

# **CERDAS MENGUASAI PYTHON**



---

# CERDAS MENGUASAI PYTHON

## Dalam 24 Jam

---

**Rolly M. Awangga**  
Informatics Research Center



**Kreatif Industri Nusantara**

***Penulis:***

Rolly Maulana Awangga

ISBN : 978-602-53897-0-2

***Editor:***

M. Yusril Helmi Setyawan

***Penyunting:***

Syafrial Fachrie Pane

Khaera Tunnisa

Diana Asri Wijayanti

***Desain sampul dan Tata letak:***

Deza Martha Akbar

***Penerbit:***

Kreatif Industri Nusantara

***Redaksi:***

Jl. Ligar Nyawang No. 2

Bandung 40191

Tel. 022 2045-8529

Email : awangga@kreatif.co.id

***Distributor:***

Informatics Research Center

Jl. Sariasih No. 54

Bandung 40151

Email : irc@poltekpos.ac.id

Cetakan Pertama, 2019

Hak cipta dilindungi undang-undang

Dilarang memperbanyak karya tulis ini dalam bentuk dan dengan cara  
apapun tanpa ijin tertulis dari penerbit

*‘Jika Kamu tidak dapat  
menahan lelahnya  
belajar, Maka kamu harus  
sanggup menahan  
perihnya Kebodohan.’  
Imam Syafi’i*

# CONTRIBUTORS

---

ROLLY MAULANA AWANGGA, Informatics Research Center., Politeknik Pos Indonesia, Bandung, Indonesia



# CONTENTS IN BRIEF

---

<b>1</b>	<b>Judul Bagian Pertama</b>	<b>1</b>
<b>2</b>	<b>Judul Bagian Kedua</b>	<b>13</b>





# DAFTAR ISI

---

Daftar Gambar	xiii
Daftar Tabel	xv
Foreword	xxi
Kata Pengantar	xxiii
Acknowledgments	xxv
Acronyms	xxvii
Glossary	xxix
List of Symbols	xxxi
Introduction	xxxiii
<i>Rolly Maulana Awangga, S.T., M.T.</i>	

<b>1</b>	<b>Judul Bagian Pertama</b>	<b>1</b>
1.1	Irvan Rizkiansyah	1
1.2	Python	1
1.2.1	Background	1
1.2.2	Problems	2

1.2.3	Objective and Contribution	2
1.2.4	Scoop and Environment	2
1.3	Luthfi Muhammad Nabil_1174035	2
1.3.1	Background	2
1.3.2	Problems	3
1.3.3	Objective and Contribution	3
1.3.4	Scoop and Environment	4
1.4	Hagan Rowlenstino/1174040	4
1.4.1	Background	4
1.4.2	Problems	4
1.4.3	Objective and Contribution	4
1.4.4	Scoop and Environment	5
1.5	Faisal Najib Abdullah 1174042	5
1.5.1	Background	5
1.5.2	Problems	5
1.5.3	Objective and Contribution	5
1.5.4	Scoop and Environment	6
1.6	Ichsan Hizman Hardy/1174034	6
1.6.1	Background	6
1.6.2	Problems	6
1.6.3	Objective and Contribution	7
1.6.4	Scope and Environment	7
1.7	Kevin Natanael Nainggolan 1174059	7
1.7.1	Background	7
1.7.2	Problems	8
1.7.3	Objective and Contribution	8
1.7.4	Scope and Environment	8
1.8	DikaSukmaPradana_1174050	8
1.8.1	Background Python	8
1.8.2	Problems	9
1.9	Muhammad Iqbal Panggabean/1174063	9
1.9.1	Background	9
1.9.2	Problems	10
1.9.3	Objective and Contribution	10
1.9.4	Scope and Environment	11

## **2 Judul Bagian Kedua 13**

2.1	IrvanRizkiansyah/1174043	13
-----	--------------------------	----

2.1.1	Teori	13
2.1.2	Keterampilan Pemrograman	15
2.1.3	Keterampilan Penanganan Error	18
2.2	Hagan Rowlenstino/1174040	18
2.2.1	Teori	18
2.2.2	Keterampilan Pemrograman	19
2.2.3	Keterampilan Penanganan Error	22
2.3	Muhammad Iqbal Panggabean	23
2.3.1	Teori	23
2.3.2	praktek	25
2.3.3	Keterampilan dan penanganan eror	26
2.4	Faisal Najib Abdullah 1174042	27
2.4.1	Teori	27
2.4.2	Keterampilan Pemrograman	28
2.4.3	Keterampilan Penanganan Error	31
Daftar Pustaka		33
Index		35



## DAFTAR GAMBAR

---



# DAFTAR TABEL

---





# Listings

---

src/chapter2/1174043_1.py	15
src/chapter2/1174043_2.py	15
src/chapter2/1174043_3.py	16
src/chapter2/1174043_4.py	16
src/chapter2/1174043_5.py	16
src/chapter2/1174043_6.py	16
src/chapter2/1174043_7.py	16
src/chapter2/1174043_8.py	16
src/chapter2/1174043_9.py	17
src/chapter2/1174043_10.py	17
src/chapter2/1174043_11.py	17
src/chapter2/1174043_2err.py	18
src/chapter2/1174040_1.py	19
src/chapter2/1174040_2.py	20
src/chapter2/1174040_3.py	20
src/chapter2/1174040_4.py	20
src/chapter2/1174040_5.py	20
src/chapter2/1174040_6.py	21

src/chapter2/1174040_7.py	21
src/chapter2/1174040_8.py	21
src/chapter2/1174040_9.py	21
src/chapter2/1174040_10.py	22
src/chapter2/1174040_11.py	22
src/chapter2/1174040_2err.py	22
src/1174063_teor.py	23
src/1174063_teor.py	23
src/1174063_teor.py	23
src/1174063_teor.py	24
src/1174063_teor.py	24
src/1174063_teor.py	24
src/1174063_teor.py	24
src/1174063_teor.py	24
src/1174063_teor.py	25
src/1174063_praktek.py	25
src/1174063_praktek.py	25
src/1174063_praktek.py	25
src/1174063_praktek.py	25
src/1174063_praktek.py	26
src/1174063_praktek.py	26
src/1174063_praktek.py	26
src/1174063_praktek.py	26
src/1174063_praktek.py	26
src/1174063_praktek.py	26
src/1174063_praktek.py	26
src/errr2.py	26
src/chapter2/1174042_1,1.py	27
src/chapter2/1174042_1,2.py	27
src/chapter2/1174042_1,3.py	27
src/chapter2/1174042_1,4.py	27
src/chapter2/1174042_1,5.py	28
src/chapter2/1174042_1,7.py	28
src/chapter2/1174042_2,1.py	28
src/chapter2/1174042_2,2.py	28
src/chapter2/1174042_2,3.py	29
src/chapter2/1174042_2,4.py	29
src/chapter2/1174042_2,5.py	29

src/chapter2/1174042_2,6.py	29
src/chapter2/1174042_2,7.py	29
src/chapter2/1174042_2,8.py	30
src/chapter2/1174042_2,9.py	30
src/chapter2/1174042_2,10.py	30
src/chapter2/1174042_2,11.py	31
src/chapter2/1174042_2err.py	31



# FOREWORD

---

Sepatah kata dari Kaprodi, Kabag Kemahasiswaan dan Mahasiswa



# KATA PENGANTAR

---

Buku ini diciptakan bagi yang awam dengan git sekalipun.

