampu melakukan instalasi interpreter python beserta IDE-nya dan mampu mengoperasikannya dengan baik.
2. Mahasiswa mampu memahami dasar-dasar pemrograman python dan membedakan variabel, objek, serta tipe data yang terdapat pada bahasa pemrograman python.

BAB II

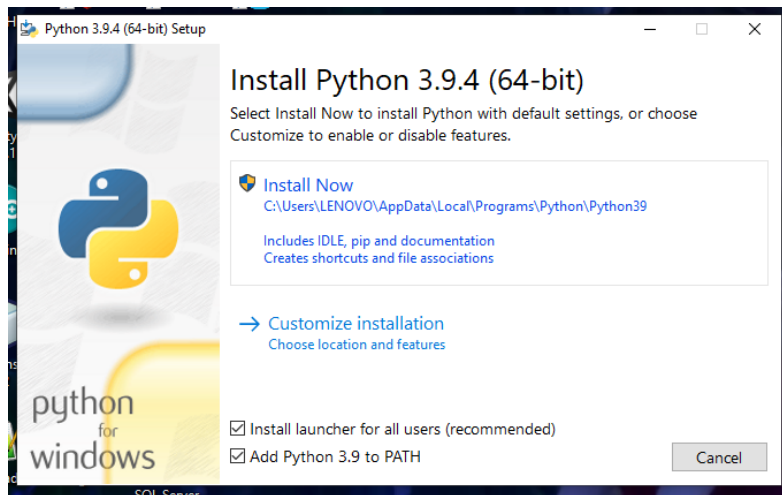
PRAKTIKUM

A. INSTALASI PYTHON

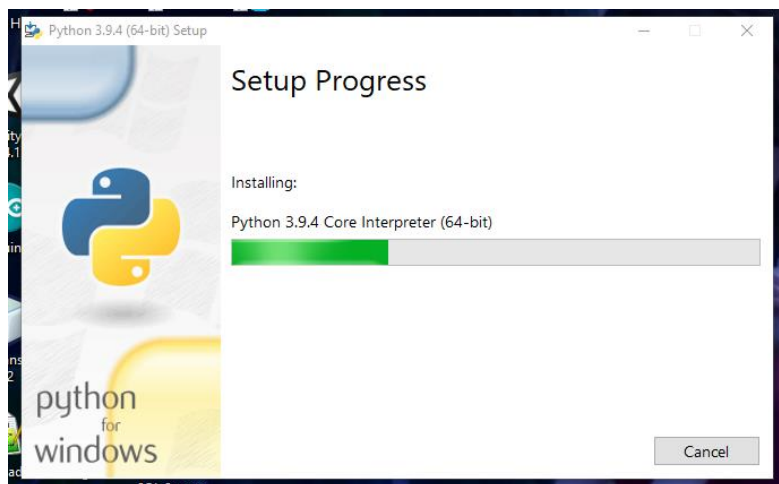
Pertama-tama sebelum membuat program, kita harus men-download interpreter python terlebih dahulu pada website : <https://www.python.org/downloads/> seperti pada gambar dibawah ini :



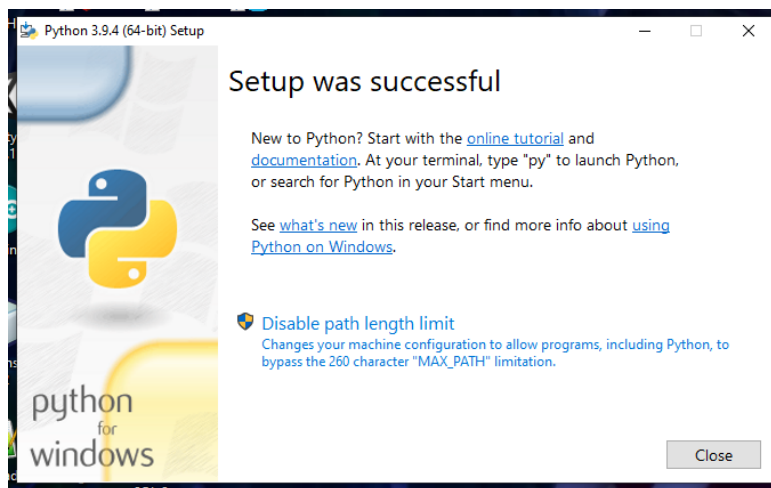
Kemudian download dan tunggu hingga selesai. Nah setelah selesai men-download, buka aplikasinya untuk melakukan instalasi. Maka akan muncul gambar seperti dibawah ini :



