

**MENCOBA PYTHON**



---

# **MENCOBA PYTHON**

## **Belajar Python untuk Pemula**

---

**Rolly Maulana Awangga**  
Politeknik Pos Indonesia

**Program Studi Sarjana Terapan Teknik Informatika.**  
Politeknik Pos Indonesia



**A JOHN WILEY & SONS, INC., PUBLICATION**

Copyright ©2017 by John Wiley & Sons, Inc. All rights reserved.

Published by John Wiley & Sons, Inc., Hoboken, New Jersey.  
Published simultaneously in Canada.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning, or otherwise, except as permitted under Section 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 646-8600, or on the web at [www.copyright.com](http://www.copyright.com). Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008.

**Limit of Liability/Disclaimer of Warranty:** While the publisher and author have used their best efforts in preparing this book, they make no representations or warranties with respect to the accuracy or completeness of the contents of this book and specifically disclaim any implied warranties of merchantability or fitness for a particular purpose. No warranty may be created or extended by sales representatives or written sales materials. The advice and strategies contained herein may not be suitable for your situation. You should consult with a professional where appropriate. Neither the publisher nor author shall be liable for any loss of profit or any other commercial damages, including but not limited to special, incidental, consequential, or other damages.

For general information on our other products and services please contact our Customer Care Department with the U.S. at 877-762-2974, outside the U.S. at 317-572-3993 or fax 317-572-4002.

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print, however, may not be available in electronic format.

***Library of Congress Cataloging-in-Publication Data:***

Mencoba Python / Rolly Maulana Awangga . . . [et al].  
p. cm.—(Wiley series in survey methodology)  
“Wiley-Interscience.”  
Includes bibliographical references and index.  
ISBN 0-471-48348-6 (pbk.)  
1. Surveys—Methodology. 2. Social sciences—Research—Statistical methods. I. Groves, Robert M. II. Series.

HA31.2.S873 2007  
001.4'33—dc22 2004044064  
Printed in the United States of America.

10 9 8 7 6 5 4 3 2 1

*Untuk Nusa Bangsa dan  
Agama*

## CONTRIBUTORS

---

JURU KETIK, Fujitsu Laboratories Ltd., Fujitsu Limited, Atsugi, Japan

JURU KODING, Center for Solid State Electronics Research, Arizona State University, Tempe, Arizona

TUKANG BEBERES, Department of Electrical and Computer Engineering, University of Notre Dame, Notre Dame, South Bend, Indiana; formerly of Center for Solid State Electronics Research, Arizona State University, Tempe, Arizona



# CONTENTS IN BRIEF

---

## **PART I SUBMICRON SEMICONDUCTOR MANUFACTURE**

<b>1 Home</b>	<b>3</b>
<b>2 Overview</b>	<b>15</b>
<b>3 Environment Setup</b>	<b>21</b>
<b>4 Basic Syntax</b>	<b>33</b>
<b>5 Basic Operator</b>	<b>43</b>
<b>6 Variable Type</b>	<b>53</b>





# CONTENTS

---

List of Figures	xiii
List of Tables	xv
Foreword	xvii
Preface	xix
Acknowledgments	xxi
Acronyms	xxiii
Glossary	xxv
List of Symbols	xxvii
Introduction	xxix
<i>Catherine Clark, PhD.</i>	
References	xxix

## **PART I SUBMICRON SEMICONDUCTOR MANUFACTURE**

<b>1 Home</b>	<b>3</b>
1.1 Home	3
	<b>ix</b>

1.1.1	Kenapa alasan memilih Pemograman Phyton	4
1.1.2	Ranah Aplikasi Python	5
1.1.3	interpreter python	5
1.1.4	Komputasi	5
1.1.5	fitur dan keunggulan python	5
1.1.6	Script Python	6
1.1.7	Kelebihan dan Kekurangan	6
1.2	python	7
1.2.1	Pembuatan class pada python	7
1.2.2	Operation	9
1.2.3	Output	9
1.2.4	Conditional	9
1.2.5	Looping	9
1.2.6	Subroutine	9
1.2.7	Sejarah	10
1.2.8	Dukungan Komunitas yang Aktif	10
1.2.9	Kelebihan dan Kekurangan	10
1.2.10	Elemen Dasar Python	12
1.2.11	Pemrograman Mode Script	13
<b>2</b>	<b>Overview</b>	<b>15</b>
2.1	Overview	15
2.1.1	Sejarah Python	17
2.1.2	Kekurangan dan Kelebihan Python	17
2.1.3	Paradigma Pemrogramman Python	17
2.1.4	Instalasi Unix dan Linux	18
2.1.5	Struktur Data	18
2.2	Fitur Overview Pada Python	19
2.3	Fitur Overview Terbaik	19
2.4	Operasi Interface	20
2.4.1	Contoh Perhitungan Matematika	20
2.4.2	ikhtisar yang sering digunakan oleh programmer	20
<b>3</b>	<b>Environtment Setup</b>	<b>21</b>
3.1	Overview Phyton	21
3.1.1	Situs Belajar Python	22
3.2	Environtment Setup For Python	23
3.2.1	Memasang Phyton	24

3.2.2	Menyiapkan PATH	24
3.2.3	installasi Python pada windows	25
3.2.4	Installasi Python pada Unix/Linux	25
3.2.5	Setting Path pada Unix/Linux	26
3.2.6	Fitur Python	26
3.2.7	Pengujian	26
3.2.8	Alasan Menggunakan Phyton	27
3.2.9	Tipe Data Phyton	27
<b>4</b>	<b>Basic Syntax</b>	<b>33</b>
4.1	Basic Syntax	33
4.1.1	Pengenal Python	33
4.1.2	Pemrograman Mode Script	34
4.1.3	Pengertian Python	35
4.1.4	Kelebihan dan Kekurangan Python	35
4.1.5	Fitur Python	37
4.1.6	Fitur Python	37
4.1.7	Pernyataan Multi-Line	38
4.1.8	Kutipan pada Python	38
4.1.9	Kutipan pada Python	39
4.1.10	Komentar pada Python	39
4.1.11	Variabel	40
4.1.12	Whitespace	40
<b>5</b>	<b>Basic Operator</b>	<b>43</b>
5.1	Python Basic Operator	43
5.2	Pengenalan Python Programming	43
5.2.1	Python Basic Operator	44
<b>6</b>	<b>Variabel Type</b>	<b>53</b>
6.1	Variable Type	53
6.1.1	Menetapkan Nilai ke Variabel	54
6.1.2	Aturan Penulisan Variabel	54
6.1.3	Membuat Variabel di Python	55
6.1.4	Menghapus Variabel	55
6.2	Tipe data	55
6.2.1	Jenis-jenis Tipe Data	56
6.2.2	Penggunaan Variabel	56



## LIST OF FIGURES

---

1.1	Logo	7
3.1	Contoh skript pada command prompt.	23
4.1	Gambar Logo Python	35
5.1	Table Operator Aritmatika.	46
5.2	Table Operator Perbandingan.	48



## LIST OF TABLES

---





# FOREWORD

---

This is the foreword to the book.



# PREFACE

---

This is an example preface. This is an example preface. This is an example preface.  
This is an example preface.

R. K. WATTS

*Durham, North Carolina*  
*September, 2007*



## ACKNOWLEDGMENTS

---

From Dr. Jay Young, consultant from Silver Spring, Maryland, I received the initial push to even consider writing this book. Jay was a constant “peer reader” and very welcome advisor during this year-long process.

To all these wonderful people I owe a deep sense of gratitude especially now that this project has been completed.

G. T. S.



## ACRONYMS

---

ACGIH	American Conference of Governmental Industrial Hygienists
AEC	Atomic Energy Commission
OSHA	Occupational Health and Safety Commission
SAMA	Scientific Apparatus Makers Association





## GLOSSARY

---

NormGibbs	Draw a sample from a posterior distribution of data with an unknown mean and variance using Gibbs sampling.
pNull	Test a one sided hypothesis from a numerically specified posterior CDF or from a sample from the posterior
sintegral	A numerical integration using Simpson's rule



## SYMBOLS

---

- $A$  Amplitude
- $\&$  Propositional logic symbol
- $a$  Filter Coefficient
  
- $\mathcal{B}$  Number of Beats



# INTRODUCTION

---

CATHERINE CLARK, PHD.

Harvard School of Public Health  
Boston, MA, USA

The era of modern began in 1958 with the invention of the integrated circuit by J. S. Kilby of Texas Instruments His first chip is shown in Fig. I. For comparison, Fig. I.2 shows a modern microprocessor chip.

This is the introduction. This is the introduction. This is the introduction. This is the introduction. This is the introduction. This is the introduction.

$$ABCDEF\alpha\beta\Gamma\Delta\sum_{def}^{abc} \tag{I.1}$$

## REFERENCES

1. J. S. Kilby, "Invention of the Integrated Circuit," *IEEE Trans. Electron Devices*, **ED-23**, 648 (1976).
2. R. W. Hamming, *Numerical Methods for Scientists and Engineers*, Chapter N-1, McGraw-Hill, New York, 1962.
3. J. Lee, K. Mayaram, an in LDD MOSFETs" *IEEE Electron Device Lett.*, **EDL-7**(3). 152 (1986).



## PART I

---

# SUBMICRON SEMICONDUCTOR MANUFACTURE

---





# CHAPTER 1

---

## HOME

---

### 1.1 Home

Pemrograman python adalah bahasa pemrograman terpopuler di tahun 2016 menurut tiobe. Python juga memiliki sintak atau aturan penulisan code pemrograman. Salah satu bagian Home merupakan halaman pengantar untuk mempelajari python . sebelum ketahapan yang baru selain home ini pembaca memerlukan pengertian yang lain yaitu seperti enverinmoment setup, syntax dan lain lain, awal untuk penulis jelakan yaitu pengertian tentang class pada python untuk mengantarkan logika dan pengetahuan apa itu class. Bahasa Pemrograman Python sudah dirasakan cukup matang untuk memecahkan masalah dalam dunia komputasi dan pengembangan sistem informasi. Manajemen paket Red Hat Linux, Blender, PyGame, ZOPE (Framework Python untuk web services) telah membuktikan kepada kita semua bahwa Python merupakan bahasa pemrograman yang perlu diajarkan pada perguruan tinggi, terutama konsep interpreter, object oriented programming dan lainnya. Phyton dapat berjalan di banyak platform atau sistem operasi seperti windows, linux/unix, mac OS X, OS/2, amiga, palm handhelds dan telepon genggam nokia. namun saat ini python juga sudah masuk kedalam virtual java dan NET. Python memiliki beberapa keunggulan, yaitu :

1. Syntaxnya sangat bersih dan mudah dibaca
2. Kemampuan melakukan pengekan syntaxnya yang kuat
3. Berorientasi objek secara intuitif
4. Kode-kode prosedur dinyatakan pada ekspresi natural
5. Modularitas yang penuh, mendukung hirarki paket
6. Penanganan error dilakukan berdasar pada eksepsi
7. Tipe-tipe data dinamis berada pada tingkat sangat tinggi
8. Library standar dapat diperluas dan modul dari pihak ketiga dapat dibuat secara virtual untuk setiap kebutuhan
9. Ekstensi dan modul-modul dapat secara mudah ditulis dalam C, C+ (atau java untuk Jython atau NET untuk IronPytho)
10. Dapat dimasukkan kedalam aplikasi sebagai antar muka skrip

#### 1.1.1 Kenapa alasan memilih Pemograman Phyton

1. Python memiliki konsep desain yang bagus dan sederhana, yang berfokus pada kemudahan dalam penggunaan. Kode Python dirancang untuk mudah dibaca, dipelajari, digunakan ulang, dan dirawat. Selain itu, Python juga mendukung pemograman berorientasi objek dan pemograman fungsional.
2. Python dapat meningkatkan produktivias dan menghemat waktu bagi para programmer. Untuk memperoleh hasil program yang sama, kode Python juga jauh lebih sedikit dibanding dengan kode yang ditulis menggunakan bahasa-bahasa pemograman lain seperti C, C++, C# maupun Java. Coba lihat ilustrasi gambar dibawah ini: Bagaimana menurut kamu ? Sudah pasti Pyhton lebih sederhana dibanding bahasa-bahasa pemograman lain.
3. Program yang ditulis menggunakan Python ,dapat dijalankan di hampir semua sistem operasi (Unix, Windows, Mac OS X, dll), termasuk untuk perangkat-perangkat mobile.
4. Python memiliki banyak dukungan pustaka yang dikembangkan oleh pihak ketiga, misalnya pustaka untuk pengembangan web, pengembangan aplikasi visual (berbasis GUI), pengembangan permainan komputer (game), dan masih banyak lagi yang lainnya.
5. Melalui mekanisme tertentu, kode Python dapat diintegrasikan dengan aplikasi yang ditulis dalam bahasa pemograman lain. Sebagai contoh, kode Python dapat dipanggil dari kode C/C++, dan begitu juga perkembangan .NET Framework.
6. Terakhir, Python bersifat gratis atau bebas (free) dan open source, meskipun digunakan untuk kepentingan komersial.