R. M. AWANGGA

*Bandung, Jawa Barat  
Februari, 2019*





# ACKNOWLEDGMENTS

---

Terima kasih atas semua masukan dari para mahasiswa agar bisa membuat buku ini lebih baik dan lebih mudah dimengerti.

Terima kasih ini juga ditujukan khusus untuk team IRC yang telah fokus untuk belajar dan memahami bagaimana buku ini mendampingi proses Intership.

R. M. A.



# ACRONYMS

---

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



# GLOSSARY

---

git	Merupakan manajemen sumber kode yang dibuat oleh linus torvald.
bash	Merupakan bahasa sistem operasi berbasiskan *NIX.
linux	Sistem operasi berbasis sumber kode terbuka yang dibuat oleh Linus Torvald



# SYMBOLS

---

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





# INTRODUCTION

---

ROLLY MAULANA AWANGGA, S.T., M.T.

Informatics Research Center  
Bandung, Jawa Barat, Indonesia

Pada era disruptif saat ini. git merupakan sebuah kebutuhan dalam sebuah organisasi pengembangan perangkat lunak. Buku ini diharapkan bisa menjadi penghantar para programmer, analis, IT Operation dan Project Manajer. Dalam melakukan implementasi git pada diri dan organisasinya.

Rumusnya cuman sebagai contoh aja biar keren[1].

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



# BAB 1

---

## JUDUL BAGIAN PERTAMA

---

### 1.1 Irvan Rizkiansyah

### 1.2 Python

#### 1.2.1 Background

Python adalah sebuah bahasa pemrograman yang bersifat interpreter, interactive, object-oriented, dan dapat beroperasi hampir pada semua platform seperti Windows, Linux, Mac. Python termasuk sebagai bahasa pemrograman yang dapat dengan mudah di pelajari karena sintaks yang jelas dan mudah dipahami, dan dapat dikombinasikan dengan penggunaan modul yang siap pakai, dan struktur data tingkat tinggi yang efisien [2].

Python memiliki kepustakaan atau biasa disebut library yang sangat luas, dan dalam distribusi Python yang telah disediakan, hal tersebut diakibatkan oleh pendistribusian Python yang bebas karena bahasa pemrograman Python merupakan bahasa pemrograman yang freeware atau bebas dalam hal pengembangannya. Python adalah sebuah bahasa pemrograman yang dapat dengan mudah dibaca dan terstruktur, hal tersebut dikarenakan penggunaan sistem identasi, yaitu pemisahan blok-blok

program susunan identasi, jadi untuk menambahkan sub-sub program dalam sebuah blok program, sub program tersebut harus diletakkan pada satu atau lebih spasi dari kolom sebuah blok [3].

Bahasa pemrograman Python dibuat oleh Guido Van Rossum. Dikarenakan para pengembang software atau perangkat lunak lebih cenderung memilih kecepatan dalam menyelesaikan suatu proyek dibandingkan dengan kecepatan proses dari program yang dijalankan, maka dari itu bahasa pemrograman Python dapat dibilang bahasa pemrograman yang kecepatannya dapat melebihi bahasa pemrograman C. Akan tetapi bahasa pemrograman Python lebih lambat dalam memproses suatu program dibandingkan bahasa pemrograman C. dengan berkembangnya kecepatan prosesor dan memori saat ini, mengakibatkan tidak terlihatnya keterlambatan dari sebuah program yang menggunakan bahasa pemrograman Python [4].

### **1.2.2 Problems**

- Kurangnya pemahaman tentang bahasa pemrograman Python
- Kurang mengerti dalam hal fungsi-fungsi yang terdapat pada bahasa pemrograman Python

### **1.2.3 Objective and Contribution**

#### **1.2.3.1 Objective**

- Dapat memahami tentang bahasa pemrograman Python
- Dapat memahami fungsi fungsi yang terdapat pada bahasa pemrograman Python

#### **1.2.3.2 Contribution**

- Dapat membangun sebuah sistem dengan menggunakan bahasa pemrograman Python
- Dapat membangun sebuah alat yang berguna, menggunakan mikrokontroler dan bahasa pemrograman python

### **1.2.4 Scoop and Environment**

- Pengenalan tentang bahasa pemrograman Python
- Pengenalan fungsi-fungsi yang terdapat pada bahasa pemrograman Python

## **1.3 Luthfi Muhammad Nabil\_1174035**

### **1.3.1 Background**

Python adalah sebuah bahasa pemrograman dengan level tinggi yang interaktif, dan mendukung berbagai paradigma pemrograman. Python sudah terkenal pada kalan-

gan programmer sebagai bahasa yang mudah dipahami dan memiliki kompleksitas yang dinamis sehingga dapat dipakai di algoritma maupun platform yang berbagai macam. Python sudah memiliki banyak komunitas pendukung karena penggunaanya yang banyak. Selain pada komunitas biasa, Python sudah diimplementasikan pada banyak perusahaan ternama dan dipasang pada aplikasi yang sudah terkenal seperti pada search engine google yang dimiliki oleh perusahaan Google.

Python mulai dirilis pada tahun 1991 oleh Guido van Rossum sebagai kelanjutan dari bahasa pemrograman ABC dengan memiliki versi yaitu 0.9.0. Nama dari bahasa Python diambil dari program televisi di Inggris bernama Monty Python. Lalu tahun 1995, Guido pindah ke CNRI di Virginia, Amerika sembari melanjutkan pengembangan Python. Versi terakhir yang dikeluarkan telah mencapai 1.6. Pada awalnya, Python adalah bahasa yang dipakai untuk Lalu pada tahun 2000, dirilis Python versi 2.0 yang memiliki peran sebagai bahasa pemrograman tidak berbayar atau open source. Van Rossum sendiri aktif pada development dari Python tetapi sudah bergabung dengan banyak penyumbang. Dibandingkan dengan bahasa lain, Python sudah melewati beberapa versi yang terbatas, mengikuti filosofi dari perubahan berurutan.

Untuk memahami bahasa Python tidak sulit, tetapi instalasi Python cukup memiliki trik tersendiri terlebih untuk pengguna yang baru memasuki lingkup programming. Pada sistem operasi windows, pengguna diharuskan untuk memasuki sistem pada windows untuk mengatur lokasi dari Python yang sudah diinstall. Selain itu, untuk yang terbiasa dengan beberapa pemrograman harus beradaptasi dengan aturan - aturan pada bahasa pemrograman Python seperti penggantian titik koma (;) dengan indentasi. Oleh karena itu, penulis akan membahas mengenai pengenalan singkat mengenai bahasa pemrograman python dan cara instalasi dari python dan library pip.

### 1.3.2 Problems

Sesuai dengan latar belakang yang telah dibahas, penulis merumuskan masalah sebagai berikut :

1. Bagaimana pemaparan singkat mengenai Python?
2. Bagaimana cara melakukan instalasi Python?

### 1.3.3 Objective and Contribution

#### 1.3.3.1 Objective

1. Untuk membahas mengenai Python.
2. Untuk menunjukkan cara instalasi Python.

**1.3.3.2 Contribution** Pada materi ini, penulis menggunakan Python.

### 1.3.4 Scoop and Environment

- Pada Chapter 1 membahas mengenai sejarah, latar belakang, dan keterangan singkat mengenai python tersebut. Chapter ini juga merangkum masalah dan mencari tujuan yang ingin dicapai penulis dalam membuat resume ini.