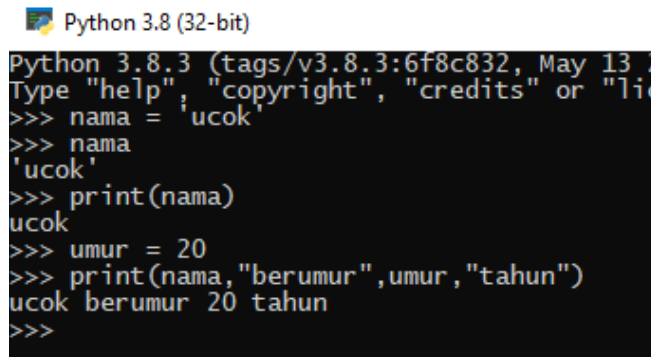
Setelah itu, ceklis tulisan “Add Python 3.9 to PATH” untuk menambahkan Path Python secara otomatis, kemudian klik install now dan tunggu proses instalasi hingga selesai. Maka setelah itu tampilannya akan menjadi seperti berikut :



Jika sudah selesai, maka tampilannya akan menjadi seperti berikut :



Jika sudah seperti ini, silahkan close saja dan proses instalasi sudah selesai. Setelah itu kita bisa coba langsung menuliskan program pada IDLE Python ini seperti pada gambar berikut ini :

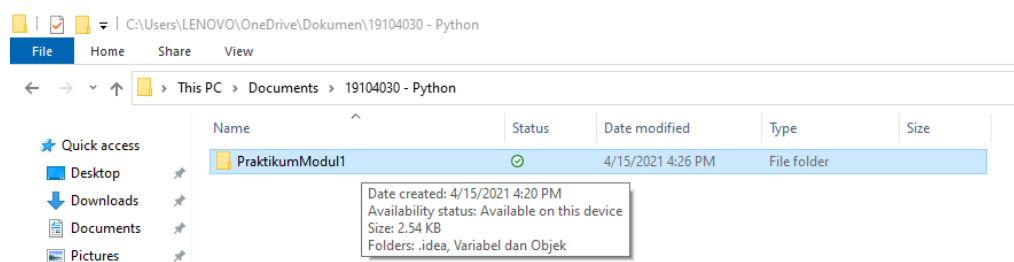


```
Python 3.8 (32-bit)
Python 3.8.3 (tags/v3.8.3:6f8c832, May 13 2020)
Type "help", "copyright", "credits" or "license()" for more
>>> nama = 'ucok'
>>> nama
'ucok'
>>> print(nama)
ucok
>>> umur = 20
>>> print(nama,"berumur",umur,"tahun")
ucok berumur 20 tahun
>>>
```

B. MEMBUAT DAN EKSEKUSI KODE PROGRAM PADA PYTHON

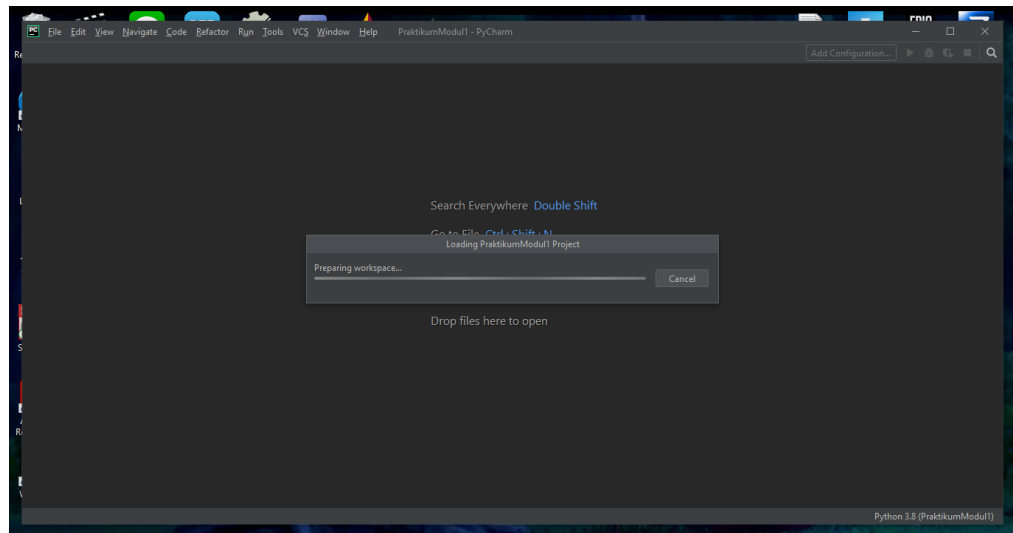
Nah disini, kita akan mencoba membuat program Python berekstensi .py pada IDE. IDE yang digunakan untuk menuliskan program Python ada berbagai macam, seperti PyCharm, PyScripter, Geany, Notepad++ dan lain-lain. Namun disini kita akan mencoba menggunakan PyCharm untuk menuliskan program python. Untuk cara men-download dan instalasinya tidak saya tunjukkan disini, karena kita akan langsung mencoba membuat kode program dengan nama hello.py, langkah-langkah yang diperlukan adalah sebagai berikut :