### 1.1.2 Ranah Aplikasi Python

Python dapat digunakan untuk membangun aplikasi-aplikasi berjalan pada banyak fungsi. diantaranya adalah sebagai berikut :

1. Pengembangan web dan internet. Dimana python mampu mengembangkan web dan internet seperti : penulisan skrip Common Gateway Internet (CGI), pengembangan frameworks seperti django dan turbo gears, python juga mendukung penuh HTML, dan XML, pemrosesan e-mail, pemrosesan RSS feeds dll.
2. Akses terhadap database. antar muka open database connectivity (ODBC) untuk MySQL, Oracle, PostgreSQL, Sybase ODBC. dan juga mampu menyediakan Application Programming Interface (API)
3. Pengembangan Graphical User Interface (GUI) pada Desktop
4. Keperluan perhitungan scientific dan numeris.
5. Pengembangan perangkat lunak komputer.
6. Pengembangan jaringan komputer.

### 1.1.3 interpreter python

Bahasa pemrograman python dilengkapi dengan suatu fasilitas seperti shell di linux, yang digunakan secara berulang dikemudian hari. untuk keperluan penulisan ekspresi kompleks, kita dapat membuatnya dalam sebuah script yang dibantu dengan adanya teks editor. berikut adalah contoh dari statement dasar, perulangan dan seleksi

### 1.1.4 Komputasi

komputasi dengan bahasa: Statistik Sederhana mari kita kembali ke eksplorasi kita tentang cara membawa sumber daya komputasi kita untuk menghasilkan teks dalam jumlah banyak. Pada bagian ini, kita mengambil pertanyaan tentang apa yang membuat teks menjadi distinct, dan gunakan metode otomatis untuk menemukan kata-kata dan ekspresi teks yang khas. Sebelum melanjutkan lebih lanjut, Anda mungkin ingin memeriksa pemahaman Anda tentang bagian terakhir dengan memprediksikan output dari kode berikut. Anda bisa menggunakan penerjemah untuk memeriksa apakah Anda benar. Jika Anda tidak yakin bagaimana melakukan tugas ini, sebaiknya Anda meninjau bagian sebelumnya sebelum melanjutkan lebih jauh.

### 1.1.5 fitur dan keunggulan python

Beberapa fitur yang dapat dikatakan sebagai keunggulan python yaitu :

1. Python is powerful and fast, artinya python merupakan suatu kumpulan modul-modul yang sangat baik dan dapat menangani secara praktis setiap domain masalah.
2. Python plays well with others, artinya python dapat berintegrasi dengan Component Object Model (COM), NET dan objek-objek Common Object Request Broker Architecture (COBRA).
3. Python runs everywhere, artinya python tersedia untuk berbagai macam sistem operasi seperti windows, unix atau linux, OS/2, mac, Amiga dll.
4. Python is friendly and easy to learn, artinya python sangat bersahabat bagi penggunaanya.
5. Python is open, artinya python dibawah lisensi open source yang membuatnya dapat disebarakan secara bebas bahkan untuk keperluan komersil.

### 1.1.6 Script Python

Seringkali pengguna harus menuliskan ekspresi yang cukup kompleks dan akan digunakan secara berulang di kemudian hari. Untuk keperluan penulisan ekspresi kompleks, kita dapat membuatnya dalam sebuah script yang dibantu dengan adanya teks editor. Penulis menggunakan vi teks editor default yang terdapat pada distro GNU/Linux.

### 1.1.7 Kelebihan dan Kekurangan

#### 1.1.7.1 Kelebihan

1. Tidak ada tahapan kompilasi dan penyambungan (link) sehingga kecepatan perubahan pada masa pembuatan sistem aplikasi meningkat.
2. Tidak ada deklarasi tipe data yang rumitkan sehingga program menjadi lebih sederhana, singkat, dan fleksible.
3. Manajemen memori otomatis yaitu kumpulan sampah memori sehingga dapat menghindari pencacatan kode.
4. Tipe data dan operasi tingkat tinggi yaitu kecepatan pembuatan sistem aplikasi menggunakan tipe objek yang telah ada.
5. Pemrograman berorientasi objek.
6. Pelekatan dan perluasan dalam C.
7. Terdapat kelas, modul, eksepsi sehingga terdapat dukungan pemrograman skala besar secara modular.

8. Pemuatan dinamis modul C sehingga ekstensi menjadi sederhana dan berkas biner yang kecil.
9. Pemuatan kembali secara dinamis modul python seperti memodifikasi aplikasi tanpa menghentikannya.



Figure 1.1 Logo

## 1.2 python

Pemrograman python adalah bahasa pemrograman terpopuler di tahun 2016 menurut tiobe. Python juga memiliki sintaks atau aturan penulisan *code* pemrograman. Salah satu bagian Home merupakan halaman pengantar untuk mempelajari python. Sebelum ketahapan yang baru selain home ini pembaca memerlukan pengertian yang lain yaitu seperti environment setup, syntax dan lain lain, awal untuk penulis jelakan yaitu pengertian tentang *class* pada python untuk mengantarkan logika dan pengetahuan apa itu *class*.

### 1.2.1 Pembuatan class pada python

Kita awali dengan sebuah kata kunci. Yaitu “class” yang kemudian diikuti dengan “nama class nya” terakhir membuat kurung buka dan tutup serta membuat tanda titik dua “()” dan “:” kalo syntax nya, seperti :

```
namaClass()
```

untuk memanggil metode kita cukup menggunakan kata kunci “class” yang kemudian diikuti dengan memanggil nama metode yang tersedia didalam class tersebut dengan dipisahkan oleh tanda titik seperti:

```
namaClass().namaMetode()
```

ingin lebih mudah kita tampung classnya dulu ke variable

```
penampung = namaClass()
```

```
penampung.namaMetode()
```

Didalam sebuah class yang dibuat biasanya terdapat init itu disediakan langsung oleh pythonnya, seperti :

```
class namaClass():
def init (self,parameter):
itu code program yang pertama kali kalian buat
def metode 1 (self,parameter):
isi metode
def metode 2 (self):
isi metode
```

Setelah menjelaskan class kita akan menjelaskan variable seperti tadi penampung class, nah variable bisa diartikan sebagai huruf atau kata tujuannya untuk mempermudah proses penulisan sebuah program. Contohnya dalam kehidupan nyata seperti gelas atau ember, ambil contoh ember kita ketahui bahwa ember bisa diisi dengan air, pasir, tanah dan lain-lain, nah ember itu sebagai variable dan isi variabelnya itu adalah air, tanah, pasir dan lain-lain. Contoh variabelnya seperti ini :

```
Variable= "ini string atau teks"

Print( "nilai isi dari variable1 adalah: ",variable1)
Print( "nilai atau isi dari variable2 adalah:",variable2)
```

Ada beberapa hal yang harus diketahui seperti input, data operation dan lain-lain seperti:

- input
- data
- opration
- output
- condutional
- looping
- subroutine

Tahapan ini merupakan proses memasukan data kedalam proses komputer melalui peralatan input. Pada bahasa Python, untuk menerima masukan dari pengguna yaitu dengan menggunakan method input() dan raw input(). Data adalah bahan mentah yang akan diolah menjadi informasi sehingga dapat berguna dan dimanfaatkan oleh pengguna. Data dapat berupa variabel, konstanta, atau yang berisi bilangan, kalimat, dan lainnya. Tipe data berupa string, number, list, tuple, dan lainnya.

### 1.2.2 Operation

Operation adalah yang akan mengubah suatu nilai menjadi nilai lain. Yang termasuk operation atau yang biasa disebut dengan operator adalah operator aritmatika, operator assignment, dan lainnya.

### 1.2.3 Output

Output adalah menuliskan informasi yang ditampilkan dilayar, disk, atau ke salah satu unit I/O. Pada Python 2.0, untuk menampilkan output dengan menulis syntax print. Sedangkan pada Python 3.0 dengan menggunakan fungsi print().

### 1.2.4 Conditional

Conditional merupakan jumlah perintah yang akan dijalankan jika kondisi tertentu sudah terpenuhi. Jika *username* dan *password* yang dimasukan benar, maka akan menampilkan halaman utama. Hal ini bisa disebut *conditional*. Pada *conditional*, Python menggunakan pernyataan if, else, dan elif.

### 1.2.5 Looping

Perintah yang akan berjalan beberapa kali, selama kondisi yang ditentukan atau kondisi yang terpenuhi. Pada looping ini, Python menggunakan pernyataan for dan while untuk melakukan perulangan.

### 1.2.6 Subroutine

Perintah yang bisa dijalankan dengan cara memanggil namanya. Sering disebut sebagai function atau method. Pada bahasa pemrograman Python, untuk menggunakan function atau method yaitu dengan menggunakan pernyataan def nama function(). Fungsi def dalam python, Penggunaan fungsi tanpa parameter Command=fungsi() Deklarasi command= def fungsi() Pemanggilan fungsi, parameter sesuai dengan kata kunci seperti tadi class Command= fungsi(arg=1, arg2=2) Deklarasi command – def fungsi (arg2,arg2) Pemanggilan fungsi, parameter sesuai dengan posisi Command= fungsi() Deklarasi command – def fungsi (x) Pemanggilan fungsi parameter sesuai dengan argument posisional tuple Command= fungsi((1,2).(1,3)) Deklarasi command – def fungsi (\*args) Pemanggilan fungsi, parameters sesuai argument kata kunci dictionary Command= fungsi (bahasa= 'python',versi='2.2') Deklarasi command = def fungsi (\*\*args) Itu adalah cara memanggil dalam code python pemrograman jadi ada nenerapa fungsi yang dibutuhkan penulisan dengan tepat maka sebab itu dengan sebab itu buaat lah penulisan yang mudah. Karena pemrograma python sangat sensitive bila ada kesalahan sedikit di penulisan atau symbol yang tertinggal.



### 1.2.7 Sejarah

Bahasa pemrograman Python adalah bahasa yang dibuat oleh seorang keturunan Belanda yaitu Guido van Rossum. Awalnya, pembuatan bahasa pemrograman ini adalah untuk membuat skrip bahasa tingkat tinggi pada sebuah sistem operasi yang terdistribusi Amoeba. Python telah digunakan oleh beberapa pengembang dan bahkan digunakan oleh beberapa perusahaan untuk pembuatan perangkat lunak komersial. Pemrograman bahasa python ini adalah pemrogram gratis atau freeware, sehingga dapat dikembangkan, dan tidak ada batasan dalam penyalinannya dan mendistribusikan.

### 1.2.8 Dukungan Komunitas yang Aktif

Python adalah salah satu pemrograman yang terus berkembang dan bertahan dikarenakan dukungan komunitas yang aktif diseluruh dunia. Banyak forum-forum ataupun blogger-blogger yang sering membagi pengalaman dalam menggunakan python. Hal ini memudahkan bagi pengguna pemula maupun pengembang untuk bertanya dan sharing tentang ilmu pemrograman ini.

### 1.2.9 Kelebihan dan Kekurangan

Kelebihan :

- (a) Tidak ada tahapan kompilasi dan penyambungan (link) sehingga kecepatan perubahan pada masa pembuatan sistem aplikasi meningkat.
- (b) Tidak ada deklarasi tipe data yang merumitkan sehingga program menjadi lebih sederhana, singkat, dan fleksible.
- (c) Manajemen memori otomatis yaitu kumpulan sampah memori sehingga dapat menghindari pencacatan kode.
- (d) Tipe data dan operasi tingkat tinggi yaitu kecepatan pembuatan sistem aplikasi menggunakan tipe objek yang telah ada.
- (e) Pemrograman berorientasi objek.
- (f) Pelekatan dan perluasan dalam C.
- (g) Terdapat kelas, modul, eksepsi sehingga terdapat dukungan pemrograman skala besar secara modular.
- (h) Pemuatan dinamis modul C sehingga ekstensi menjadi sederhana dan berkas biner yang kecil
- (i) Pemuatan kembali secara dinamis modul python seperti memodifikasi aplikasi tanpa menghentikannya.
- (j) Model objek universal kelas Satu.
- (k) Konstruksi pada saat aplikasi berjalan.

- (l) Interaktif, dinamis dan alamiah.
- (m) Akses hingga informasi interpreter.
- (n) Portabilitas secara luas seperti pemrograman antar platform tanpa ports
- (o) Kompilasi untuk portable kode byte sehingga kecepatan eksekusi bertambah dan melindungi kode sumber.
- (p) Antarmuka terpasang untuk pelayanan keluar seperti perangkat Bantu system, GUI, persistence, database, dll.

Kekurangan:

- (a) 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.
- (b) Disebabkan python merupakan interpreter, python bukan merupakan perangkat bantu terbaik untuk pengantar komponen performa kritis.
- (c) Python tidak dapat digunakan sebagai dasar bahasa pemrograman implementasi untuk beberapa komponen, tetapi dapat bekerja dengan baik sebagai bagian depan skrip antarmuka untuk mereka.
- (d) 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.
- (e) Banyak terdapat referensi lama terutama dari pencarian google, python adalah pemrograman yang sangat lambat. Namun belum lama ini ditemukan bahwa Google, Youtube, DropBox dan beberapa software sistem banyak menggunakan Python.
- (f) Kini Python menjadi salah satu bahasa pemrograman yang populer digunakan oleh pengembangan web, aplikasi web, aplikasi perkantoran, simulasi, dan masih banyak lagi. Hal ini disebabkan karena Python bahasa pemrograman yang dinamis dan mudah dipahami.
- (g) Selain itu, sekarang telah tersedia berbagai situs kursus yang bagus untuk mempelajari bahasa pemrograman Python ini sehingga pembaca maupun developer pemula yang akan mempelajari bahasa ini akan menjadi lebih mudah karena dapat berlatih dimanapun dan kapanpun selama terhubung dengan Internet.
- (h) Menariknya, berbagai situs kursus gratis ini menawarkan metode pembelajaran yang interaktif sehingga mudah dimengerti oleh pesertanya.

