

CERDAS MENGUASAI PYTHON

Penulis:

Rolly Maulana Awangga

ISBN : 978-602-53897-0-2

Editor:

M. Yusril Helmi Setyawan

Penyunting:

Syafrial Fachrie Pane

Khaera Tunnisia

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. Sariasisih 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

1 Mengenal Python dan Anaconda	1
2 Pemograman Dasar	11
3 Praktek Pemograman Dasar	19
4 Fungsi dan Kelas	23
5 Praktek Fungsi dan Kelas	27
6 Library CSV dan Pandas	33
7 Praktek Library CSV dan Pandas	51
8 Komunikasi Perangkat Keras	65

DAFTAR ISI

Daftar Gambar	xiii
Daftar Tabel	xvii
Foreword	xxv
Kata Pengantar	xxvii
Acknowledgments	xxix
Acronyms	xxxi
Glossary	xxxiii
List of Symbols	xxxv
Introduction	xxxvii
<i>Rolly Maulana Awangga, S.T., M.T.</i>	
1 Mengenal Python dan Anaconda	1
1.1 D. Irga B. Naufal Fakhri D4 TI 2C	1
1.1.1 Sejarah Python	1
1.1.2 Tanggal Rilis Python	2
1.1.3 Perbedaan Python 2 dengan Python 3	3

1.1.4	Penggunaan Python di perusahaan dunia	4
1.1.5	Cara menginstall Anaconda	5
1.1.6	Cara menggunakan Spyder pada Anaconda	8
1.1.7	Membuat Hello World di Spyder	9
2	Pemrograman Dasar	11
2.1	D. Irga B. Naufal Fakhri	11
2.1.1	Teori	11
3	Praktek Pemrograman Dasar	19
3.1	D. Irga B. Naufal Fakhri	19
3.1.1	Ketrampilan Pemrograman	19
3.1.2	Ketrampilan Penanganan Error	22
4	Fungsi dan Kelas	23
4.1	D. Irga B. Naufal Fakhri	23
4.1.1	Pemahaman Teori	23
5	Praktek Fungsi dan Kelas	27
5.1	D. Irga B. Naufal Fakhri	27
5.1.1	Keterampilan Pemrograman	27
5.1.2	Ketrampilan Penanganan Error	30
6	Library CSV dan Pandas	33
6.1	D. Irga B. Naufal Fakhri	33
6.1.1	Pemahaman Teori	33
6.2	Fanny Shafira Damayanti 1174069	36
6.2.1	Pemahaman Teori	36
6.3	Aulyardha Anindita 1174054	39
6.3.1	Pemahaman Teori	39
6.4	Nurul Izza Hamka 1174062	42
6.4.1	Pemahaman Teori	42
6.5	Chandra Kirana Poetral 1174079	45
6.5.1	Teori	45
7	Praktek Library CSV dan Pandas	51
7.1	D. Irga B. Naufal Fakhri	51
7.1.1	Soal 1	51

7.1.2	Soal 2	52
7.1.3	Soal 3	52
7.1.4	Soal 4	52
7.1.5	Soal 5	52
7.1.6	Soal 6	52
7.1.7	Soal 7	53
7.1.8	Soal 8	53
7.1.9	Soal 9	53
7.1.10	Keterampilan Penanganan Error	54
7.2	Fanny Shafira Damayanti 1174069	54
7.2.1	Keterampilan Pemrograman	54
7.2.2	Penanganan Error	55
7.3	Aulyardha Anindita 1174054	56
7.3.1	Keterampilan Pemrograman	56
7.3.2	Keterampilan Penanganan Error	57
7.4	Nurul Izza Hamka 1174062 Teori	58
7.5	Keterampilan Pemrograman	58
7.6	Ketrampilan Penanganan Error	60
7.7	Chandra Kirana Poetra	60
7.7.1	Soal 1	60
7.7.2	Soal 2	61
7.7.3	Soal 3	61
7.7.4	Soal 4	61
7.7.5	Soal 5	61
7.7.6	Soal 6	61
7.7.7	Soal 7	62
7.7.8	Soal 8	62
7.7.9	Soal 9	62
7.7.10	Keterampilan Penanganan Error	63
8	Komunikasi Perangkat Keras	65
8.1	D. Irga B. Naufal Fakhri	65
8.1.1	Pemahaman Teori	65
8.2	Difa Al Fansha	68
8.2.1	Soal Nomor 1	68
8.2.2	Soal Nomor 2	69
8.2.3	Soal Nomor 3	72
8.2.4	Soal Nomor 4	72

8.2.5	Soal Nomor 5	72
8.2.6	Soal Nomor 6	72
8.2.7	Soal Nomor 7	72
8.3	Fanny Shafira Damayanti 1174069	73
8.3.1	Pemahaman Teori	73
8.4	Aulyardha Anindita 1174054	76
8.4.1	Pemahaman Teori	76

DAFTAR GAMBAR

1.1	Tampilan Instalasi 1	5
1.2	Tampilan Instalasi 2	5
1.3	Tampilan Instalasi 3	6
1.4	Tampilan Instalasi 4	6
1.5	Tampilan Instalasi 5	6
1.6	Tampilan Instalasi 6	7
1.7	Tampilan Instalasi 7	7
1.8	Tampilan Instalasi 8	7
1.9	Tampilan Instalasi 9	8
1.10	Tampilan awal Anaconda	8
1.11	Tampilan spider	8
1.12	Tampilan new file pada spider	9
1.13	print("Hello World")	9

1.14	Dialog simpan file	9
1.15	Hasil Program	10
6.1	Buat data CSV	46
6.2	Menyimpan CSV	47
6.3	Menyimpan CSV	48
8.1	Setup	66
8.2	Instalasi	66
8.3	Instalasi Berhasil	66
8.4	Device Manager	67
8.5	Arduino Terdeteksi	67
8.6	Plagiarisme	68
8.7	Licence Agreement	69
8.8	Installation Options	70
8.9	Installation Folder	70
8.10	Installation	71
8.11	Installing	71
8.12	Driver baru	73
8.13	Buka device manager	73
8.14	klik "unknown device"	74
8.15	Klik update driver software	74
8.16	Pilih browse	74
8.17	Cari folder instalasinya	74
8.18	install	75
8.19	selesai	75
8.20	plagiarisme	76
8.21	Setup	77
8.22	Instalasi	77
8.23	Search Device Manager	78

8.24	Unknown Device	78
8.25	Update Driver Software	78
8.26	Mencari folder untuk driver software	79
8.27	Lokasi folder	79
8.28	Install PAada tampilan WIndows Security	79
8.29	Pesan Peringatan Berhasil	80
8.30	Peringatan COM Arduino Uno	80
8.31	Plagiarisme	81

DAFTAR TABEL

Listings

2.1	Contoh kode variable Boolean.	11
2.2	Contoh kode variable String.	11
2.3	Contoh kode variable Integer.	12
2.4	Contoh kode variable Float.	12
2.5	Contoh kode variable Hexadecimal.	12
2.6	Contoh kode variable Complex.	12
2.7	Contoh kode variable List.	12
2.8	Contoh kode variable Tuple.	12
2.9	Contoh kode variable Set.	12
2.10	Contoh kode variable Dictionary.	13
2.11	Contoh kode input dan outputnya.	13
2.12	Contoh kode operasi pertambahan.	13
2.13	Contoh kode operasi pengurangan.	13
2.14	Contoh kode operasi perkalian.	13
2.15	Contoh kode operasi pembagian.	14
2.16	Contoh kode operasi modulus.	14
2.17	Contoh kode operasi perpangkatan.	14
2.18	Contoh kode operasi pembulatan hasil pembagian kebawah.	14

src/3/1174066/1174066.py	29
src/3/1174066/1174066.py	29
src/3/1174066/1174066.py	29
src/3/1174066/main_1174066.py	30
src/3/1174066/main_1174066.py	30
src/3/1174066/main_1174066.py	31
src/4/1174066/Teori/1174066.csv	33
src/4/1174066/Teori/1174066.csv	34
src/4/1174066/Teori/1174066_csv.py	34
src/4/1174066/Teori/1174066_csv.py	35
src/4/1174066/Teori/1174066_csv.py	35
src/4/1174066/Teori/1174066_pandas.py	36
src/4/1174066/Teori/1174066_pandas.py	36
src/4/1174066/Teori/1174066_pandas.py	36
src/4/1174069/Teori/1174069_csvteori.py	37
src/4/1174054/Teori/contohcsv.py	40
src/4/1174062/Teori/CSV.py	42
src/4/1174079/Teori/1174079.csv	45
src/4/1174079/Teori/contohcsvchandra.py	49
src/4/1174079/Teori/contohcsvwrite.py	49
src/4/1174079/Teori/contohcsvwriteparameter.py	49
src/4/1174079/Teori/pandasreadchandra.py	50
src/4/1174079/Teori/pandaswritechandra.py	50
src/4/1174066/Praktek/lib_1174066_csv.py	51
src/4/1174066/Praktek/lib_1174066_csv.py	52
src/4/1174066/Praktek/lib_1174066_pandas.py	52
7.1 lib_1174066_csv.py	53
7.2 main.py	53
7.3 main2.py	53
7.4 main2.py	54
src/4/1174069/Praktek/1174069_csv.py	54
src/4/1174069/Praktek/1174069_csv.py	54
src/4/1174069/Praktek/1174069_pandas.py	54
src/4/1174069/Praktek/1174069_pandas.py	54

src/4/1174069/Praktek/1174069_pandas.py	55
src/4/1174069/Praktek/1174069_pandas.py	55
src/4/1174069/Praktek/1174069_pandas.py	55
src/4/1174069/Praktek/main.py	55
src/4/1174069/Praktek/main2.py	55
src/4/1174069/Praktek/1174069.py	56
src/4/1174054/Praktek/1174054csv.py	56
src/4/1174054/Praktek/1174054csv.py	56
src/4/1174054/Praktek/1174054pandas.py	56
src/4/1174054/Praktek/1174054pandas.py	57
src/4/1174054/Praktek/main.py	57
src/4/1174054/Praktek/main2.py	57
src/4/1174054/Praktek/1174054.py	58
src/4/1174062/Praktek/1174062csv.py	58
src/4/1174062/Praktek/1174062csv.py	58
src/4/1174062/Praktek/1174062Pandas.py	59
src/4/1174062/Praktek/main.py	59
src/4/1174062/Praktek/main2.py	59
src/4/1174062/Praktek/1174062.py	60
src/4/1174079/Praktek/lib_1174079_csv.py	60
src/4/1174079/Praktek/lib_1174079_csv.py	61
src/4/1174079/Praktek/lib_1174079_pandas.py	62
src/4/1174079/Praktek/lib_1174079_pandas.py	62
src/4/1174079/Praktek/lib_1174079_csv.py	62
7.5 main.py	62
7.6 main2.py	62
7.7 main2.py	63
src/4/1174079/Praktek/contoherrorpandas.py	63

src/5/1174066/Teori/1174066.py	68
src/5/1174069/Teori/1174069_teori.py	76
src/5/1174054/Teori/1174054.py	81

FOREWORD

Sepatah kata dari Kaprodi, Kabag Kemahasiswaan dan Mahasiswa

KATA PENGANTAR

Buku ini diciptakan bagi yang awam dengan flask 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

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[?].

$$ABC\mathcal{DEF}\alpha\beta\Gamma\Delta \sum_{def}^{abc} \quad (I.1)$$

BAB 1

MENGENAL PYTHON DAN ANACONDA

1.1 D. Irga B. Naufal Fakhri D4 TI 2C

1.1.1 Sejarah Python

Python adalah bahasa pemrograman interpretatif multiguna dengan filosofi perancangan yang berfokus pada tingkat keterbacaan kode. Python diklaim sebagai bahasa yang menggabungkan kapabilitas, kemampuan, dengan sintaksis kode yang sangat jelas, dan dilengkapi dengan fungsionalitas pustaka standar yang besar serta komprehensif.

Python diciptakan oleh Guido van Rossum di Scitchting Mathematisch Centrum (CWI) di Belanda pada tahun 1990-an. Bahasa python terinspirasi dari bahasa pemrograman ABC dan merupakan kelanjutan dari bahasa tersebut. Nama python sendiri bukan berasal dari nama ular python namun karena Guido adalah penggemar grup komedi Inggris bernama Monty Python. Guido masih menjadi penulis utama untuk python, walaupun python bersifat open source sehingga ribuan orang juga berkontribusi dalam mengembangkan python.

Di tahun 1995, Guido melanjutkan pembuatan python di Corporation for National Research Initiative (CNRI) di Virginia Amerika, dimana dia merilis beberapa versi

dari python. Pada Mei 2000, Guido dan tim Python pindah ke BeOpen.com dan membentuk tim BeOpen PythonLabs. Di bulan Oktober pada tahun yang sama, tim python pindah ke Digital Creation (sekarang menjadi Perusahaan Zope). Pada tahun 2001, dibentuklah Organisasi Python yaitu Python Software Foundation (PSF). PSF merupakan organisasi nirlaba yang dibuat khusus untuk semua hal yang berkaitan dengan hak intelektual Python. Perusahaan Zope menjadi anggota sponsor dari PSF.

1.1.2 Tanggal Rilis Python

Semua versi python yang dirilis bersifat open source. Dalam sejarahnya, hampir semua rilis python menggunakan lisensi GFL-compatible. Berikut adalah versi major dan minor python berikut tanggal rilisnya.

- Python 1.0 – Januari 1994
- Python 1.2 – 10 April 1995
- Python 1.3 – 12 Oktober 1995
- Python 1.4 – 25 Oktober 1996
- Python 1.5 – 31 Desember 1997
- Python 1.6 – 5 September 2000
- Python 2.0 – 16 Oktober 2000
- Python 2.1 – 17 April 2001
- Python 2.2 – 21 Desember 2001
- Python 2.3 – 29 Juli 2003
- Python 2.4 – 30 Nopember 2004
- Python 2.5 – 19 September 2006
- Python 2.6 – 1 Oktober 2008
- Python 2.7 – 3 Juli 2010
- Python 3.0 – 3 Desember 2008
- Python 3.1 – 27 Juni 2009
- Python 3.2 – 20 Februari 2011
- Python 3.3 – 29 September 2012
- Python 3.4 – 16 Maret 2014
- Python 3.5 – 13 September 2015
- Python 3.6 – 23 Desember 2016

1.1.3 Perbedaan Python 2 dengan Python 3

Pada Python 2 dan Python 3 memiliki kesamaan kapabilitas namun cara penggunaannya berbeda

- Print

Pada python2, print lebih seperti statement daripada fungsi

```
1 print "Saya Belajar Python"
```

sedangkan pada python3, print digunakan sebagai fungsi

```
1 print("Saya Belajar Python")
```

- Pembagian pada Interger

Pada Python 2, semua tipe data angka yang tidak mengandung desimal akan diperlakukan sebagai integer. Terlihat mudah pada awalnya, ketika mencoba untuk membagi kedua integer akan didapatkan tipe data float.

```
1 3 / 2 = 1.5
```

Python 2 menggunakan floor division atau dibulatkan ke nilai paling rendah misalnya 1.5 jadi 1, 2.6 jadi 2 dan seterusnya. Pada Python 2.7 akan menjadi seperti ini:

```
1 3
2 4
3 x = 3 / 2
4 print a
5 #Output
6 1
```

Untuk desimal maka tambahkan .0 setelah bilangan dan menjadi seperti ini 3.0 / 2.0 untuk mendapatkan hasil 1.5 Pada Python 3, pembagian pada bilangan integer lebih intuitif:

```
1 a = 3 / 2
2 print(a)
3 #Output
4 1.5
```

Kita juga masih bisa melakukan 3.0 / 2.0 untuk mendapatkan 1.5 namun untuk mendapatkan floor division maka pada Python 3 gunakan //:

```
1 b = 3 // 2
2 print(b)
3 #Output
4 1
```

Fitur pada Python 3 ini tidak bisa digunakan pada Python 2.7

- Dukungan Unicode

Ketika bahasa pemrograman menangani tipe data string (yang mana merupakan sekumpulan karakter), mereka bisa melakukan beberapa cara berbeda sehingga komputer dapat mengubah angka ke huruf dan simbol lain. Python 2 menggunakan alfabet ASCII secara default, sehingga ketika kita mengetik "Halo!" maka Python 2 menangani string sebagai ASCII. Terbatas pada beberapa ratus karakter, ASCII mungkin bukan pilihan yang fleksibel untuk menangani proses encoding terutama yang non English.

Untuk menggunakan unicode yang lebih luwes, mendukung lebih dari 128,000 karakter maka kita harus mengetik u"Halo!", dengan tambahan u di depannya yang mana berarti Unicode.

Python 3 menggunakan Unicode secara default, yang mana menyelamatkan programmer dari tambahan kode lagi, lebih hemat waktu dan mudah untuk diisikan dan ditampilkan. Karena Unicode mendukung berbagai karakter linguistik yang beragam termasuk menampilkan emoji, penggunaan karakter secara default dengan encoding memastikan perangkat mobile didukung oleh program yang kita buat.

Jika kita ingin kode Python 3 kita mendukung Python 2, tambahkan u di depan string.

1.1.4 Penggunaan Python di perusahaan dunia

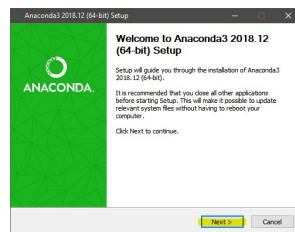
1. Google adalah perusahaan besar yang menggunakan banyak kode Python di dalam mesin pencarinya. Dan mesin pencari google adalah yang paling terkenal di dunia.
2. Youtube, situs video terbesar dan terpopuler di dunia, sebagian besar kodennya ditulis dalam bahasa Python.
3. Facebook, media sosial terbesar di dunia, menggunakan Tornado, sebuah framework Python untuk menampilkan timeline.
4. Instagram, siapa yang tidak kenal. Instagram menggunakan Django, framework python sebagai mesin pengolah sisi server dari aplikasinya.
5. Pinterest, banyak menggunakan python untuk membangun aplikasinya.
6. Dropbox, barangkali Anda adalah salah seorang pengguna layanan ini. Dropbox menggunakan python baik di sisi server maupun di sisi pengguna layanannya.
7. Quora, salah satu situs tanya jawab terbesar di dunia, dibangun menggunakan Python.
8. NASA, badan antariksa Amerika ini menggunakan Python untuk bidang sainsnya.
9. NSA, badan mata – mata Amerika banyak menggunakan Python untuk analisa kriptografi dan intelijen

10. Blender, Maya, software pembuat animasi 3D terkenal, menggunakan Python sebagai salah satu bahasa skrip pemrogramannya.
11. Raspberry Pi, komputer mini yang banyak digunakan sebagai mikrokontroller, menggunakan Python sebagai bahasa utamanya.
12. ESRI, produsen terkenal pembuat software pemetaan GIS banyak menggunakan Python di produknya.