1. Buat direktori tempat penyimpanan Anda

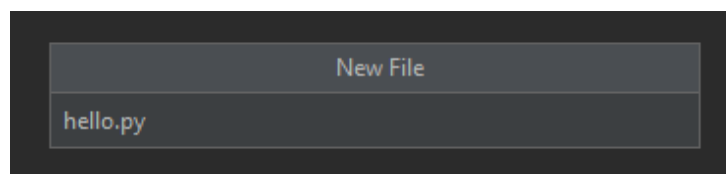
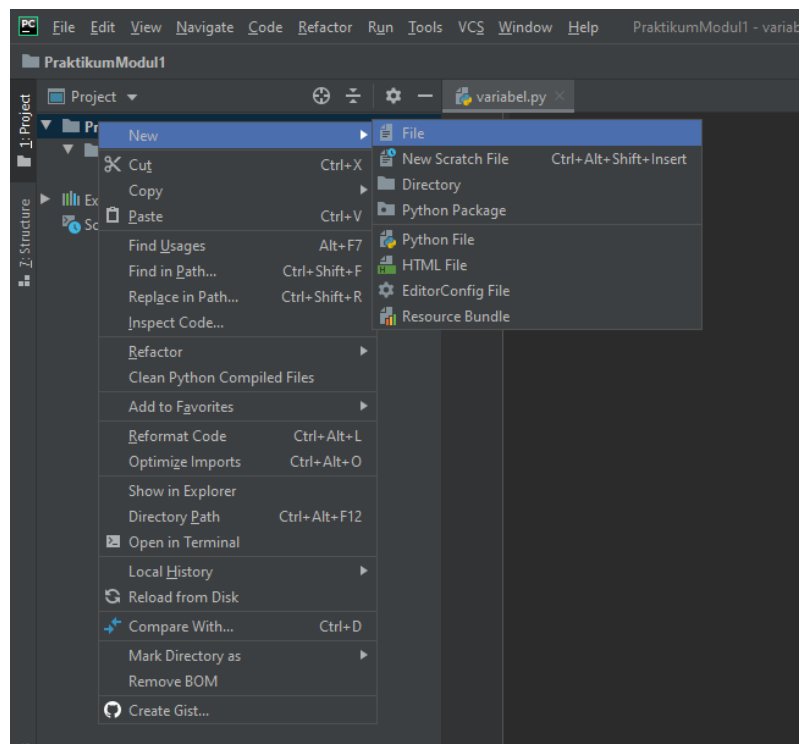