## 1.4 Hagan Rowlenstino/1174040

### 1.4.1 Background

Python di desain sebagai bahasa pemrograman yang dapat digunakan sehari-hari. Pencipta python, Guido van Rossum, telah menulis seri lengkap tentang sejarah bahasa tersebut. Python diciptakan di awal 1990 di CWI (the Centrum voor Wiskunde and Informatica), tempat kelahiran ALGOL (Algorithmic Language 68). Sebelumnya, Rossum juga telah mengerjakan bahasa pemrograman ABC, yang dikembangkan di CWI sebagai bahasa pengajaran yang menekankan kejelasan. Walaupun project ABC telah di tutup, Rossum banyak belajar dari hal tersebut saat dia mulai membuat Python sebagai alat untuk multimedia dan project penelitian sistem operasi. Dia ingin Python mempunyai tingkatan yang cukup tinggi agar mudah untuk dibaca dan ditulis, juga mirip dengan Java, dan menawarkan portabilitas serta error model yang terdefinisi dengan baik.

Python juga kaya akan vocabulary yang berguna untuk membuat algoritma yang kompleks dengan efisien dikarenakan punya dictionaries yang memiliki string yang kuat dan asosiasi array yang fleksibel. Python menggabungkan antara fleksibilitas tingkat tinggi, kemampuan membaca, dan interface yang terdefinisi dengan baik. Kombinasi tersebut membuat Python cocok untuk menyelesaikan masalah komputasi non-algoritma seperti integrasi dengan web, format data, atau hardware kelas rendah. Python mudah untuk dipelajari karena strukturnya sederhana dan sintaksnya jelas, punya library yang portable dan dapat digunakan di beda perangkat, dan dapat terintegrasi dengan bahasa pemrograman lain seperti C, C++, dan Java.

### 1.4.2 Problems

1. Banyak pemrograman yang penggunaannya kompleks

### 1.4.3 Objective and Contribution

#### 1.4.3.1 Objective

1. Dapat memudahkan pemrograman dengan bahasa pemrograman yang tepat

#### 1.4.3.2 Contribution

1. Menggunakan Python sebagai bahasa pemrograman

### 1.4.4 Scoop and Environment

1. Mengimplementasikan Python dalam pemrograman

## 1.5 Faisal Najib Abdullah 1174042

### 1.5.1 Background

Python lahir pada akhir tahun 1980 an dan implementasinya dimulai pada Desember 1989 oleh Guido van Rossum di CWI di Belanda sebagai penerus bahasa ABC (itu sendiri terinspirasi oleh SETL) yang mampu menangani pengecualian dan berinteraksi dengan sistem operasi Amuba. Van Rossum adalah penulis utama Python, dan peran sentralnya yang berkelanjutan dalam menentukan arah Python tercermin dalam judul yang diberikan kepadanya oleh komunitas Python, Benevolent Dictator for Life (BDFL).

Python adalah bahasa pemrograman interpretatif multiguna dengan filosofi desain yang berfokus pada keterbacaan kode dan python sendiri diklaim sebagai bahasa yang menggabungkan kapabilitas, kemampuan, dengan kode sintaksis yang sangat jelas, dan dilengkapi dengan fungsi pustaka standar yang besar dan komprehensif. Python juga didukung oleh komunitas besar.

Python mendukung pemrograman multi paradigma, terutama tetapi tidak terbatas pada pemrograman berorientasi objek, pemrograman imperatif, dan pemrograman fungsional. Salah satu fitur yang tersedia di Python adalah sebagai bahasa pemrograman dinamis yang dilengkapi dengan manajemen memori otomatis. Python menggunakan bahasa scripting yang sama seperti bahasa pemrograman dinamis, meskipun dalam praktiknya penggunaan bahasa ini lebih luas mencakup konteks penggunaan yang umumnya tidak dilakukan menggunakan bahasa skrip. Python dapat digunakan untuk keperluan pengembangan perangkat lunak dan dapat berjalan di berbagai platform sistem operasi.

CPython, implementasi referensi Python, adalah perangkat lunak bebas dan open source dan memiliki model pengembangan berbasis komunitas, seperti halnya hampir semua implementasi alternatifnya. CPython dikelola oleh Yayasan Perangkat Lunak Python nirlaba [5].

### 1.5.2 Problems

- Mahasiswa D4 TI belum dapat memahami apa itu python
- Mahasiswa D4 TI belum mengerti fungsi fungsi apa saja yang terdapat pada python
- Mahasiswa D4 TI belum dapat menjalankan fungsi python

### 1.5.3 Objective and Contribution

#### 1.5.3.1 Objective



- Mahasiswa D4 TI dapat memahami apa itu python
- Mahasiswa D4 TI dapat memahami fungsi fungsi yang terdapat pada python
- Mahasiswa D4 TI dapat menjalankan fungsi python

### **1.5.3.2 Contribution**

- Mahasiswa D4 TI dapat membangun suatu aplikasi yang mengimplementasikan bahasa python
- Mahasiswa D4 TI dapat membangun alat yang terhubung dengan aplikasi menggunakan bahasa python

### **1.5.4 Scoop and Environtment**

- Mengenali apa itu python pada mahasiswa
- Mengenali fungsi fungsi dasar python dan menjalankannya

## **1.6 Ichsan Hizman Hardy/1174034**

### **1.6.1 Background**

Python merupakan bahasa pemrograman interpretatif multiguna. Python pertama kali, diciptakan oleh Guido van Rossum di Stichting Mathematisch Centrum atau CWI di Belanda pada tahun 1990. pada tahun 1995, Guido melanjutkan karyanya pada Python di Virginia, dimana ia telah merilis beberapa versi perangkat lunak. Tidak seperti bahasa lain yang sulit dibaca dan dipahami, python menekankan keterbacaan kode untuk membuatnya lebih mudah untuk memahami sintaksis. Ini membuat Python sangat mudah dipelajari untuk pemula dan mereka yang telah menguasai bahasa pemrograman lain.

Python dengan desain yang sangat mudah di baca dan dipahami, karena sama seperti bahasa pemrograman yang lainnya yaitu dengan menggunakan bahasa inggris. selain itu juga lebih sedikit dalam penggunaan rumus atau syntac.

Pyton juga mendukung sistem teknik pemrograman yang merangkum kode dalam objek. Bahasa Python mendukung hampir semua sistem operasi, termasuk operasi Linux.

Dengan kode yang simpel dan mudah diimplementasikan, seorang programmer dapat lebih mengutamakan pengembangan aplikasi yang dibuat. Kamu bisa menggunakan Python untuk membuat aplikasi berbasis web, game, atau bahkan sebuah search engine

### **1.6.2 Problems**

1. Mahasiswa D4TI2B belum bisa menggunakan bahasa python.

2. Bagaimana pengaruh bahasa python terhadap mahasiswa D4TI2B.
3. Bagaimana penggunaan bahasa python terhadap web service.

### **1.6.3 Objective and Contribution**

#### **1.6.3.1 Objective**

1. Mahasiswa D4TI2B mampu memahami bahasa pemrograman python secara bertahap.
2. Bahasa pemrograman python mampu mempengaruhi mahasiswa D4TI2B menjadi lebih semangat dalam belajar web service.
3. Penggunaan bahasa python mampu mempermudah mahasiswa dalam membuat web service.