Untuk lebih lengkapnya bisa mengunjungi www.python.org/about/success/

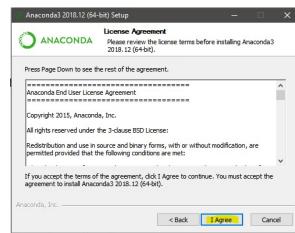
1.1.5 Cara menginstall Anaconda

1. Pastikan anda telah menginstall python dan anda mengetahui versi dari python yang telah anda install
2. Download Anaconda dari website www.anaconda.com/distribution
3. pilih sesuai dengan versi python anda, jika versi anda python3 maka pilih python3
4. Setelah itu buka file yang telah anda download
5. Setelah muncul gambar dibawah ini, tekan next



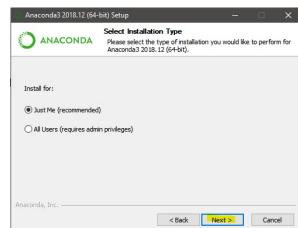
Gambar 1.1 Tampilan Instalasi 1

6. Baca license agreement lalu tekan 'I Agree'



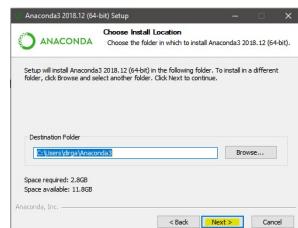
Gambar 1.2 Tampilan Instalasi 2

- Setelah itu pilih mau diinstall pada user yang sedang anda pakai atau kesemua user, direkomendasikan untuk memilih just me yaitu hanya user yang sedang dipakai saja



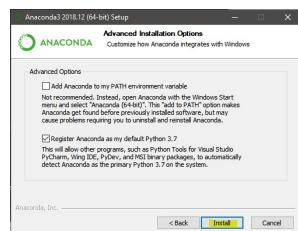
Gambar 1.3 Tampilan Instalasi 3

- Catat tempat dimana anda akan menginstall anaconda, lalu tekan 'Next'



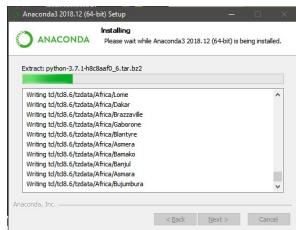
Gambar 1.4 Tampilan Instalasi 4

- Setelah itu anda diberi pilihan, direkomendasikan untuk tidak mengubah pilihan tersebut, lalu tekan 'Install'

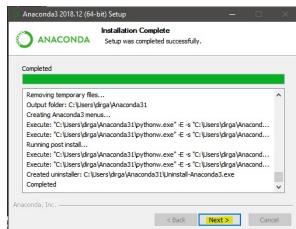


Gambar 1.5 Tampilan Instalasi 5

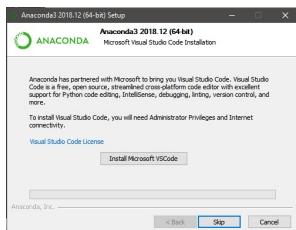
- Tunggu sampai instalasi selesai
- Setelah selesai tekan 'Next'



Gambar 1.6 Tampilan Instalasi 6

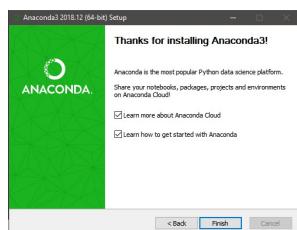


Gambar 1.7 Tampilan Instalasi 7



Gambar 1.8 Tampilan Instalasi 8

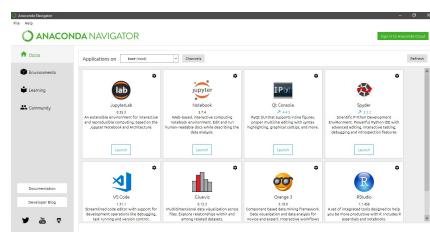
12. Setelah itu ada opsi untuk memilih untuk meinstall visual studio code, jika anda berminat klik 'Install VSCode' jika tidak tekan 'Skip'
13. Tekan 'Finish' untuk menyelesaikan instalasi



Gambar 1.9 Tampilan Instalasi 9

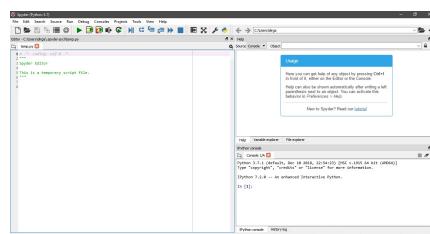
1.1.6 Cara menggunakan Spyder pada Anaconda

Pertama buka aplikasi Anaconda sampai muncul seperti ini



Gambar 1.10 Tampilan awal Anaconda

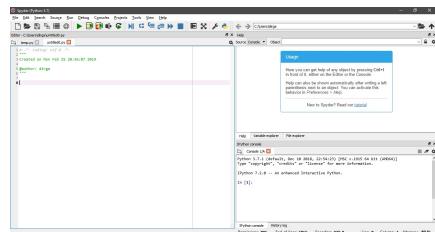
Setelah itu tekan Launch dibawah logo Spyder Tunggu sampai muncul seperti ini



Gambar 1.11 Tampilan spider

1.1.7 Membuat Hello World di Spyder

Setelah membuka spyder seperti gambar di section sebelumnya tekan menu File lalu klik New File atau bisa menggunakan kombinasi tombol Ctrl + N sampai muncul seperti ini



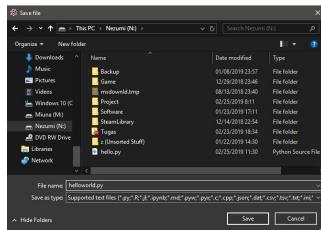
Gambar 1.12 Tampilan new file pada spider

Karena kita menggunakan Python3.7 maka kita menggunakan fungsi print() untuk memunculkan teks Hello World yang akan kita buat, tuliskan print("Hello World") pada teks editor di Spyder

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Mon Feb 25 20:36:07 2019
4
5 @author: dirga
6 """
7
8 print("Hello World")
```

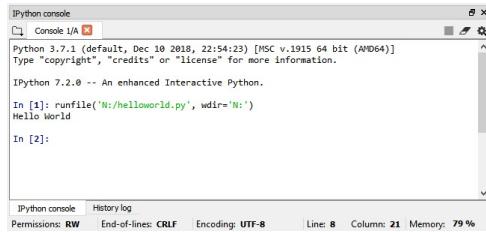
Gambar 1.13 print("Hello World")

setelah itu tekan tombol play berwarna hijau diatas, karena kita belum save file yang kita buat maka akan muncul dialog simpan file, pilih tempat dan nama file yang akan disimpan contohnya helloworld.py



Gambar 1.14 Dialog simpan file

setelah itu tekan run maka hasil dari program yang kita buat tadi ada dibagian console yang berada di pinggir kanan bawah



The screenshot shows an IPython console window. The title bar says "IPython console". The main area displays the following text:

```
Python 3.7.1 (default, Dec 10 2018, 22:54:23) [MSC v.1915 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.2.0 -- An enhanced Interactive Python.

In [1]: runfile('N:/helloworld.py', wdir='N:')
Hello World

In [2]:
```

At the bottom of the window, there are tabs for "IPython console" and "History log". Below the tabs, status information is displayed: Permissions: RW, End-of-lines: CRLF, Encoding: UTF-8, Line: 8, Column: 21, Memory: 79 %.

Gambar 1.15 Hasil Program

BAB 2

PEMOGRAMAN DASAR

2.1 D. Irga B. Naufal Fakhri

2.1.1 Teori

1. Jenis-jenis variabel pada python dan cara penggunaannya:

(a) Boolean

```
1 #Boolean  
2 varbool = True  
3 print(varbool)
```

Listing 2.1 Contoh kode variable Boolean.

(b) String

```
1 #String  
2 varstr = "D. Irga B. Naufal Fakhri"  
3 print(varstr)
```

Listing 2.2 Contoh kode variable String.

(c) Integer

```

1 #Integer
2 varint = 1174066
3 print(varint)

```

Listing 2.3 Contoh kode variable Integer.

(d) Float

```

1 #Float
2 varflt = 1.1
3 print(varflt)

```

Listing 2.4 Contoh kode variable Float.

(e) Hexadecimal

```

1 #Hexadecimal
2 varhex = 0x1
3 print(varhex)

```

Listing 2.5 Contoh kode variable Hexadecimal.

(f) Complex

```

1 #Complex
2 varcmp = 7j
3 print(varcmp)

```

Listing 2.6 Contoh kode variable Complex.

(g) List

```

1 #List
2 varli = [1,2,3]
3 print(varli)
4 print(varli[2])

```

Listing 2.7 Contoh kode variable List.

(h) Tuple

```

1 #Tuple
2 vartu = (1,2,3)
3 print(vartu)
4 print(vartu[2])

```

Listing 2.8 Contoh kode variable Tuple.

(i) Set

```

1 #Set
2 varset = {1,2,3}
3 print(varset)

```

Listing 2.9 Contoh kode variable Set.

(j) Dictionary

```

1 #Dictionary
2 vardic = {1: 'satu', 2: 'dua', 'tiga':3}
3 print(vardic)
4 print(vardic[1])

```

Listing 2.10 Contoh kode variable Dictionary.

2. Permintaan Input dari user dan Outputnya

```

1 #Input dan Output
2 nama = input('Tolong masukan nama anda: ')
3 print('Hallo, '+nama)

```

Listing 2.11 Contoh kode input dan outputnya.

3. Operator dasar aritmatika dan perubahan tipe data variable

Operator dasar aritmatika

- (a) Perjumlahan (+) Operator ini berfungsi untuk melakukan operasi perjumlahan.

```

1 #Input dan Output
2 nama = input('Tolong masukan nama anda: ')
3 print('Hallo, '+nama)
4
5 #Operator Dasar Aritmatika
6 #Perjumlahan
7 var1 = 1
8 var2 = 2
9 hasil = var1 + var2
10 print(hasil)

```

Listing 2.12 Contoh kode operasi pertambahan.

- (b) Pengurangan (-) Operator ini berfungsi untuk melakukan operasi pengurangan.

```

1 #Pengurangan
2 var1 = 4
3 var2 = 3
4 hasil = var1 - var2
5 print(hasil)

```

Listing 2.13 Contoh kode operasi pengurangan.

- (c) Perkalian (*) Operator ini dipergunakan untuk melakukan operasi perkalian.

```

1 #Perkalian
2 var1 = 5
3 var2 = 6
4 hasil = var1 * var2
5 print(hasil)

```

Listing 2.14 Contoh kode operasi perkalian.

- (d) Pembagian (/) Operator ini dipergunakan untuk melakukan operasi pembagian.

```

1 #Pembagian
2 var1 = 8
3 var2 = 2
4 hasil = var1 / var2
5 print(hasil)

```

Listing 2.15 Contoh kode operasi pembagian.

- (e) Modulus (Operator ini dipergunakan untuk melakukan operasi modulus.

```

1 #Modulus
2 var1 = 7
3 var2 = 3
4 hasil = var1 % var2
5 print(hasil)

```

Listing 2.16 Contoh kode operasi modulus.

- (f) Perpangkatan (***) Operator ini dipergunakan untuk melakukan operasi perpangkatan.

```

1 #Perpangkatan
2 var1 = 10
3 var2 = 2
4 hasil = var1 ** var2
5 print(hasil)

```

Listing 2.17 Contoh kode operasi perpangkatan.

- (g) Pembulatan Kebawah Pada Hasil Pembagian (//) Operator ini dipergunakan untuk melakukan operasi pembulatan hasil bagi.

```

1 #Pembulatan_Kebawah_Pada_Hasil_Pembagian
2 var1 = 11
3 var2 = 3
4 hasil = var1 // var2
5 print(hasil)

```

Listing 2.18 Contoh kode operasi pembulatan hasil pembagian kebawah.

Perubahan tipe data variable

- (a) String menjadi Integer

```

1 #String menjadi Integer
2 varstr = '1'
3 varint = int(varstr)
4 print(varint)

```

Listing 2.19 Contoh kode variable string menjadi integer.

- (b) Integer menjadi String

```

1 #Integer menjadi String
2 varint = 2
3 varstr = str(varint)
4 print(varstr)

```

Listing 2.20 Contoh kode variable integer menjadi string.

4. Sintak perulangan (looping), jenis-jenisnya, dan penggunaannya.

- (a) While Loop While Loop adalah perulangan yang mengeksekusi statement terus menerus selama kondisi bernilai True.

```

1 #While Loop
2 hitung = 1
3 while (hitung < 6):
4     print(hitung)
5     hitung += 1

```

Listing 2.21 Contoh kode penggunaan while loop.

- (b) For Loop For Loop adalah pengulangan berdasarkan kondisi yang telah ditentukan biasanya kondisi pertambahan seperti 1 sampai 5

```

1 #For Loop
2 angka = [1, 2, 3, 4, 5]
3 for a in angka:
4     print(a)

```

Listing 2.22 Contoh kode penggunaan for loop.

- (c) Nested Loop Nested Loop merupakan pengulangan yang ada di dalam pengulangan

```

1 #Nested Loop
2 i = 1
3 while(i < 5):
4     j = 1
5     while(j <= 5):
6         print('TEST')
7         j += 1
8     i += 1

```

Listing 2.23 Contoh kode penggunaan nested loop.

5. Sintak kondisi dan penggunaannya.

- (a) If Kondisi ini digunakan untuk mengecek apabila kondisi tersebut dipenuhi akan mengeksekusi kode didalamnya.

```

1 #Kondisi
2 #If
3 var = 1
4 if var > 0:
5     print(var, "adalah angka lebih dari 0")

```

Listing 2.24 Contoh kode penggunaan if.

- (b) If Else Kondisi ini digunakan untuk mengecek apabila kondisi tersebut dipenuhi akan mengeksekusi kode didalamnya dan didalamnya memiliki dua kondisi.

```

1 #If Else
2 var = -1
3 if var > 0:
4     print(var, "adalah angka kurang dari 0")
5 else:
6     print(var, "adalah angka lebih dari 0")

```

Listing 2.25 Contoh kode penggunaan if else.

- (c) Elif Kondisi ini digunakan untuk mengecek apabila kondisi tersebut dipenuhi akan mengeksekusi kode didalamnya dan didalamnya memiliki dua kondisi atau lebih.

```

1 #Elif
2 var = 0
3 if var > 0:
4     print(var, "adalah angka lebih dari 0")
5 elif var < 0:
6     print(var, "adalah angka kurang dari 0")
7 else:
8     print(var, "adalah angka yang sama dengan 0")

```

Listing 2.26 Contoh kode penggunaan elif.

- (d) Kondisi di dalam kondisi Kondisi ini digunakan saat kondisi memerlukan kondisi lagi didalamnya

```

1 #Kondisi di dalam kondisi
2 if var > 0:
3     print(var, "adalah angka lebih dari 0")
4     if var > 10:
5         print(var, "adalah angka lebih dari 10.")
6     else:
7         print(var, "adalah angka lebih dari 10.")
8 elif var < 0:
9     print(var, "adalah angka kurang dari 0")
10 else:
11     print(var, "adalah angka yang sama dengan 0")

```

Listing 2.27 Contoh kode penggunaan kondisi di dalam kondisi.

6. Jenis-jenis error pada python dan cara mengatasinya.

- Syntax Errors Syntax Errors adalah kesalahan pada penulisan syntax atau kode. Solusinya adalah memperbaiki penulisan syntax atau kode
- Zero Division Error ZeroDivisionError adalah exceptions yang terjadi saat eksekusi program menghasilkan perhitungan matematika pembagian dengan angka nol (0). Solusinya adalah tidak membagi suatu yang hasilnya nol.

- Name Error NameError adalah exception saat kode melakukan eksekusi terhadap local name atau global name yang tidak terdefinisi atau tidak ada. Solusinya adalah memastikan variabel atau function yang akan dipanggil ada didalam program atau tidak salah mengetikannya.
- Type Error TypeError adalah exception saat melakukan eksekusi terhadap suatu operasi atau fungsi dengan type object yang tidak sesuai. Solusinya adalah mengkoversi varibelnya sesuai dengan tipe data sesuai dengan yang akan digunakan.

7. Cara pemakaian Try Except.

```
1 #Try Except
2 try :
3     print(tidakada)
4 except:
5     print("Terjadi kesalahan penulisan kode")
6 finally:
7     print("Try except telah selesai")
```

Listing 2.28 Contoh kode penggunaan try except.

BAB 3

PRAKTEK PEMOGRAMAN DASAR

3.1 D. Irga B. Naufal Fakhri

3.1.1 Ketrampilan Pemrograman

1. Jawaban Soal 1

```
1 #Jawaban No. 1
2
3 print(1174066%3)
4 #mod3 = 1
5
6 print(" ##  ## #######  ###  #####  #####  #####")
7 print("###  ### #######  #####  ##  ##  ##")
8 print(" ##  ##  ##  ##  ##  ##  ##  ##  ##")
9 print(" ##  ##  ## #######  ##  ## #######  #####")
10 print(" ##  ##  ##  ##  ##  ##  ##  ##  ##  ##")
11 print(" ##  ##  ##  ##  ##  ##  ##  ##  ##  ##")
12 #Cara kedua Penulisan NPM tanda pagar menurut https://
13 #      stackoverflow.com/questions/9632995/how-to-easily-print-ascii
14 #      -art-text
15 text = "1174066"
16 from PIL import Image, ImageDraw, ImageFont
```

```

15 import numpy as np
16 myfont = ImageFont.truetype("verdanab.ttf", 12)
17 size = myfont.getsize(text)
18 img = Image.new("1", size, "black")
19 draw = ImageDraw.Draw(img)
20 draw.text((0, 0), text, "white", font=myfont)
21 pixels = np.array(img, dtype=np.uint8)
22 chars = np.array([' ', '#'], dtype="U1")[pixels]
23 strings = chars.view('U' + str(chars.shape[1])).flatten()
24 print("\n".join(strings))

```

2. Jawaban Soal 2

```

1 #Jawaban No. 2
2
3 npm = input("Input : ")
4 i = 1
5 while(i <= int(npm[5:])):
6     print("Hallo, "+str(npm)+" apa kabar?")
7     i += 1

```

3. Jawaban Soal 3

```

1 #Jawaban No. 3
2
3 npm = input("Input : ")
4 i = 1
5 akhir = int(npm[4:5]) + int(npm[5:6]) + int(npm[6:7])
6 while(i <= akhir):
7     print("Hallo, "+str(npm[4:7])+" apa kabar?")
8     i += 1

```

4. Jawaban Soal 4

```

1 #Jawaban No. 4
2
3 npm = input("Input : ")
4 print("Hallo, "+str(npm[-3])+" apa kabar?")

```

5. Jawaban Soal 5

```

1 #Jawaban No. 5
2
3 a = 1
4 b = 1
5 c = 7
6 d = 4
7 e = 0
8 f = 6
9 g = 6
10
11 npm = [a,b,c,d,e,f,g]
12
13 for n in npm:
14     print(n, end="")
15 print()

```

6. Jawaban Soal 6

```
1 #Jawaban No. 6
2
3 print(a+b+c+d+e+f+g)
```

7. Jawaban Soal 7

```
1 #Jawaban No. 7
2
3 print(a*b*c*d*e*f*g)
```

8. Jawaban Soal 8

```
1 #Jawaban No. 8
2
3 for n in npm:
4     print(n)
```

9. Jawaban Soal 9

```
1 #Jawaban No. 9
2
3 for n in npm:
4     if(n % 2 == 0):
5         if(n != 0):
6             print(n, end =",")
7
8 print()
```

10. Jawaban Soal 10

```
1 #Jawaban No. 10
2
3 for n in npm:
4     if(n % 2 != 0):
5         print(n, end =",")
6
7 print()
```

11. Jawaban Soal 11

```
1 #Jawaban No. 11
2
3 for n in npm:
4     for x in range(2,n):
5         if n%x != 0:
6
7             break
8         print(n, "Bilangan prima")
9         break
```

3.1.2 Ketrampilan Penanganan Error

1. Jawaban Soal No. 1

- Syntax Errors Syntax Errors adalah kesalahan pada penulisan syntax atau kode. Solusinya adalah memperbaiki penulisan syntax atau kode
- Zero Division Error ZeroDivisonError adalah exceptions yang terjadi saat eksekusi program menghasilkan perhitungan matematika pembagian dengan angka nol (0). Solusinya adalah tidak membagi suatu yang hasilnya nol.
- Name Error NameError adalah exception saat kode melakukan eksekusi terhadap local name atau global name yang tidak terdefinisi atau tidak ada. Solusinya adalah memastikan variabel atau function yang akan dipanggil ada didalam program atau tidak salah mengetikannya.
- Type Error TypeError adalah exception saat melakukan eksekusi terhadap suatu operasi atau fungsi dengan type object yang tidak sesuai. Solusinya adalah mengkoversi varibelnya sesuai dengan tipe data sesuai dengan yang akan digunakan.

2. Jawaban Soal No. 2

```
1 a = "1"
2 b = 1
3
4 try :
5     print(a+b)
6 except TypeError:
7     print("Berbeda type datanya sehingga tidak bisa dijumlahkan")
```

BAB 4

FUNGSI DAN KELAS

4.1 D. Irga B. Naufal Fakhri

4.1.1 Pemahaman Teori

1. Fungsi

Fungsi adalah blok blok kode yang terorganisir yang dapat digunakan kembali didalam program yang digunakan untuk melakukan suatu perintah yang telah diberikan. Untuk membuat fungsi kita harus menggunakan def kemudian nama fungsinya dan (variable)nya diakhiri oleh tanda :

```
1 #Contoh fungsi inputan
2 def printteks( str ):
3     print (str)
4     return
5
6 printteks("Ini adalah Contoh fungsi")
```

Listing 4.1 Contoh kode fungsi inputan ke fungsi.

Fungsi juga berguna untuk melemparkan variable contohnya

```

1 #Contoh fungsi outputan
2 def output( str ):
3     full = "Ini adalah contoh dari " + str
4     return full
5
6 print(output("fungsi outputan"))

```

Listing 4.2 Contoh kode fungsi outputan ke fungsi.

2. Paket(Package) atau Library

Paket atau yang biasa disebut dengan library adalah kumpulan kode-kode fungsi atau method pada python yang dapat dipanggil kedalam program python yang kita buat. Package berada di file terpisah dari main program cara memanggil package: Pastikan file package ada didalam folder yang sama lalu ditambah import dengan nama filenya tanpa extensi (.py)

```

1 #Contoh import library atau package
2 import fungsi_11174066
3
4 input(fungsi_11174066.hello("Dirga"))

```

Listing 4.3 Contoh import package atau library.

3. Kelas (Class), Objek (Object), Atribut (Attribute), dan Method

Kelas(Class) adalah sebuah blueprint(cetakan) dari sebuah objek. Objek(Object) adalah hasil cetakan dari sebuah kelas(class). Atribut(Attribute) adalah nilai data yang ada didalam sebuah object. Method adalah sesuatu yang bisa dilakukan oleh object.

```

1 #Contoh Class
2 class Mahasiswa:
3     def __init__(self ,nama ,npm ,kelas):
4         self.nama = nama
5         self.npm = npm
6         self.kelas = kelas
7
8     def datadiri(self):
9         print ("Nama: ", self.nama, " , NPM: ", self.npm, " ,
10        Kelas: ", self.kelas)
11
12 #Pembuatan Object
13 mhs1 = Mahasiswa("D. Irga",11174066,"D4TI2C")
14 mhs1.datadiri()

```

Listing 4.4 Contoh import package atau library.

4. Cara memanggil library dari instansiasi

Cara memanggilnya:

- Pertama kita import filenya
- kemudian buat variablenya jika menggunakan variable untuk menampung data

- Kemudian panggil nama classnya(file) dan panggil fungsinya
- Kemudian menggunakan perintah print untuk menampilkan data

```

1 #Contoh package nomor 4
2 def Tambah(x,y):
3     z = x + y
4     return z

```

Listing 4.5 Contoh package atau library.