2. Jalankan program teks editor

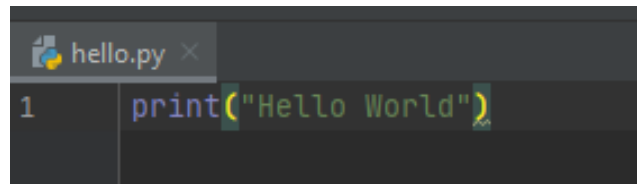




3. Simpan file tersebut dengan nama hello.py

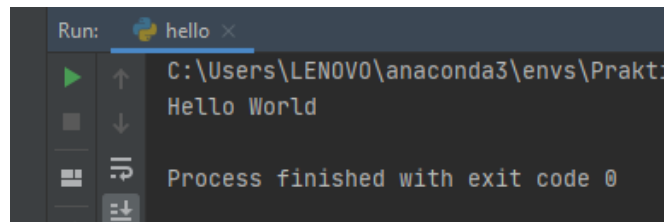


4. Tuliskan kode sebagai berikut `print("Hello World!")`



```
hello.py x
1 print("Hello World!")
```

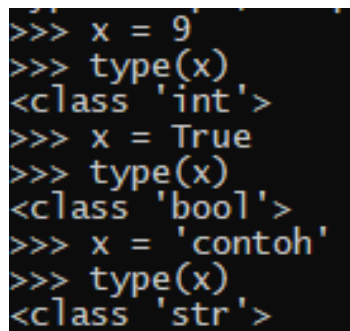
Berikut Outputnya :



```
Run: hello x
C:\Users\LENOVO\anaconda3\envs\Prakti
Hello World
Process finished with exit code 0
```

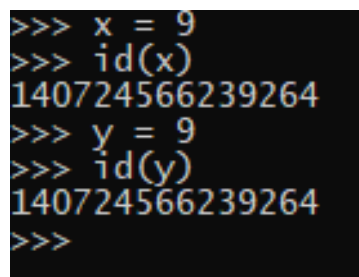
C. VARIABEL DAN OBJEK

Python merupakan Bahasa dengan dynamic typing yaitu variabelnya tidak dibatasi oleh tipe datanya. Sebagai contoh, variabel yang sudah diisi dengan tipe bilangan bulat bisa diisi dengan bilangan riil, string, ataupun tipe data yang lain.



```
>>> x = 9
>>> type(x)
<class 'int'>
>>> x = True
>>> type(x)
<class 'bool'>
>>> x = 'contoh'
>>> type(x)
<class 'str'>
```

Pada contoh kode diatas bisa dilihat ketika variabel (x) berisi tipe data integer yaitu “9”, maka hasilnya ia akan menampilkan tipe data integer seperti pada output “<class ‘int’>” begitu juga dengan variabel (x) ketika diisi tipe data lainnya. Sehingga satu variable dapat berubah-ubah tipe datanya sesuai dengan kebutuhan.



```
>>> x = 9
>>> id(x)
140724566239264
>>> y = 9
>>> id(y)
140724566239264
>>>
```

Dari potongan kode di atas, jika kita memanggil id untuk variable x maupun y maka akan muncul id yang sama. Hal ini menunjukkan bahwa variable x maupun y memiliki id referensi yang sama karena nilai pada variabel x maupun y adalah sama-sama sebuah objek yang bernilai 9.

Berdasarkan potongan kode yang telah kita buat sebelumnya yang memiliki id yang sama. Jika kita menggunakan perintah del untuk menghapus variable y, maka yang akan dihapus adalah referensinya saja, bukan objek '9' yang tadi ditunjuk oleh variable x dan y.

```
>>> x = 9
>>> id(x)
140724566239264
>>> y = 9
>>> id(y)
140724566239264
>>> del y
>>> y
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'y' is not defined
>>> x
9
>>> id(x)
140724566239264
```

Berdasarkan kode program tersebut, tambahkan kode berikut

```
>>> x = True
```

Dengan menambahkan kode baris di atas, maka referensi objek variabel x akan dipindahkan dari objek '9' ke objek 'True'. Dengan demikian objek lama (9) akan diklaim sebagai sampah karena objek tersebut tidak ditunjuk oleh variable apapun.

D. PYTHON BERSIFAT CASE-SENSITIVE

Kode program pada python bersifat *case-sensitive*. Dengan demikian misal variable **Posisi** akan berbeda dengan variable **posisi**. Seperti pada gambar berikut :

```
>>> posisi = (300,300)
>>> posisi
(300, 300)
>>> Posisi
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'Posisi' is not defined
```

E. PERINTAH PROGRAM (STATEMENT)

Pada python setiap kode program yang dituliskan tidak harus diakhiri dengan sebuah statement (biasanya tanda titik koma) seperti pada Java dan C. Titik koma pada python hanya diberikan pada saat ada dua atau lebih statement pada satu baris yang sama, seperti tanda koma pada tulisan.

```
>>> a = 1; b = 2; c = 3
>>> print(a); print(b); print(c)
1
2
3
```

Secara umum perintah program ditulis dalam satu baris kode, tetapi jika perintah yang dituliskan panjang maka anda dapat memecah perintah tersebut menjadi beberapa baris. Dimana setiap baris harus dihubungkan dengan tanda backslash (\). Contohnya seperti program dibawah ini.

```
>>> x = 9
>>> if isinstance(x,int) and \
... x > 0 and \
... x % 2 == 1:
...     print("%d adalah bilangan bulat ganjil positif" %x)
... 9 adalah bilangan bulat ganjil positif
```