#### **1.6.3.2 Contribution**

1. Membantu mahasiswa D4TI2B dalam menyelesaikan masalah pada python.
2. Membantu mahasiswa D4TI2B memahami bahasa pemrograman python.
3. Mempelajari bahasa python dalam proses pembuatan web service.

### **1.6.4 Scope and Environment**

1. Mahasiswa D4TI2B memahami bahasa pemrograman python.
2. Mahasiswa D4TI2B mampu menjalankan fungsi python.
3. Mahasiswa D4TI2B mampu membuat web service menggunakan python.

## **1.7 Kevin Natanael Nainggolan 1174059**

### **1.7.1 Background**

Python adalah bahasa pemrograman tingkat tinggi yang interpretatif dan memiliki tujuan umum. Python dibuat oleh Guido Van Rossum dan perilisan pertamanya di tahun 1991, Python di desain untuk memberikan performa dalam membaca kode, hal ini memberikan kontribusi untuk membuat pemrograman yang jelas dalam skala besar maupun kecil. Python sendiri sudah dipahami sejak tahun 1980-an oleh Guido Van Rossum di Centrum Wiskunde and Informatic di Belanda sebagai bahasa penerus ABC. Bahasa ini dapat berinteraksi pada dengan sistem operasi Amoeba dan mulai diimplementasikan di bulan Desember 1989.

Pengembangan python ada beberapa, seperti perilisan Python 2.0 menyediakan fitur pengumpul sampah pendeteksi siklus dan dapat digunakan dalam Unicode,

python 3.0 yang dirilis pada 3 Desember 2008, memberikan fitur yang memberikan banyak fitur yang dibuat untuk revisi python versi 2.6.x dan 2.7.x dan penerjemah dari kode python 2 ke kode python 3, lalu python 2.7s yang ditetapkan tahun 2015 ditunda hingga 2020 mendatang dikarenakan kekhawatiran bahwa kode yang ada sebagian besar tidak dapat di-porting ke python 3. Januari 2017, google mengumumkan pengerjaan python 2.7 untuk melakukan pengembangan lebih lanjut.

### **1.7.2 Problems**

1. Pemahaman yang salah tentang Logic error dan Syntax error.
2. Sulit membedakan mana jenis error antara Logic error dan Syntax error.

### **1.7.3 Objective and Contribution**

#### **1.7.3.1 Objective**

1. Orang-orang yang belum paham tentang perbedaan Logic Error dan Syntax error dapat memahami perbedaanya.
2. Dapat menentukan dan menyelesaikan error pada Logic error maupun Syntax error.

#### **1.7.3.2 Contribution**

1. Menjelaskan definisi dari Logic error dan Syntax error.
2. Memberikan gambaran tentang masing-masing error agar dapat membedakannya.

### **1.7.4 Scope and Environment**

1. Pemahaman Logic error dan Syntax error dalam python.

## **1.8 DikaSukmaPradana.1174050**

### **1.8.1 Background Python**

Python adalah bahasa pemrograman tingkat tinggi untuk keperluan umum yang filosofi desainnya menekankan keterbacaan kode. Sintaksis Python memungkinkan programmer untuk mengekspresikan konsep dalam lebih sedikit baris kode daripada yang mungkin dilakukan dalam bahasa seperti C, dan bahasa tersebut menyediakan konstruksi yang dimaksudkan untuk memungkinkan program yang jelas pada skala kecil dan besar[5].

Python mendukung banyak paradigma pemrograman, termasuk gaya pemrograman berorientasi objek, imperatif dan fungsional. Ini fitur sistem tipe yang sepenuhnya dinamis dan manajemen memori otomatis, mirip dengan Skema, Ruby, Perl dan Tclm dan memiliki perpustakaan standar yang besar dan komprehensif[5].

Seperti bahasa dinamis lainnya, Python sering digunakan sebagai bahasa scripting, tetapi juga digunakan dalam berbagai konteks non-scripting. Menggunakan alat pihak ketiga, kode Python dapat dikemas ke dalam program yang dapat dieksekusi mandiri. Penerjemah python tersedia untuk banyak sistem operasi[5].

CPython, implementasi referensi Python, adalah perangkat lunak bebas dan open source dan memiliki model pengembangan berbasis komunitas, seperti halnya hampir semua implementasi alternatifnya. CPython dikelola oleh Yayasan Perangkat Lunak Python nirlaba[5].

Python dikandung pada akhir 1980-an dan implementasinya dimulai pada Desember 1989 oleh Guido van Rossum di CWI di Belanda sebagai penerus bahasa ABC (itu sendiri terinspirasi oleh SETL) yang mampu menangani pengecualian dan berinteraksi dengan sistem operasi Amuba. Van Rossum adalah penulis utama Python, dan peran sentralnya yang terus menerus dalam menentukan arah Python adalah komunitas Python, Diktator Kebajikan untuk Hidup (BDFL)[5].

Python 2.0 dirilis pada 16 Oktober 2000, dengan banyak fitur baru termasuk pengumpul sampah penuh dan dukungan untuk Unicode. Dengan rilis ini, proses pengembangan diubah dan menjadi lebih transparan dan didukung masyarakat[5].

Python 3.0 (juga disebut Python 3000 atau py3k), rilis utama yang tidak kompatibel dengan versi terbelakang, dirilis pada 3 Desember 2008 setelah periode pengujian yang panjang. Banyak fitur utamanya telah di-backport ke Python 2.6 dan 2.7 yang kompatibel dengan backwards[5].

## 1.8.2 Problems

1. Bagaimana mahasiswa D4TI2B bisa menggunakan bahasa python.
2. Bagaimana pengaruh bahasa python terhadap mahasiswa D4TI2B.

## 1.9 Muhammad Iqbal Panggabean/1174063

### 1.9.1 Background

Python adalah salah satu bahasa pemrograman tingkat tinggi yang bersifat interpreter, interactive, objectoriented, dan dapat beroperasi hampir di semua platform: Mac, Linux, dan Windows. Python termasuk bahasa pemrograman yang mudah dipelajari karena sintaks yang jelas, dapat dikombinasikan dengan penggunaan modulmodul siap pakai, dan struktur data tingkat tinggi yang efisien [2].

Distribusi Python dilengkapi dengan suatu fasilitas seperti shell di Linux. Lokasi penginstalan Python biasa terletak di /usr/bin/Python, dan bisa berbeda. Menjalankan Python, cukup dengan mengetikkan Python, tunggu sebentar lalu muncul

tampilan `!!!`, berarti Python telah siap menerima perintah. Ada juga tanda ... yang berarti baris berikutnya dalam suatu blok prompt `'!!!'`. Text editor digunakan untuk modus skrip [?].

Untuk membangun penelitian ini digunakan wxPython yang merupakan toolkit GUI untuk bahasa pemrograman Python. wxPython memungkinkan programmer Python untuk membuat aplikasi dengan pondasi kuat, grafis antarmuka dengan pengguna yang sangat fungsional, sederhana, dan mudah. wxPython diimplementasikan sebagai modul ekstensi oleh Python (kode asli). wxPython membungkus wxWidget sebagai salah satu GUI library populer yang ditulis dalam bahasa C++. Selain itu, digunakan pula Boa Constructor yang merupakan Integrated Development Environment (IDE) untuk Python dan wxPython GUI Builder yang cross-platform. Boa Constructor mampu membuat, memanipulasi frame secara visual (tanpa skrip), dan ada banyak object inspector seperti: browser objek, hirarki warisan, debugger yang canggih, dan bantuan yang sudah terintegrasi [?].