```

1 #Contoh cara memanggil library atau package
2 import fungsi_1174066
3
4 x = 3
5 y = 7
6 print(fungsi_1174066.Tambah(x,y))

```

Listing 4.6 Contoh import package atau library.

5. Contoh pemakaian paket dengan perintah from kalkulator import Penambahan
Pemakaian package(paket) dengan perintah from namafilenya import berfungsi untuk memanggil fungsi dari nama filenya

```

1 #Contoh cara memanggil library atau package dari from import
2 from fungsi_1174066 import Tambah
3
4 x = 3
5 y = 7
6 print(Tambah(x,y))

```

Listing 4.7 Contoh import package atau library.

6. Jelaskan dengan contoh kode, pemakaian paket fungsi didalam folder
Jika file paket ada didalam folder maka kita harus menambahkan lokasi filenya ada didalam folder apa dengan cara menggunakan namafolder.namafile

```

1 #Contoh cara memanggil library atau package dari folder
2 from folder_1174066.fungsi_1174066 import Tambah
3
4 x = 3
5 y = 7
6 print(Tambah(x,y))

```

Listing 4.8 Contoh import package atau library didalam folder.

7. Jelaskan dengan contoh kode, pemakaian paket fungsi didalam folder
Jika file paket ada didalam folder maka kita harus menambahkan lokasi filenya ada didalam folder apa dengan cara menggunakan namafolder.namafile

```

1 #Contoh cara memanggil library atau package dari folder
2 from folder_1174066.fungsi_1174066 import Tambah
3
4 x = 3

```

```
5 y = 7  
6 print(Tambah(x,y))
```

Listing 4.9 Contoh import package atau library didalam folder.

BAB 5

PRAKTEK FUNGSI DAN KELAS

5.1 D. Irga B. Naufal Fakhri

5.1.1 Keterampilan Pemrograman

1. Jawaban nomor 1

```
1 #Jawaban nomor 1
2 def npm(npm):
3     text = str(npm)
4     from PIL import Image, ImageDraw, ImageFont
5     import numpy as np
6     myfont = ImageFont.truetype("verdanab.ttf", 12)
7     size = myfont.getsize(text)
8     img = Image.new("1",size,"black")
9     draw = ImageDraw.Draw(img)
10    draw.text((0, 0), text, "white", font=myfont)
11    pixels = np.array(img, dtype=np.uint8)
12    if int(text) % 3 == 0:
13        chars = np.array([' ', '*', '#'], dtype="U1")[pixels]
14    elif int(text) % 3 == 1:
15        chars = np.array([' ', '#', '*'], dtype="U1")[pixels]
16    elif int(text) % 3 == 2:
```

```

17     chars = np.array([' ', '+'], dtype="UI")[pixels]
18     strings = chars.view('U' + str(chars.shape[1])).flatten()
19     print("\n".join(strings))

```

2. Jawaban nomor 2

```

1 #Jawaban nomor 2
2 def loop(npm):
3     i = 1
4     while(i <= int(npm[5:])):
5         print("Hallo, "+str(npm)+" apa kabar?")
6         i += 1
7
8 loop(input("Masukan NPM : "))

```

3. Jawaban nomor 3

```

1 #Jawaban nomor 3
2 def looptambah(npm):
3     i = 1
4     akhir = int(npm[4:5]) + int(npm[5:6]) + int(npm[6:7])
5     while(i <= akhir):
6         print("Hallo, "+str(npm[4:7])+" apa kabar?")
7         i += 1
8
9 looptambah(input("Input: "))

```

4. Jawaban nomor 4

```

1 #Jawaban nomor 4
2 def hallo(npm):
3     print("Hallo, "+str(npm[-3:])+" apa kabar?")
4
5 hallo(input("Input: "))

```

5. Jawaban nomor 5

```

1 #Jawaban nomor 5
2 def array(npm):
3     for x in npm:
4         print(x)

```

6. Jawaban nomor 6

```

1 #Jawaban nomor 6
2 def jumlah(npm):
3     npm = list(map(int, npm))
4     hasil = 0
5     for x in npm:
6         hasil += x
7     print("Hasil perjumlahan dari npm anda adalah: " + str(hasil))
8
9 jumlah(input("NPM: "))

```

7. Jawaban nomor 7

```

1 #Jawaban nomor 7
2 def kali(npm):
3     npm = list(map(int, npm))
4     hasil = 0
5     for x in npm:
6         hasil *= x
7     print("Hasil perkalian dari npm anda adalah: " + str(hasil))
8
9 kali(input("NPM: "))

```

8. Jawaban nomor 8

```

1 #Jawaban nomor 8
2 def printgenap(npm):
3     npm = list(map(int, npm))
4     for x in npm:
5         if(x % 2 == 0):
6             if(x != 0):
7                 print(x, end = " ")
8
9 printgenap(input("NPM: "))

```

9. Jawaban nomor 9

```

1 #Jawaban nomor 9
2 def printganjil(npm):
3     npm = list(map(int, npm))
4     for x in npm:
5         if(x % 2 != 0):
6             print(x, end = " ")
7
8 printganjil(input("NPM: "))

```

10. Jawaban nomor 10

```

1 #Jawaban nomor 10
2 def printprima(npm):
3     npm = list(map(int, npm))
4     prima = []
5     for n in npm:
6         bilPrima = True
7         if n == 0 or n == 1:
8             bilPrima = False
9         for x in range(2, n):
10            if n % x == 0:
11                bilPrima = False
12        if bilPrima:
13            prima.append(n)
14
15    for p in prima:
16        print(p, end = " ")
17
18 printprima(input("NPM: "))

```

11. Jawaban nomor 11

```

1 #Pemanggilan library 3lib
2 from lib_1174066 import *
3
4 npm(1174066)
5 loop(input("Masukan NPM : "))
6 looptambah(input("Input: "))
7 hallo(input("Input: "))
8 array(input("NPM: "))
9 jumlah(input("NPM: "))
10 kali(input("NPM: "))
11 printgenap(input("NPM: "))
12 printganjil(input("NPM: "))
13 printprima(input("NPM: "))

```

12. Jawaban nomor 12

```

1 #Pemanggilan kelas 3lib
2 from kelas3lib_1174066 import kelas3lib
3 npm = "1174066"
4 kelas = kelas3lib(npm)
5
6 kelas.npmprint()
7 kelas.loop()
8 kelas.looptambah()
9 kelas.hallo()
10 kelas.array()
11 kelas.jumlah()
12 kelas.kali()
13 kelas.printgenap()
14 kelas.printganjil()
15 kelas.printprima()

```

5.1.2 Ketrampilan Penanganan Error

- Syntax Errors

Syntax Errors adalah kesalahan pada penulisan syntax atau kode. Solusinya adalah memperbaiki penulisan syntax atau kode

- Zero Division Error

ZeroDivisionError adalah exceptions yang terjadi saat eksekusi program menghasilkan perhitungan matematika pembagian dengan angka nol (0). Solusinya adalah tidak membagi suatu yang hasilnya nol.

- Name Error

NameError adalah exception saat kode melakukan eksekusi terhadap local name atau global name yang tidak terdefinisi atau tidak ada. Solusinya adalah memastikan variabel atau function yang akan dipanggil ada didalam program atau tidak salah mengetikannya.

- Type Error

TypeError adalah exception saat melakukan eksekusi terhadap suatu operasi atau fungsi dengan type object yang tidak sesuai. Solusinya adalah mengkoversi varibelnya sesuai dengan tipe data sesuai dengan yang akan digunakan.

```
1 #Pemanggilan kelas 3lib
2 from kelas3lib_1174066 import kelas3lib
3 npm = "1174066"
4 kelas = kelas3lib(npm)
5
6 kelas.npmprint()
7 kelas.loop()
8 kelas.looptambah()
9 kelas.hallo()
10 kelas.array()
11 kelas.jumlah()
12 kelas.kali()
13 kelas.printgenap()
14 kelas.printganjil()
15 kelas.printprima()
```


BAB 6

LIBRARY CSV DAN PANDAS

6.1 D. Irga B. Naufal Fakhri

6.1.1 Pemahaman Teori

1. CSV

CSV (Comma Separated Values file) adalah sebuah tipe file text biasa yang memiliki penataan khusus yang biasanya berfungsi untuk mengelola data. sesuai dengan namanya file csv memisahkan setiap data menggunakan koma (,).

Format data CSV pertama kali digunakan pada tahun 1978 pada complier FORTRAN 77, kemudian nama CSV baru muncul dan mulai digunakan pada tahun 1983

Contoh data pada csv:

```
1 npm, nama , kelas , tanggal lahir  
2 11714066,D. Irga B. Naufal Fakhri ,D4 Teknik Informatika 2C  
   ,11/07/98
```

2. Aplikasi yang bisa menciptakan CSV

Semua aplikasi teks editor seperti notepad++, vscode, sublime ataupun notepad dapat menciptakan CSV termasuk aplikasi spreadsheet seperti Microsoft Excel, Libre Office

3. Jelaskan bagaimana cara menulis dan membaca file csv di Excel atau spreadsheet

- Buka Microsoft Excel 2019-nya lalu buat dokumen baru
- Isikan data sesuai dengan kebutuhan, yang paling atas akan menjadi header dari file csv
- Setelah memasukkan data, klik file lalu klik Save As
- Pilih Browse dan pilih tempat menyimpannya akan dimana
- Masukkan nama file pada File Name
- Lalu pada Save As Type pilih CSV (comma delimited) (*.csv)
- Maka hasil file akan seperti ini

```

1 npm , nama , kelas , tanggal lahir
2 11714066,D. Irga B. Naufal Fakhri ,D4 Teknik Informatika 2C
   ,11/07/98

```

4. Jelaskan sejarah library csv

Module csv mengimplementasikan kelas untuk membaca dan menulis data kedalam format CSV. Hal ini memungkinkan programmer untuk "tulis data ini dalam format yang disukai oleh Excel," atau "baca data dari file yang dihasilkan oleh Excel," tanpa mengetahui detail yang tepat dari format CSV yang digunakan oleh Excel. Pemrogram juga dapat menggambarkan format CSV yang dipahami oleh aplikasi lain atau menentukan format CSV tujuan khusus untuk mereka sendiri.

5. Jelaskan sejarah library pandas

pandas adalah sebuah library open source dan berlisensi BSD yang menyediakan performa yang tinggi, mudah digunakan struktur data dan data analisis untuk python.

6. Jelaskan fungsi-fungsi yang terdapat di library csv

- csv.reader

Berfungsi untuk membaca dan mengembalikan data kedalam variable dari file csv. Fungsi reader dirancang untuk mengambil data pada setiap baris didalam file dan membuat daftar semua kolom. Kemudian, tinggal dipilih kolom mana yang diinginkan untuk data variabel.

```

1 def read(namofile):
2     with open(namofile) as csv_file:
3         csv_reader = csv.reader(csv_file, delimiter=',')
4         line_count = 0
5         for row in csv_reader:

```

```

6     if line_count == 0:
7         print(f'Column names are {", ".join(row)}')
8         line_count += 1
9     else:
10        print(f'\tNomor NPM: {row[0]} Nama: {row[1]}
Kelas: {row[2]}.')
11        line_count += 1
12        print(f'Processed {line_count} lines.')
13
14 read('1174066.csv')

```

- **csv.writer**

Berfungsi untuk menuliskan data dari variable kedalam file csv. Fungsi writer akan membuat objek yang cocok untuk menulis. Untuk mengulang data yang ada di atas baris, gunakan fungsi writerow.

```

1 def write(namofile):
2     with open(namofile, mode='w') as employee_file:
3         employee_writer = csv.writer(employee_file, delimiter=
4             ', ', quotechar='"', quoting=csv.QUOTE_MINIMAL)
5
6         employee_writer.writerow(['npm', 'nama', 'kelas'])
7         employee_writer.writerow(['1174066', 'D. Irga 3', 'D4
Teknik Informatika 2C'])
7
8 write('1174066_baru.csv')

```

- **csv.register_dialect**

Mendaftarkan dialect pada csv

- **csv.unregister_dialect**

Menghapus dialect yang diasosiasi dengan nama dari registry dialect

- **csv.list_dialects**

Mengembalikan dialect yang diasosiasi dengan nama

- **csv.field_size_limit**

Mengembalikan ukuran field maksimum yang diizinkan oleh parser.

- **csv.DictReader**

Berfungsi untuk membaca dan mengembalikan data kedalam variable dictionary dari file csv.

```

1 def readdict(namofile):
2     with open(namofile, mode='r') as csv_file:
3         csv_reader = csv.DictReader(csv_file)
4         line_count = 0
5         for row in csv_reader:
6             if line_count == 0:
7                 print(f'Column names are {", ".join(row)}')
8                 line_count += 1
9                 print(f'\tNomor NPM: {row["npm"]} Nama: {row["nama"]}
']) Kelas: {row["kelas"]}.')
10                line_count += 1
11                print(f'Processed {line_count} lines.')

```

```

12
13 readdict('1174066.csv')

```

7. Jelaskan fungsi-fungsi yang terdapat di library pandas

- `pandas.read_csv`

Berfungsi untuk membaca dan mengembalikan data kedalam format DataFrame dari file csv.

```

1 def readpanda(namafile):
2     df = pandas.read_csv(namafile)
3     print(df)
4
5 readpanda('1174066.csv')

```

- `to_dict`

Berfungsi untuk membaca dan mengembalikan data kedalam format dictionary dari file csv.

```

1 def readpandadict(namafile):
2     df = pandas.read_csv(namafile).to_dict()
3     print(df)
4
5 readpandadict('1174066.csv')

```

- `to_csv`

Berfungsi untuk mengedit data didalam csv dan menulisnya kedalam file csv

```

1 def writepanda(namafile):
2     df = pandas.read_csv(namafile,
3                           index_col='npm',
4                           parse_dates=['tanggallahir'],
5                           header=0,
6                           names=['npm', 'nama', 'kelas', 'tanggallahir'])
7     df.to_csv('1174066_edit.csv')
8 writepanda('1174066.csv')

```

6.2 Fanny Shafira Damayanti | 1174069

6.2.1 Pemahaman Teori

1. Fungsi CSV, sejarah dan contoh

CSV (comma separated value) digunakan pada tahun 1983 merupakan suatu tipe file yang digunakan untuk pengolahan informasi yang dihasilkan spreadsheet yang di proses melalui mesin analitik. CSV juga digunakan sebagai file yang agnostik karena bisa digunakan oleh berbagai database untuk backup data.

Contoh :

```
1 import csv
2
3 siswa = [
4     ('fanny', 'A', 90),
5     ('shafira', 'B', 85),
6     ('dama', 'A', 80),
7     ('yanti', 'B', 90),
8
9 ]
10
11 # tentukan lokasi file , nama file , dan inisialisasi csv
12 f = open('siswa.csv', 'w')
13 w = csv.writer(f)
14 w.writerow(['Nama', 'Kelas', 'Nilai'])
15
16 # menulis file csv
17 for s in siswa:
18     w.writerow(s)
19
20 # menutup file csv
21 f.close()
```

2. Aplikasi untuk membuat file CSV

- Microsoft Excel
- Spyder
- Apple Number
- LibreOffice
- Apple Office Calc
- Apache Open Office Calc

3. Cara menulis dan membaca file .csv di Excel atau Spreadsheet

Cara menulis :

Buka Ms. Excel, lalu buat file nya, ketika akan di save ganti jenis filenya menjadi .csv

- Download Template CSV.
- Buka file spreadsheet Anda di Excel.
- Buat dokumen baru di Excel.
- Tambahkan judul kolom, lalu ketikkan informasi dalam kolom tersebut
- Klik File , dan pilih Save As
- Masukkan nama file, lalu pilih CSV (Comma delimited) (* csv) dari dropdown Save as type .

Cara membaca :

- Buka Microsoft Excel.
- Mulai / buka spreadsheet
- Pilih tab Data
- Pilih opsi Dari Teks. (Jika opsi berwarna abu-abu, Anda mungkin perlu membuka spreadsheet / workbook baru).
- Temukan dan pilih file .csv yang telah Anda unduh dari Kotive. Klik pada file dan kemudian klik Impor.
- Panduan impor Teks akan terbuka. Pastikan opsi Dibatasi dipilih. Klik tombol Berikutnya.
- Pilih Koma di bawah Pembatas. Kualifikasi Teks harus menunjukkan “(tanda kutip ganda). Klik tombol Selesai. Anda mungkin ditanya Di mana Anda ingin meletakkan data? Klik pada sel kiri atas. Klik tombol OK.
- Excel menampilkan data di buku kerja Anda

4. Sejarah Libary CSV Inisiatif standardisasi utama - mentransformasikan "definisi fuzzy de facto" menjadi definisi yang lebih tepat dan de jure - adalah pada tahun 2005, dengan RFC4180, mendefinisikan CSV sebagai Tipe Konten MIME. Kemudian, pada 2013, beberapa kekurangan RFC4180 ditangani oleh rekomendasi W3C.

Pada 2014 IETF menerbitkan RFC7111 yang menjelaskan aplikasi fragmen URI pada dokumen CSV. RFC7111 menentukan bagaimana rentang baris, kolom, dan sel dapat dipilih dari dokumen CSV menggunakan indeks posisi.

Pada 2015 W3C, dalam upaya untuk meningkatkan CSV dengan semantik formal, mempublikasikan draft rekomendasi pertama untuk standar metadata CSV, yang dimulai sebagai rekomendasi pada bulan Desember tahun yang sama.

5. Sejarah Libarary Pandas

Pandas muncul ketika ada bahasa pemrograman R dan Matlab. Pengembang Wes McKinney mulai mengerjakan pandas pada 2008 ketika di AQR Capital Management karena kebutuhan akan alat kinerja tinggi yang fleksibel untuk melakukan analisis kuantitatif pada data keuangan. Sebelum meninggalkan AQR, dia bisa meyakinkan manajemen untuk mengizinkannya membuka sumber perpustakaan.

Pegawai AQR lainnya, Chang She, bergabung dengan upaya ini pada 2012 sebagai kontributor utama kedua ke perpustakaan.

Pada 2015, panda menandatangani sebagai proyek NumFOCUS yang disponsori secara fiskal, sebuah badan amal nirlaba 501 (c) (3) di Amerika Serikat.

6. Fungsi yang terdapat pada Library CSV

- csv.reader berfungsi untuk membaca modul csv.

- csv.writer berfungsi untuk menulis modul csv.
- csv.writerows berfungsi untuk menambahkan baris baru.

7. Fungsi yang terdapat pada Library Pandas

Dengan panda kita dapat dengan mudah merubah data (CSV, excel, JSON atau SQL) menjadi sebuah object data yang terdiri dari baris dan kolom yang disebut dengan DataFrame. Fitur :

- DataFrame Object untuk manipulasi data dengan pengindeksan terintegrasi.
- Alat untuk membaca dan menulis data antara struktur data dalam memori dan berbagai format file.
- Penyelarasan data dan penanganan terpadu pada kehilangan data.
- Membentuk kembali dan memutar set data.
- Seleksi berbasis label, pengindeksan fantastis, dan melakukan subset kumpulan data besar.
- Penyisipan dan penghapusan kolom struktur data.
- Memungkinkan operasi split-apply-combine pada Data set.
- Menghubungkan dan menggabungkan Data set.
- Pengindeksan hierarki untuk bekerja dengan data dimensi tinggi dalam struktur data dimensi rendah.
- Fungsionalitas seri waktu: Pembuatan rentang tanggal dan konversi frekuensi.
- Menyediakan penyaringan data (sorting and filtering).

6.3 Aulyardha Anindita | 1174054

6.3.1 Pemahaman Teori

1. Fungsi File CSV, Sejarah, dan Contoh

CSV (Comma Separated Value) merupakan salah satu tipe file yang sering digunakan secara luas didalam dunia programming. CSV berfungsi dalam pengolahan informasi yang dihasilkan spreadsheet untuk diproses lebih lanjut menggunakan mesin analitik. CSV juga digunakan oleh berbagai database untuk proses backup data.

Pada tahun 1972 CSV memberikan format data berupa tanggal lebih awal pada komputer pribadi lebih dari satu dekade : kompiler IBM Fortran dibawah OS /360 yang kemudian disetujui pada tahun 1978. CSV pertama kali digunakan pada tahun 1983. CSV lebih mudah untuk diketik daripada data yang selaras dengan kolom tetap dan cenderung menghasilkan hasil yang salah jika suatu nilai ditinjau satu kolom dari lokasi yang dituju. pada tahun 2014 IETF menerbitkan RFC7111 yang menjelaskan fragmen URI pada dokumen CSV. RFC7111

menentukan bagaimana rentang baris, kolom, dan sel dapat dipilih dari dokumen CSV menggunakan indeks posisi. dan pada tahun 2015 W3C dalam upaya meningkatkan CSV dengan semantik formal mempublikasikan draft rekomendasi pertama untuk standar metadata CSV yang dimulai sebagai rekomendasi pada bulan desember di tahun yang sama.

Contoh CSV :