Python merupakan pemrograman yang tidak pernah dicompile secara full. Jika kamu sudah menyelesaikan programnya dan kamu ingin mengirim ke teman atau dibagikan ke internet maka teman atau orang lain dapat mengubah kode

diprogram kamu karena program dibuka di notepad, python akan tetap berbentuk kode yang sama tidak acak acakan sehingga orang lain dapat memahami pemrograman yang kamu buat.

```
>>> Python 2.4.3 ( #1, Nov 11 2010, 13:34:43) [GCC 4.1.2 20080704 (Red
Hat 4.1.2-48)] on linux2 Type "help", "copyright", "credits" or "license" for
more information.
```

Ketik teks berikut pada prompt Python dan lalu tekan Enter:

```
print ``Hello, Python!``
```

Jika Anda menjalankan versi baru Python, Anda perlu menggunakan pernyataan cetak dengan tanda kurung seperti pada cetak ("Halo, Python;"). Namun dengan versi Python 2.4.3, ini menghasilkan hasil sebagai berikut:

Hello, Python!

### 1.2.10 Elemen Dasar Python

Bahasa pemrograman python memiliki beberapa elemen penting, diantaranya yaitu :

- (a) Himpunan Karakter Semua karakter di python terdiri dari huruf besar, huruf kecil, angka dan simbol-simbol lainnya yang terdapat pada keyword dapat digunakan untuk menandai atau memberikan nama terhadap sebuah data atau object.
- (b) Pengenal (identifier) merupakan suatu nama yang biasa dipakai oleh user (programmer) untuk menamakan : variabel, konstanta, tipe data, fungsi, label, objek, serta hal-hal lain. Karena python memiliki penamaan case sensitive maksudnya huruf besar dan kecil berbeda. Contoh : No\_mahasiswa berbeda dengan no\_mahasiswa
- (c) Nama Variabel harus diawali dengan huruf atau karakter garis bawah, selanjutnya dapat diikuti dengan huruf maupun angka atau tanda garis bawah. Nama Variabel tidak boleh diawali dengan angka.
- (d) Nama Variabel tidak boleh menggunakan operator-operator aritmatika dan karakter khusus.
- (e) jika nama variabel terdiri dari dua atau lebih, maka antar kata tidak diperbolehkan menggunakan spasi
- (f) Nama Variabel tidak boleh menggunakan kata-kata yang telah memiliki arti khusus dalam bahasa python.
- (g) Penggunaan huruf kecil dan huruf besar dibedakan.
- (h) Panjang maksimal suatu variabel adalah 32 karakter sehingga jika mendeklarasikan suatu variabel yang panjangnya lebih dari 32 karakter, secara otomatis sistem tetap akan mengenali sepanjang 32 karakter saja.

#### (i) Kata Kunci

Kata kunci (keywords) merupakan pengenalan ke sistem python yang mempunyai arti khusus bagi interpreter python dan kata kunci juga merupakan kumpulan kata yang dicadangkan. kata-kata ini tidak boleh digunakan sebagai identifier.

### 1.2.11 Pemrograman Mode Script

Memohon interpreter dengan parameter script memulai eksekusi script dan berlanjut sampai script selesai. Saat skrip selesai, juru bahasa tidak lagi aktif. Mari kita tuliskan program Python sederhana dalam sebuah naskah. File Python memiliki ekstensi .py. Ketik kode sumber berikut di file.

Objek Dengan Python, seperti semua bahasa berorientasi objek, ada kumpulan kode dan data yang disebut objek, yang biasanya mewakili potongan dalam model konseptual suatu sistem. Objek dengan Python dibuat (yaitu, instantiated) dari template yang disebut kelas (yang akan dibahas kemudian, sebanyak bahasa dapat digunakan tanpa memahami kelas). Mereka memiliki atribut, yang mewakili berbagai potongan kode dan data yang membentuk objek. Untuk mengakses atribut, seseorang menuliskan nama objek yang diikuti oleh suatu periode (selanjutnya disebut titik), diikuti dengan nama atribut.

Contohnya adalah atribut 'atas' dari string, yang mengacu pada kode yang mengembalikan salinan string di mana semua huruf adalah huruf besar. Untuk mendapatkan ini, perlu untuk memiliki cara untuk merujuk ke objek (dalam contoh berikut, jalan adalah string literal yang membangun objek).

Paradigma : Multi-paradigm: object-oriented, imperative, functional, procedural, reflective Muncul Tahun : 1991 Perancang : Guido van Rossum Pengembangan : Python Software Foundation Rilis terbaru : 3.2.3 / 11 April 2012; 46 hari lalu 2.7.3 / 11 April 2012; 46 hari lalu / Sistem pengetikan: duck, dynamic, strong Implementasi : CPython, IronPython, Jython, Python for S60, PyPy Dialek : Cython, RPython, Stackless Python Terpengaruh oleh : ABC, ALGOL 68, C, C++, Dylan Haskell, Icon, Java, Lisp, Modula-3, Perl Mempengaruhi : Boo, Cobra, D, Falcon, Groovy, JavaScript, Ruby Sistem operasi : Cross-platform Lisensi : Python Software Foundation License, GNU GPL

Situs web : [python.org](http://python.org)



## CHAPTER 2

---

## OVERVIEW

---

### 2.1 Overview

Python adalah bahasa open source yang modern, ditafsirkan, berorientasi objek, yang digunakan dalam semua jenis rekayasa perangkat lunak. Meskipun telah ada selama dua dekade, namun telah mulai digunakan dalam ilmu atmosfer beberapa tahun yang lalu setelah komunitas pengembang berkumpul di atas paket ilmiah standar (misalnya, penanganan array) yang dibutuhkan untuk karya ilmu atmosfer. Python sekarang merupakan platform integrasi yang kuat untuk semua jenis ilmu pengetahuan atmosfer, mulai dari analisis data sampai komputasi terdistribusi, dan antarmuka pengguna grafis dengan sistem informasi geografis. Di antara fitur yang menonjol, Python memiliki sintaks yang ringkas namun alami untuk kedua array dan nonarrays, membuat program sangat jelas dan mudah dibaca; seperti kata pepatah, "Python adalah pseudocode yang dapat dieksekusi." Juga, karena bahasa itu ditafsirkan, perkembangannya jauh lebih mudah; Anda tidak perlu menghabiskan waktu ekstra memanipulasi compiler dan linker. Selain itu, struktur data modern dan sifat object-oriented bahasa membuat kode Python lebih kuat dan kurang rapuh. Akhirnya, silsilah open

source Python, dibantu oleh basis pengguna dan pengembang besar di industri dan juga sains, berarti program Anda dapat memanfaatkan puluhan ribu paket Python yang ada. Ini termasuk visualisasi, perpustakaan numerik, interkoneksi dengan bahasa yang dikompilasi dan lainnya, caching memori, layanan Web, pemrograman antarmuka pengguna grafis mobile dan desktop, dan lain-lain. Dalam banyak kasus, beberapa paket ada di masing-masing area domain di atas. Anda tidak terbatas hanya pada apa yang bisa diberikan oleh satu vendor atau bahkan hanya yang bisa diberikan oleh komunitas ilmiah.

Python bahasa script tingkat tinggi, ditafsirkan, interaktif dan berorientasi objek. Python dirancang agar mudah dibaca. Ini menggunakan kata kunci bahasa Inggris sering di mana bahasa lainnya menggunakan tanda baca, dan memiliki konstruksi sintaksis lebih sedikit daripada bahasa lainnya. Python is interpreted : diproses pada saat runtime oleh interpreter. Anda tidak perlu mengkompilasi program anda sebelum menjalankannya. Ini mirip dengan PERL dan PHP. Tidak perlu untuk mengkompilasi program anda sebelum mengeksekusi itu. Hal ini merupakan mirip dengan php. Python is Interactive: Anda dapat benar-benar duduk di prompt Python dan berinteraksi dengan penafsir langsung untuk menulis program Anda. Python is Object-Oriented: Python mendukung gaya Berorientasi Objek atau teknik pemrograman yang merangkum kode di dalam objek. Python is a Beginner's Language: Python adalah bahasa yang besar untuk programmer tingkat pemula dan mendukung pengembangan berbagai aplikasi dari pengolahan teks sederhana untuk browser WWW untuk game. Fitur overview dalam python itu adalah :

- (a) Easy-to-learn: Python memiliki beberapa kata kunci, struktur sederhana, dan sintaks yang jelas. Hal ini memungkinkan siswa untuk mengambil bahasa dengan cepat.
- (b) Easy-to-read: kode Python lebih jelas dan terlihat mata.
- (c) Easy-to-maintain: kode sumber Python cukup mudah-untuk-menjaga.

Fitur overview terbaik adalah:

- (a) IT mendukung metode pemrograman fungsional dan terstruktur serta OOP.
- (b) Hal ini dapat digunakan sebagai bahasa scripting atau dapat dikompilasi untuk byte-kode untuk membangun aplikasi besar.
- (c) Ini memberikan tingkat tinggi sangat tipe data dinamis dan mendukung memeriksa jenis dinamis.
- (d) IT mendukung pengumpulan sampah otomatis.
- (e) Hal ini dapat dengan mudah diintegrasikan dengan C, C ++, COM, ActiveX, CORBA, dan Java.

Fitur overview dalam python itu adalah:

- (a) Easy-to-learn: Python memiliki beberapa kata kunci, struktur sederhana, dan sintaks yang jelas. Hal ini memungkinkan siswa untuk mengambil bahasa dengan cepat.

- (b) Easy-to-read: kode Python lebih jelas dan terlihat mata.
- (c) Easy-to-maintain: kode sumber Python cukup mudah-untuk-menjaga.

### 2.1.1 Sejarah Python

Python dikembangkan oleh Guido van Rossum pada akhir tahun delapan puluhan dan awal tahun sembilan puluhan di National Research Institute for Mathematics and Computer Science di Belanda. Python berasal dari banyak bahasa lain, termasuk ABC, Modula-3, C, C ++, Algol-68, SmallTalk, dan shell Unix dan bahasa script lainnya. Fitur overview terbaik adalah IT mendukung metode pemrograman fungsional dan terstruktur serta OOP. Hal ini dapat digunakan sebagai bahasa scripting atau dapat dikompilasi untuk byte-kode untuk membangun aplikasi besar. Ini memberikan tingkat tinggi sangat tipe data dinamis dan mendukung memeriksa jenis dinamis. IT mendukung pengumpulan sampah otomatis. Hal ini dapat dengan mudah diintegrasikan dengan C, C ++, COM, ActiveX, CORBA, dan Java. Hal tersebut menjadi terpopuler karena kemudahan bagi programmer yang menjadikan python pemrograman terbaik pada tahun 2016.

### 2.1.2 Kekurangan dan Kelebihan Python

Kelebihan :

- (a) Tidak ada tahapan kompilasi dan penyambungan (link) sehingga kecepatan perubahan pada masa pembuatan sistem aplikasi meningkat.
- (b) Tidak ada deklarasi tipe data yang rumit sehingga program menjadi lebih sederhana, singkat, dan fleksible.

### 2.1.3 Paradigma Pemrograman Python

Pada program yang selama ini kita buat, kita mendesain program kita berdasarkan fungsi (blok statemen yang memanipulasi data). Hal ini disebut pemrograman procedure-oriented. Ada cara lain untuk mengorganisasi program dengan menggabungkan data dan operasi yang dibungkus dalam suatu obyek yaitu paradigma pemrograman berorientasi obyek. Dalam perancangan bahasa, para perancang bahasa mengikuti paradigma-paradigma tertentu yang merupakan bentuk pemecahan masalah mengikuti aliran atau “genre” tertentu dari program dan bahasa. Berikut ini merupakan paradigma-paradigma pemrograman yang utama:

- (a) Imperative programming- $\bar{t}$  program terdiri dari instruksi yang membentuk perhitungan, menerima input dan menghasilkan output. Contoh bahasa: Fortran, C, dan C++.