## 1.9.2 Problems

- (a) Bagaimana mahasiswa Teknik Informatika Politeknik Pos Indonesia bisa menggunakan bahasa pemrograman python
- (b) Kenapa mahasiswa Teknik Informatika Politeknik Pos Indonesia harus belajar bahasa pemrograman python
- (c) Bagaimana penggunaan bahasa python terhadap web service.

## 1.9.3 Objective and Contribution

### 1.9.3.1 Objective

- (a) Mahasiswa D4TI2B mampu memahami bahasa pemrograman python secara bertahap.
- (b) Bahasa pemrograman python mampu mempengaruhi mahasiswa D4TI2B menjadi lebih semangat dalam belajar web service.
- (c) Mahasiswa Teknik Informatika Politeknik Pos Indonesia mampu memahami bahasa pemrograman python secara bertahap.
- (d) Bahasa pemrograman python Dapat dijalankan di Linux, Mac, Windows dan termasuk perangkat mobile.
- (e) Penggunaan bahasa python mampu mempermudah mahasiswa dalam membuat web service.

### 1.9.3.2 Contribution

- (a) Membantu mahasiswa D4TI2B dalam menyelesaikan masalah pada python.

- (b) Membantu mahasiswa D4TI2B memahami bahasa pemrograman python.
- (c) Mempelajari bahasa python dalam proses pembuatan web service.
- (d) Membantu mahasiswa Teknik Informatika dalam menyelesaikan masalah pada python.
- (e) Membantu mahasiswa Teknik Informatika memahami bahasa pemrograman python.
- (f) Mempelajari bahasa Python dalam proses pembuatan web service.

#### **1.9.4 Scope and Environment**

- (a) Mahasiswa D4TI2B memahami bahasa pemrograman python.
  - (b) Mahasiswa D4TI2B mampu menjalankan fungsi python.
  - (c) Mahasiswa D4TI2B mampu membuat web service menggunakan python.
3. Mahasiswa Teknik Informatika Politeknik Pos Indonesia memahami bahasa pemrograman python.
  4. Mahasiswa Teknik Informatika Politeknik Pos Indonesia mampu menjalankan fungsi python.
  5. Mahasiswa Teknik Informatika Politeknik Pos Indonesia mampu membuat web service menggunakan python.



## BAB 2

---

# JUDUL BAGIAN KEDUA

---

### 2.1 IrvanRizkiansyah/1174043

#### 2.1.1 Teori

1. Pada python variabel tidak perlu dideklarasikan, pendeklarasian terjadi secara otomatis pada saat memberikan suatu nilai atau data ke variabel. Terdapat beberapa jenis tipe data variabel pada python, diantaranya :
  - Python Numbers, dimana akan menyimpan data yang berupa angka. Penggunaan pada python sebagai berikut : `var1 = 5 var2 = 48.9`
  - Python Text, dimana akan menyimpan data yang berupa teks ataupun karakter. Penggunaan pada python harus diapitkan oleh tanda petik ("..."), contohnya : `nama = "Irvan" jnskelamin = "L"`
  - Python Boolean, dimana yang hanya memiliki 2 nilai yaitu True dan False saja. penggunaan pada python huruf pertama harus kapital, contohnya : `var3 = True var4 = False`
2.
  - Meminta input pada user `nama = input("Masukkan Nama Anda : ")`



- menampilkan output print "Hello Nama Saya Adalah",nama
3.
    - Operator tambah  $a = b + c$
    - Operator kurang  $a = b - c$
    - Operator kali  $a = b * c$
    - Operator bagi  $a = b / c$
    - Konversi integer ke string `konvVar = str(var1)`
    - Konversi string ke integer `konvVar = int(var2)`
  4.
    - Pengulangan for, kemampuan mengulang proses data menggunakan urutan apapun, seperti list. contoh penggunaan pada Python dan contoh kode adalah :

```
for i in range(10):
    print(i)
```

- Pengulangan while, kemampuan mengulang proses data yang akan terus berlanjut jika kondisinya True. contoh penggunaan pada Python dan contoh kode adalah :

```
i= 0
while i < 10 :
    i=i+1
    print ("loop ke =", i)
```

5. Pengambilan keputusan berguna untuk menentukan tindakan apa yang akan diambil sesuai dengan kondisi yang ada. Contohnya :

```
nilai = 9
if(nilai > 7):
    print("Selamat Anda Lulus")
else:
    print("Maaf Anda Tidak Lulus")
```

Dan untuk kondisi di dalam kondisi contohnya :

```
gaji = 10000000
berkeluarga = True
if gaji > 3000000:
    print "Gaji sudah diatas UMR"
    if berkeluarga:
        print "Wajib ikutan asuransi dan menabung untuk pensiun"
    else:
        print "Tidak perlu ikutan asuransi"
else:
    print "Gaji belum UMR"
```

6.
  - Syntax Errors, Salahnya dalam penulisan sintaks. cara penanganannya adalah dengan menganalisa bagian kode yang error dan memperbaiki sintaks tersebut.
  - Exceptions, error yang terjadi karena sintaks tidak dapat dieksekusi. cara penanganannya adalah dengan menganalisa bagian kode yang error dan memperbaiki sintaks tersebut.
7. Try Except adalah cara penanganan error pada Python. Contohnya :

```
x = 0
try:
    x = 1 / 0
except Exception, e:
    print e
```

## 2.1.2 Keterampilan Pemrograman

```
1 NPM = 1174043
2 char = "#"
3 if NPM%3 == 0:
4     char = "*"
5 if NPM%3 == 1:
6     char = "#"
7 if NPM%3 == 2:
8     char = "+"
9 baris = []
10 baris.append("  ##      ##      #####  ##      #####  ##
      ##### ")
11 baris.append(" #####  #####  ##      ##  ##      ##  ##  ##
      ##  ##  ## ")
12 baris.append("  ##      ##      ##      ##      ##      ##  ##
      ##      ## ")
13 baris.append("  ##      ##      ##      ##      ##      ##  ##
      ##      ##### ")
14 baris.append("  ##      ##      ##      #####  ##      ##
      #####  ## ")
15 baris.append("  ##      ##      ##      ##      ##      ##
      ##      ## ")
16 baris.append(" #####  #####  ##      ##      #####
      ##      ##### ")
17 a = 0
18 for x in baris:
19     print(baris[a].replace("#", char))
20     a+=1
```

```
21 print('Masukkan NPM Anda : ')
22 NPM = input()
23 y = NPM[5:7]
24 g = int(y)
25 for i in range(g):
26     print('Halo ' + NPM, 'Apa Kabar?')
```

```

31 print('Masukkan NPM Anda : ')
2 NPM = input()
3 y = NPM[4:7]
4 x = NPM[4:5]
5 y = NPM[5:6]
6 z = NPM[6:7]
7 g = (int(x) + int(y) + int(z))
8 for i in range(g):
9     print('Halo ' + y, 'Apa Kabar?')

```

```

41 print('Masukkan NPM Anda : ')
2 NPM = input()
3 y = NPM[4:5]
4 print('Halo ' + y, 'Apa Kabar?')