```
1 import csv
2
3 siswa = [
4     ('Aulya', 'A', 90),
5     ('Ardha', 'B', 85),
6     ('Dita', 'A', 80),
7     ('Anin', 'B', 90),
8     ('Dilan', 'C', 70)
9 ]
10
11 # tentukan lokasi file , nama file , dan inisialisasi csv
12 f = open('siswa.csv', 'w')
13 w = csv.writer(f)
14 w.writerow(('Nama', 'Kelas', 'Nilai'))
15
16 # menulis file csv
17 for s in siswa:
18     w.writerow(s)
19
20 # menutup file csv
21 f.close()
```

2. Aplikasi Yang Menciptakan File CSV

- a. Microsoft Excel
- b. Python (Spyder)
- c. Apple Numbers
- d. LibrareOffice
- e. Apple Office Calc
- f. Apache Open Office Calc.

3. Cara Menulis dan Membaca File CSV di Excel

a. Cara Menulis File CSV

- Downloadlah terlebih dahulu Template CSV.
- Kemudian Buka file spreadsheet Anda di Excel.
- Selanjutnya, Buat dokumen baru di Excel.
- Tambahkan judul kolom, lalu ketikkan informasi dalam kolom tersebut
- Klik File lalu pilih Save As
- Masukkan nama file, lalu pilih CSV (Comma delimited) (* csv) dari dropdown Save as type .

b. Cara Membaca File CSV

- Pertama, buka microsoft excelnnya
- Kemudian buka spreadsheet
- Selanjutnya, pilihlah tab Data
- Kemudian pilih opsi dari teks. (Jika opsi berwarna abu-abu, Anda mungkin perlu membuka spreadsheet / workbook baru).
- Cari dan pilih file .csv yang telah Anda unduh dari Kotive. Klik pada file dan kemudian klik Impor. Panduan impor Teks akan terbuka. Pastikan opsi Dibatasi dipilih.
- Klik tombol Berikutnya. Pilih Koma di bawah Pembatas. Kualifikasi Teks harus menunjukkan “(tanda kutip ganda).
- Klik tombol Selesai. Anda mungkin ditanya Di mana Anda ingin meletakkan data? Klik pada sel kiri atas. Klik tombol OK.
- Excel menampilkan data di buku kerja Anda

4. Sejarah Library CSV

Inisiatif standardisasi utama - mentransformasikan ”definisi fuzzy de facto” menjadi definisi yang lebih tepat dan de jure - adalah pada tahun 2005, dengan RFC4180, mendefinisikan CSV sebagai Tipe Konten MIME. Kemudian, pada 2013, beberapa kekurangan RFC4180 ditangani oleh rekomendasi W3C.

Pada 2014 IETF menerbitkan RFC7111 yang menjelaskan aplikasi fragmen URI pada dokumen CSV. RFC7111 menentukan bagaimana rentang baris, kolom, dan sel dapat dipilih dari dokumen CSV menggunakan indeks posisi.

Pada 2015 W3C, dalam upaya untuk meningkatkan CSV dengan semantik formal, mempublikasikan draft rekomendasi pertama untuk standar metadata CSV, yang dimulai sebagai rekomendasi pada bulan Desember tahun yang sama.

5. Sejarah Library Pandas

Library pandas pertama kali muncul ketika ada R dan Matlab. R dan Matlab merupakan bahasa pemrograman yang berfokus pada data yang besar.

Pengembang Library Pandas adalah Wes McKinney. Dia mulai mengerjakan pandas pada tahun 2008 ketika di AQR Capital Management karena kebutuhan akan alat kinerja tinggi yang fleksibel untuk melakukan suatu analisis kuantitatif pada data keuangan. Pegawai AQR lainnya adalah Chang She bergabung pada tahun 2012 sebagai kontributor utama kedua ke perpustakaan. Dan pada tahun 2015 pandas menandatangani sebagai proyek NumFOCUS yang disponsori secara fiskal, yang merupakan badan amal nirlaba 501 (c) (3) di Amerika Serikat.

6. Fungsi-fungsi Yang Terdapat di Library CSV

- reader() berfungsi untuk membaca data oleh module CSV

- write() berfungsi untuk menulis isi data
- writerows() berfungsi untuk menambahkan baris baru pada file

7. Fungsi-fungsi Yang Terdapat di Library Pandas

- DataFrame, Object untuk manipulasi data dengan pengindeksan terintegrasi.
- Alat untuk membaca dan menulis data antara struktur data dalam memori dan berbagai format file.
- Penyelarasan data dan penanganan terpadu pada kehilangan data.
- Membentuk kembali dan memutar set data.
- Seleksi berbasis label, pengindeksan fantastis, dan melakukan subset kumpulan data besar.
- Penyisipan dan penghapusan kolom struktur data.
- Memungkinkan operasi split-apply-combine pada Data set.
- Menghubungkan dan menggabungkan Data set.
- Pengindeksan hierarki untuk bekerja dengan data dimensi tinggi dalam struktur data dimensi rendah.
- Fungsionalitas seri waktu: Pembuatan rentang tanggal dan konversi frekuensi.
- Menyediakan penyaringan data (sorting dan filtering).

6.4 Nurul Izza Hamka | 1174062

6.4.1 Pemahaman Teori

1. Sejarah CSV, Fungsi File CSV , dan Contoh :

Sejarah : CSV (Comma Separated Values) adalah nilai yang dipisahkan oleh tanda koma, CSV ini mulai digunakan pada tahun 1983. Pada tahun 2005 dengan RCF4180, mendefinisikan CSV sebagai konten MIME. Kemudian pada tahun 2013 ada beberapa kekurangan pada RCF4180 dan ini berhasil ditrangkan oleh W3C. Kemudian pada tahun 2004 IETF menerbitkan RCF7111, kemudian pada tahun berikutnya yaitu 2015 W3C untuk meningkatkan CSV mempublikasikan drafts of recommendations iuntuk CSV.

Fungsi : File CSV dapat dibuat atau diedit di Excel. File CSV ini menyimpan Informasi yang dipisahkan oleh tanda koma. Jika data yang kita simpan dalam bentuk CSV akan sangat mudah untuk memindahkan dari satu program ke program lainnya.

```
1 #Contoh No 1
2
3 import csv
4
```

```

5 Mahasiswa = [
6     ('Izzah', 'C', 'TI', 2017),
7     ('Fanny', 'C', 'TI', 2017),
8     ('Dita', 'C', 'TI', 2017),
9     ('Nadia', 'A', 'MB', 2017),
10    ('Sri', 'B', 'LB', 2017)
11 ]
12
13 # tentukan lokasi file , nama file , dan inisialisasi csv
14 f = open('Mahasiswa.csv', 'w')
15 w = csv.writer(f)
16 w.writerow(['Nama', 'Kelas', 'Jurusan', 'Angkatan'])
17
18 # menulis file csv
19 for s in Mahasiswa:
20     w.writerow(s)
21
22 # menutup file csv
23 f.close()

```

2. Aplikasi yang bisa menciptakan file CSV

Format file CSV bisa dibuat di Microsoft Excel, Apple Numbers, LibreOffice, dan Apple Office Calc, dan Apache Open Office Calc.

3. Cara menulis dan membaca file CSV di Excel atau Spreadsheet Cara Menulis CSV di Excel :

- Download Template CSV,
- Buka file spreadsheet Anda di Excel,
- Buat dokumen baru di Excel,
- Tambahkan judul kolom, lalu ketikkan informasi dalam kolom tersebut,
- Klik File , dan pilih Save As ,
- Masukkan nama file, lalu pilih CSV (Comma delimited) (* csv) dari drop-down Save as type.

Cara Menimport CSV di Excel :

- Mulai / buka spreadsheet ,

- Pilih tab Data,
 - Pilih opsi Dari Teks. (Jika opsi berwarna abu-abu, Anda mungkin perlu membuka spreadsheet / workbook baru),
 - Temukan dan pilih file .csv yang telah Anda unduh dari Kotive.
 - Klik pada file dan kemudian klik Impor,
 - Panduan impor Teks akan terbuka. Pastikan opsi Dibatasi dipilih.
 - Klik tombol Berikutnya,
 - Pilih Koma di bawah Pembatas. Kualifikasi Teks harus menunjukkan “(tanda kutip ganda). Klik tombol Selesai,
 - Anda mungkin ditanya Di mana Anda ingin meletakkan data? Klik pada sel kiri atas. Klik tombol OK,
 - Excel menampilkan data di buku kerja Anda.
4. Sejarah Library CSV Tahun 2005, dengan RFC4180, mendefinisikan CSV sebagai Tipe Konten MIME. Kemudian, pada 2013, beberapa kekurangan RFC4180 ditangani oleh rekomendasi W3C. Kemudian Pada 2014 IETF menerbitkan RFC7111 yang menjelaskan aplikasi fragmen URI pada dokumen CSV. Dan Pada 2015 W3C, dalam upaya untuk meningkatkan CSV dengan semantik formal, mempublikasikan draft rekomendasi pertama untuk standar metadata CSV, yang dimulai sebagai rekomendasi pada bulan Desember tahun yang sama.
5. Sejarah Library Pandas
- Pandas di Tulis oleh Wes McKinney, dan Pandas ini pertama kali di luncurkan pada tanggal 11 Januari 2008. Ditulis dengan Python, Cython, C. Sistem operasi yang digunakan adalah Lintas-platform.
6. Fungsi Dalam Library CSV
- Ada beberapa Fungsi dari library CSV yaitu Perpustakaan csv berisi objek dan kode yang menyediakan fungsionalitas untuk membaca dan menulis ke file CSV. File CSV ini dirancang dan dihasilkan Excel.
7. Fungsi Dalam Library Pandas Fungsi dari File pandas ini kita dapat mengelolah suatu data dan bentuk pengelolaannya seperti join, distinct, group by, agregasi. Pandas ini juga dapat membaca file dari berbagai format seperti CSV. Pandas adalah open source dari python yang menyediakan analisis data dan struktur yang sangat mudah digunakan, dan pandas ini tersedia untuk semua instalasi Python. Dengan panda kita dapat dengan mudah merubah data (CSV, excel, JSON atau SQL) menjadi sebuah object data yang terdiri dari baris dan kolom yang disebut dengan DataFrame.

6.5 Chandra Kirana Poetra| 1174079

6.5.1 Teori

1. Apa itu fungsi file csv, jelaskan sejarah dan contoh?

CSV (Comma separated Values) adalah format data yang telah ada sebelum personal komputer hadir 10 tahun sebelumnya, CSV file pada tahun 1972 telah didukung oleh IBM Fortran, nama CSV sendiri mulai digunakan pada tahun 1983, menggunakan koma dan spasi sebagai pembatas, jadi karakter string yang tidak dikutip tidak bisa mengandung koma atau spasi, list CSV lebih mudah untuk diketik daripada data yang diatur berdasarkan kolumn, dan lebih terhindar dari hasil yang salah jika di cetak pada media seperti punched card.

1 NO;Nama;Kelas
2 1;Chandra Kirana P;D4TI2C
3 2;Bakti Qilan Mufid;D4TI2C
4 3;Gani ;D4TI2C

2. Aplikasi-Aplikasi apa saja yang bisa menciptakan file csv?

- Notepad
- OpenOffice Calc
- Excel
- Google docs

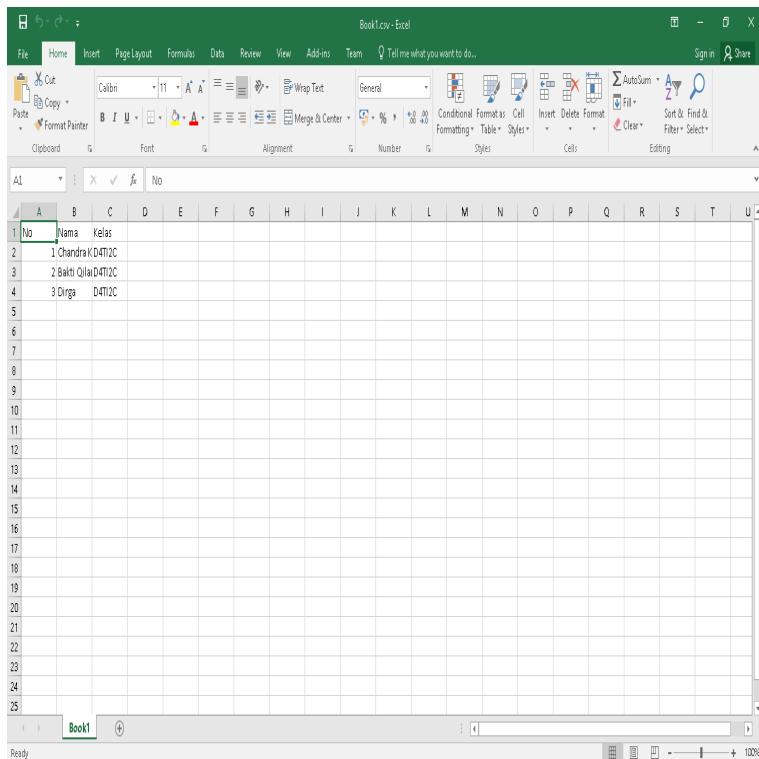
CSV file adalah sebuah file berisi text jadi CSV file bisa dibuat oleh text editor apapun.

3. Jelaskan bagaimana cara menulis dan membaca file csv di excel atau spreadsheet

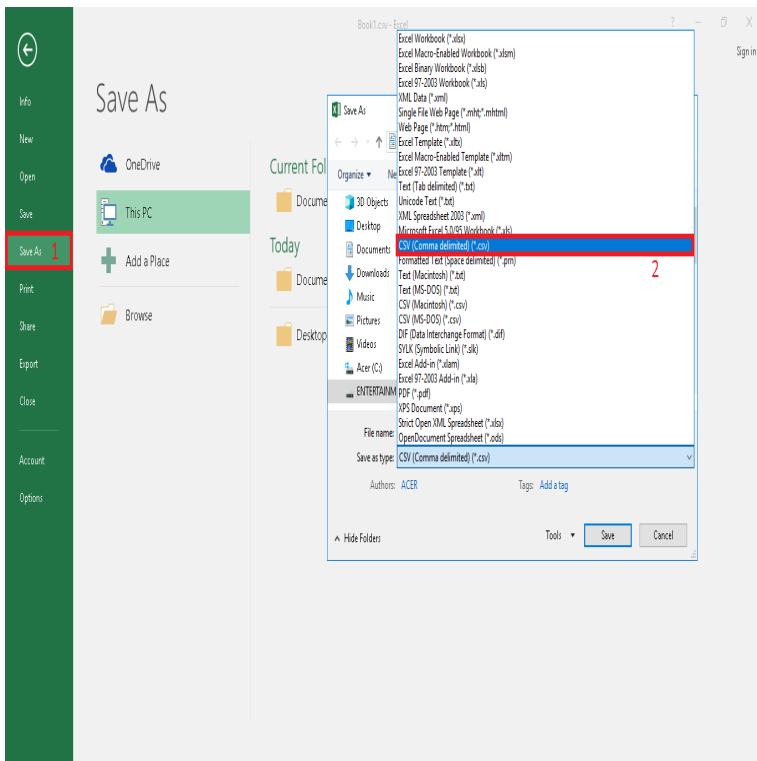
- Pertama buka terdahulu Program Excel anda
- Lalu sertakan data contoh seperti screenshot di bawah
- Simpan dengan ekstensi csv
- untuk membuka file csv, cari file yang anda simpan tadi

4. Jelaskan sejarah library csv

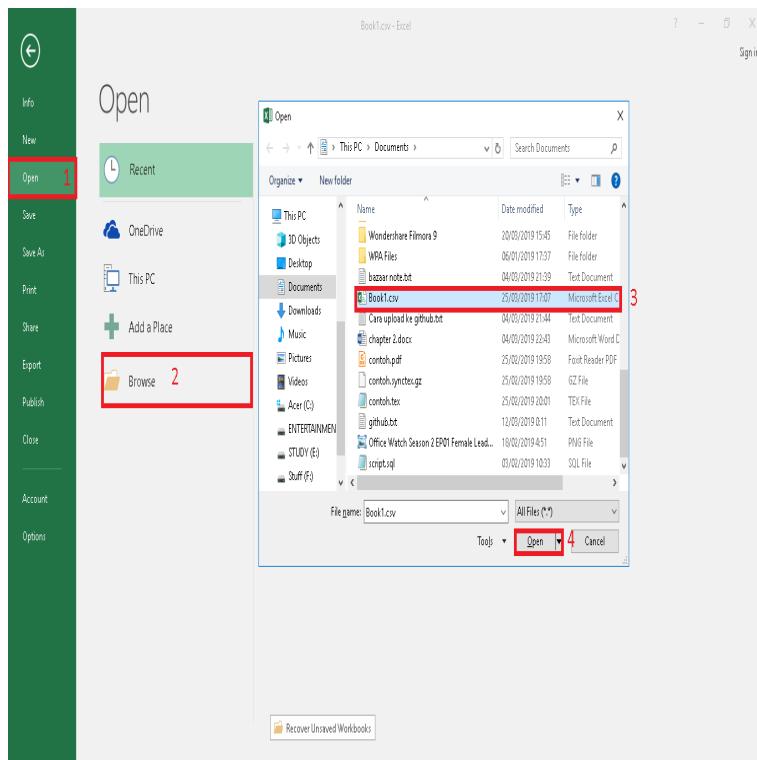
CSV (Comma separated Values) adalah format data yang telah ada sebelum personal komputer hadir 10 tahun sebelumnya, CSV file pada tahun 1972 telah didukung oleh IBM Fortran, nama CSV sendiri mulai digunakan pada tahun 1983, menggunakan koma dan spasi sebagai pembatas, jadi karakter string yang tidak dikutip tidak bisa mengandung koma atau spasi, list CSV lebih mudah untuk diketik daripada data yang diatur berdasarkan kolumn, dan lebih terhindar dari hasil yang salah jika di cetak pada media seperti punched card



Gambar 6.1 Buat data CSV



Gambar 6.2 Menyimpan CSV



Gambar 6.3 Menyimpan CSV

5. Jelaskan sejarah library pandas

Pandas (Python Data Analysis Library) nama ini didapat dari istilah panel data dan juga istilah econometrik untuk data terstruktur multidimensi, Developer Wes McKinney mulai mengerjakan pandas pada tahun 2008, ketika saat AQR capital management sedang membutuhkan alat yang memiliki performa tinggi untuk melakukan analisis kuantitatif tentang data finansial, sebelum keluar dari AQR, dia bisa membuat management percaya untuk supaya dia bisa membuat library open source. Chang She, bergabung pada tahun 2012 sebagai kontributor ke 2 untuk library pandas.

6. Jelaskan fungsi-fungsi yang terdapat di library csv

- fungsi open di CSV

Digunakan untuk membuka dan membaca file csv

```

1 with open('contohchandra.txt') as csv_file:
2     csv_reader = csv.reader(csv_file, delimiter=',')
3     line_count = 0
4     for row in csv_reader:
5         if line_count == 0:
6             print(f'Column names are {" ".join(row)}')
7             line_count += 1
8         else:
9             print(f'\t{row[0]} works in the {row[1]}'
10                department, and was born in {row[2]}.')
11             line_count += 1
12     print(f'Processed {line_count} lines.')

```

- fungsi write di csv

Digunakan untuk menulis data ke file csv menggunakan writerow

```

1 import csv
2 with open('contohchandra.csv', mode='w') as employee_file:
3     employee_writer = csv.writer(employee_file, delimiter=',',
4                                   quotechar='"', quoting=csv.QUOTE_MINIMAL)
5
6     employee_writer.writerow(['aaaaaaaaa', 'Accounting', 'November'])
6     employee_writer.writerow(['bbbbbb Meyers', 'IT', 'March'])

```

- fungsi write dengan parameter di csv

Digunakan untuk menulis data ke file csv menggunakan writerow

```

1 import csv
2 with open('employee_file2.csv', mode='w') as csv_file:
3     fieldnames = ['emp_name', 'dept', 'birth_month']
4     writer = csv.DictWriter(csv_file, fieldnames=fieldnames)
5
6     writer.writeheader()
7     writer.writerow({'emp_name': 'John Smith', 'dept': 'Accounting',
8                      'birth_month': 'November'})
8     writer.writerow({'emp_name': 'Erica Meyers', 'dept': 'IT',
9                      'birth_month': 'March'})

```

7. Jelaskan fungsi-fungsi yang terdapat di library pandas

- fungsi pandas read csv

Digunakan untuk membuka, menganalisis, dan membaca file csv

```
1 import pandas  
2 df = pandas.read_csv('contohchandrapandas.csv')  
3 print(df)
```

- fungsi pandas to csv Digunakan untuk menulis data ke file csv

```
1 import pandas
2 df = pandas.read_csv('contohchandrapandas.csv',
3                     index_col='Employee',
4                     parse_dates=['Hired'],
5                     header=0,
6                     names=['Employee', 'Hired', 'Salary', 'Sick Days']
7                )
7 df.to_csv('hrdata_modified.csv')
```

BAB 7

PRAKTEK LIBRARY CSV DAN PANDAS

7.1 D. Irga B. Naufal Fakhri

7.1.1 Soal 1

Buatlah fungsi untuk membuka file csv dengan lib csv mode list

```
1 def read(namafile):
2     with open(namafile) as csv_file:
3         csv_reader = csv.reader(csv_file, delimiter=',')
4         line_count = 0
5         for row in csv_reader:
6             if line_count == 0:
7                 print(f'Column names are {" ".join(row)}')
8                 line_count += 1
9             else:
10                 print(f'\tNomor NPM: {row[0]} Nama: {row[1]} Kelas: {row[2]}.')
11                 line_count += 1
12             print(f'Processed {line_count} lines.')
```

7.1.2 Soal 2

Buatlah fungsi untuk membuka file csv dengan lib csv mode dictionary

```

1 def readdict(namafile):
2     with open(namafile, mode='r') as csv_file:
3         csv_reader = csv.DictReader(csv_file)
4         line_count = 0
5         for row in csv_reader:
6             if line_count == 0:
7                 print(f'Column names are {" ".join(row)}')
8                 line_count += 1
9             print(f'\tNomor NPM: {row["npm"]} Nama: {row["nama"]}')
10            Kelas: {row["kelas"]})
11            line_count += 1
12            print(f'Processed {line_count} lines.')

```

7.1.3 Soal 3

Buatlah fungsi untuk membuka file csv dengan lib pandas mode list

```

1 def readpanda(namafile):
2     df = pandas.read_csv(namafile)
3     print(df)

```

7.1.4 Soal 4

Buatlah fungsi untuk membuka file csv dengan lib pandas mode dictionary

```

1 def readpandadict(namafile):
2     df = pandas.read_csv(namafile).to_dict()
3     print(df)
4
5 #readpandadict('1174066.csv')

```

7.1.5 Soal 5

Buat fungsi baru untuk mengubah format tanggal menjadi standar dataframe

```

1 def readrubahTanggal(namafile):
2     df = pandas.read_csv(namafile, index_col='npm', parse_dates=[0,
3     'tanggallahir'])
4     print(df)

```

7.1.6 Soal 6

Buat fungsi baru untuk mengubah index kolom

```

1 def changeIndex(namafile):
2     df = pandas.read_csv(namafile, index_col='npm')
3     print(df)

```

7.1.7 Soal 7

Buat fungsi baru untuk mengubah atribut atau nama kolom

```

1 def renamekolom(namafile):
2     df = pandas.read_csv(namafile, header=0, names=['Nomor Induk
3         Mahasiswa', 'Nama Lengkap', 'Kelas dan Jurusan', 'Tanggal Lahir'])
4     print(df)

```

7.1.8 Soal 8

Buat program main yang menggunakan library NPM csv yang membuat dan membaca file csv

```

1 def write(namafile):
2     with open(namafile, mode='w') as employee_file:
3         tulis_file = csv.writer(employee_file, delimiter=',',
4             quotechar='"', quoting=csv.QUOTE_MINIMAL)
5
6         tulis_file.writerow(['npm', 'nama', 'kelas', 'tanggallahir'])
7         tulis_file.writerow(['1174066', 'D. Irga 3', 'D4 Teknik
8             Informatika 2C', '11/07/98'])

```

Listing 7.1 lib_1174066.csv.py