- (b) Object-oriented (OO) programming-ꦏꦺꦴꦥꦸꦭꦤ꧀ objek yang saling berinteraksi melalui pesan yang mengubah state mereka. Contoh bahasa: Java, C++.
- (c) Functional programming-ꦏꦸꦩꦸꦭꦤ꧀ fungsi matematika dengan input (domain) dan hasil (range). Fungsi-fungsi saling berinteraksi dan berkombinasi menggunakan komposisi fungsional, kondisional, dan rekursif. Contoh bahasa: Lisp, Scheme, ML
- (d) Logic (declarative) programming -ꦏꦺꦴꦩꦸꦭꦤ꧀ masalah menggunakan bahasa deklaratif, yang terdiri dari fakta dan aturan. Contoh bahasa : Prolog
- (e) Event-driven programming-ꦏꦸꦩꦸꦭꦤ꧀ program merupakan sebuah loop yang secara kontinu merespon event yang timbul oleh perintah yang tidak terduga. Event ini berasal dari aksi user pada layar atau sumber lainnya. Contoh bahasa: Visual Basic dan Java.
- (f) Concurrent programming-ꦏꦸꦩꦸꦭꦤ꧀ program merupakan sekumpulan proses yang bekerjasama, saling berbagi informasi dari waktu ke waktu tapi biasanya beroperasi secara tidak serempak. Contoh bahasa : SR, Linda, dan HPF.

#### 2.1.4 Instalasi Unix dan Linux

Sistem operasi LINUX adalah salah satu alternatif yang digunakan untuk menggantikan sistem operasi Windows. Perkembangan LINUX sangat cepat karena sistem operasi ini dikembangkan oleh semua orang yang ingin mengembangkannya. Tentu saja membuat sistem operasi LINUX terasa semakin user friendly. Bagi sekolah-sekolah yang sudah mempunyai fasilitas, maka sistem operasi LINUX lebih banyak memiliki kelebihan untuk koneksi ke internet karena LINUX hingga saat ini jarang sekali terkena virus sehingga cukup aman digunakan untuk internet. Berikut adalah langkah-langkah sederhana untuk menginstal Python di mesin Unix / Linux :

- (a) Ikuti link untuk mendownload kode sumber zip yang tersedia untuk Unix / Linux.
- (b) Download dan ekstrak file.
- (c) Mengedit Modul / Setup file jika Anda ingin menyesuaikan beberapa pilihan.
- (d) Jalankan ./configure script33
- (e) Ini menginstal Python di lokasi standar / usr / local / bin
- (f) dan pustakanya di / usr / local / lib / pythonXX dimana XX adalah versi Python.

#### 2.1.5 Struktur Data

Python adalah bahasa yang diketik secara dinamis , nilai Python , bukan variabel, tipe carry. Ini berimplikasi pada banyak aspek dari cara fungsi bahasa.

Semua variabel dalam Python menyimpan referensi ke objek, dan referensi ini dilewatkan ke fungsi; sebuah fungsi tidak dapat mengubah nilai referensi variabel dalam fungsi pemanggilannya. Beberapa orang telah memanggil skema parameter-passing ini "Call by object reference." Di antara bahasa yang diketik secara dinamis, Python cukup tipe-check. Konversi implisit didefinisikan untuk tipe numerik (dan juga boolean), jadi seseorang dapat secara sah mengalikan bilangan kompleks dengan bilangan bulat panjang (misalnya) tanpa pengecoran eksplisit. Namun, tidak ada konversi implisit antara (misalnya) angka dan string; string adalah argumen yang tidak valid terhadap fungsi matematis yang mengharapkan sebuah angka.

## 2.2 Fitur Overview Pada Python

- (a) Easy-to-learn: Python memiliki beberapa kata kunci, struktur sederhana, dan sintaks yang jelas. Hal ini memungkinkan siswa untuk mengambil bahasa dengan cepat.
- (b) Easy-to-read: kode Python lebih jelas dan terlihat mata.
- (c) Easy-to-maintain: kode sumber Python cukup mudah-untuk-menjaga.
- (d) A broad standard library: bulk Python perpustakaan sangat portabel dan cross-platform yang kompatibel pada UNIX, Windows, dan Macintosh.
- (e) Interactive Mode: Python memiliki dukungan untuk mode interaktif yang memungkinkan pengujian interaktif dan debugging dari potongan kode.
- (f) Portable: Python dapat dijalankan pada berbagai macam platform perangkat keras dan memiliki antarmuka yang sama pada semua platform.
- (g) Extendable: Anda dapat menambahkan modul tingkat rendah ke interpreter Databases: Python menyediakan antarmuka untuk semua database komersial utama.
- (h) GUI Programming: Python mendukung aplikasi GUI yang dapat dibuat dan porting ke banyak panggilan sistem, perpustakaan dan sistem jendela, seperti Windows Scalable: Python menyediakan struktur dan dukungan yang lebih baik untuk program besar dari shell scripting.

## 2.3 Fitur Overview Terbaik

IT mendukung metode pemrograman fungsional dan terstruktur serta OOP. Hal ini dapat digunakan sebagai bahasa scripting atau dapat dikompilasi untuk byte-kode untuk membangun aplikasi besar. Ini memberikan tingkat tinggi sangat tipe data dinamis dan mendukung memeriksa jenis dinamis. IT mendukung pengumpulan sampah otomatis. Hal ini dapat dengan mudah diintegrasikan dengan C, C ++, COM, ActiveX, CORBA, dan Java. Hal tersebut menjadi terpopuler karena kemudahan bagi programmer yang menjadikan python pemrograman terbaik pada tahun 2016.

## 2.4 Operasi Interface

### 2.4.1 Contoh Perhitungan Matematika

```
>>>import math
>>>math.cos(math.pi / 4)
>>>0.70710678118654757
>>>math.log(1024, 2)
>>>10.0
```

### 2.4.2 ikhtisar yang sering digunakan oleh programmer

- (a) pengenalan python
- (b) mengisntal python
- (c) matematika + numbers
- (d) string
- (e) list
- (f) if elif else
- (g) for loop
- (h) for else range breake
- (i) pass dan continue
- (j) while loop
- (k) function
- (l) function dan arguments
- (m) return velue
- (n) lamda function
- (o) sope global dan local
- (p) more on list
- (q) sticks dan queues

## CHAPTER 3

---

# ENVIRONMENT SETUP

---

### 3.1 Overview Python

Python adalah bahasa script tingkat tinggi, ditafsirkan, interaktif dan berorientasi objek. Python dirancang agar mudah dibaca. Ini menggunakan kata kunci bahasa Inggris sering di mana bahasa lainnya menggunakan tanda baca, dan memiliki konstruksi sintaksis lebih sedikit daripada bahasa lainnya. Python diinterpretasikan: Python diproses pada saat runtime oleh interpreter. Anda tidak perlu mengkompilasi program Anda sebelum menjalankannya. Ini mirip dengan PERL dan PHP. Python mendukung multi paradigma pemrograman, utamanya; namun tidak dibatasi; pada pemrograman berorientasi objek, pemrograman imperatif, dan pemrograman fungsional. Salah satu fitur yang tersedia pada python adalah sebagai bahasa pemrograman dinamis yang dilengkapi dengan manajemen memori otomatis. Seperti halnya pada bahasa pemrograman dinamis lainnya, python umumnya digunakan sebagai bahasa skrip meski pada praktiknya penggunaan bahasa ini lebih luas mencakup konteks pemanfaatan yang umumnya tidak dilakukan dengan menggunakan bahasa skrip. Python da-

pat digunakan untuk berbagai keperluan pengembangan perangkat lunak dan dapat berjalan di berbagai platform sistem operasi.

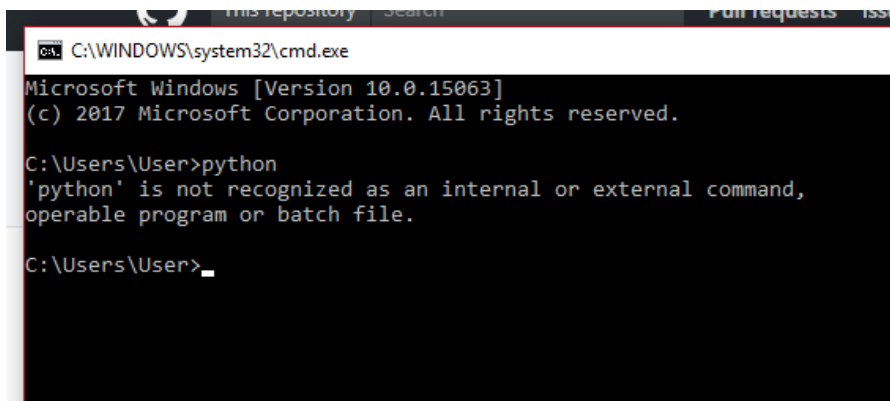
### 3.1.1 Situs Belajar Python

- (a) Learn Python Bagi pembaca yang tertarik untuk belajar bahasa pemrograman Python secara gratis, situs ini menjadi salah satu pilihan yang bijak. Dalam situs Learn Python ini pembaca akan dihadapkan pada halaman utama yang berisi penjelasan, tutorial, dan kolom pembelajaran interaktif. Disini, pembaca dapat belajar bahasa pemrograman Python mulai dari dasar hingga tingkat lanjut dengan berbagai penjelasan serta tutorial dasar untuk memahami bahasa pemrograman Python. Developer dapat langsung memasukkan kode-kode latihan pada kolom pembelajaran interaktif yang nantinya dapat dijalankan untuk melihat apakah kode tersebut bisa berjalan atau terjadi kesalahan. Selain itu, pembaca dapat juga mengetahui keinginan dari hasil-hasil kode latihan yang diberikan oleh Learn Python pada kolom pembelajaran interaktif.
- (b) Codecademy Bisa dibilang ini salah satu situs yang menawarkan pembelajaran dan latihan untuk beberapa bahasa situs yang populer di dunia seperti Python, JavaScript, jQuery, Ruby, PHP, Python, HTML, dan CSS dengan tingkatan level yang disesuaikan. Developer dapat mempelajari bahasa pemrograman Python di situs ini dengan interaktif dan baik. Nantinya developer akan diberikan halaman latihan dua kolom yang terdiri dari pengenalan pada kolom kiri dan latihan pada kolom kanan. Pada kolom kanan ini developer dapat langsung mengetikkan baris kode untuk pemrograman dan dapat langsung dijalankan secara langsung. Nantinya developer dapat melihat persentase mengenai tingkatan bahasa pemrograman yang telah dipelajari.
- (c) Treehouse Treehouse merupakan salah satu situs yang mengajarkan beragam bahasa pemrograman mulai dari web hingga aplikasi mobile. Beberapa materi kursus yang ditawarkan oleh situs ini di antaranya Learn Python, Android Development, Web Designer, dan masih banyak lagi. Dalam situs ini, pembaca maupun developer akan disuguhkan jalur latihan bahasa pemrograman Python mulai dari dasar hingga tingkat lanjut. Dengan menggunakan situs ini, developer dapat melakukan latihan pemrograman secara interaktif. Nantinya developer akan meraih skor dari kemampuannya dalam menghadapi latihan yang ditawarkan Treehouse.
- (d) Trinket Situs ini menghadirkan kursus bahasa pemrograman Python yang bisa dibilang lengkap. Pembaca maupun developer akan diberikan berbagai materi dan tutorial yang interaktif. Dalam halaman materi yang disampaikan akan terdapat kolom interaktif yang akan membuat developer dapat mempelajari Python dengan lebih mudah. Untuk materi bahasa pemrograman Python yang disampaikan akan bertahap mulai dari dasar hingga tingkat lanjut. Developer juga dapat belajar bersama visual kura-kura yang

akan disajikan pada kolom interaktif setiap developer menjalankan kode-kode yang telah diselesaikan. Kura-kura tersebut akan bergerak yang disertai dengan hasil akhir dari kode-kode yang dijalankan. Bisa dibilang metode pembelajaran yang ditawarkan oleh situs Trinket ini menyenangkan.

### 3.2 Environment Setup For Python

Bahasa pemrograman python tersedia diberbagai platform, termasuk juga terdapat pada linux dan max os x. Untuk mengecek apakah python sudah terinstall atau belum, pertama tama bukalah command prompt atau terminal komputer dan ketikan python. Akan muncul info atau status mengenai apakah python sudah terinstall dan berapa versinya atau belum terinstall sama sekali. Pada



**Figure 3.1** Contoh skript pada command prompt.

gambar 3.1 memberitahukan bahwa pada komputer yang saya pakai sekarang ini belum ada atau belum terinstall python. Ketika python belum terinstall maka sistem akan mendeteksi bahwa comand yang diberikan tidak cocok dengan yang ada disistem. Akan tetapi kalau sudah terinstall maka akan memunculkan nama python begitu pula versi python yang di install. Python menyediakan dukungan yang kuat untuk integrasi dengan bahasa pemrograman lain dan alat-alat bantu lainnya. Python hadir dengan pustakapustaka standar yang dapat diperluas serta dapat dipelajari hanya dalam beberapa hari. Sudah banyak programmer Python yang menyatakan bahwa mereka mendapatkan produktivitas yang lebih tinggi. Mereka juga merasakan bahwa Python meningkatkan kualitas

```
#!/usr/bin/python
import sys
try:
```

```

    # open file stream
    file = open(file_name, "w")
except IOError:
    print "There was an error writing to", file_name
    sys.exit()
print "Enter '", file_finish,
print "' When finished"
while file_text != file_finish:
    file_text = raw_input("Enter text: ")
    if file_text == file_finish:
        # close the file
        file.close()
        break
    file.write(file_text)
    file.write("\n")
file.close()
file_name = raw_input("Enter filename: ")
if len(file_name) == 0:
    print "Next time please enter something"
    sys.exit()
try:
    file = open(file_name, "r")
except IOError:
    print "There was an error reading file"
    sys.exit()
file_text = file.read()
file.close()
print file_text

```

### 3.2.1 Memasang Phyton

Distribusi Python tersedia untuk berbagai macam platform. Anda hanya perlu mendownload kode biner yang berlaku untuk platform Anda dan menginstal Python. Jika kode biner untuk platform Anda tidak tersedia, Anda memerlukan kompiler C untuk mengkompilasi kode sumber secara manual. Kompilasi kode sumber menawarkan fleksibilitas lebih dalam hal pilihan fitur yang Anda butuhkan dalam instalasi Anda. Berikut adalah ikhtisar singkat tentang menginstal Python di berbagai platform :

### 3.2.2 Menyiapkan PATH

Program dan file eksekusi lainnya bisa berada di banyak direktori, jadi sistem operasi menyediakan jalur pencarian yang mencantumkan direktori yang dicari

OS untuk executable. Path disimpan dalam variabel lingkungan, yang merupakan string bernama yang dikelola oleh sistem operasi. Variabel ini berisi informasi yang tersedia untuk perintah shell dan program lainnya. The path variabel disebut sebagai PATH di Unix atau jalan pada Windows (Unix adalah kasus sensitif, Windows tidak). Di Mac OS, installer menangani detail jalur. Untuk meminta juru bahasa Python dari direktori tertentu, Anda harus menambahkan direktori Python ke path Anda.