Tetapi tanda backslash tidak diperlukan jika kita menulis perintah kode dalam bentuk array atau kode yang terdapat diantara tanda (...), [...] atau {...}.

```
>>> print("Pemrograman GUI" +
... "dengan Python dan PyQt")
Pemrograman GUIDengan Python dan PyQt
>>> data = [
... 100,
... 200,
... 300
... ]
>>> kamus = {
... 'one': 'satu',
... 'two': 'dua',
... 'three': 'tiga'
... }
>>> data
[100, 200, 300]
>>> kamus
{'one': 'satu', 'two': 'dua', 'three': 'tiga'}
>>> _
```

F. TIPE NUMERIK

Sama seperti bahasa pemrograman lain, python mendukung beberapa tipe data, salah satunya adalah tipe data numerik yang meliputi bilangan bulat, bilangan riil.

A. Bilangan Bulat

Dalam python terapat dua tipe bilangan bulat yaitu int dan bool. Selain tipe integral primitive python juga dapat menggunakan bilangan integral dengan basis desimal, biner, octal maupun heksadesimal. Berikut contoh bilangan bulat integral pada Python.

```
>>> # bilangan biner
>>> a = 0b1001
>>> # bilangan oktal
>>> b = 0o23
>>> # bilangan heksadesimal
>>> c = 0x2f
>>> a
9
>>> b
19
>>> c
47
>>> _
```

Tipe bilangan bulat yang kedua adalah tipe Boolean seperti yang telah kita ketahui tipe data boolean bernilai true atau false saja. Berikut contoh bilangan bulat Boolean.

```
>>> a = True
>>> type(a)
<class 'bool'>
>>> int(a)
1
>>> _
```

Proses perhitungan dan penambahan bilangan pada python akan menghasilkan objek baru, hal ini terlihat dari id nya, seperti pada gambar berikut.

```
>>> a = 15
>>> id(a)
140737229498592
>>> a += 5
>>> a
20
>>> id(a)
140737229498752
>>> _
```

B. Bilangan Riil

Untuk tipe bilangan riil, python menyediakan tipe float, decimal.Decimal dan complex. Tipe bilangan float menggunakan titik untuk tanda desimalnya. Dapat dilihat pada gambar berikut.

```
>>> a = 123.456
>>> a
123.456
>>> a * 2
246.912
```

Sedangkan untuk tipe decimal hampir sama dengan tipe data float, akan tetapi tipe desimal digunakan untuk melakukan perhitungan dengan nilai koma yang lebih presisi.

G. TIPE STRING

Tipe data string dalam python direpresentasikan dengan tipe **str**. Objek string dapat dibuat dengan tiga cara yaitu :

- Menggunakan tanda petik tunggal
- Menggunakan tanda petik ganda
- Menggunakan tanda petik tunggal ataupun ganda yang direpetisi sebanyak tiga kali

Berikut contoh kode programnya :

```
>>> s1 = 'pemrograman python'
>>> s2 = "pemrograman python 2"
>>> s3 = '''pemrograman
... python 3'''
```

Objek dalam string tidak dapat dirubah, tiap karakter di dalam string dapat diakses dengan tanda [] diikuti nomor array-nya. Sekarang kita akan coba liat hasilnya menggunakan tanda [] seperti gambar berikut.

```
>>> s1[0], s1[1], s1[2]
('p', 'e', 'm')
```

Dalam string kita dapat memberikan karakter khusus antara lain **\n** untuk memberikan enter, **\'** untuk memberikan petik tunggal, dan **\t** untuk memberikan tab. Pada dasarnya karakter khusus dalam python harus diawali dengan backslash (\) diikuti dengan karakter khususnya, seperti pada contoh berikut.

```
>>> data = 'p001\tspidol\t\t9000\np002\tpensil\t\t6000'
>>> print(data)
p001      spidol          9000
p002      pensil         6000
```

Python juga dapat menggabungkan dua objek string menjadi satu dengan operator +. Sekarang kita akan coba gabung data string 'harga' dengan kode diatas, dan berikut adalah hasilnya.

```
>>> data = '\tharga\n' + data
>>> print(data)
        harga
p001      spidol          9000
p002      pensil         6000
```