```

1 import lib_1174066_csv
2
3 # Membuat file csv baru yang isinya npm, nama, kelas dan tanggallahir
4 lib_1174066_csv.write('1174066_csv_baru.csv')
5
6 # Membaca file csv
7 # List
8 lib_1174066_csv.read('1174066.csv')
9 # Dictionary
10 lib_1174066_csv.readdict('1174066.csv')

```

Listing 7.2 main.py

7.1.9 Soal 9

Buat program main2.py yang menggunakan library NPM pandas.py yang membuat dan membaca file csv

```

1 def writepanda(namafile, filebaru):
2     df = pandas.read_csv(namafile,
3             index_col='npm',
4             parse_dates=['tanggallahir'],
5             header=0,
6             names=['npm', 'nama', 'kelas', 'tanggallahir'])
7     df.to_csv(filebaru)

```

Listing 7.3 main2.py

```

1 import lib_1174066_pandas
2
3 # Membuat file baru
4 lib_1174066_pandas.writepanda('1174066.csv','1174066_baru_pandas.csv')
5
6 # Membaca file csv panda
7 # Dataframe
8 lib_1174066_pandas.readpanda('1174066.csv')
9 # Dictionary
10 lib_1174066_pandas.readpandadict('1174066.csv')

```

Listing 7.4 main2.py

7.1.10 Keterampilan Penanganan Error

Pada praktikum saat ini saya tidak mendapatkan error

7.2 Fanny Shafira Damayanti | 1174069

7.2.1 Keterampilan Pemrograman

1. NO 1

```

1 #No. 1
2 def bukaModeListCsv():
3     with open('teori.csv') as csv_file:
4         csv_reader = csv.reader(csv_file, delimiter=',')
5         for row in csv_reader:
6             print(row[0], row[1], row[2])

```

2. NO 2

```

1 #No. 2
2 def bukaModeDictCsv():
3     with open('teori.csv', mode='r') as csv_file:
4         csv_reader = csv.DictReader(csv_file)
5         for row in csv_reader:
6             print(row['npm'], row['nama'], row['kelas'])

```

3. NO 3

```

1 #No. 3
2 def bukaModeListPandas():
3     df = pandas.read_csv('teori.csv')
4     print(df)

```

4. NO 4

```

1 #No. 3
2 def bukaModeListPandas():
3     df = pandas.read_csv('teori.csv')
4     print(df)

```

5. NO 5

```

1 #No. 4
2 def bukaModeDictPandas():
3     df = pandas.read_csv('teori.csv')
4     dt = pandas.DataFrame.from_dict(df)
5     print(dt)

```

6. NO 6

```

1 #No. 5
2 def ubahFormatTanggal():
3     df = pandas.read_csv('teori.csv', parse_dates=['tanggal_lahir',
4     ''])
5     print(df)

```

7. NO 7

```

1 #No. 6
2 def ubahIndexKolom():
3     df = pandas.read_csv('teori.csv')
4     df.index = ['Row_1', 'Row_2']
5     print(df)

```

8. NO 8

```

1 lib = __import__('1174069_csv')
2
3 lib.bukaModeListCsv()
4 lib.bukaModeDictCsv()
5
6 lib.tulisCsv()

```

9. NO 9

```

1 lib = __import__('1174069_pandas')
2
3 lib.bukaModeListPandas()
4 lib.bukaModeDictPandas()
5
6 lib.tulisCsvPandas()

```

7.2.2 Penanganan Error

1. Peringatan error yang terdapat pada praktikum chapter 4 ini yaitu :

- Syntax Errors Syntax Errors terjadi ketika ada kesalahan dalam menuliskan kode. Solusinya adalah memperbaiki penulisan kode yang salah.
- Name Error NameError terjadi ketika salah mengetikan kode local name yang tidak terdefinisi. Solusinya adalah menuliskan kode dengan benar agar function nya dapat terpanggil.

- Type Error TypeError terjadi pada saat eksekusi terhadap fungsi dengan tipe objek tidak sesuai. Solusinya mengkonversi variablenya harus sesuai dengan tipe datanya.

Contoh Penggunaan TryExcept

```

1 #Fungsi Try Except
2 def bacaCsvPandas():
3     try:
4         df = pandas.read_csv('teori.csv')
5         print(dt)
6     except SyntaxError:
7         print("Kesalahan penulisan syntax")
8     except NameError:
9         print("Variable tersebut tidak ada")
10    except TypeError:
11        print("Tipe data salah")
12    except:
13        print("Terjadi sebuah kesalahan")

```

7.3 Aulyardha Anindita | 1174054

7.3.1 Keterampilan Pemrograman

1. Jawaban Soal No. 1

```

1 #Jawaban No. 1
2 def bukaModeListCsv():
3     with open('teori.csv') as csv_file:
4         csv_reader = csv.reader(csv_file, delimiter=',')
5         for row in csv_reader:
6             print(row[0], row[1], row[2])

```

2. Jawaban Soal No. 2

```

1 #Jawaban No. 2
2 def bukaModeDictCsv():
3     with open('teori.csv', mode='r') as csv_file:
4         csv_reader = csv.DictReader(csv_file)
5         for row in csv_reader:
6             print(row['npm'], row['nama'], row['kelas'])

```

3. Jawaban Soal No. 3

```

1 #Jawaban No. 3
2 def bukaModeListPandas():
3     df = pandas.read_csv('teori.csv')
4     print(df)

```

4. Jawaban Soal No. 4

```

1 #Jawaban No. 3
2 def bukaModeListPandas():
3     df = pandas.read_csv('teori.csv')
4     print(df)

```

5. Jawaban Soal No. 5

```

1 #Jawaban No. 4
2 def bukaModeDictPandas():
3     df = pandas.read_csv('teori.csv')
4     dt = pandas.DataFrame.from_dict(df)
5     print(dt)

```

6. Jawaban Soal No. 6

```

1 #Jawaban No. 5
2 def ubahFormatTanggal():
3     df = pandas.read_csv('teori.csv', parse_dates=['tanggal_lahir'])
4     print(df)

```

7. Jawaban Soal No. 7

```

1 #Jawaban No. 6
2 def ubahIndexKolom():
3     df = pandas.read_csv('teori.csv')
4     df.index = ['Row_1', 'Row_2']
5     print(df)

```

8. Jawaban Soal No. 8

```

1 lib = __import__('1174054csv')
2
3 lib.bukaModeListCsv()
4 lib.bukaModeDictCsv()
5
6 lib.tulisCsv()

```

9. Jawaban Soal No. 9

```

1 lib = __import__('1174054pandas')
2
3 lib.bukaModeListPandas()
4 lib.bukaModeDictPandas()
5
6 lib.tulisCsvPandas()

```

7.3.2 Keterampilan Penanganan Error

Peringatan error di praktik keempat ini, yaitu:

- Syntax Errors Syntax Errors adalah keadaan dimana pada kode python mngalami kesalahan dalam penulisan. Untuk mengatasinya yaitu dengan memperbaiki penulisan kode yang salah
- Name Error NameError adalah suatu keadaan atau exception yang terjadi ketika kode melakukan eksekusi terhadap local name atau global name yang tidak terdefinisi. Untuk mengatasinya yaitu dengan memastikan variabel atau function yang dipanggil ada atau tidak salah ketik.
- Type Error TypeError adalah suatu keadaan atau exception yang akan terjadi apabila pada saat dilakukannya eksekusi terhadap suatu operasi atau fungsi dengan type object yang tidak sesuai. Untuk mengatasinya yaitu dengan mengkoversi variabelnya sesuai dengan tipe data yang akan digunakan.

Fungsi yang menggunakan try except

```

1 #Fungsi Try Except
2 def bacaCsvPandas () :
3     try :
4         df = pandas.read_csv ('teori.csv')
5         print (dt)
6     except SyntaxError :
7         print ("Kesalahan penulisan syntax")
8     except NameError :
9         print ("Variable tersebut tidak ada")
10    except TypeError :
11        print ("Tipe data salah")
12    except :
13        print ("Terjadi sebuah kesalahan")
```

7.4 Nurul Izza Hamka | 1174062 | Teori

7.5 Keterampilan Pemrograman

1. NO1

```

1 #Jawaban No. 1
2 def bukaModeListCsv () :
3     with open ('teori.csv') as csv_file :
4         csv_reader = csv.reader (csv_file , delimiter = ',')
5         for row in csv_reader :
6             print (row [0] , row [1] , row [2])
```

2. NO2

```

1 #Jawaban No. 2
2 def bukaModeDictCsv () :
3     with open ('teori.csv' , mode = 'r') as csv_file :
4         csv_reader = csv.DictReader (csv_file)
5         for row in csv_reader :
6             print (row [ 'npm' ] , row [ 'nama' ] , row [ 'kelas' ])
```

3. NO3

```
1 #Jawaban No. 3
2 def bukaModeListPandas():
3     df = pandas.read_csv('teori.csv')
4     print(df)
```

4. NO4

```
1 #Jawaban No. 3
2 def bukaModeListPandas():
3     df = pandas.read_csv('teori.csv')
4     print(df)
```

5. NO5

```
1 #Jawaban No. 4
2 def bukaModeDictPandas():
3     df = pandas.read_csv('teori.csv')
4     dt = pandas.DataFrame.from_dict(df)
5     print(dt)
```

6. NO6

```
1 #Jawaban No. 5
2 def ubahFormatTanggal():
3     df = pandas.read_csv('teori.csv', parse_dates=['tanggal lahir'])
4     print(df)
```

7. NO7

```
1 #Jawaban No. 6
2 def ubahIndexKolom():
3     df = pandas.read_csv('teori.csv')
4     df.index = ['Row_1', 'Row_2']
5     print(df)
```

8. NO8

```
1 lib = __import__('1174006csv')
2
3 lib.bukaModeListCsv()
4 lib.bukaModeDictCsv()
5
6 lib.tulisCsv()
```

9. NO9

```
1 lib = __import__('1174006pandas')
2
3 lib.bukaModeListPandas()
4 lib.bukaModeDictPandas()
5
6 lib.tulisCsvPandas()
```

7.6 Ketrampilan Penanganan Error

- NameError Error yang terjadi adalah NameError yang mana terjadi kesalahan pada saat melakukan eksekusi dan tidak dapat terdefinisi. Penanganan yang dapat dilakukan adalah memastikan bahwa Variable dan Function yang akan kita panggil pastikan semua benar.
- SyntaxError Tipe Error ini adalah ada kesalahan pada penulisan, untuk itu pastikan penulisannya benar.
- TypeError Error ini terjadi pada saat eksekusi terhadap dan operasi dan fungsi masuk kedalam objek sedangkan typenya salah.
- Menggunakan Try Except

```

1 #Fungsi Try Except
2 def bacaCsvPandas () :
3     try :
4         df = pandas.read_csv ('teori.csv')
5         print(dt)
6     except SyntaxError:
7         print("Kesalahan penulisan syntax")
8     except NameError:
9         print("Variable tersebut tidak ada")
10    except TypeError:
11        print("Tipe data salah")
12    except:
13        print("Terjadi sebuah kesalahan")

```

7.7 Chandra Kirana Poetra

7.7.1 Soal 1

Buatlah fungsi (file terpisah/library dengan nama NPM csv.py) untuk membuka file csv dengan lib csv mode list

```

1 def readcsvfile (namafайл):
2     with open(namafайл) as csv_file:
3         csv_reader = csv.reader(csv_file , delimiter=',')
4         line_count = 0
5         for row in csv_reader:
6             if line_count == 0:
7                 print(f'Column names are {", ".join(row)}')
8                 line_count += 1
9             else:
10                print(f'\tNomor No: {row[0]} Nama: {row[1]} NPM: {row
11 [2]}.')
12                line_count += 1
13                print(f'Processed {line_count} lines .')

```

7.7.2 Soal 2

Buatlah fungsi (file terpisah/library dengan nama NPM csv.py) untuk membuka file csv dengan lib csv mode dictionary

```

1 def readdengandict(a):
2     with open(a, mode='r') as csv_file:
3         csv_reader = csv.DictReader(csv_file)
4         line_count = 0
5         for row in csv_reader:
6             if line_count == 0:
7                 print(f'Column names are {", ".join(row)}')
8                 line_count += 1
9             print(f'\tNomor No: {row["No"]} Nama: {row["Nama"]} NPM:
{row["NPM"]}')
10            line_count += 1
11            print(f'Processed {line_count} lines.')

```

7.7.3 Soal 3

Buatlah fungsi (file terpisah/library dengan nama NPM csv.py) untuk membuka file csv dengan lib csv mode dictionary

```

1 def exampleofhowtoreadpanda(a):
2     df = pandas.read_csv(a)
3     print(df)

```

7.7.4 Soal 4

Buatlah fungsi (file terpisah/library dengan nama NPM csv.py) untuk membuka file csv dengan lib csv mode dictionary

```

1 def exampleofreadpandausingdict(a):
2     df = pandas.read_csv(a).to_dict()
3     print(df)

```

7.7.5 Soal 5

Buat fungsi baru di NPM pandas.py untuk mengubah format tanggal menjadi standar dataframe

```

1 def exampleofreadchangedateformatpandas(a):
2     df = pandas.read_csv(a, index_col='npm', parse_dates=['
tanggallahir'])
3     print(df)

```

7.7.6 Soal 6

Buat fungsi baru di NPM pandas.py untuk mengubah index kolom

```

1 def exampleofhowtochangeindexinpandas(a):
2     df = pandas.read_csv(a, index_col='npm')
3     print(df)

```

7.7.7 Soal 7

Buat fungsi baru di NPM pandas.py untuk mengubah atribut atau nama kolom

```

1 def exampleofhowtorenameacolumn(a):
2     df = pandas.read_csv(a, header=0, names=['Student ID', 'Full Name',
3     'Class', 'Date of birth'])
4     print(df)

```

7.7.8 Soal 8

Buat program main.py yang menggunakan library NPM csv.py yang membuat dan membaca file csv

```

1 def write(a):
2     with open(a, mode='w') as exampleofhowtowrite:
3         tulis_file = csv.writer(exampleofhowtowrite, delimiter=',',
4         quotechar='"', quoting=csv.QUOTE_MINIMAL)
5
6         tulis_file.writerow(['No', 'Nama', 'NPM', 'tanggallahir'])
7         tulis_file.writerow(['1', 'Chandra Kirana Poetra', '1174079',
8         '05/09/99'])
9
10    import lib_1174079_csv
11    # buat file baru
12    lib_1174079_csv.write('1174079_csv_baru.csv')
13    # read dengan
14    # Dictionary
15    lib_1174079_csv.readdengandict('1174079.csv')
16    # List
17    lib_1174079_csv.readcsvfile('1174079.csv')

```

Listing 7.5 main.py

7.7.9 Soal 9

Buat program main2.py yang menggunakan library NPM pandas.py yang membuat dan membaca file csv

```

1 def examplehowtowritepanda(a,b):
2     df = pandas.read_csv(a,
3             index_col='npm',
4             parse_dates=['tanggallahir'],
5             header=0,
6             names=['No', 'Nama', 'NPM', 'tanggallahir'])
7     df.to_csv(b)

```

Listing 7.6 main2.py

```
1 import lib_1174079_pandas
2 # filebaru
3 lib_1174079_pandas . writepanda('117079.csv','1174066_baru_pandas.csv')
4 # baca file csv dengan library panda
5 # Dataframe
6 lib_1174079_pandas . exampleofhowtoreadpanda('117079.csv')
7 # Dictionary
8 lib_1174079_pandas . exampleofreadpandausingdict('117079.csv')
```

Listing 7.7 main2.py

7.7.10 Keterampilan Penanganan Error

Unicode Decode Error Terjadi ketika kesalahan decode untuk unicode terdeteksi, typo semacamnya

```
1 def nyobaintrycatch():
2     try :
3         da = pandas.read_csv("1174079.csv")
4         print(da)
5     except SyntaxError:
6         print("Cek apabila ada typo")
7     except NameError:
8         print("Variable tidak ditemukan")
9     except UnicodeDecodeError:
10        print("ada yang salah di penulisan")
11    except :
12        ("terjadi kesalahan")
```


BAB 8

KOMUNIKASI PERANGKAT KERAS

8.1 D. Irga B. Naufal Fakhri

8.1.1 Pemahaman Teori

1. Device Manager pada Windows dan folder /dev pada Linux

Device manager adalah system tools yang fungsinya untuk mengidentifikasi dan mengelola hardware serta driver yang diperlukan oleh hardware yang dihubungkan. Device Manager juga bisa mengenali hardware dan menginstall drivernya secara otomatis, tapi kalo driver dari hardware tersebut tidak ada pada koleksi driver Windows, maka Device Manager akan memberikan notifikasi dan tanda khusus bahwa hardware tersebut membutuhkan driver terpisah agar bisa terinstall dengan benar.

/dev pada Linux merupakan direktori yang berfungsi untuk menyimpan konfigurasi device atau hardware dari sistem, seperti harddisk (hda, sda), terminal (tty) dll.

2. Jelaskan langkah-langkah instalasi driver dari arduino

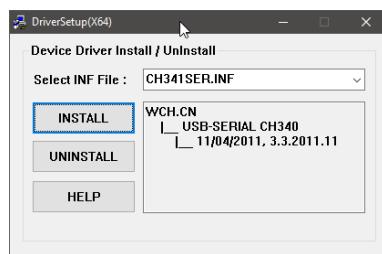
Apabila anda menggunakan Arduino versi original maka drivernya akan diinstall saat anda menginstall Arduino IDE saat anda menhubungkan Arduino anda ke PC menggunakan kabel USB. Jika anda menggunakan Arduino Uno SMD Clone seperti saya maka anda harus menginstall drivernya terlebih dahulu setelah menginstall Arduino Uno, Caranya:

- Buka program setup.exe pada folder DRIVER1



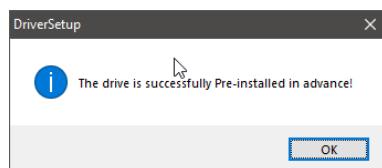
Gambar 8.1 Setup

- Setelah itu klik Install



Gambar 8.2 Instalasi

- Setelah muncul seperti dibawah ini tekan OK

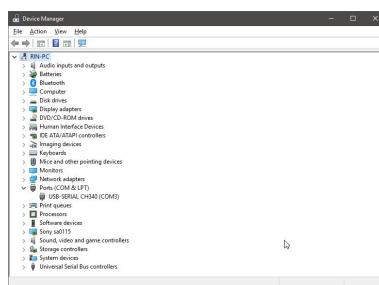


Gambar 8.3 Instalasi Berhasil

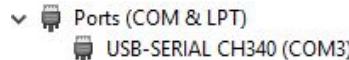
- Hubungkan Arduino ke PC, lalu buka device manager
- Lalu pilih pada bagian port apabila seperti ini maka driver arduino telah terinstall

3. Jelaskan bagaimana cara membaca baudrate dan port dari komputer yang sudah terinstall driver

Cara membaca baudrate adalah dengan cara membuka arduino ide lalu mengclik serial monitor yang iconnya seperti kaca pembesar(Cari). Untuk mengecek



Gambar 8.4 Device Manager



Gambar 8.5 Arduino Terdeteksi

port kita bisa melihatnya melalui device manager pada bagian ports, yang ada tulisan COMXX (XX adalah angka dari COM) itu adalah portnya

4. Jelaskan sejarah library pyserial

pySerial adalah modul API Python untuk mengakses port serial. pySerial menyediakan API yang seragam di berbagai sistem operasi, termasuk Windows, Linux, dan BSD.

5. Jelaskan fungsi-fungsi apa saja yang dipakai dari library pyserial

- **Serial()**

Berfungsi untuk membuka port serial.

- **Write()**

Berfungsi untuk mengirimkan data string ke port serial dan mengembalikan nomor bytes yang terkirim.

- **Read(size)**

Berfungsi untuk membaca data dari port serial.

- **Readline(size)**

Berfungsi untuk membaca line sampai line terakhir (EOL).

- **Close()**

Berfungsi untuk menutup pembacaan port serial.

6. Jelaskan kenapa butuh perulangan dan tidak butuh perulangan dalam membaca serial

Perulangan dalam python berfungsi untuk mengulangi kode/perintah yang ada didalam perulangan tersebut. ada dua macam perulangan pada python, yang

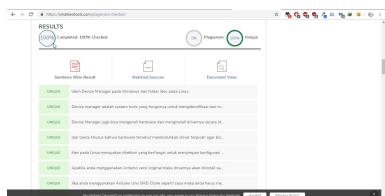
pertama for dan yang kedua while. Perbedaan for dan while adalah for yaitu perulangan yang menghitung (Counted Loop) sedangkan while adalah perulangan yang tidak terhitung (Uncounted Loop). Penggunaan for itu biasanya jika untuk mengulangi kode yang akan ditentukan berapa kali diulangnya sedangkan while digunakan jika ada syarat tertentu untuk mengulangi kode itu dan tidak menentu berapa banyak perulangannya. Mengapa diperlukan perulangan, karena agar data yang dibaca tidak hanya satu kali saja namun berkali kali, dengan adanya perulangan kita bisa membaca datanya berulang kali. Sehingga data yang kita baca dapat muncul lebih dari satu. Sedangkan kalau kita tidak memakai perulangan maka data yang muncul hanya satu.

7. Jelaskan bagaimana cara membuat fungsi yang menggunakan pyserial

Pembuatan fungsi sama seperti pembuatan fungsi seperti biasanya namun method dari pyserial dimasukkan kedalam fungsi dan dipanggil fungsi yang kita buat tadi