### 3.2.3 instalasi Python pada windows

- (a) Unduh Python 2.7 di python.org.
- (b) Jalankan file instalasi Python yang telah diunduh. Secara default Python akan terinstall di folder C:\Python27.
- (c) Selanjutnya mengatur variable environment, buka Control Panel->System and Security->System.
- (d) Klik Advanced system settings, Environment Variables.
- (e) Pada System variables, pilih Path, klik Edit.
- (f) Pada Variable value, tambahkan ;C:\Python27.

#### 3.2.3.1 Pengujian Python pada windows

- (a) Cara pertama: ketik python untuk menjalankan Python Interpreter. Jika berhasil akan ditampilkan versi Python yang digunakan beserta interpreternya. Coba ketikkan print "hello world" untuk menampilkan teks hello world.
- (b) Cara kedua: buat file hello.py, yang isinya print "hello world", simpan di mana saja suka-suka kamu, misalnya di D: python-code. Buka command prompt, Run -> cmd. Kemudian masuk ke folder D: python-code, lalu ketik python hello.py untuk menjalankan file Python yang telah dibuat.

### 3.2.4 Instalasi Python pada Unix/Linux

Berikutini adalah simple steps untuk menginstall python pada Unix/Linux.

- (a) Buka web browser dan ketik alamat <https://www.python.org/downloads/>
- (b) Ikuti link untuk mendownload source code yang bisa digunakan oleh Unix/Linux
- (c) Download dan kemudian ekstrak file
- (d) Edit modules/setup file jika mau customize beberapa pengaturan
- (e) Jalankan perintah ./configure script
- (f) Instalasi

Lokasi standar penginstallan python ada pada /usr/local/bin dan libraries nya berada /usr/local/lib/pythonXX dimana XX adalah versi python yang di install.



### 3.2.5 Setting Path pada Unix/Linux

Untuk menambahkan direktori python di path pada Unix

- (a) In the csh shell – ketik `setenv PATH PATH:/usr/local/bin/python` lalu tekan enter
- (b) In the bash shell(Linux) – ketik `export PATH=PATH:/usr/local/bin/python` lalu tekan enter
- (c) In the sh or ksh shell – ketik `PATH=PATH:/usr/local/bin/python` lalu tekan enter
- (d) Note - `/usr/local/bin/python` adalah path untuk direktori Python

### 3.2.6 Fitur Python

Fitur-Fitur Python Python memiliki beberapa fitur yang menjadikan bahasa pemrograman ini berbeda dari bahasa lain antara lain :

- (a) Memiliki kepustakaan yang luas, dalam distribusi Python telah disediakan modul- modul siap pakai untuk berbagai keperluan.
- (b) Memiliki tata bahasa yang jernih dan mudah dipelajari.
- (c) Memiliki aturan layout kode sumber yang memudahkan
- (d) pengecekan, pembacaan kembali dan penulisan ulang kode sumber.
- (e) Berorientasi obyek.
- (f) 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++.
- (g) Memiliki fasilitas pengumpulan sampah otomatis, seperti halnya pada bahasa pemrograman Java, python memiliki fasilitas pengaturan penggunaan memory komputer sehingga para pemrogram tidak perlu melakukan pengaturan memory komputer secara langsung.
- (h) Memiliki banyak fasilitas pendukung sehingga mudah dalam pengoprasian-nya.

### 3.2.7 Pengujian

Cara pertama: ketik `python` untuk menjalankan Python Interpreter. Jika berhasil akan ditampilkan versi Python yang digunakan beserta interpreternya. Coba ketikkan `print "hello world"` untuk menampilkan teks hello world.

```
C:>python Python 2.7.11 (v2.7.11:6d1b6a68f775, Dec 5 2015, 20:40:30) [MSC
v.1500 64 bit (AMD64)] on win32 Type "help", "copyright", "credits" or "li-
cense" for more information.
```

```
>>> print "hello world"
>>> hello world
```

### 3.2.8 Alasan Menggunakan Python

Python memiliki konsep desain yang bagus dan sederhana, yang berfokus pada kemudahan dalam penggunaan. Kode Python dirancang untuk mudah dibaca, dipelajari, digunakan ulang, dan dirawat. Selain itu, Python juga mendukung pemrograman berorientasi objek dan pemrograman fungsional.

### 3.2.9 Tipe Data Python

Menggunakan tipe Number

Tipe data Number digunakan untuk menyimpan nilai-nilai numerik. Tipe ini merupakan tipe data immutable, yang artinya jika kita mengubah nilai dari sebuah data, maka kita akan mengalokasikan obyek baru. Sama seperti tipe data lainnya, obyek Number dibuat ketika kita memberikan sebuah nilai padanya. Contoh:

```
>>> data = 1
```

Kita juga dapat mengubah nilai yang ada dalam variable data tersebut.

```
>>> data = data + 1
>>> data=3.50
>>> floatdat = 7.5
>>> data = floatdat
```

Kita dapat menghapus sebuah obyek ataupun banyak obyek dengan menggunakan pernyataan del. Misalnya:

```
>>> del data
>>> del data, floatdat
```

Python mengelompokkan tipe Number dalam 4 macam, yaitu:

**Plain Integer** Plain integer atau bilangan bulat merupakan tipe data yang sering kita temui pada semua bahasa pemrograman. Integer ini mempunyai range nilai antara  $-2^{32}$  sampai  $2^{31}-1$ . Tipe ini juga dapat ditulis dalam bentuk octal (di tanda awalan "0") maupun hexadesimal (ditandai awalan "0x" atau "0X"). Contoh: 10 100 6542 -784 083 -042 -0x43 0X61

**Long Integer** Long integer sangat membantu kita untuk perhitungan di luar range nilai integer. Secara virtual, tidak ada batasan nilai tergantung besar virtual memory yang kita gunakan. Akhiran 'l' atau 'L' disetiap nilai bilangan bulat menandakan bahwa data tersebut bertipe long integer. 562718819L -0x526718L 012L -567299101L

Floating Point Real Number Tipe ini sering disebut sebagai tipe real (atau float). Tipe ini sama dengan tipe double di C. Nilai float mempunyai dua bagian, bagian titik desimal dan bagian eksponensial. Tanda positif atau negatif diantara “e” merupakan tanda eksponen. Contoh nilai float: 0.0 14.5 -15.4 32.3+e18 -90.76712 -90. -32.54e100 70.2-E12

Complex Number Sebuah bilangan kompleks biasanya ditunjukkan oleh bentuk  $a + bj$ , dimana  $a$  adalah bagian real dan  $b$  adalah bagian imajiner. Bagian imajiner merupakan bilangan di awal tanda “j” atau “J”. Berikut ini contoh bilangan kompleks: 3.14j 45j 54.56+12.1J 3e+36J

Bagian real dan imajiner dari bilangan kompleks dapat kita pisahkan menggunakan data atribut, yaitu menggunakan real dan imag. Sedangkan untuk mendapatkan konjugasi dari bilangan kompleks tersebut, kita dapat menggunakan metode `conjugate()`.

```
>>> kompleks = 23.45-1.23J
>>> kompleks.real
23.45
>>> kompleks.imag
-1.23
>>> kompleks.conjugate()
(23.45+1.23j)
```

String merupakan salah satu tipe data yang sering digunakan dalam pemrograman Python. Sebuah string dapat dinyatakan sebagai kumpulan karakter yang dibatasi oleh satu atau dua tanda petik. Inilah contohnya,

```
>>> nama = "Rolly Maulana Awangga"
>>> nama
'Rolly Maulana Awangga'
>>> slm = 'Salam Python Dahsyat!'
>>> slm
'Salam Python Dahsyat!'
>>> print slm
Salam Python Dahsyat!
```

Dari contoh di atas, ketika kita memanggil variabel secara langsung maka akan ditampilkan isi dari variabel tersebut dengan sebuah tanda petik. Namun jika kita menggunakan pernyataan `print`, maka tanda petik tersebut akan dihilangkan.

#### Menampilkan Tanda Petik Sebagai String

Di dalam sebuah string tidak dapat berisi tanda petik yang sama dengan tanda petik yang digunakan oleh string tersebut. Misalkan, ketika kita menuliskan 'Py'thon' maka akan muncul pesan kesalahan (syntax error). Agar tidak muncul

pesan kesalahan, kita bisa mengganti tanda petik luarnya dengan tanda petik ganda, misalnya "Py'thon". Tanda petik juga dapat ditulis setelah tanda backslash agar dapat ditampilkan sebagai string.

```
>>> str = "Py'thon"
>>> str
"Py'thon"
>>> str2 = 'Py"thon'
>>> str2
'Py"thon'
>>> "ÖK, sampai ketemu lagi."
""ÖK, " sampai ketemu lagi."
```

Kode n atau Triple Quotes Juga Boleh!

Jika kita ingin menuliskan string yang panjang dalam beberapa baris, maka kita perlu menambahkan tanda backslash diikuti huruf n (n) sebagai tanda baris baru.

```
>>> teks = "Python adalah bahasa pemrograman yang powerfull.nTerbukti
Python bisa dijalankan di segala platform OS.nJadi, saatnya kita menggunakan
Python sebagai bahasa permrograman nsehari-hari. Salam Python Dahsyat!"
```

Tanda n akan memberikan perintah membuat baris baru jika kita memanggil teks dengan pernyataan print.

```
>>> print teks
```

Python adalah bahasa pemrograman yang powerfull. Terbukti Python bisa dijalankan di segala platform OS. Jadi, saatnya kita menggunakan Python sebagai bahasa permrograman sehari-hari. Salam Python Dahsyat!

Kita juga dapat menggunakan tanda petik tiga (triple quotes), """ atau ''', untuk menuliskan sebuah string yang panjang dalam beberapa baris.

```
>>> teks = """Python adalah bahasa pemrograman yang powerfull. ... Ter-
bukti Python bisa dijalankan di segala platform OS. ... Jadi, saatnya kita meng-
gunakan Python sebagai bahasa pemrograman ... sehari-hari. Salam Python
Dahsyat!"""
```

Tampilan teks dengan pernyataan print seperti di bawah ini,

```
>>> print teks
Python adalah bahasa pemrograman yang powerfull. Terbukti
Python bisa dijalankan di segala platform OS. Jadi, saatnya kita menggunakan
Python sebagai bahasa pemrograman sehari-hari. Salam Python Dahsyat!
```

Menggabungkan String

Untuk menggabungkan dua buah string atau lebih, kita dapat menggunakan operator +. Sedangkan untuk menggandakan string, kita gunakan operator \*.

```
>>> Tampan = 'Rolly' + 'Ganteng'
```

```
>>> Tampan 'RollyGanteng'
>>> newTampan = Tampan*5
>>> newTampan
'RollyGantengRollyGantengRollyGantengRollyGantengRollyGanteng'
>>> Tampan *= 4
>>> print Tampan
RollyGantengRollyGantengRollyGantengRollyGanteng
```

Jika dua string ditulis secara berurutan, maka secara langsung kedua string tersebut akan digabungkan.

```
>>> Tampan = 'Rolly"Ganteng'
>>> Tampan
'RollyGanteng'
```

#### Menentukan Panjang String

Panjang dari sebuah string dapat kita temukan dengan menggunakan fungsi `len()`.

```
>>> len(blog) 12
```

#### Memecah String

Tidak seperti bahasa lainnya, Python tidak mendukung tipe Karakter. Untuk mengambil satu karakter atau lebih dari sebuah string, kita dapat memecah string tersebut menggunakan indeks (disebut Metode Irisan). Irisan terdiri dari dua indeks yang dipisahkan tanda koma.

```
>>> buah = 'Nanas'
>>> buah[0]
'N'
>>> buah[0:2]
'Na'
>>> buah[0:4]
'Nana'
>>> buah[0:5]
'Nanas'
```

Dari contoh di atas, panjang string buah adalah 5. Ketika kita menghitung maju, indeks bernilai 0 sampai (panjang-string - 1) dimulai dari kiri ke kanan. Maka dari itu, kita dapat mengakses setiap karakter dalam range 0 sampai 4.