A. Membandingkan String

Untuk membandingkan kesamaan string python menggunakan operator ==. Sedangkan untuk membandingkan id objek string menggunakan is. Selain kedua operator tersebut, python juga dapat menggunakan operator lainnya untuk membandingkan tipe data string. Maka kita akan coba membuat kode program dan menggunakan operator >, <=, >=, <, ==, dan !=. Berikut adalah hasilnya.

```
>>> s1 = 'python'
>>> s2 = 'PYTHON'
>>> s1 == s2
False
>>> s1 != s2
True
>>> s1 < s2
False
>>> s1 > s2
True
>>> s1 <= s2
False
>>> s1 >= s2
True
```

B. Mengekstrak Substring

Substring di dalam string dapat diekstrak dengan menggunakan operator slice (:) dengan menyertakan indeks awal dan akhir sebagai penanda. Berikut contoh kode programnya.

```
>>> s = 'Pemrograman Python dan PyQt'
>>> s1 = s[0:11]
>>> s1
'Pemrograman'
>>> len(s1)
11
```

Kode tersebut mengambil substring dari variable s mulai dari indeks ke 0 sampai indeks ke 11. Jika kita tidak menyertakan indeks maka string yang akan diekstrak adalah sepanjang string tersebut / string sisanya. Hasilnya akan menjadi seperti berikut.

```
>>> S = s[:11]
>>> print(S)
Pemrograman
>>> S = s[:8]
>>> print(S)
Pemrogra
>>> S = s[8:]
>>> print(S)
man Python dan PyQt
>>> S = s[0:11:2]
>>> print(S)
Pmormn
>>> S = s[0:11:1]
>>> print(S)
Pemrograman
>>> S = s[0:11:3]
>>> print(S)
Prpra
```

C. Membuat String dengan format tertentu

Python pada dasarnya juga dapat menggabungkan tipe data atau format lain ke dalam string yang telah dibuat. Antara lain dengan menggunakan \$d, %f, %s dan lain sebagainya. Berikut contoh kode programnya.

```
>>> S = 'balonku ada %d, kempes %d tinggal %f' %(5,1,4)
>>> S
'balonku ada 5, kempes 1 tinggal 4.000000'
```

H. TIPE KOLEKSI

Tipe koleksi biasa disebut dengan tipe container. Beberapa tipe koleksi antara lain list, dictionary, tuple dan set. Objek list dibuat dengan menggunakan tanda [], setiap objek yang berada di dalamnya dipisahkan dengan menggunakan koma dan dapat terdiri dari berbagai macam tipe data, seperti pada gambar berikut.

```
>>> list = ['balon', 'budi', 'ada', 5]
```

Model dan cara akses list dapat digabungkan dengan fungsi perulangan dasar seperti for, while dan lain sebagainya.

```

>>> list = ['balon', 'budi', 'ada', 5]
>>> for item in list:
...     print (item)
...
balon
budi
ada
5

```

Untuk menghapus atau merubah elemen pada list anda dapat menggunakan perintah **del nama_list['indeks_list']** Hasilnya akan menjadi seperti berikut.

```

>>> del list[0]
>>> for item in list:
...     print(item)
...
budi
ada
5

```

Untuk merubah dapat menggunakan perintah **namaList['indeks'] = value baru**. Sehingga hasilnya menjadi seperti berikut.

```

>>> list[1] = 'tidak ada'
>>> for item in list:
...     print(item)
...
budi
tidak ada
5

```

Serta, untuk menambahkan elemen pada list anda dapat menggunakan perintah **nama_list.extend([list/tuple])**. Dan hasilnya akan menjadi seperti berikut.

```

>>> barang = ('balon', 'uang')
>>> list.extend(barang)
>>> for item in list:
...     print(item)
...
budi
tidak ada
5
balon
uang

```

BAB III

KESIMPULAN

Kesimpulan yang dapat saya ambil adalah, bahasa pemrograman python ternyata cukup mudah dipahami daripada bahasa pemrograman lainnya, serta kode program pada pemrograman python tidak panjang dan bertele-tele, melainkan sangat singkat dan jelas, sehingga hal ini memudahkan kita dalam menulis program yang banyak. Walaupun python masih terdapat case-sensitive dan indentasi, namun bahasa pemrograman python tetap terlihat lebih simple dan mudah dipahami ketimbang bahasa pemrograman lainnya seperti Java, Javascript dan lain-lain.