```
1 import serial
2
3 def ReadSerial():
4     ser = serial.Serial("COM3", 9600)
5     read = ser.readline()
6     print(read)
7
8 ReadSerial()
```

8. Scan Plagiarisme



Gambar 8.6 Plagiarisme

8.2 Difa Al Fansha

8.2.1 Soal Nomor 1

Apa itu fungsi device manager di windows dan folder /dev di linux?

Jawab :

- Fungsi Device Manager di Sistem Operasi Windows

Device manager merupakan program yang mengatur device atau perangkat yang terhubung dengan komputer / laptop.

- Fungsi Folder /dev di Sistem Operasi Linux

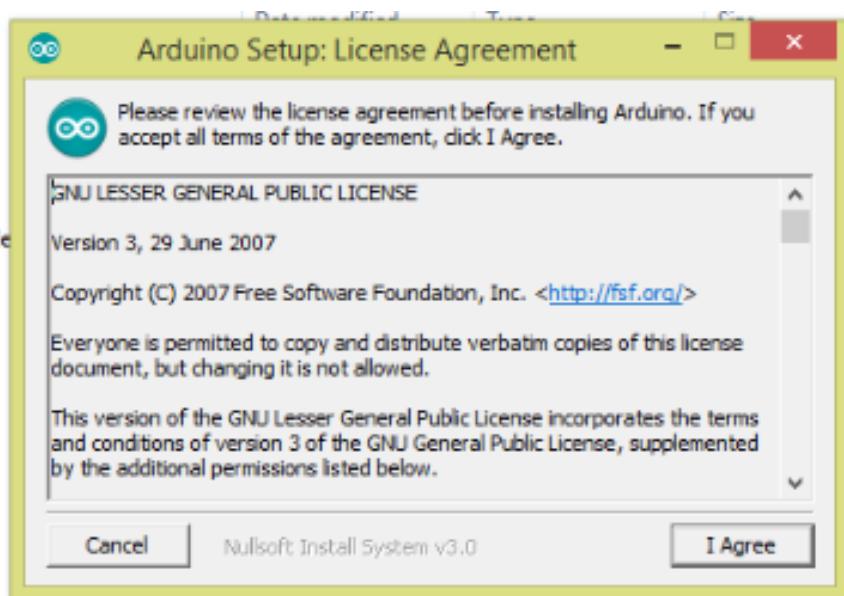
Device manager pada linux berada pada folder /dev yang mempunyai arti device. folder ini berisi konfigurasi device pada sistem.

8.2.2 Soal Nomor 2

Jelaskan langkah-langkah instalasi driver dari arduino!

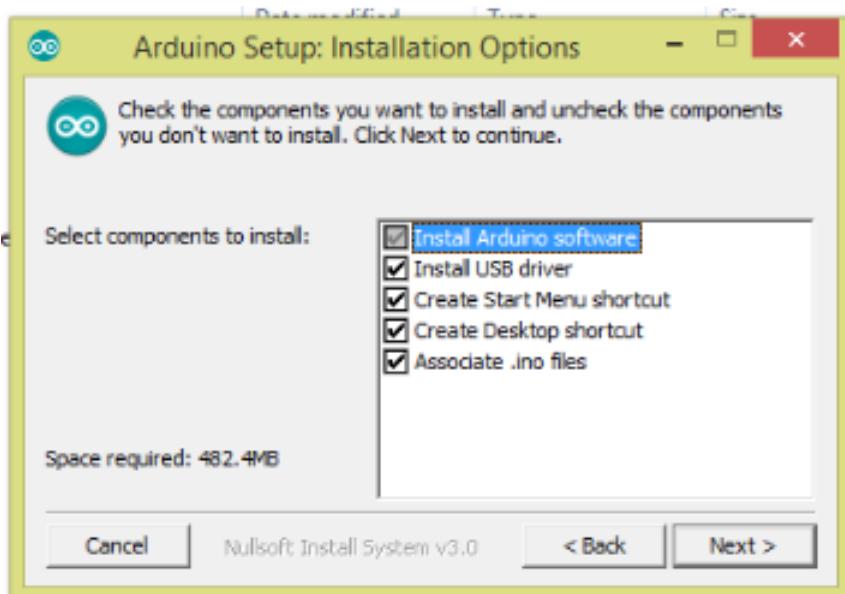
Jawab :

1. Download Software IDE Arduino di <https://www.arduino.cc/en/Main/Software>
2. Setelah di download jalankan software tersebut.
3. Pilih I Agree.

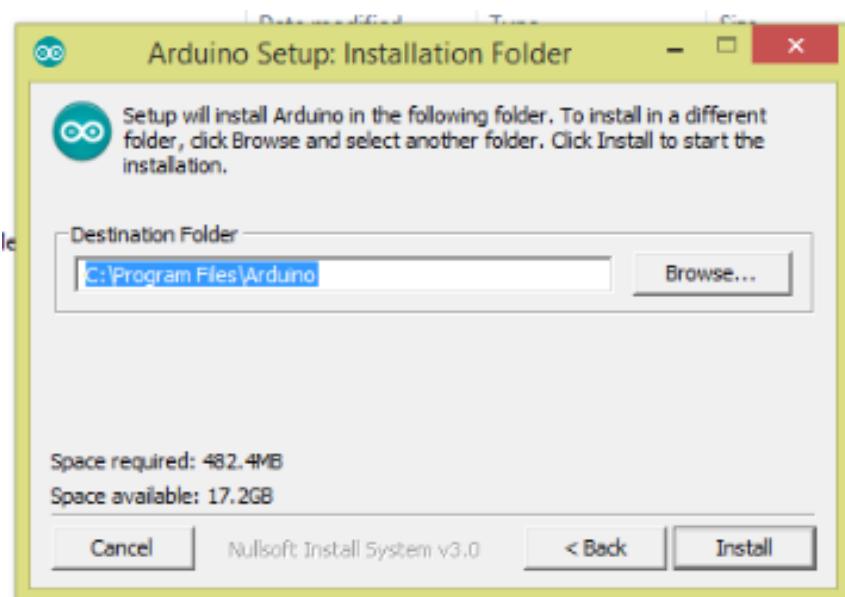


Gambar 8.7 Licence Agreement

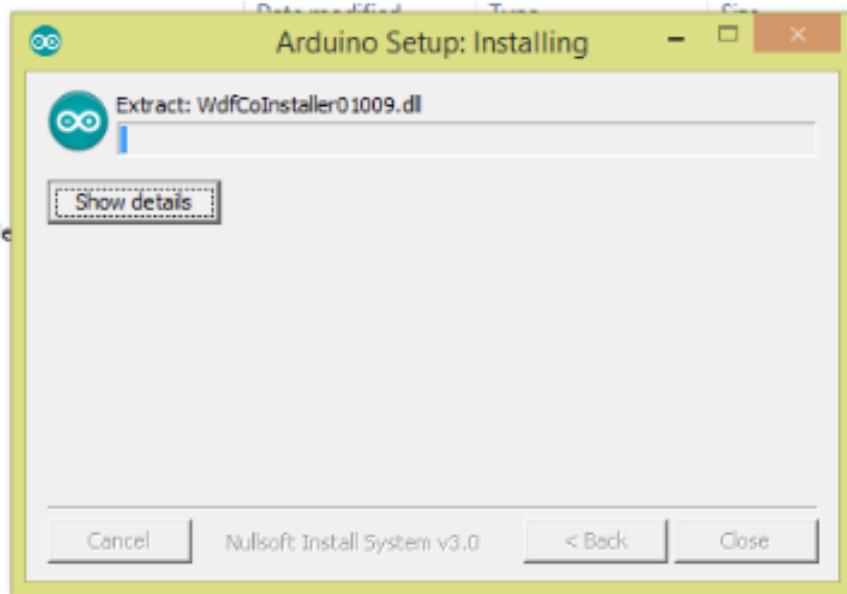
4. Lalu Next.
5. Pilih directory, lalu tekan Install.
6. Tunggu hingga proses selesai
7. Jika sudah selesai, tekan close



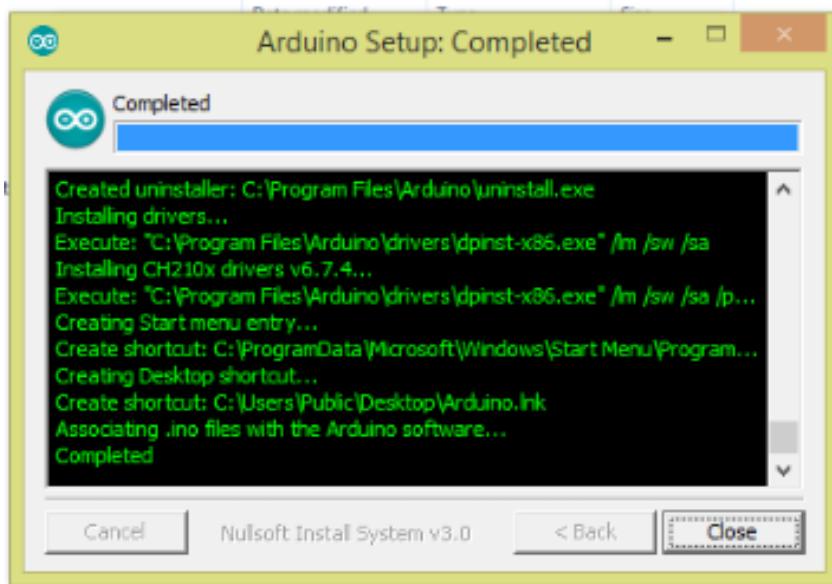
Gambar 8.8 Installation Options



Gambar 8.9 Installation Folder



Gambar 8.10 Installation



Gambar 8.11 Installing

8.2.3 Soal Nomor 3

Jelaskan bagaimana cara membaca baud rate dan port dari komputer yang sudah terinstall driver!

Jawab :

baud rate dan port akan langsung terbaca saat arduino dicolokkan ke komputer, sistem akan membaca informasi device yang terhubung.

8.2.4 Soal Nomor 4

Jelaskan sejarah library pyserial! Jawab :

Salah satu cara untuk melakukan komunikasi melalui serial menggunakan python adalah dengan menggunakan modul pyserial. PySerial diluncurkan pada tahun 2002.

8.2.5 Soal Nomor 5

Jelaskan fungsi-fungsi apa saja yang dipakai dari library pyserial!

Jawab :

- stop() berguna untuk menghentikan program yang berjalan.
- readline() berguna untuk membaca sebuah string dari port serial.
- read(size)berguna untuk membaca jumlah byte dari port serial.
- close()berguna untuk menutup port serial.

8.2.6 Soal Nomor 6

Jelaskan kenapa butuh perulangan dan tidak butuh perulangan dalam membaca serial!

Jawab :

- Perulangan

Perulangan diperlukan untuk untuk mengulangi perintah agar lebih mudah dan tidak terjadi penumpukan kodingan. Perulangan dijalankan jika kondisi benar dan akan berhenti jika kondisi salah.

- Tidak membutuhkan perulangan

Apabila perintah dijalankan sekali, kita tidak memerlukan perulangan.

8.2.7 Soal Nomor 7

Jelaskan bagaimana cara membuat fungsi yang menggunakan pyserial!

Jawab :

Definisikan nama fungsi dengan cara def namaFungsi(): lalu masukkan isi fungsi

8.3 Fanny Shafira Damayanti | 1174069

8.3.1 Pemahaman Teori

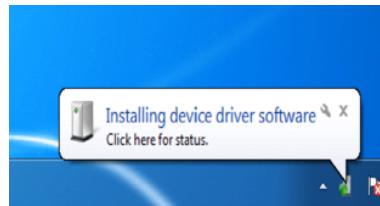
1. Fungsi Device manager di Windows dan folder /dev di Linux

Device Manager di Windows berfungsi untuk mengelola semua Hardware yang berada di Windows.

Folder /dev berisikan file device pada Linux.

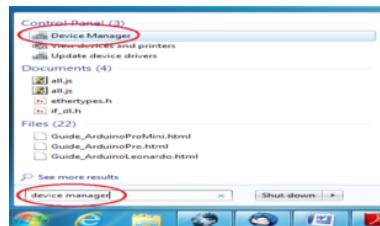
2. Langkah-langkah Instalasi driver dari arduino

- Download Software Arduino IDE
- Hubungkan port USB Arduino ke port pc.
- setelah itu pc akan mendeteksi driver baru



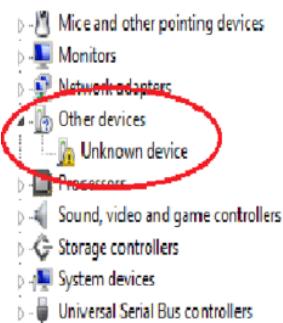
Gambar 8.12 Driver baru

- Buka device manager

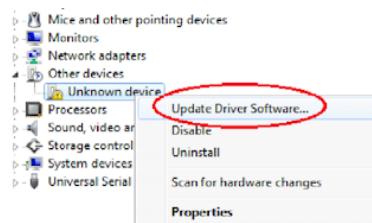


Gambar 8.13 Buka device manager

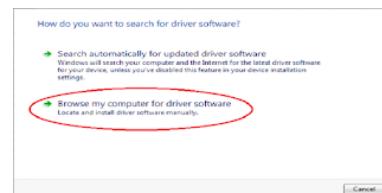
- setelah itu, cari "Unknown device"
- Klik kanan pada "Unknown device" lalu klik Update driver software
- Pilih Browse my computer for driver software
- Cari folder installan Arduino nya
- Lalu klik next
- setelah itu klik install



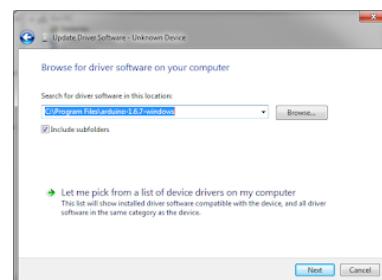
Gambar 8.14 klik "unknown device"



Gambar 8.15 Klik update driver software



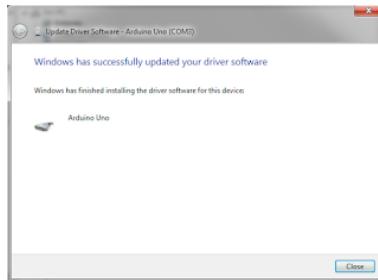
Gambar 8.16 Pilih browse



Gambar 8.17 Cari folder instalasinya



Gambar 8.18 install



Gambar 8.19 selesai

- Arduino telah berhasil di install
3. Cara membaca baudrate dan port dari komputer yang sudah terinstall driver
Untuk membaca baudrate dan port yaitu dengan cara :
- menginstall Arduino IDE
 - setelah itu buka menu serial monitor yang berada di tab tools
 - lalu akan terlihat baudrate dan port yang sedang digunakan.
4. Sejarah library pyserial PySerial merupakan sebuah library yang digunakan untuk komunikasi ke port serial terutama untuk mikrokontroller. PySerial pertama kali diluncurkan pada tahun 2002 yang makin berkembang dalam setiap versinya hingga tahun 2017 lalu.
5. Fungsi-fungsi yang di pakai dari libarary pyserial
- `stop()`
: untuk menghentikan pembacaan program
 - `serial.to_bytes(sequence)`
: berfungsi untuk mengubah sequence ke dalam bytes agar dapat dikirim ke dalam arduino.
 - `close()`
: untuk menutup port dan menghentikan pembacaan program

6. kenapa butuh perulangan dan tidak butuh perulangan dalam membaca serial

Karena perulangan digunakan untuk membaca seluruh data pada serial yang ada setiap baris. Perulangan digunakan agar data dapat muncul secara terus menerus atau realtime. Sedangkan kalau tidak memakai perulangan, maka data hanya muncul satu kali, tidak berulang.

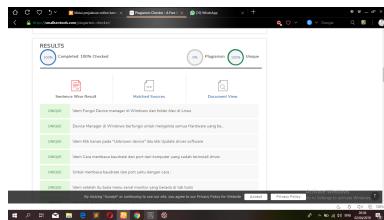
7. Cara membuat fungsi menggunakan pyserial

```

1 import serial
2
3 def testArduino():
4     ser = serial.Serial("COM5", 115200)
5     print(ser.name)
6
7 testArduino()

```

8. Scan Plagiarisme



Gambar 8.20 plagiarisme

8.4 Aulyardha Anindita | 1174054

8.4.1 Pemahaman Teori

1. Fungsi Device Manager dan folder /dev

- Fungsi Device Manager

Device manager dalam windows merupakan perluasan dari microsoft management console. Device manager mampu menampilkan seluruh hardware yang bisa diinisialisasi (dikenali) oleh windows. Device manager disini memiliki beberapa fungsi diantaranya :

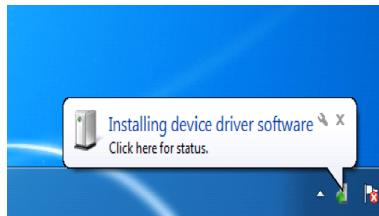
- a. Menunjukkan status suatu hardware
- b. Menunjukkan informasi detail suatu hardware
- c. Mengelola driver hardware
- d. Disable dan enable hardware
- e. Memberikan pesan error terjadinya problem status device

- Fungsi Folder /dev

Folder/dev adalah suatu representasi dari drive yang terhubung ke dalam sistem operasi linux dan sistem menganggapnya sebagai file-file direktori. Biasanya sering ditampilkan direktori seperti /dev/sdal yang mewakili Drive SATA pertama dalam sistem.

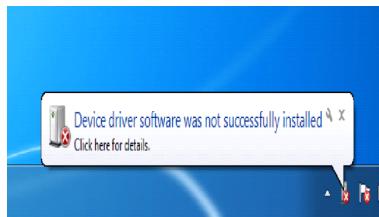
2. Langkah-langkah instalasi driver dari arduino

- Pertama, download dan instal terlebih dahulu Arduino IDE
- Hubungkan port USB Arduino UNO ke Port USB PC, maka PC akan mendekati keberadaan perangkat baru dan akan muncul pop up berupa pesan install device driver software. Seperti pada gambar berikut :



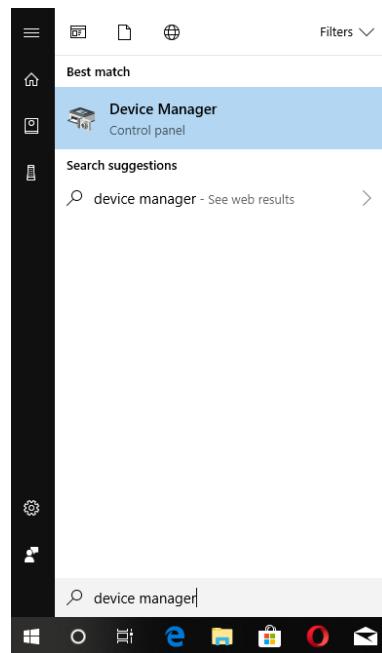
Gambar 8.21 Setup

- Dalam Windows tidak menyediakan menyediakan driver untuk Arduino Uno seperti yang terlihat pada gambar dibawah ini, sehingga proses instalasinya harus dilakukan secara manual.



Gambar 8.22 Instalasi

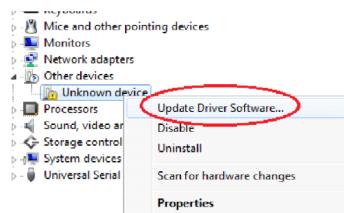
- Selanjutnya, buka start windows kemudian ketikkan device manager lalu tekan enter, maka akan muncul device manager. Klik untuk menjalankan. Seperti pada gambar berikut :
- Kemudian Carilah Unknown device pada bagian Other device, biasanya terdapat tanda seru berwarna kuning hal itu disebabkan karena penginstallan tidak berjalan dengan sempurna.
- Klik kanan pada “Unknown device” kemudian pilih Update Driver Software



Gambar 8.23 Search Device Manager

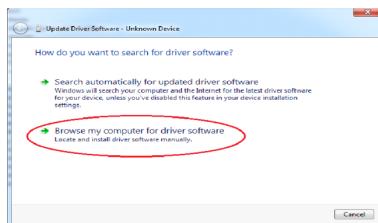


Gambar 8.24 Unknown Device



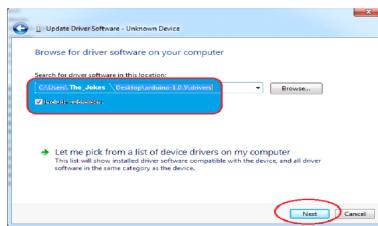
Gambar 8.25 Update Driver Software

- Pilih Browse my computer for driver software



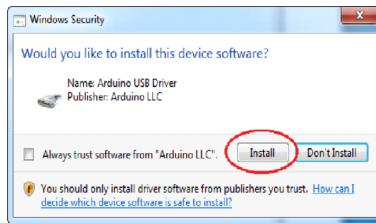
Gambar 8.26 Mencari folder untuk driver software

- Arahkan lokasi folder ke folder arduino kemudian checkbox lalu centang include subfolders. setelah itu, Klik Next untuk melanjutkan instalasi driver.



Gambar 8.27 Lokasi folder

- Kemudian lanjutkan dengan mengklik Install pada tampilan Windows Security

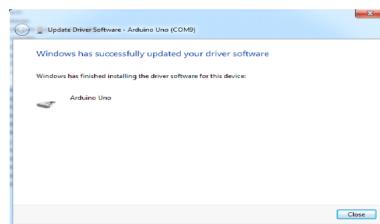


Gambar 8.28 Install PAda tampilan WIndows Security

- Jika instalasi driver berhasil maka akan muncul Windows has successfully updated your driver software.
- Perhatikan dan ingat nama COM Arduino Uno, karena nama COM ini yang akan digunakan untuk meng-upload program nantinya

3. Cara Membaca Baudrate dan Port

Cara membaca baudrate yaitu dengan membuka arduino ide yang telah diinstal lalu klik serial monitor dimana iconnya seperti kaca pembesar (cari).



Gambar 8.29 Pesan Peringatan Berhasil



Gambar 8.30 Peringatan COM Arduino Uno

Cara membaca port yaitu dengan melalui device manager pada bagian port yang terdapat tulisan COMXX dimana itu adalah portnya.

4. Sejarah Library Pyserial

PySerial merupakan sebuah library yang digunakan untuk komunikasi ke port serial terutama untuk mikrokontroller. PySerial pertama kali diluncurkan pada tahun 2002 yang makin berkembang dalam setiap versinya hingga tahun 2017 lalu.

PySerial menyediakan antarmuka untuk berkomunikasi melalui protokol komunikasi serial. Komunikasi serial adalah salah satu protokol komunikasi komputer tertua. Protokol komunikasi serial mendahului spesifikasi USB yang digunakan oleh komputer dan perangkat keras lain seperti mouse, keyboard, dan webcam. USB adalah singkatan dari Universal Serial Bus. USB dan dibangun di atas dan memperluas antarmuka komunikasi serial asli.

5. Fungsi-fungsi pada Library Pyserial

- `Serial()`, berfungsi untuk membuka port serial
- `Write()`, berfungsi untuk menulis data lewat port serial
- `readline(size)`, berfungsi untuk membaca sebuah string dari port serial
- `read(size)`, berfungsi untuk membaca jumlah byte dari port serial
- `close()`, berfungsi untuk menutup port serial

6. Mengapa butuh perulangan dan tidak butuh perulangan dalam membaca serial

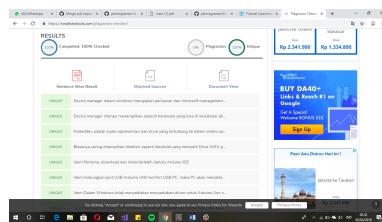
Ketika kita akan membaca serial di Arduino diperlukan perulangan yang berfungsi

untuk membaca data secara berulang kali sehingga data yang muncul banyak. Sedangkan apabila tidak membutuhkan perulangan maka Arduino hanya akan membaca data satu kali saja.

7. Cara Membuat fungsi menggunakan Pyserial