Sebuah string juga dapat dihitung mundur, dengan indeks -1 sampai (negatif panjang-string) dimulai dari kanan ke kiri. Berikut gambaran lengkapnya, baik itu penghitungan maju atau mundur.

Contoh penggunaan penghitungan mundur,

```
>>> buah[-1]
's'
>>> buah[-5]
'N'
>>> buah[-5:-1]
'Nana'
>>> buah[1:-1]
'ana'
```

Jika kita lupa berapa nilai indeks awala atau indeks akhir, kita dapat kosongkan indeks tersebut.

```
>>> buah[:3]
'Nan'
>>> buah[2:]
'nas'
```

Pengosongan indeks akan menyebabkan semua string ditampilkan.

```
>>> buah[:]
'Nanas'
```

#### String Bersifat Immutable

Tipe data String pada Python bersifat immutable, yang artinya sekali dibuat maka tidak dapat diubah kembali. Ada pertanyaan bagus, jika string bersifat immutable, mengapa kita bisa mengubah nilai dari variabel string tersebut? Jawabannya sangat sederhana. Ketika kita memberikan nilai yang berbeda pada variabel string, sebuah obyek baru berhasil kita buat. Lihat contoh berikut,

```
>>> nama = "Klinik"
>>> nama
'Klinik'
>>> id(nama)
3076962016L
>>> nama = "Python"
>>> id(nama)
3077289888L
```

Catat bahwa ketika string nama telah kita buat, maka identitas dari variabel ini dapat kita ketahui dengan menggunakan fungsi `id()`. Jika kita ubah nilai dari variabel nama tersebut, maka identitasnya juga berubah. Hal ini menandakan

bahwa obyek baru telah dibuat. Penggantian nilai pada string pada posisi indeks tertentu akan menghasilkan pesan kesalahan.

```
>>> nama[0] = 'K'
```

Traceback (most recent call last):

File "", line 1, in

TypeError: 'str' object does not support item assignment

Kita juga dapat menambahkan sebuah string baru pada string lama.

```
>>> 'Si' + nama[0]
```

```
'SiP'
```

```
>>> 'J' + nama[1:]
```

```
'Jython'
```

## CHAPTER 4

---

# BASIC SYNTAX

---

### 4.1 Basic Syntax

#### 4.1.1 Pengenal Python

Pengenal Python Pengenal Python adalah nama yang digunakan untuk mengidentifikasi variabel, fungsi, kelas, modul atau objek lainnya. Pengenal dimulai dengan huruf A sampai Z atau huruf a sampai z atau garis bawah diikuti oleh nol atau lebih huruf, garis bawah dan angka (0 sampai 9). Python tidak mengizinkan karakter tanda baca seperti , dan % dalam pengenal. Python adalah bahasa pemrograman yang sensitif. Dengan demikian, `Tenaga Kerja` dan `tenaga kerja` adalah dua pengidentifikasi yang berbeda dengan Python. Berikut adalah konvensi penamaan untuk pengenal Python - Nama kelas dimulai dengan huruf besar. Semua pengenal lainnya mulai dengan huruf kecil. Memulai pengenal dengan satu garis bawah terkemuka menunjukkan bahwa pengenal bersifat pribadi. Memulai pengenal dengan dua garis bawah terkemuka menunjukkan pengenal yang sangat pribadi. Jika pengenal juga diakhiri dengan dua tanda garis bawah, `__idntier` adalah nama khusus yang ditentukan bahasa. Bahasa Python memiliki



banyak kesamaan dengan Perl, C, dan Java. Namun, ada beberapa perbedaan yang pasti antara bahasa. Program Python Pertama Mari kita jalankan program dalam mode pemrograman yang berbeda. Pemrograman Mode Interaktif Memohon interpreter tanpa melewati le script sebagai parameter menampilkan prompt berikut :

Python

```
Python 2.4.3 (#1, Nov 11 2010, 13:34:43)
[GCC 4.1.2 20080704 (Red Hat 4.1.2-48)] on linux2 Type "help", "copyright", "credits"
```

Ketik teks berikut pada prompt Python dan tekan Enter:

```
Python 2.4.3 ( #1, Nov 11 2010, 13:34:43)
[GCC 4.1.2 20080704 (Red Hat 4.1.2-48)]
on linux2 Type "help", "copyright",
"credits" or "license" for more information.
```

Ketik teks berikut pada prompt Python dan tekan Enter:

```
print "Hello, Python!"
```

Jika Anda menjalankan versi baru Python, Anda perlu menggunakan pernyataan cetak dengan tanda kurung seperti pada cetak ("Halo,Python!");. Namun dengan versi Python 2.4.3, ini menghasilkan hasil sebagai berikut:

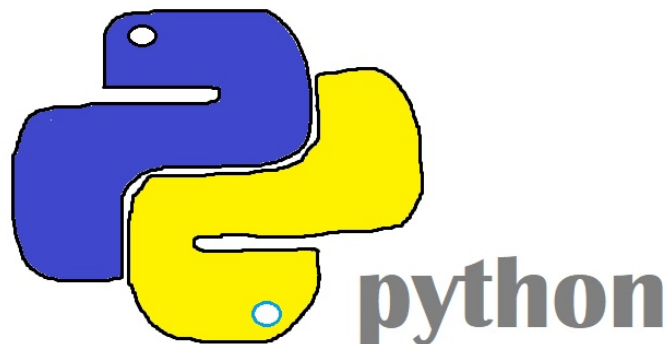
```
Hello, Python!
```

#### 4.1.2 Pemrograman Mode Script

Memohon interpreter dengan parameter script memulai eksekusi script dan berlanjut sampai script selesai. Saat skrip selesai, juru bahasa tidak lagi aktif. Mari kita tuliskan program Python sederhana dalam sebuah naskah. File Python memiliki ekstensi .py. Ketik kode sumber berikut di le. Objek Dengan Python, seperti semua bahasa berorientasi objek, ada kumpulan kode dan data yang disebut objek, yang biasanya mewakili potongan dalam model konseptual suatu sistem. Objek dengan Python dibuat (yaitu, instantiated) dari template yang disebut kelas (yang akan dibahas kemudian, sebanyak bahasa dapat digunakan tanpa memahami kelas). Mereka memiliki atribut, yang mewakili berbagai potongan kode dan data yang membentuk objek. Untuk mengakses atribut, seseorang menuliskan nama objek yang diikuti oleh suatu periode (selanjutnya disebut titik), diikuti dengan nama atribut. Contohnya adalah atribut 'atas' dari string, yang mengacu pada kode yang mengembalikan salinan string di mana semua huruf adalah huruf besar. Untuk mendapatkan ini, perlu untuk memiliki cara untuk merujuk ke objek (dalam contoh berikut, jalan adalah string literal yang membangun objek).

### 4.1.3 Pengertian Python

Python adalah salah satu pemrograman yang terus berkembang dan bertahan dikarenakan dukungan komunitas yang aktif diseluruh dunia. Banyak forum-forum ataupun blogger-blogger yang sering membagi pengalaman dalam menggunakan python. Hal ini memudahkan bagi pengguna pemula maupun pengembang untuk bertanya dan sharing tentang ilmu pemrograman ini. Gambar berikut



**Figure 4.1** Gambar Logo Python

4.1 adalah gambar logo python dengan lambang 2 ular python yang saling membelakangi. ular diatas berwarna biru dan ular dibawah berwarna kuning.

### 4.1.4 Kelebihan dan Kekurangan Python

Kelebihan yang dimiliki oleh Python :

- Tidak ada tahapan kompilasi dan penyambungan (link) sehingga kecepatan perubahan pada masa pembuatan sistem aplikasi meningkat.
- Tidak ada deklarasi tipe data yang merumitkan sehingga program menjadi lebih sederhana, singkat, dan eksible.
- Manajemen memori otomatis yaitu kumpulan sampah memori sehingga dapat menghindari pencacatan kode.
- Tipe data dan operasi tingkat tinggi yaitu kecepatan pembuatan sistem 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.

Kekurangan yang dimiliki Python :

- Beberapa penugasan terdapat diluar dari jangkauan python, seperti bahasa pemrograman dinamis lainnya, python tidak secepat atau esien 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 esisiensi dan eksibilitas 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.
- Banyak terdapat referensi lama terutama dari pencarian google, python adalah pemrograman yang sangat lambat. Namun belum lama ini ditemukan bahwa Google, Youtube, DropBox dan beberapa software sistem banyak menggunakan Python.

Dibalik kelebihan dan kekurangan yang dimiliki, Kini Python menjadi salah satu bahasa pemrograman yang populer digunakan oleh pengembangan web, aplikasi web, aplikasi perkantoran, simulasi, dan masih banyak lagi. Hal ini disebabkan karena Python bahasa pemrograman yang dinamis dan mudah dipahami. Python memiliki hak cipta. Seperti Perl, kode sumber Python sekarang tersedia di bawah GNU General Public License (GPL). Python sekarang dikelola oleh tim pengembangan inti di institut tersebut, walaupun Guido van Rossum masih memegang peran penting dalam mengarahkan kemajuannya.

#### 4.1.5 Fitur Python

Python memiliki fitur yang meliputi:

- (a) Mudah dipelajari: Python memiliki beberapa kata kunci, struktur sederhana, dan sintaks yang jelas. Hal ini memungkinkan siswa untuk mengambil bahasa dengan cepat.
- (b) Mudah dibaca: kode Python lebih jelas dan terlihat oleh mata.
- (c) Mudah dipelihara: kode sumber Python cukup mudah untuk dipelihara.
- (d) Perpustakaan standar yang luas: sebagian besar perpustakaan Python sangat portabel dan kompatibel dengan platform cross-platform di UNIX, Windows, dan Macintosh.
- (e) Mode Interaktif: Python memiliki dukungan untuk mode interaktif yang memungkinkan pengujian interaktif dan debugging dari cuplikan kode.
- (f) Portable: Python dapat berjalan di berbagai platform perangkat keras dan memiliki antarmuka yang sama pada semua platform.
- (g) Dapat diperpanjang: Anda dapat menambahkan modul tingkat rendah ke penerjemah Python. Modul ini memungkinkan programmer untuk menambahkan atau menyesuaikan alat mereka agar lebih efisien.
- (h) Database: Python menyediakan antarmuka untuk semua database komersial utama.

#### 4.1.6 Fitur Python

Fitur Python meliputi:

- (a) Mudah dipelajari: Python memiliki beberapa kata kunci, struktur sederhana, dan sintaks yang jelas. Hal ini memungkinkan siswa untuk mengambil bahasa dengan cepat.
- (b) Mudah dibaca: kode Python lebih jelas dan terlihat oleh mata.
- (c) Mudah dipelihara: kode sumber Python cukup mudah untuk dipelihara. Perpustakaan standar yang luas: sebagian besar perpustakaan Python sangat portabel dan kompatibel dengan platform cross-platform di UNIX, Windows, dan Macintosh.
- (d) Mode Interaktif: Python memiliki dukungan untuk mode interaktif yang memungkinkan pengujian interaktif dan debugging dari cuplikan kode.
- (e) Portable: Python dapat berjalan di berbagai platform perangkat keras dan memiliki antarmuka yang sama pada semua platform.
- (f) Dapat diperpanjang: Anda dapat menambahkan modul tingkat rendah ke penerjemah Python. Modul ini memungkinkan programmer untuk menambahkan atau menyesuaikan alat mereka agar lebih efisien.
- (g) Database: Python menyediakan antarmuka untuk semua database komersial utama.

#### 4.1.7 Pernyataan Multi-Line

Pernyataan di Python biasanya diakhiri dengan baris baru. Python, bagaimanapun, memungkinkan penggunaan karakter kelanjutan baris ( `\` ) untuk menunjukkan bahwa garis tersebut harus dilanjutkan. Misalnya –

```
\subsection {Pernyataan Multi-Line}
Pernyataan di Python biasanya diakhiri dengan baris baru. Python, bagaimanapun, memu
\begin {equation}

total = item_one + \
item_two + \
item_three
\end {verbatim}
```

Pernyataan yang ada di dalam kurung [], {}, atau () tidak perlu menggunakan karakter

```
\begin{verbatim}
days = ['Monday', 'Tuesday', 'Wednesday',
'Thursday', 'Friday']
```

#### 4.1.8 Kutipan pada Python

Python menerima kutipan tunggal ( `'` ), ganda ( `"` ) dan triple ( `' '` atau `' " "` ) untuk menunjukkan literal string, selama jenis kutipan yang sama dimulai dan mengakhiri string. Tanda kutip triple digunakan untuk membentang string di beberapa baris. Misalnya:

```
word = 'word'
sentence = "This is a sentence."
paragraph = """This is a paragraph. It is
made up of multiple lines and sentences."""
\end {verbatim}
```

```
\subsection{Komentar pada Python}
Tanda hash (\#)\ yang tidak berada di dalam string literal memulai sebuah komentar.
\begin{verbatim}
#!/usr/bin/python
```

```
# First comment
print "Hello, Python!" # second comment
```

Dan hasilnya sebagai berikut :

Hello, Python!