```

```

51 a = 1
2 b = 1
3 c = 7
4 d = 4
5 e = 0
6 f = 4
7 g = 3
8 print(str(a)+str(b)+str(c)+str(d)+str(e)+str(f)+str(g))

```

```

61 a = 1
2 b = 1
3 c = 7
4 d = 4
5 e = 0
6 f = 4
7 g = 3
8 print(a+b+c+d+e+f+g)

```

```

71 a = 1
2 b = 1
3 c = 7
4 d = 4
5 e = 0
6 f = 4
7 g = 3
8 print(a*b*c*d*e*f*g)

```

```

81 a = 1
2 b = 1
3 c = 7
4 d = 4
5 e = 0
6 f = 4
7 g = 3
8 print(a)
9 print(b)
10 print(c)
11 print(d)

```

```

12 print(e)
13 print(f)
14 print(g)

```

```

91 a = 1
2  b = 1
3  c = 7
4  d = 4
5  e = 0
6  f = 4
7  g = 3
8  array = []
9  array.append(a)
10 array.append(b)
11 array.append(c)
12 array.append(d)
13 array.append(e)
14 array.append(f)
15 array.append(g)
16 for x in array:
17     if x != 0:
18         if x%2==0:
19             print(x, end = " ")

```

```

101 a = 1
2  b = 1
3  c = 7
4  d = 4
5  e = 0
6  f = 4
7  g = 3
8  array = []
9  array.append(a)
10 array.append(b)
11 array.append(c)
12 array.append(d)
13 array.append(e)
14 array.append(f)
15 array.append(g)
16 for x in array:
17     if x != 0:
18         if x%2==1:
19             print(x, end = " ")

```

```

111 a = 1
2  b = 1
3  c = 7
4  d = 4
5  e = 0
6  f = 4
7  g = 3
8  array = []
9  array.append(a)
10 array.append(b)

```

```

11 array.append(c)
12 array.append(d)
13 array.append(e)
14 array.append(f)
15 array.append(g)
16 for x in array:
17     if x != 0:
18         i = 1
19         bil = 0
20         while i <= x:
21             if x%i==0:
22                 bil+=1
23                 i+=1
24             if bil == 2:
25                 print(x)

```

### 2.1.3 Keterampilan Penanganan Error

1. TypeError yaitu error di dalam tipe data saat melakukan substring dan ingin memasukkannya ke dalam kondisi for yang hanya menerima tipe int. jadi harus merubah tipe inputan yaitu string menjadi integer.

```

21 a = 2
2 b = 'tes'
3 try:
4     print(a + b)
5 except TypeError:
6     print("Integer dan String Tidak Dapat Dijumlah Karena Berbeda Tipe Data")

```

## 2.2 Hagan Rowlenstino/1174040

### 2.2.1 Teori

1. tipe data teks : ada string yaitu kumpulan karakter dan char adalah karakter. penulisannya harus diapit dengan tanda petik 1,2, ataupun 3 ('..'), (".."), ("..." ), (""""...""")

tipe data angka : ada float yaitu bilangan pecahan dan integer yaitu bilangan bulat. penulisannya yaitu dengan menginisialisasikan nama variable lalu masukkan angka (x = 30)

tipe data boolean : tipe yang memiliki dua nilai yaitu true dan false. penggunaannya huruf pertamanya harus kapital True dan False.

2. input().inisialisasikan input tersebut x = input() lalu print(x)
3. +,\*,-/ . misal a = '10' maka integerr = int(a) dan misal a= 10 maka stringg = string(a)

#### 4. while : untuk perulangan yang tidak pasti

```
i = 0
while True:
    if i < 10:
        print "Saat ini i bernilai: ", i
        i = i + 1
    elif i >= 10:
        break

for : untuk perulangan yang pasti
for i in range(0, 10):
    print i
```

#### 5. if kondisi: hasil

```
    dan
    if kondisi:
hasil
if kondisi:
    hasil
```

#### 6. type error = ubah tipe str jadi int

#### 7. taruh try : diatas sintaks yang ingin diketahui jika terjadi error lalu enter dan tulis except: lalu tekan enter dan masukkan tulisan yang akan ditampilkan.

```
a = 2
b = 'as'
try:
    print(a + b)
except TypeError:
    print("Integer dan String Tidak Dapat
        Dijumlah Karena Berbeda Tipe")
```

## 2.2.2 Keterampilan Pemrograman

```
1 NPM = 1174040
2 char = ""#""
3 if NPM%3 == 0:
4     char = ""*""
5 if NPM%3 == 1:
6     char = ""#""
7 if NPM%3 == 2:
```

```

8     char = "+"
9     row = []
10    row.append("    ##          ##          ##          ##          ##          ##
                #####")
11    row.append(" #####          ##          ##          ##          ##          ##
                ##          ##")
12    row.append("    ##          ##          ##          ##          ##          ##
                ##          ##")
13    row.append("    ##          ##          ##          ##          ##          ##
                ##          ##")
14    row.append("    ##          ##          #####          ##          ##
                #####          ##")
15    row.append("    ##          ##          ##          ##          ##          ##
                ##          ##")
16    row.append(" #####          ##          ##          ##          ##          ##
                #####")
17    a = 0
18    for x in row:
19        print(row[a].replace("#", char))
20        a+=1

```

```

21 print('input NPM')
2 NPM = input()
3 y = NPM[5:7]
4 g = int(y)
5 for i in range(g):
6     print('Halo ' + NPM, 'Apa Kabar?')

```

```

31 print('input NPM')
2 NPM = input()
3 y = NPM[4:7]
4 x = NPM[4:5]
5 y = NPM[5:6]
6 z = NPM[6:7]
7 g = (int(x) + int(y) + int(z))
8 for i in range(g):
9     print('Halo ' + y, 'Apa Kabar?')

```

```

41 print('input NPM')
2 NPM = input()
3 y = NPM[4:5]
4 print('Halo ' + y, 'Apa Kabar?')

```

```

51 a = 1
2 b = 1
3 c = 7
4 d = 4
5 e = 0
6 f = 4
7 g = 0
8 print(str(a)+str(b)+str(c)+str(d)+str(e)+str(f)+str(g))

```

```
6: a = 1
  2 b = 1
  3 c = 7
  4 d = 4
  5 e = 0
  6 f = 4
  7 g = 0
  8 print(a+b+c+d+e+f+g)
```

```
7: a = 1
  2 b = 1
  3 c = 7
  4 d = 4
  5 e = 0
  6 f = 4
  7 g = 0
  8 print(a*b*c*d*e*f*g)
```

```
8: a = 1
  2 b = 1
  3 c = 7
  4 d = 4
  5 e = 0
  6 f = 4
  7 g = 0
  8 print(a)
  9 print(b)
 10 print(c)
 11 print(d)
 12 print(e)
 13 print(f)
 14 print(g)
```

```
9: a = 1
  2 b = 1
  3 c = 7
  4 d = 4
  5 e = 0
  6 f = 4
  7 g = 0
  8 array = []
  9 array.append(a)
 10 array.append(b)
 11 array.append(c)
 12 array.append(d)
 13 array.append(e)
 14 array.append(f)
 15 array.append(g)
 16 for x in array:
 17     if x != 0:
 18         if x%2==0:
 19             print(x, end = " ")
```



```

10_1 a = 1
    2 b = 1
    3 c = 7
    4 d = 4
    5 e = 0
    6 f = 4
    7 g = 0
    8 array = []
    9 array.append(a)
   10 array.append(b)
   11 array.append(c)
   12 array.append(d)
   13 array.append(e)
   14 array.append(f)
   15 array.append(g)
   16 for x in array:
   17     if x != 0:
   18         if x%2==1:
   19             print(x, end = " ")