```
1 #Teori Membuat Fungsi
2 import serial
3
4 def testArduino():
5     ser = serial.Serial("COM5", 115200)
6     print(ser.name)
7
8 testArduino()
```

8. Cek Plagiarisme



Gambar 8.31 Plagiarisme

8.5 Nurul Izza Hamka | 1174062 | Teori

1. Fungsi Device Manager di Windows dan Folder / Dev di Linux Device manager ini adalah untuk menampilkan semua hardware yang bisa dikenali oleh windows, tampilan di device manager ini sudah dikelompokkan dengan rapi serta teratur,dengan adanya device manager ini akan sangat membantu dalam memanajemen setiap hardware yang ada dalam windows.

Fungsi dari device manager ini adalah seperti :

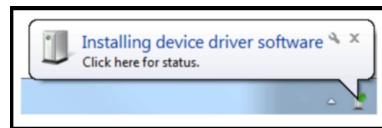
1. Mengelola setiap driver didalam hardware,
2. Mengidentifikasi konfig yang terjadi dalam hardware,
3. Menunjukka setiap informasi secara detail suatu hardware,
4. Menampilkan status dalam hardware.

Sedangka pada Linux ia melihat hardware apa saja yang yang telah terpasang melalui command dan terminal ataupun secara Visual.

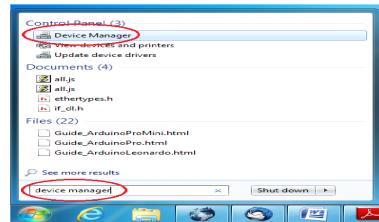
2. Langkah-Langkah Instalasi Driver dari Arduino :

Adapun langkah-langkah dalam menginstal arduino di Windows :

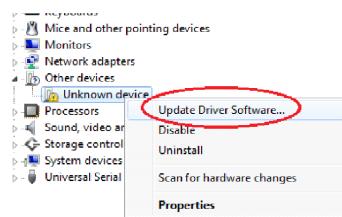
- (a) Pertama Hubungkan Port USB arduino UNO ke komputer atau Laptop,



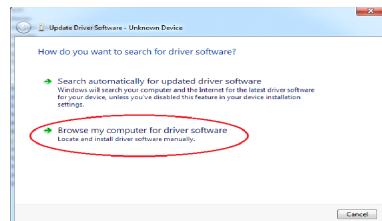
- (b) Setelah terhubung, laptop akan mengidentifikasi terlebih dahulu,
- (c) Selanjutnya windows akan menginstall di driver,pastikan device manager kita telah terinstall,



- (d) Di device manager pada bagian Ports, akan muncul nama baru yaitu Arduino UNO,
- (e) Klik kanan Arduino Uno tersebut kemudian pilih Update Driver Software,



- (f) Setelah itu Browser my Computer for driver software,lalu klik Next untuk install



- (g) Jika berhasil, berarti Arduino telah berhasil di install.

3. Cara membaca baudrate dan Port dari Komputer yang telah Terinstall Driver:
 1. Buka device manager,



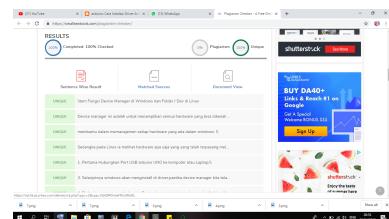
2. Pilih port (COM & LPT),
3. Kemudian double klik yang COM,
4. Pilih tab yang Port Setting, dan lihat pada Bit per Second,

Kemudian Cara Membaca Port Dari Komputer : 1. Buka device manager,
2. Pilih port (COM & LPT),
3. Selanjutnya akan muncul jika port arduino telah terbaca oleh PC.

4. Sejarah Library Pyserial: Pyserial adalah perpustakaan yang menyediakan dukungan untuk sebuah koneksi serial "RS-232" melalui berbagai perangkat yang berbeda. Pyserial ini adalah paket python yang memfasilitasi komunikasi serial antara PC dengan perangkat keras eksternal. Pyserial ini menyediakan interface yang dapat digunakan untuk berkomunikasi melalui protokol komunikasi serial.
5. Fungsi-Fungsi Yang Dipakai Dari Library Pyserial: 1. Serial,digunakan untuk membuka port serial,
2. Write(data), digunakan untuk menulis data lewat port serial,
3. Readline(),digunakan untuk membaca sebuah string dari port serial,
4. Read(size),digunakan untuk membaca jumlah byte dari port serial,
5. Close(),digunakan untuk menutup port serial.
6. Kenapa Butuh Perulangan dan Tidak Butuh Perulangan Dalam Membaca Serial:
ketika membaca serial pada arduino maka akan dibutuhkan perulangan untuk bisa membaca setiap data secara berulang, sehingga data-data yang muncul bisa banyak.
Sedangkan jika kita tidak membutuhkan perulangan maka Arduino hanya akan membaca data sekali saja.
7. Bagaimana Cara Membuat Fungsi Yang Menggunakan pyserial.
8. Plagiarisme

8.6 Tia Nur Candida | 1174086

Pemahaman Teori



8.6.1 Soal No. 1

Apa itu fungsi device manager di windows dan folder /dev di linux?

Device manager merupakan perangkat lunak untuk menampilkan seluruh perangkat keras yang di-inisialisasi atau dikenali oleh sistem operasi Windows. Device Manager membantu dalam mengelola atau me-manage semua perangkat keras yang terspasang dan terdeteksi dalam sistem Windows. Perangkat keras tersebut bisa berupa harddisk, kartu VGA, sound, keyboard, perangkat USB dan lain-lainnya.

Fungsi device manager antara lain :

1. Menunjukkan status mengenai suatu perangkat keras.
2. Menunjukkan informasi detail mengenai suatu perangkat keras.
3. Mengelola driver perangkat keras.
4. Menonaktifkan dan mengaktifkan perangkat keras.
5. Mengidentifikasi konflik antar perangkat keras.
6. Memberitahukan terjadinya masalah pada perangkat keras.

Folder /dev merupakan representasi dari drive yang terhubung ke sistem operasi Linux dan oleh sistem dianggap sebagai file-file direktori. Biasanya sering ditampilkan direktori seperti /dev/sda1 yang mewakili Drive SATA pertama dalam sistem.

8.6.2 Soal No. 2

Jelaskan langkah-langkah instalasi driver dari arduino!

Berikut ini adalah langkah-langkah instalasi driver dari Arduino UNO di Windows:

1. Pertama pastikan Arduino IDE telah terinstall.
2. Lalu hubungkan port USB Arduino Uno ke port USB PC.

3. Kemudian PC anda akan mendeteksi perangkat baru yang terpasang dan akan muncul pop seperti ini.
4. Karena Arduino Uno baru pertama kali terpasang, maka akan muncul pop up error seperti ini.
5. Buka "Start" lalu cari Device Manager, kemudian klik "Device Manager".
6. Setelah Device Manager terbuka, silahkan cari "Unknown Device" yang berada di Other Device.
7. Kemudian klik kanan pada "Unknown Device", lalu pilih "Update Driver Software".
8. Setelah itu muncul window baru, lalu pilih "Browse my computer for driver software".
9. Lalu cari folder yang terinstall Arduino IDE dengan mengklik browse. Kemudian klik "Next".
10. Windows akan mencari dan menginstall driver yang berada pada folder tersebut.
11. Setelah itu akan muncul window, lalu klik "Install".
12. Jika berhasil terinstal maka akan muncul window seperti ini.

8.6.3 Soal No. 3

Jelaskan bagaimana cara membaca baudrate dan port dari komputer yang sudah terinstall driver!

Membaca Baudrate dari Komputer

1. Pertama buka "Start". Cari "Device Manager", lalu klik.
2. Kemudian pilih "Ports (COM & LPT)".
3. Klik dua kali pada COM yang terhubung.
4. Pilih tab "Port Settings", lalu lihat di "Bit per second".

Membaca Port dari Komputer

1. Pertama buka "Start". Cari "Device Manager", lalu klik.
2. Kemudian pilih "Ports (COM & LPT)".
3. Port dari Arduino telah terbaca oleh PC.

8.6.4 Soal No. 4

Jelaskan sejarah library pyserial!

PySerial adalah paket Python yang menfasilitasi komunikasi serial antara PC dengan perangkat keras eksternal. PySerial menyediakan antarmuka untuk berkomunikasi melalui protokol komunikasi serial. Komunikasi serial adalah salah satu protokol komunikasi komputer tertua. Protokol komunikasi serial mendahului spesifikasi USB yang digunakan oleh komputer dan perangkat keras lain seperti mouse, keyboard, dan webcam. USB adalah singkatan dari Universal Serial Bus. USB dan dibangun di atas dan memperluas antarmuka komunikasi serial asli.

8.6.5 Soal No. 5

Jelaskan fungsi-fungsi apa saja yang dipakai dari library pyserial!

Fungsi-fungsi yang dipakai dari library PySerial, yaitu:

1. Serial - fungsi ini untuk membuka port serial.
2. write(data) - fungsi ini menulis data lewat port serial.
3. readline() - fungsi ini membaca sebuah string dari port serial.
4. read(size) - fungsi ini untuk membaca jumlah byte dari port serial.
5. close() - fungsi ini untuk menutup port serial.

8.6.6 Soal No. 6

Jelaskan kenapa butuh perulangan dan tidak butuh perulangan dalam membaca serial!

Pada saat membaca serial di Arduino diperlukan perulangan agar bisa membaca data secara berulang kali sehingga data yang muncul banyak. Sedangkan apabila tidak membutuhkan perulangan maka Arduino hanya akan membaca data sekali saja.

8.6.7 Soal No. 7

Jelaskan bagaimana cara membuat fungsi yang menggunakan pyserial!