Anda bisa mengetikkan komentar di baris yang sama setelah sebuah pernyataan atau ungkapan –

```
name = \"Madisetti\" # This is again comment
```

Anda dapat mengomentari beberapa baris sebagai berikut –

```
# This is a comment.
# This is a comment, too.
# This is a comment, too.
# I said that already.
```

#### 4.1.8.1 Variabel

```
days = ['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday']
```

(4.1)

#### 4.1.9 Kutipan pada Python

Python menerima kutipan tunggal (`'`), ganda (`"`) dan triple (`' '` atau `''' '`) untuk menunjukkan literal string, selama jenis kutipan yang sama dimulai dan mengakhiri string. Tanda kutip triple digunakan untuk membentang string di beberapa baris. Misalnya:

```
word = ' word' sentence = "This is a sentence."
```

(4.2)

#### 4.1.10 Komentar pada Python

Tanda hash yang tidak berada di dalam string literal memulai sebuah komentar. Semua karakter setelah dan sampai akhir garis fisik adalah bagian dari komentar dan penafsir Python mengabaikannya.

```
#!/usr/bin/pythonprint "Hello, Python!"
```

(4.3)

Dan hasilnya sebagai berikut –

```
Hello, Python!
```

(4.4)

Anda bisa mengetikkan komentar di baris yang sama setelah sebuah pernyataan atau ungkapan –

```
name = "Madisetti" This is again comment
```

(4.5)

Anda dapat mengomentari beberapa baris sebagai berikut –

```
This is a comment. I said that already.
```

(4.6)

#### 4.1.11 Variabel

Jadi variable itu untuk menyimpan data sebagai penampung nya dan nilai adalah isi dari variable itu sendiri. Contoh variable katakana saja my int = 7 itu juga bisa di ubah menjadi my int =3 Code merubah nilai variabelnya My int = 7 My int = Print my int

#### 4.1.12 Whitespace

Whitespace adalah penyusun kode untuk penulisan oleh karena itu harus berhati hati karena sangat sensitive Contoh code nya:

```
Def spam()
Eggs + 12
Return eggs
Print spam()
```

Itu merupakan kode penulisan yang salah maka akan muncul pesan error. Pesan error yang muncul tentang penulisan dan bisa di edit agar tidak terjadi kesalahan penulisan lagi. Contoh yang benar :

```
Def spam():
Eggs = 12
Return eggs
Print spam()
```

Kita coba contohkan penulisan oprasi matematika nya. Code nya adalah:

```
Jumlah = 10+10
Print jumlah
Pejumlahan = 72 + 23
Pengurangan 108 { 204
Perkalian = 108 * 0,5
Pembagian = 108 / 9
```

Maka akan muncul print yang sesuai inputan yang kalian buat. Selain itu bisa juga perpangkatan

Kalkulator pangkatan meggunakan eight

Contohnya eight = \2\*\* 3\

Eight itu adalah variable baru yang kita buat untuk mengeset nilai menjadi 8.

Contoh: stopwatch sederhana yang biasa nya di gunakan untuk ujian perak-tek. Penjelasan tentang stopwatch yaitu alat untuk mengukur kecepatan suatu benda secara akurat. Sebelum membuat nya kita harus mengetahui cara kerja stopwatch itu sendiri. Memulai angka nya dari 0 karena angka 0 adalah angka pertama dalam waktu. Code nya antara lain:

```
Depanjam = 0
Jam = 0
Depanmenit = 0
Menit = 0
Depandetik = 0
Detik = 0
```

Lalu gunakan import time Untuk system loop jadi setelah detik mencapai 9 maka menit akan bertambah 1 seperti itu.

```
While true :
    Time.sleep(1)
    Detik += 1
    If detik == 9:
        Detik = 0
        Depandetik += 1
        If depandetik == 6:
            Menit +=1
            Depandetik = 0
            Detik = 0
            If menit == 9:
                Menit = 0
                Depanmenit += 1
                If depanmenit == 6:
                    Jam +=1
                    Depanmenit = 0
                    Menit = 0
                    If ja, == 9 :
                        Depanjam += 1
                        Jam = 0
            Print("{0}{1}{2}{3}{4}{5vert} ".Format(Depanjam, Jam, Depanmenit,
            Menit, Depandetik, Detik), end="setminus r" )
```

Run maka akan jadi program simple tentang stopwatch sederhana.





## CHAPTER 5

---

# BASIC OPERATOR

---

### 5.1 Python Basic Operator

### 5.2 Pengenalan Python Programming

Menurut Van Python adalah bahasa pemrograman yang menggabungkan kapabilitas, kemampuan, dengan sintaksis kode yang sangat jelas, dan dilengkapi dengan fungsionalitas pustaka standar yang besar serta komprehensif. Python mendukung multi paradigma pemrograman, utamanya namun tidak dibatasi pada pemrograman berorientasi objek, pemrograman imperatif, dan pemrograman fungsional. Salah satu fitur yang tersedia pada python adalah sebagai bahasa pemrograman dinamis yang dilengkapi dengan manajemen memori otomatis. Seperti halnya pada bahasa pemrograman dinamis lainnya, python umumnya digunakan sebagai bahasa skrip meski pada praktiknya penggunaan bahasa ini lebih luas mencakup konteks pemanfaatan yang umumnya tidak dilakukan dengan menggunakan bahasa skrip. Python dapat digunakan untuk berbagai keperluan pengembangan perangkat lunak dan dapat berjalan di berbagai platform sistem operasi. Manfaat besar dari sistem operasi seperti py-

ton adalah portabilitas aplikasi, setidaknya aplikasi yang dibuat dengan Python murni. Seluruh Sistem Operasi akan berjalan di mana pun dapat mengkompilasi Kernel Linux dan Python, dengan sedikit usaha daripada mengatakan bahwa porting seluruh Linux ke chipset yang berbeda. Aplikasi Python tidak perlu dikompilasi ulang untuk CPU target, dan jika sebagian besar Sistem Operasi dibuat dengan Python, port dibuat dengan mudah. Proses mengubah dari bentuk bahasa tingkat tinggi dalam bahasa pemrograman ada dua tipe, yaitu interpreter dan compiler. Fungsi Interpreter yaitu membaca program berbahasa tingkat tinggi dan kemudian mengeksekusi program tersebut. Sedangkan fungsi compiler membaca program dan menerjemahkannya secara keseluruhan kemudian baru dieksekusi. Bisa juga mengompilasi suatu program yang merupakan bagian program lain. Dalam kasus ini, program tingkat tinggi ini dinamakan kode sumber (source code) dan hasil terjemahannya sering dinamakan object code atau executable. [?]

### 5.2.1 Python Basic Operator

Peran operator dalam proses perhitungan matematika sangatlah penting. Setiap bahasa pemrograman selalu memiliki sebuah operator. Walaupun kadang kala ada beberapa contoh operator yang memiliki perbedaan fungsi atau simbol. Namun hanya sebagian saja, bukan seluruhnya. Operator adalah konstruksi yang dapat memanipulasi nilai operan. Operator merupakan simbol khusus yang mempresentasikan perhitungan seperti penambahan dan perkalian. Nilai yang digunakan oleh operator sering disebut operand. Ekspresi merupakan kombinasi dari operator dan operandnya. Dalam sebuah eksekusi program, suatu ekspresi akan di evaluasi sehingga menghasilkan suatu nilai tunggal. Di dalam bahasa pemrograman Python ada beberapa tipe operator seperti operator Aritmatika, Perbandingan, Penugasan, Logika, Bitwise, Keanggotaan dan Identitas. Memasukkan data dan menampilkannya merupakan hal yang sering dilakukan dalam pemrograman. Ada beberapa fungsi dasar dalam fasilitas I/O, yaitu print dan input. Intruksi print digunakan khusus untuk menampilkan sebuah karakter ke layar. Sedangkan input digunakan untuk membaca sebuah karakter. Bentuk Umum intruksi print adalah: Print ("string kontrol"), argumen.

Memberi nilai dari bagian kanan operator ke bagian kiri operator. Operator penunjukan dalam bahasa python menggunakan tanda sama dengan, termasuk dalam:

$$+ =, - =, * =, / = . \quad (5.1)$$

Instruksi

$$x = x + 1 \quad (5.2)$$

sama artinya dengan

$$x+ = 1 \quad (5.3)$$

Operator membandingkan kesamaan dua nilai digunakan tanda == dan menghasilkan sebuah ekspresi boolean Selain operator Aritmatika, Python juga men-

dukung operator berkondisi yang berfungsi untuk membandingkan suatu nilai dengan nilai yang lain. Operator-operator yang didukung oleh Python yaitu operator Unari ( + dan - ) dan operator Binari ( +, -, \*, /). Pada ekspresi Aritmatika berikut:

$$x = y + z \quad (5.4)$$

y dan z disebut sebagai operan dari operator +. Sebagai contoh operasi

$$3 + 2 = 5 \quad (5.5)$$

Disini 3 dan 2 adalah operan dan + adalah operator.

Bahasa pemrograman Python mendukung berbagai macam operator, diantaranya :

- (a) Operator Aritmatika(Arithmetic Operators)
- (b) Operator Perbandingan(Comparison Relational Operators)
- (c) Operator Logika (Logical Operators)
- (d) Operator Bitwise (Bitwise Operators)
- (e) Operator Identitas (Identity Operators)

**5.2.1.1 Operator Aritmatika** Operator Aritmatika sendiri merupakan operator yang dipergunakan dalam melakukan operasi matematika, seperti pengurangan, pembagian, penjumlahan, perkalian, pangkat, modulus dll.

Di bawah ini adalah tabel 5.1 yang terdapat pada bahasa pemrograman Python.

Berikut contoh penggunaan operator aritmatika pada bahasa pemrograman python:

(a) Penjumlahan

```
print(13 + 2)
apel = 7
jeruk = 9
buah = apel + jeruk
print(buah)
```

(b) Pengurangan

```
hutang = 10000
bayar = 5000
sisahutang = hutang - bayar
print("Sisa hutang Anda adalah ", sisahutang)
```

(c) Perkalian

```
panjang = 15
```

Operator	Contoh	Penjelasan
Penjumlahan +	$1 + 3 = 4$	Menjumlahkan nilai dari masing-masing operan atau bilangan
Pengurangan -	$4 - 1 = 3$	Mengurangi nilai operan di sebelah kiri menggunakan operan di sebelah kanan
Perkalian *	$2 * 4 = 8$	Mengalikan operan/bilangan
Pembagian /	$10 / 5 = 2$	Untuk membagi operan di sebelah kiri menggunakan operan di sebelah kanan
Sisa Bagi %	$11 \% 2 = 1$	Mendapatkan sisa pembagian dari operan di sebelah kiri operator ketika dibagi oleh operan di sebelah kanan
Pangkat **	$8 ** 2 = 64$	Memangkatkan operan disebelah kiri operator dengan operan di sebelah kanan operator
Pembagian Bulat //	$10 // 3 = 3$	Sama seperti pembagian. Hanya saja angka dibelakang koma dihilangkan

Figure 5.1 Table Operator Aritmatika.

```

lebar = 8
luas = panjang * lebar
print(luas)

```

## (d) Pembagian

```

kue = 16
anak = 4
kuePerAnak = kue / anak
print("Setiap anak akan mendapatkan bagian kue sebanyak ", kuePerAnak)

```

## (e) Sisa Bagi / Modulus

```

bilangan1 = 14
bilangan2 = 5
hasil = bilangan1 / bilangan2
print("Sisa bagi dari bilangan ", bilangan1, " dan ", bilangan2, " adalah ", hasil)

```

## (f) Pangkat

```

bilangan3 = 8
bilangan4 = 2
hasilPangkat = bilangan3 ** bilangan4
print(hasilPangkat)

```

## (g) Pembagian Bulat

```

print(10//3)

```

10 dibagi 3 adalah 3.3333.

#### (h) Perhitungan Maksimum dan Minimum Pada Operator Aritmatika

```
def __add__(x,y):
    return n: (x(n+2)+y(n+2)+2)/4
def __sqrt__(x):
    return n: sqrt(x(2 n))
def cmp(x,y):
    n=0
    while 1:
        xn,yn=x(n),y(n)
        if xn<yn+1: return 1 ==> x <y
        if xn>yn+1: return 1 ==> x >y
        n+=1
    def minmax(x,y):
        s,d=x+y,abs(xy)
        return (sd)/2,(s+d)/2
```

Contoh penggunaan operator aritmatika pada bahasa pemrograman python

Perhatikan perintah pada baris pertama tanpa harus menuliskan tanda kurung pun python sudah melakukan operasi matematika sesuai dengan aturan prioritas matematika. Operator yang biasa dipergunakan pada aritmatika ini pun beberapa dapat dipergunakan dalam melakukan manipulasi string contohnya :

```
print 'Pintarcoding.com!' * 3
print 'hallo' + ' ' + 'Pintarcoding.com!'
```

**5.2.1.2 Operator Perbandingan** Operator perbandingan adalah operator yang digunakan untuk membandingkan nilai dari masing-masing operand. Operator ini juga dikenal dengan operator relasi dan sering digunakan untuk membuat sebuah logika atau kondisi. Operator perbandingan akan mengembalikan sebuah nilai boolean yaitu true atau false. Operator ini juga dikenal dengan operator relasi dan sering digunakan untuk membuat sebuah logika atau kondisi.