```

```

11_1 a = 1
    2 b = 1
    3 c = 7
    4 d = 4
    5 e = 0
    6 f = 4
    7 g = 0
    8 array = []
    9 array.append(a)
   10 array.append(b)
   11 array.append(c)
   12 array.append(d)
   13 array.append(e)
   14 array.append(f)
   15 array.append(g)
   16 for x in array:
   17     if x != 0:
   18         i = 1
   19         bil = 0
   20         while i <= x:
   21             if x%i==0:
   22                 bil+=1
   23             i+=1
   24         if bil == 2:
   25             print(x)

```

### 2.2.3 Keterampilan Penanganan Error

1. TypeError yaitu error di dalam tipe data disaat melakukan substring dan ingin memasukkannya ke dalam kondisi for yang hanya menerima tipe int. jadi harus merubah tipe inputan yaitu string menjadi integer.

```

2_1 a = 2
    2 b = 'as'

```

```

3 try:
4     print(a + b)
5 except TypeError:
6     print("Integer dan String Tidak Dapat Dijumlah Karena Berbeda Tipe")

```

## 2.3 Muhammad Iqbal Panggabean

### 2.3.1 Teori

1. Jenis jenis variable phyton dan cara pemakaiannya Variabel merupakan tempat menyimpan data. Dalam Phyton terdapat beberapa variabel dengan berbagai type data diantaranya adalah variabel dengan type data number, string, dan boolean. Dalam phyton kita dapat membuat variable dengan cara sebagai gambar berikut

```

1 c = 7
2 d = "Gabe Tamvan"
3
4 print(c)
5 print(d)

```

2. Kode untuk meminta input dari user dan bagaimana melakukan output ke layar

```

1 npm = input()
2 print(npm)

```

3. Operator dasar aritmatika Ada operator penambahan, pengurangan perkalian, perkalian, pembagian, modulus, perpangkatan, dan pembulatan decimal.

```

1 x = 5
2 y = 3
3
4 print(x+y)
5
6 x = 5
7 y = 3
8
9 print(x-y)
10
11 x = 12
12 y = 3
13
14 print(x*y)
15
16 x = 8
17 y = 2
18
19 print(x/y)
20
21 x = 5
22 y = 2

```

```

23
24 print(x%y)

```

4. Perulangan Terdapat dua jenis perulangan di dalam python yaitu perulangan while dan perulangan for

```

1 ulang = ["topi", "celana", "baju"]
2 for i in range(ulang):
3     print(i)

```

```

1 coba = 1
2 while i < 6 :
3     print(i)
4     i += 1

```

5. sintak Untuk memilih kondisi, dan kondisi didalam kondisi Pengambilan kondisi If yang digunakan untuk mengantisipasi kondisi yang terjadi saat program dijalankan dan menentukan tindakan apa yang akan diambil sesuai dengan kondisi.

```

1 a = 40
2 b = 300
3 if b > a :
4     print("b lebih besar dari a")

```

```

1 a = 40
2 b = 40
3 if b > a :
4     print("b besar dari a")
5 elif a == b :
6     print("a sama dengan b")

```

```

1 a = 300
2 b = 40
3 if b > a :
4     print("b besar dari a")
5 elif a == b :
6     print("a sama dengan b")
7 else :
8     print("a lebih dari b")

```

6. Jenis-jenis error pada python Syntax Errors adalah keadaan dimana kode python mengalami kesalahan penulisan. ZeroDivisonError adalah eror yang terjadi saat eksekusi program menghasilkan perhitungan matematika pembagian dengan angka nol. NameError adalah eror yang terjadi saat kode di eksekusi terhadap local name atau global name yang tidak terdefinisi. TypeError adalah eror yang terjadi saat dilakukan eksekusi pada suatu operasi atau fungsi dengan type object yang tidak sesuai.

7. Cara memakai try except Cara pemakaian try except adalah sebagai berikut :

```

1 x = 0
2 y = "1"
3 try :
4     z = x + y
5     print(z)
6 except TypeError :
7     print("Perbedaan data")

```

## 2.3.2 praktek

### 1. Jawaban soal no 1

```

1 print ("+++  +++  ++++++++  +++  +++  ++++++++")
2 print ("+++  +++  ++++++++  +++  +++  ++++++++")
3 print ("+++  +++  +++  +++  +++  +++  +++  +++")
4 print ("+++  +++  +++  +++  +++  +++  +++  +++")
5 print ("+++  +++  +++  ++++++++  +++  +++")
6 print ("+++  +++  +++  ++++++++  +++  +++  +++")
7 print ("+++  +++  +++  ++++++++  +++  +++  +++  +++")
8 print ("+++  +++  +++  ++++++++  +++  +++  +++  +++")
9 print ("+++  +++  +++  ++++++++  +++  ++++++++")
10 print ("+++  +++  +++  ++++++++  +++  ++++++++")

```

### 2. Jawaban soal no 2

```

1 npm = input("Masukkan NPM :")
2 hitung = 0
3 while(hitung <= 63):
4     print("Hallo , " + str(npm) + "Apa kabar?")
5     hitung = hitung +1

```

### 3. Jawaban soal no 3

```

1 npm = input("Masukkan NPM :")
2 hitung = 0
3 while(hitung <= 6):
4     print("Halo , " + str(npm[4:7]) + " Apa kabar?")
5     hitung = hitung +1

```

### 4. Jawaban soal no 4

```

1 npm = input("Masukkan NPM : ")
2 print("Hello , " + str(npm[-3]) + " Apa kabar?")

```

## 5. Jawaban soal no 5

```

1 a = 1
2 b = 1
3 c = 7
4 d = 4
5 e = 0
6 f = 6
7 g = 3
8 h = a+b+c+d+e+f+g
9 i = a*b*c*d*e*f*g
10
11 print("Hasil no 5")
12 print("NPM anda")
13 print(a, b, c, d, e, f, g, h)

```

## 6. Jawaban soal no 6

```

1 print("Hasil no 6")
2 print(h)

```

## 7. Jawaban soal no 7

```

1 print("Hasil no 7")
2 print(i)

```

## 8. Jawaban soal no 8

```

1 print("Hasil no 8")
2 thisset = (a, b, c, d, e, f, g)
3
4 for x in thisset:
5     print(x)

```

## 9. Jawaban soal no 9

```

1 print(d,e,f)

```

## 10. Jawaban soal no 10

```

1 print(a,b,c,g)

```

## 11. Jawaban soal no 11

```

1 print(a,b,c,g)

```

### 2.3.3 Keterampilan dan penanganan error

```

1 c = 1
2 d = "4"
3
4 try:
5     a = c - d
6     print(a)
7 except:
8     print("hampura euy edit errornya")

```

## 2.4 Faisal Najib Abdullah 1174042

### 2.4.1 Teori

1. Variabel merupakan tempat menyimpan data, sedangkan tipe data adalah jenis data yang tersimpan dalam variabel.

```

1 x = "aku"
2 y = "sayang"
3 z = "najib"
4 a = x + y + z
5 print(z)
6
7 #hasil aku sayang najib

```

Variabel x memiliki nilai aku, variable y memiliki nilai sayang, dan variabel z memiliki nilai najib. karna memiliki type data string maka kata kata tersebut jika di tambahkan berubah menjadi sebuah kalimat

2. Input, untuk membuat kode input, pertama buat variabel x yang berisi input seperti pada contoh jika di run maka langsung diminta untuk memasukan NIM ketika di enter hasilnya berupa Hello, 1174042

```

1 print("NPM:")
2 x = input()
3 print("Hello , " + x)

```

3. Untuk merubah type data dari string ke integer, tambahkan kata int lalu kurung buka lalu nama variabel yang akan dirubah dan kurung tutup seperti pada contoh.

```

1 x = 5
2 y = 3
3 q = "7"
4 z = "8"
5
6 print(x + y)
7 print(x - y)
8 print(x * y)
9 print(x / y)
10 print(int(q) + int(z))
11 print(str(x) + str(y))

```

4. untuk perulangan disini menggunakan while variabel i bernilai 0, kemudian while i lebih kecil dari 6 jika benar maka akan terus dilakukan pengulangan dan jika salah tidak akan dilakukan pengulangan, i selalu bertambah 1, dan menampilkan nilai i.

```

1 x = 5
2 i = 0
3 while i < 6:
4     i += 1
5     print(i)

```

5. membuat 2 variabel a dan b, variabel a bernilai 200 dan b bernilai 33 jika b lebih besar dari a maka akan menampilkan sesuai perintah seperti contoh.

```

1 a = 200
2 b = 33
3 if b > a:
4     print("b lebih kecil dari a")
5 elif a == b:
6     print("a dan b tidak sama")
7 else:
8     print("a lebih besar dari b")

```

6. Jenis error yang sering di alami pada python

- menjumlahkan bilangan yang berbeda type data. Solosinya rubah dan sesuaikan type data yang dibutuhkan
- sepasi pada kondisi yang harus sejajar. Sejajarkan posisi sesuai kondisi
- Typo. Cek kembali agar tidak terjadi kesalahan code

7. untuk menggunakan try, pertama tuliskan coba terlebih dahulu code apakah terjadi error atau tidak. Jika terjadi error copy TypeError kemudian tuliskan try sebelum line yg error, dibawah line yg error tuliskan except dan paste typeerror yang sebelumnya sudah di copy, kemudian tuliskan kenapa bisa terjadi error menggunakan katakata sendiri.

```

1 x = 1
2 c = "7"
3
4 try :
5     d = x + c
6 except TypeError:
7     print("type data berbeda")

```

## 2.4.2 Keterampilan Pemrograman

```

1 print(1174042%3)
2 print("# # ### # # # #")
3 print("# # # ## # # ## #")
4 print("# # # ### # # ### #")
5 print("# # # # # # ###")

```

```

2 print("NPM1:")
2 x = input()
3 a = 0
4 b = int(x[5:7])
5 while a < b :
6     a += 1
7     print("Hello , " + x + " Apa Kabar?")

```

```

31 print("NPM1:")
2  x = input()
3  a = 0
4  k = int(x[4])
5  p = int(x[5])
6  i = int(x[6])
7  b = x[4:7]
8  c = k+p+i
9
10 while a < c:
11     a += 1
12     print("Hello , " + b + " Apa Kabar?")

```

```

41 print("NPM1:")
2  x = input()
3  p = x[4]
4  print("Hello , " + p + " Apa Kabar?")

```

```

51 print("NPM1:")
2  a = 1
3  b = 1
4  c = 7
5  d = 4
6  e = 0
7  f = 4
8  g = 2
9  x = str(a)+str(b)+str(c)+str(d)+str(e)+str(f)+str(g)
10 print("Hello , " + x + " Apa Kabar?")

```

```

61 print("NPM1:")
2  a = 1
3  b = 1
4  c = 7
5  d = 4
6  e = 0
7  f = 4
8  g = 2
9  x = a+b+c+d+e+f+g
10 print(x)

```

```

71 print("NPM1:")
2  a = 1
3  b = 1
4  c = 7
5  d = 4
6  e = 0
7  f = 4
8  g = 2
9  x = a*b*c*d*e*f*g
10 print(x)

```



```

8. print("NPM1:")
2 a = 1
3 b = 1
4 c = 7
5 d = 4
6 e = 0
7 f = 4
8 g = 2
9 print(a)
10 print(b)
11 print(c)
12 print(d)
13 print(e)
14 print(f)
15 print(g)

```

```

9. print("NPM1:")
2 a = 1
3 b = 1
4 c = 7
5 d = 4
6 e = 0
7 f = 4
8 g = 2
9
10 if a%2 == 0:
11     print(a)
12
13 if b%2 == 0:
14     print(b)
15
16 if c%2 == 0:
17     print(c)
18
19 if d%2 == 0:
20     print(d)
21
22 if e%2 == 0:
23     print(e)
24
25 if f%2 == 0:
26     print(f)
27
28 if g%2 == 0:
29     print(g)

```

```

10. print("NPM1:")
2 a = 1
3 b = 1
4 c = 7
5 d = 4
6 e = 0
7 f = 4
8 g = 2
9

```

```

10 if a%2 == 1:
11     print(a)
12
13 if b%2 == 1:
14     print(b)
15
16 if c%2 == 1:
17     print(c)
18
19 if d%2 == 1:
20     print(d)
21
22 if e%2 == 1:
23     print(e)
24
25 if f%2 == 1:
26     print(f)
27
28 if g%2 == 1:
29     print(g)

```

```

11 a = 1
2  b = 1
3  c = 7
4  d = 4
5  e = 0
6  f = 4
7  g = 0
8  array = []
9  array.append(a)
10 array.append(b)
11 array.append(c)
12 array.append(d)
13 array.append(e)
14 array.append(f)
15 array.append(g)
16 for x in array:
17     if x != 0:
18         i = 1
19         bil = 0
20         while i <= x:
21             if x%i == 0:
22                 bil += 1
23             i += 1
24         if bil == 2:
25             print(x)

```

### 2.4.3 Keterampilan Penanganan Error

1. Pada saat mengerjakan praktek kedua ini error hanya pada kesalahan type data yaitu `TypeError`;, solusinya yaitu merubah type data.

```

21 a = 1
2  b = 10

```

```
3 c = "7"  
4  
5 try :  
6     d = a + c  
7 except TypeError:  
8     print("ini teh beda atuhh type datanya mana bisa bersatu")
```

# DAFTAR PUSTAKA

---

1. R. Awangga, "Sampeu: Servicing web map tile service over web map service to increase computation performance," in *IOP Conference Series: Earth and Environmental Science*, vol. 145, no. 1. IOP Publishing, 2018, p. 012057.
2. D. A. Prasetya and I. Nurviyanto, "Deteksi wajah metode viola jones pada opencv menggunakan pemrograman python," 2012.
3. T. R. Perkasa, H. Widyantara, and P. Susanto, "Rancang bangun pendeteksi gerak menggunakan metode image subtraction pada single board computer (sbc)," *Journal JCONES*, vol. 3, no. 2, 2014.
4. M. Miftakhuddin, W. Suadi, and B. A. Pratomo, "Implementasi key-value store dengan struktur data list dan tree menggunakan python," *Institut Teknologi Sepuluh Nopember*.
5. G. Van Rossum *et al.*, "Python programming language." in *USENIX annual technical conference*, vol. 41, 2007, p. 36.



# Index

---

disruptif, xxxiii  
modern, xxxiii