Di bawah ini adalah tabel 5.2 macam-macam operator perbandingan yang terdapat pada bahasa pemrograman Python.

Di bawah ini adalah contoh implementasi operator perbandingan pada program di Python:

```
a = 20 b = 10 c = 0
```

```
print ("a = ",a)
print ("b = ",b)
```

```
if ( a == b ):
```

Operator	Contoh	Penjelasan
Sama dengan ==	1 == 1 bernilai True	Jika masing-masing operan memiliki nilai yang sama, maka kondisi bernilai benar atau True.
Tidak sama dengan !=	2 != 2 bernilai False	Akan menghasilkan nilai kebalikan dari kondisi sebenarnya.
Tidak sama dengan <>	2 <> 2 bernilai False	Akan menghasilkan nilai kebalikan dari kondisi sebenarnya.
Lebih besar dari >	5 > 3 bernilai True	Jika nilai operan kiri lebih besar dari nilai operan kanan, maka kondisi menjadi benar.
Lebih kecil dari <	5 < 3 bernilai True	Jika nilai operan kiri lebih kecil dari nilai operan kanan, maka kondisi menjadi benar.
Lebih besar atau sama dengan >=	5 >= 3 bernilai True	Jika nilai operan kiri lebih besar dari nilai operan kanan, atau sama, maka kondisi menjadi benar.
Lebih kecil atau sama dengan <=	5 <= 3 bernilai True	Jika nilai operan kiri lebih kecil dari nilai operan kanan, atau sama, maka kondisi menjadi benar.

Figure 5.2 Table Operator Perbandingan.

```

    print ("Baris 1 - a sama dengan b")
else:
    print ("Baris 1 - a tidak sama dengan b")

if ( a != b ):
    print ("Baris 2 - a tidak sama dengan b")
else:
    print ("Baris 2 - a sama dengan b")

if ( a < b ):
    print ("Baris 4 - a kurang dari b")
else:
    print ("Baris 4 - a lebih dari b")

if ( a > b ):
    print ("Baris 5 - a lebih dari b")
else:
    print ("Baris 5 - a kurang dari b")

if ( a >= b ):
    print ("Baris 6 - a lebih besar dari atau
sama dengan b")
else:

```

```

    print ("Baris 6 - a tidak lebih besar dari
atau sama dengan b")

if ( a <= b ):
    print ("Baris 7 - a lebih kecil dari
atau sama dengan b")

else:
    print ("Baris 7 - a tidak lebih kecil dari
atau sama dengan b")

```

Jika program diatas dijalankan, maka akan menghasilkan output di bawah ini:

a = 20 b = 10 Baris 1 - a tidak sama dengan b Baris 2 - a tidak sama dengan b  
 Baris 4 - a lebih dari b Baris 5 - a lebih dari b Baris 6 - a lebih besar dari atau  
 sama dengan b Baris 7 - a tidak lebih kecil dari atau sama dengan b

**5.2.1.3 Bitwise Operator** Operator bitwise (Bitwise Operators) adalah operator khusus yang disediakan PHP untuk menangani proses logika untuk bilangan biner. Bilangan biner atau binary adalah jenis bilangan yang hanya terdiri dari 2 jenis angka, yakni 0 dan 1. Jika operand (sebuah objek yang ada pada operasi matematika yang dapat digunakan untuk melakukan operasi) yang digunakan bukan bilangan biner, maka akan dikonversi secara otomatis oleh PHP menjadi bilangan biner. Operator Bitwise adalah operator untuk melakukan operasi berdasarkan bit/biner. Konsepnya memang hampir sama dengan operator Logika. Namun, Bitwise digunakan untuk biner.

```

AND : &
OR : |
XOR : ^
Negasi/kebalikan : ~
Left Shift : <<
Right Shift : >>

```

Hasil operasi dari operator ini agak sulit dipahami, kalau kita belum paham operasi bilangan biner. Misalnya, kita punya variabel a = 60 dan b = 13.

Bila dibuat dalam bentuk biner, akan menjadi seperti ini: a = 00111100 b = 00001101 Kemudian, dilakukan operasi bitwise

- (a) Operasi AND a = 00111100 b = 00001101 a & b = 00001100
- (b) Operasi OR a = 00111100 b = 00001101 a — b = 00111101
- (c) Operasi XOR a = 00111100 b = 00001101 a ^ b = 00110001
- (d) Operasi NOT (Negasi/kebalikan) a = 00111100 a = 11000011



**5.2.1.4 Operator Logika** Operator logika merupakan operator yang akan menghasilkan sebuah output true atau false, di dalam python memberikan tiga operator logika yaitu, 'and', 'or', dan 'not'. Operator 'and' dan 'or' memberikan hasil bukan boolean, kecuali merupakan operasi boolean. Python memiliki sedikit perbedaan dalam mendeklarasikan operator logika, perbedaannya adalah angka yang bukan bilangan nol (0) dianggap sebagai kondisi true (benar) atau memiliki nilai satu (1). Adapun cara penggunaan dari operator-operator tersebut adalah sebagai berikut:

```
print 10 > 2 and 10 > 5 # bernilai True
print True or False # bernilai True
print not False # bernilai True
perintah diatas digunakan untuk ketika dijalankan maka akan menghasilkan nilai True
```

**5.2.1.5 Operator Identitas** operator identitas membandingkan lokasi memori dari dua benda. Ada dua operator Identitas seperti yang dijelaskan di bawah ini :

- (a) Operator is, Mengevaluasi ke true jika variabel di kedua sisi operator menunjuk ke objek yang sama dan salah sebaliknya Contoh : x adalah y, berikut adalah hasil dalam 1 jika id(x) sama dengan id(y) .
- (b) Operator is not, Mengevaluasi false jika variabel di kedua sisi operator menunjuk ke objek yang sama dan benar sebaliknya. Contoh : x tidak y, di sini bukan hasil dalam 1 jika id(x) tidak sama dengan id(y) .

Di bawah ini adalah contoh implementasi operator identitas pada program di Python:

```
#!/usr/bin/python

a = 20
b = 20

if ( a is b ):
    print "Line 1 - a and b have same identity"
else:
    print "Line 1 - a and b do not have same identity"

if ( id(a) == id(b) ):
    print "Line 2 - a and b have same identity"
else:
    print "Line 2 - a and b do not have same identity"

b = 30
if ( a is b ):
```

```
    print "Line 3 - a and b have same identity"
else:
    print "Line 3 - a and b do not have same identity"

if ( a is not b ):
    print "Line 4 - a and b do not have same identity"
else:
    print "Line 4 - a and b have same identity"
```

Hasil dalam menjalankan program di atas menghasilkan hasil sebagai berikut:

Line 1 - a and b have same identity

Line 2 - a and b have same identity

Line 3 - a and b do not have same identity

Line 4 - a and b do not have same identity



## CHAPTER 6

---

## VARIABEL TYPE

---

### 6.1 Variable Type

Variabel merupakan tempat menyimpan data, sedangkan tipe data adalah jenis data yang tersimpan dalam variabel. Variabel menyimpan data yang dilakukan selama program dieksekusi, yang nantinya isi dari variabel tersebut dapat diubah oleh operasi - operasi tertentu pada program yang menggunakan variabel. Variabel juga dapat dikatakan sebagai entitas yang memiliki nilai dan berbeda satu dengan yang lain. Variabel mengalokasikan memori untuk menyimpan nilai. Hal ini berarti ketika membuat variabel, maka memesan beberapa ruang di memori.

Tipe data di bahasa pemrograman python dibagi menjadi dua kelompok yaitu sebagai berikut ini:

- (a) Immutable, merupakan tipe data yang tidak bisa diubah (string dan bilangan)
- (b) Mutable, merupakan tipe data yang bisa diubah (list dan dictionary)

Variabel bisa digunakan untuk menyimpan bilangan bulat, desimal atau juga karakter. Variabel dapat menyimpan berbagai macam tipe data. Variabel bersifat

mutable, artinya nilainya bisa berubah-ubah. Tidak seperti pemrograman lainnya, variabel pada Python tidak harus dideklarasikan secara eksplisit. Pendeklarasian variabel terjadi secara otomatis ketika kita memberikan sebuah nilai pada suatu variabel. Untuk pemberian nilai, bisa langsung dengan tanda `=`. Berdasarkan tipe data sebuah variabel, penafsir mengalokasikan memori dan memutuskan apa yang dapat disimpan dalam memori yang dipesan. Oleh karena itu, dengan menetapkan tipe data yang berbeda ke variabel, Anda dapat menyimpan bilangan bulat, desimal atau karakter dalam variabel ini.

### 6.1.1 Menetapkan Nilai ke Variabel

Variabel Python tidak memerlukan deklarasi eksplisit untuk memesan ruang memori. Deklarasi terjadi secara otomatis saat Anda menetapkan nilai ke variabel. Tanda sama (`=`) digunakan untuk menetapkan nilai pada variabel.

Operan di sebelah kiri operator `=` adalah nama dari variabel dan operan di sebelah kanan operator `=` adalah nilai yang disimpan dalam variabel. Misalnya:

```
#!/usr/bin/python

counter = 100          # An integer assignment
miles   = 1000.0       # A floating point
name    = "John"       # A string

print counter
print miles
print name
```

Di sini, 100, 1000,0 dan John adalah nilai-nilai ditugaskan untuk counter, miles, dan name variabel, masing-masing. Ini menghasilkan hasil berikut - 100 1000.0 John

### 6.1.2 Aturan Penulisan Variabel

Berikut ini merupakan aturan-aturan yang digunakan dalam penulisan variabel pada bahasa pemrograman python, yaitu sebagai berikut :

- (a) Nama variabel boleh diawali menggunakan huruf atau garis bawah.
- (b) Karakter selanjutnya dapat berupa huruf, garis bawah.
- (c) Karakter pada nama variabel bersifat sensitif (case-sensitif). Artinya huruf besar dan kecil dibedakan.
- (d) variabel ku, keduanya adalah variabel yang berbeda.
- (e) Nama variabel tidak boleh menggunakan kata kunci yang sudah ada dalam python seperti if, while, for, dsb.

### 6.1.3 Membuat Variabel di Python

Variabel di python dapat dibuat dengan format seperti ini:

```
nama\_variabel = <nilai>
Contoh:
variabel\_ku = "ini isi variabel"
variabel2 = 20
Kemudian untuk melihat isi variabel, kita dapat menggunakan fungsi print.
print variabel\_ku
print variabel2
```

### 6.1.4 Menghapus Variabel

Ketika sebuah variabel tidak dibutuhkan lagi, maka kita bisa menghapusnya dengan fungsi del(). Contoh:

```
>>> nama = "petanikode"
>>> print nama
petanikode
>>> del(nama)
>>> print nama
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'nama' is not defined
```

## 6.2 Tipe data

Cara mengisi nilai variabel ditentukan dengan jenis datanya, misalkan untuk tipe data teks (string) maka harus diapit dengan tanda petik (...). Sedangkan untuk angka (integer) dan boolean tidak perlu diapit dengan tanda petik.

Contoh:

```
nama\_ku = "Petani Kode"
umur = 20
tinggi = 183.22
```

Python akan secara otomatis mengenali jenis data atau tipe data yang tersimpan dalam sebuah variabel. Untuk memeriksa tipe data pada suatu variabel, kita bisa menggunakan fungsi type().

Contoh:

```

>>> usia = 20
>>> type(usia)
<type 'int'>
>>> usia = "20"
>>> type(usia)
<type 'str'>
>>> usia = '20'
>>> type(usia)
<type 'str'>
>>> usia = 20.5
>>> type(usia)
<type 'float'>
>>> usia = true
Traceback (most recent call last):
File "<stdin>", line 1, in <module>
NameError: name 'true' is not defined
>>> usia = True
>>> type(usia)
<type 'bool'>

```

### 6.2.1 Jenis-jenis Tipe Data

Secara umum, tipe data primitif dalam python dibagi menjadi tiga jenis:

- (a) Tipe data angka
- (b) Tipe data teks
- (c) Tipe data boolean

### 6.2.2 Penggunaan Variabel

Contoh penggunaan variabel dalam bahasa pemrograman Python adalah sebagai berikut :

```

#proses memasukan data ke dalam variabel
nama = "John Doe"
#proses mencetak variabel
print(nama)

#nilai dan tipe data dalam variabel dapat diubah
umur = 20 #nilai awal
print(umur) #mencetak nilai umur
type(umur) #mengecek tipe data umur
umur = "dua puluh satu" #nilai setelah diubah

```

```
print(umur)          #mencetak nilai umur
type(umur)           #mengecek tipe data umur

namaDepan = "Budi"
namaBelakang = "Susanto"
nama = namaDepan + " " + namaBelakang
umur = 22
hobi = "Berenang"
print("Biodata\n", nama, "\n", umur, "\n", hobi)

#contoh variabel lainya
inivariabel = "Halo"
ini\_juga\_variabel = "Hai"
\_inivariabeljuga = "Hi"
inivariabel222 = "Bye"

panjang = 10
lebar = 5
luas = panjang * lebar
print(luas)
```