```

1 import serial
2
3 def testArduino():
4     ser = serial.Serial("COM5", 115200)
5     print(ser.name)
6
7 testArduino()

```

Listing 8.1 Fungsi yang menggunakan pyserial.

8.6.8 Cek Plagiat

8.6.9 Kode Program

8.7 Chandra Kirana Poetra

8.7.1 Pemahaman Teori

1. Apa itu fungsi device manager di windows dan folder /dev di linux

Device manager adalah suatu aplikasi kecil bagian dari control panel yang ada di sistem operasi windows, aplikasi ini membuat user bisa melihat dan mengontrol hardware yang terpasang ke komputer. ketika suatu hardware tidak bekerja, hardware yang terdeteksi akan terdapat notifikasi bagi user untuk ditangani. di deivce manager ini user bisa enable atau disable hardware, melihat informasi technical, dan lain lain. device manager sudah ada di windows sejak versi 95

/dev di linux merupakan lokasi spesial atau tempat penyimpanan file file perangkat. didalamnya anda bisa menemukan hda1 dan hda2 yang merupakan partisi system, cdrom dan fd0 adalah drive CD-ROM dan juga floppy drive, dan lain lain.

2. Jelaskan langkah-langkah instalasi driver dari arduino

Download dari halaman sini dulu <https://www.arduino.cc/en/main/software> lalu ikuti saja langkah nya, langkah dibawah ini adalah langkah alternatif jika terjadi error, langkah ini akan menginstall driver arduino dari device manager

- Buka driver manager
- Buka bagian port
- Klik Kanan lalu update driver
- pilih Browse my computer
- Cari tempat driver berada, jika anda sebelumnya sudah menginstall arduino ide maka driver ada di folder instalasi arduino
- Klik next saja lalu pilih install
- Driver selesai terinstall
- Cek hasil instalasi

3. Jelaskan bagaimana cara membaca baudrate dan port dari komputer yang sudah terinstall driver

Baudrate (bits per second) adalah suatu takaran yang digunakan untuk mengukur informasi yang akan di sampaikan atau dikirim dan cara membacanya sendiri adalah dengan cara membuka aplikasi arduino IDEnya lalu mengklik icon serial monitor. Port sendiri bisa kita lihat di bagian device manager pada opsi port, tinggal dibuka saja dropdownnya maka akan tampil port number berapanya

4. Jelaskan sejarah library pyserial

pySerial adalah modul yang digunakan untuk mengakses port, menyediakan backend untuk python pada platform windwos, OSX, linux, BSD dan lain lain,

5. Jelaskan fungsi-fungsi apa saja yang dipakai dari library pyserial

- Serial()

Berfungsi untuk mendefinisikan port, diikuti dengan parameter lainnya seperti baudrate, bytesize dan lain lain

- open()

Berfungsi untuk membuka port

- close()

Berfungsi untuk Menutup port

- write(data)

Berfungsi untuk menulis data berdasarkan parameter

- flush()

Operasi Flush memaksa data file untuk dikirim ke file cache di ram dan disana akan dialihkan ke sistem operasi untuk dikirimkan ke disk datayang akan di simpan

6. Jelaskan kenapa butuh perulangan dan tidak butuh perulangan dalam membaca serial

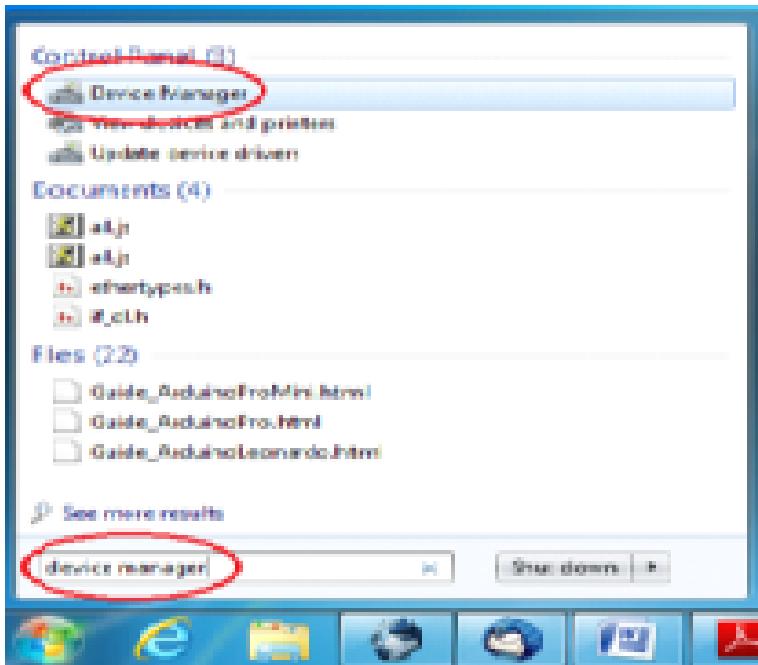
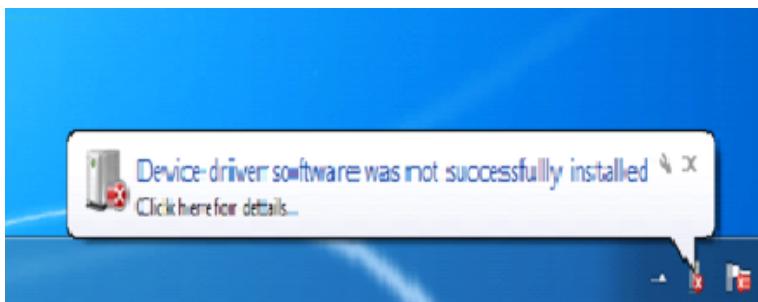
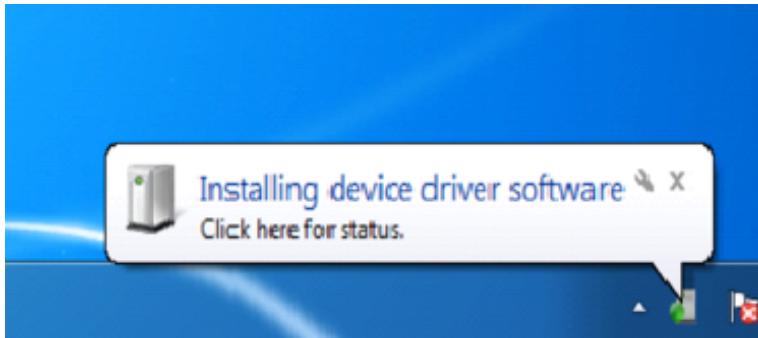
Perulangan sendiri merupakan kode/perintah yang digunakan untuk melakukan suatu kegiatan/tujuan yang sama secara berulang ulang. for dan while adalah perulangan dalam python. For dan while merupakan perulangan namun memiliki fungsi yang berbeda, for menggunakan perhitungan dalam penggunaannya jadi kode akan dieksekusi sesuai dengan jumlah yang telah ditentukan, sedangkan while akan terus mengulang perintah-perintahnya ketika kondisi masih sesuai dengan apa yang ditulis di kondisi Mengapa diperlukan perulangan? Karena kode program akan menjadi lebih sederhana dibandingkan dengan cara manual satu persatu dan juga lebih efficient dengan meminimalisir baris kode

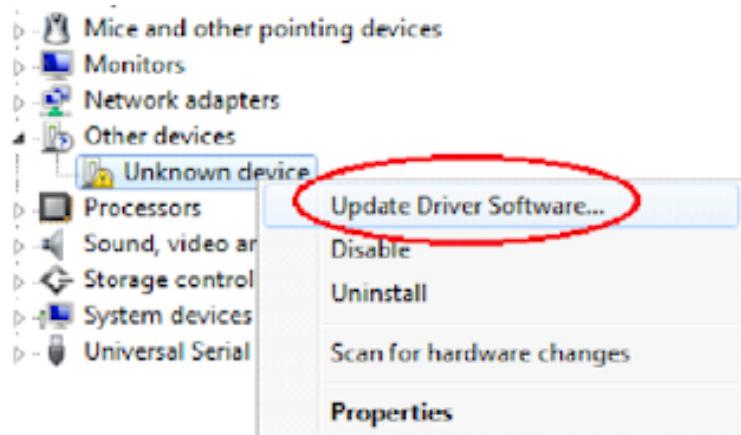
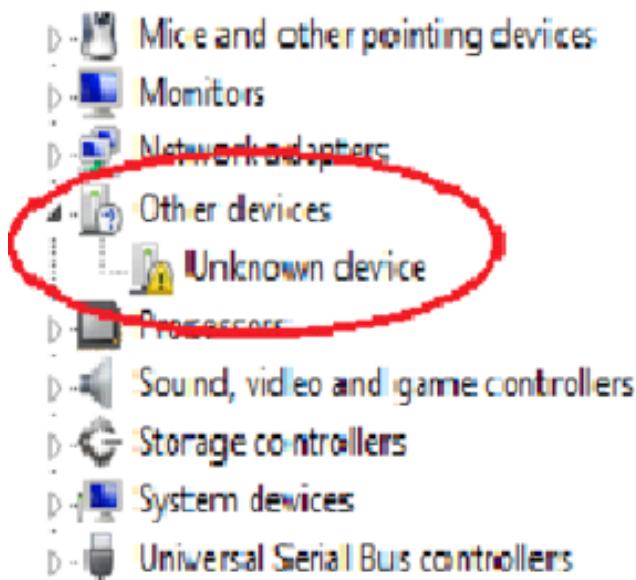
7. Jelaskan bagaimana cara membuat fungsi yang menggunakan pyserial

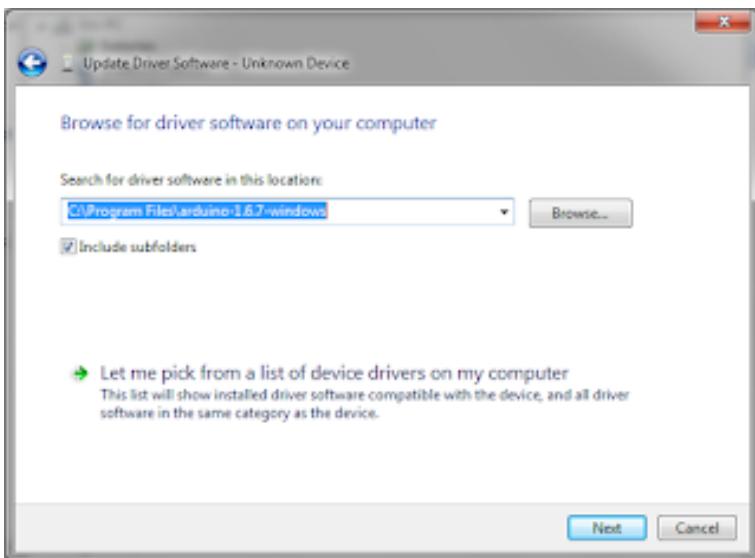
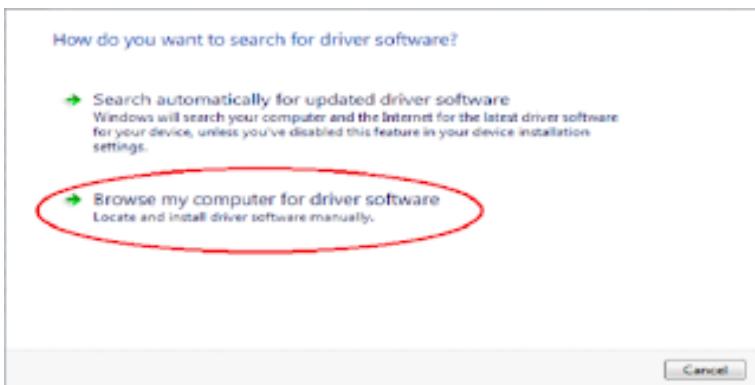
Membuat fungsi sama seperti sebelumnya namun method dari pyserial dimasukkan kedalam fungsi dan dipanggil fungsi yang kita buat tadi

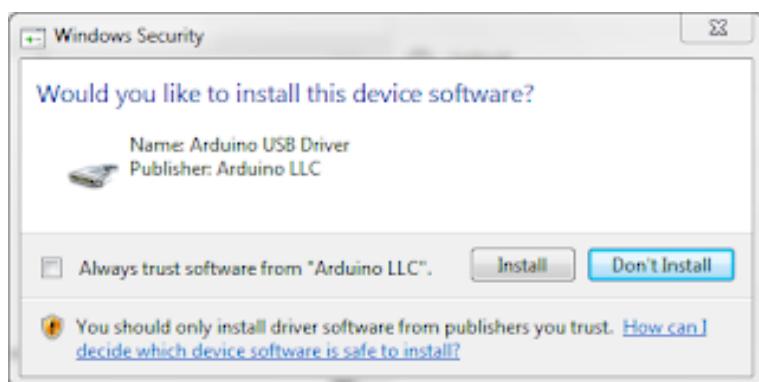
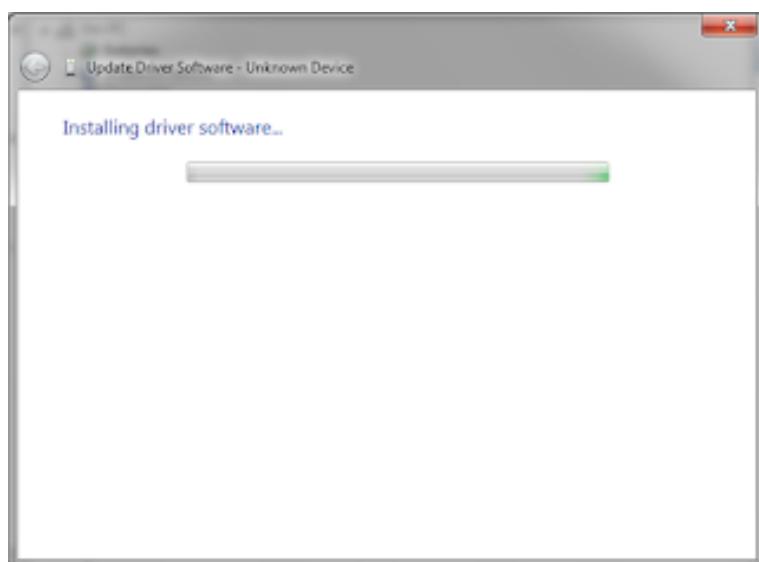
```
1 import serial
2
3 def bacaserialarduino():
4     variableserial = serial.Serial("COM1", 9600)
5     baca = variableserial.readline()
6     print(baca)
7
8 bacaserialarduino()
```

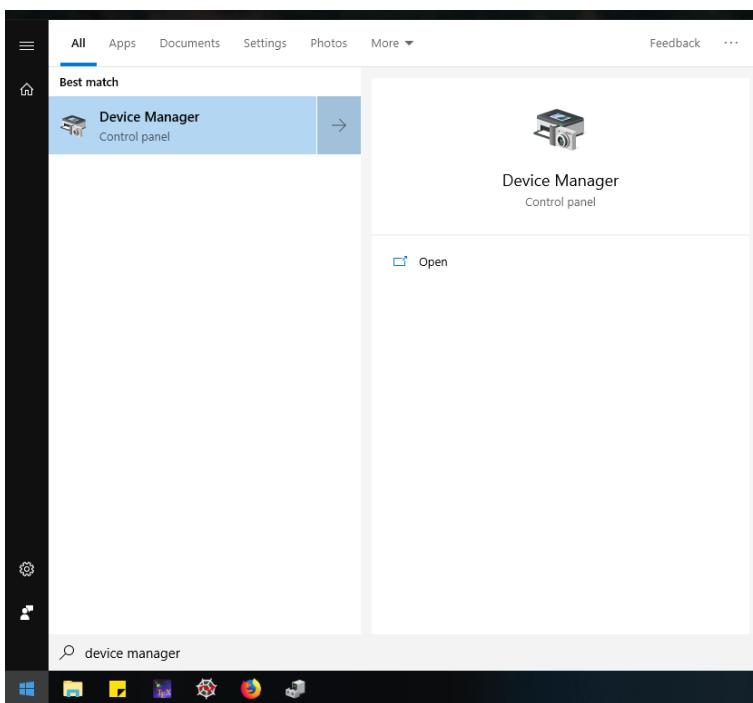
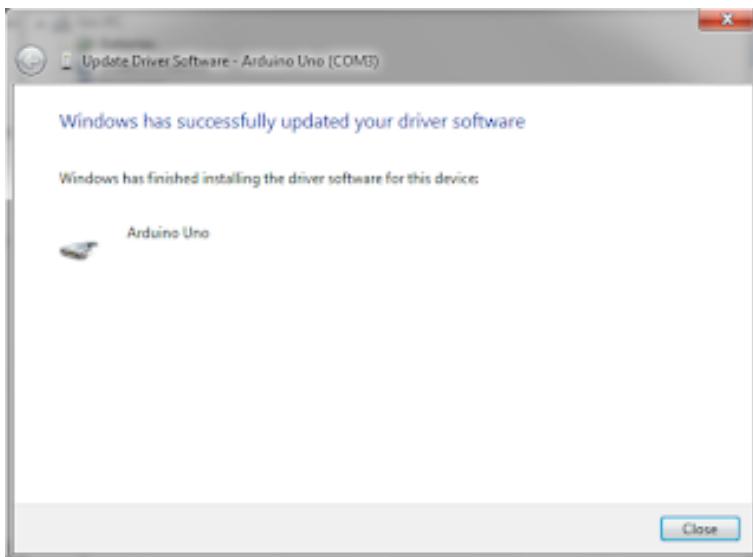
8. Scan Plagiarisme

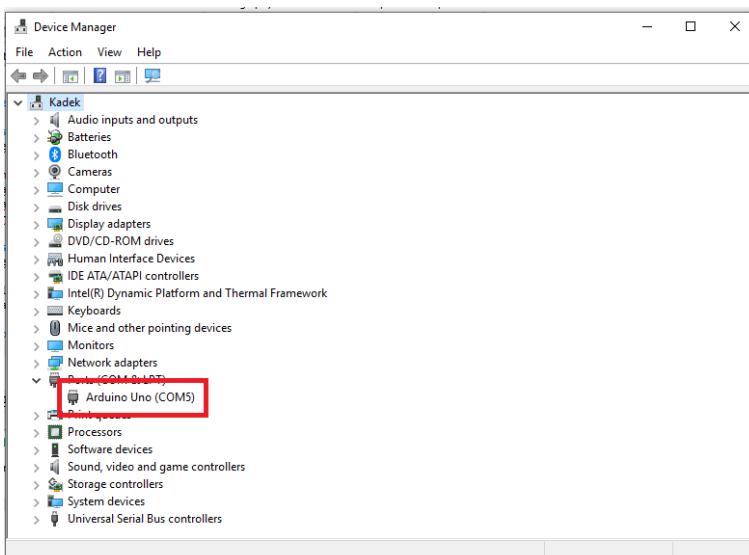
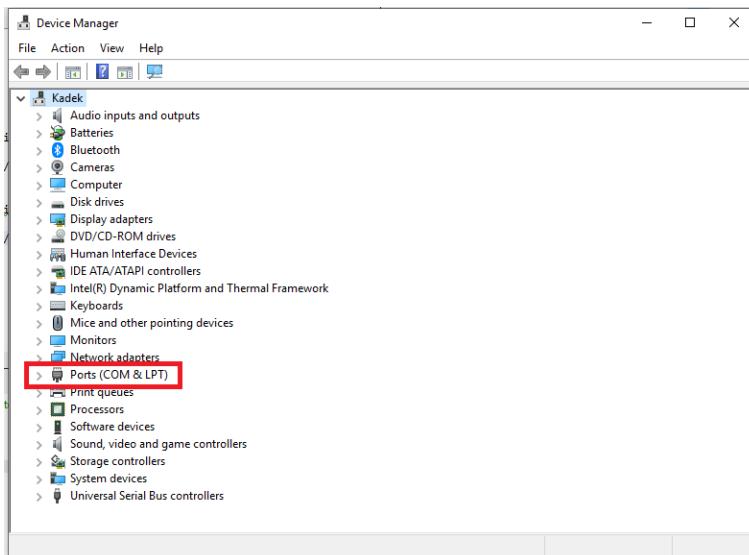


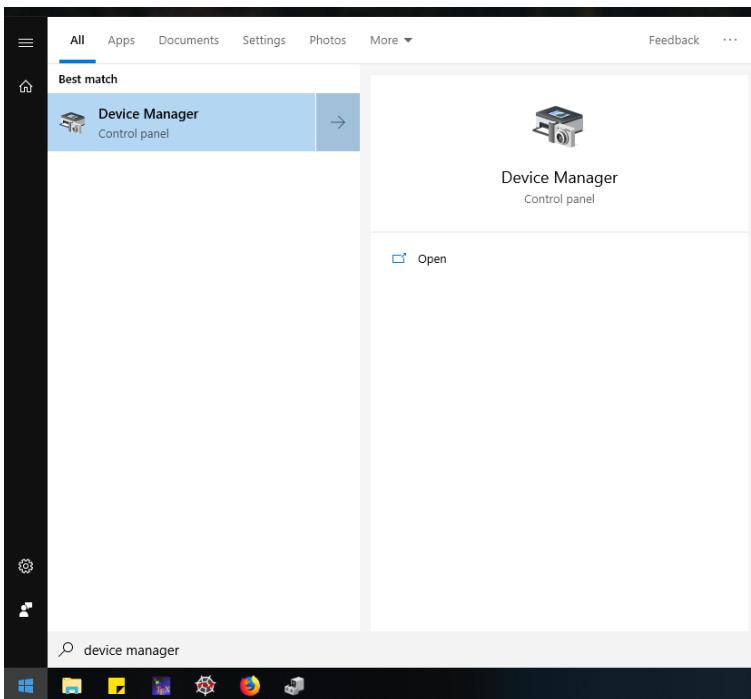
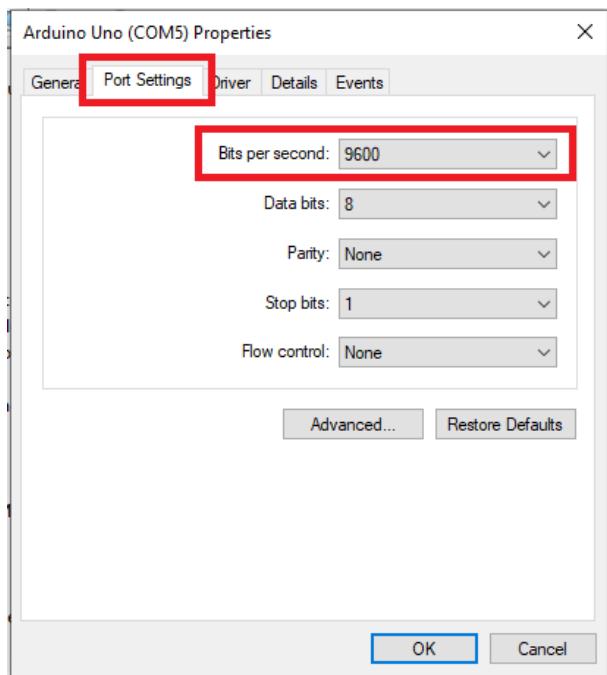


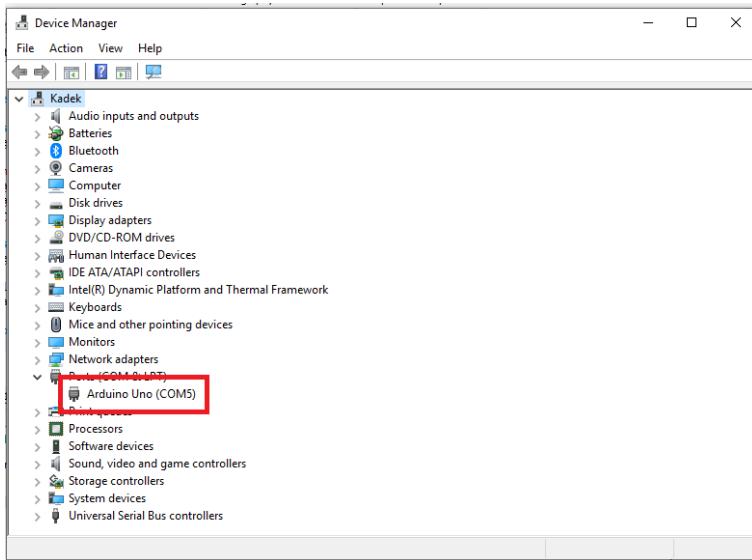
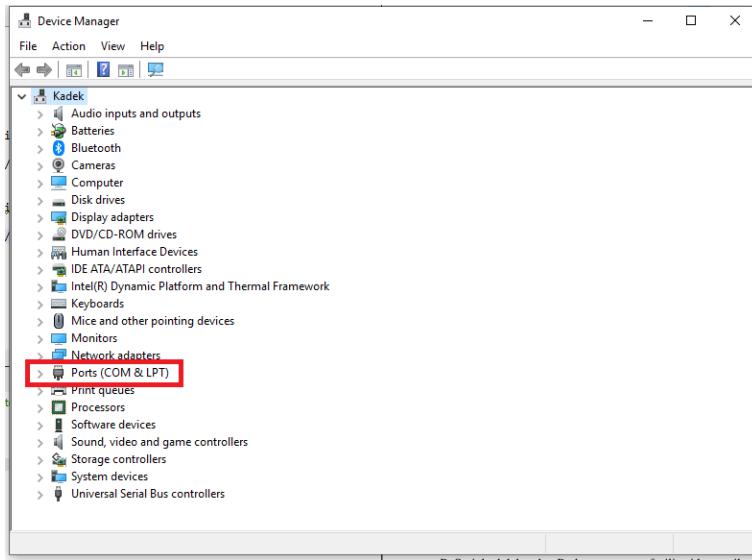






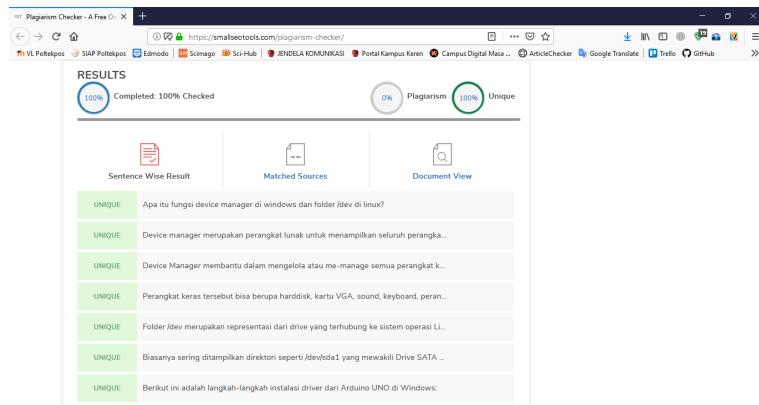




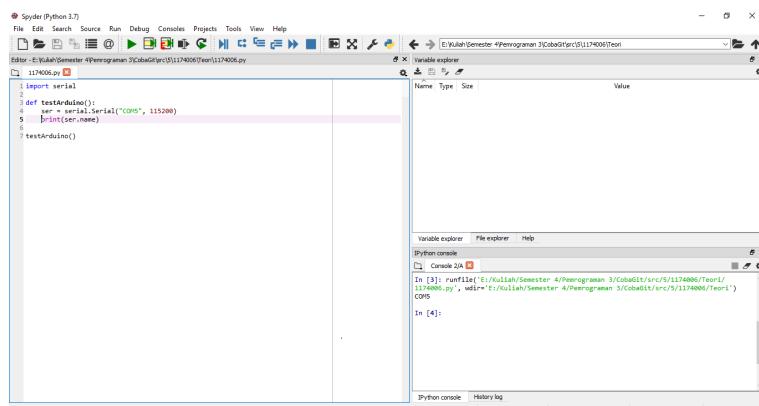


```
Console 2/A
In [3]: runfile('E:/Kuliah/Semester 4/Pemrograman 3/CobaGit/src/5/1174006/Teori/1174006.py', wdir='E:/Kuliah/Semester 4/Pemrograman 3/CobaGit/src/5/1174006/Teori')
COM5
```

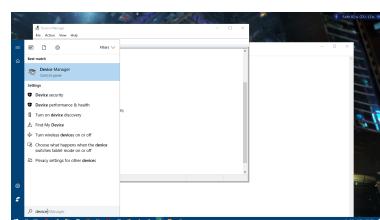
Gambar 8.32 Hasil pembuatan fungsi pyserial.



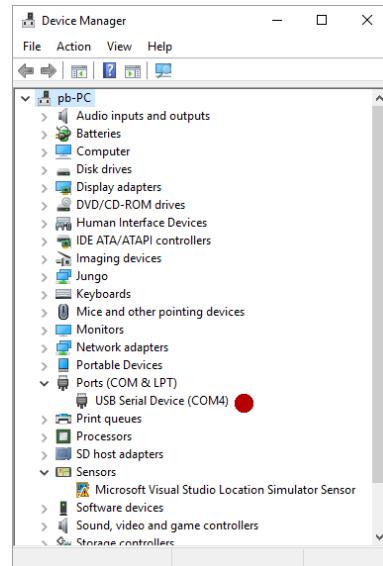
Gambar 8.33 Hasil cek plagiat.



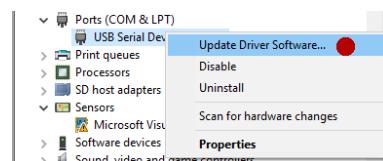
Gambar 8.34 Kode program file 1174086.py.



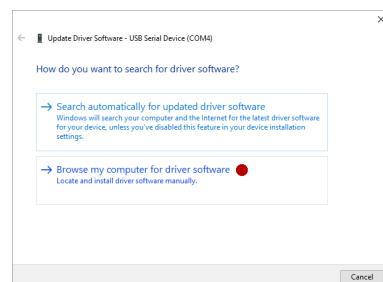
Gambar 8.35 Driver Manager



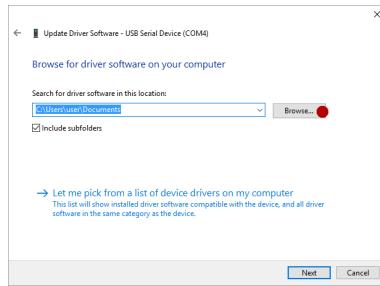
Gambar 8.36 Port



Gambar 8.37 Instalasi



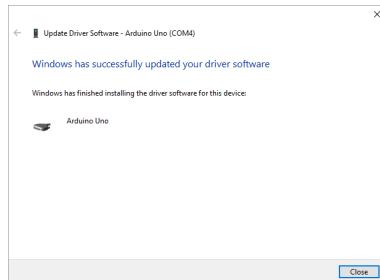
Gambar 8.38 browse my computer



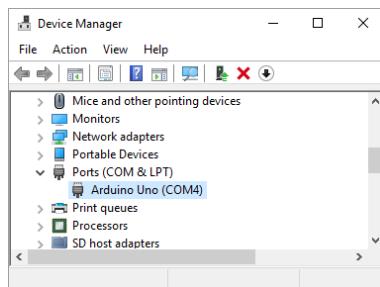
Gambar 8.39 Driver Arduino



Gambar 8.40 Install driver



Gambar 8.41 Selesai



Gambar 8.42 Result



Gambar 8.43 Plagiarisme

BAB 9

PRAKTEK KOMUNIKASI PERANGKAT KERAS

9.1 D. Irga B. Naufal Fakhri

Kode Arduino untuk pembacaan temprature dan humidity menggunakan sensor DHT1:

```
1 #include <dht.h>
2
3 dht DHT;
4
5 #define DHT11_PIN 6
6
7 void setup(){
8     Serial.begin(9600);
9     Serial.println("Ready");
10 }
11
12 void loop()
13 {
14     int chk = DHT.read11(DHT11_PIN);
15     Serial.print("Temperature = ");
16     Serial.println(DHT.temperature);
17     Serial.print("Humidity = ");
18     Serial.println(DHT.humidity);
```

```

19     delay(2000);
20 }
```

9.1.1 Soal 1

Buatlah fungsi (file terpisah/library dengan nama NPM realtime.py) untuk mendapatkan data langsung dari arduino

```

1 import serial
2
3 def ReadSerial():
4     ser = serial.Serial("COM3", 9600)
5     read = ser.readline().decode("utf-8").strip("\n").strip("\r")
6     print(read)
7
8 ReadSerial()
```

Gambar 9.1 Membaca Serial tanpa loop

9.1.2 Soal 2

Buatlah fungsi (file terpisah/library dengan nama NPM save.py) untuk mendapatkan data langsung dari arduino dengan looping

```

1 import serial
2
3 def ReadSerialLoop():
4     ser = serial.Serial("COM3", 9600)
5     while (1):
6         read = ser.readline().decode("utf-8").strip("\n").strip("\r")
7         print(read)
8
9 ReadSerialLoop()
```

Gambar 9.2 Membaca Serial dengan loop

9.1.3 Soal 3

Buatlah fungsi (file terpisah/library dengan nama NPM realtime.py) untuk mendapatkan data dari arduino dan langsung ditulis kedalam file csv

```

1 import csv
2 def write():
```

```

3   with open("1174066.csv", mode='w') as nama_file:
4       tulis_file = csv.writer(nama_file, delimiter=',', quotechar='''',
5           quoting=csv.QUOTE_MINIMAL)
6       ser = serial.Serial("COM3", 9600)
7       read = ser.readline().decode("utf-8").strip('\n').strip('\'r')
8       tulis_file.writerow([read])
9   write()

```

Listing 9.1 "Kode python"

```

1 Ready
2
3 Temperature = 30
4
5 Humidity = 66
6
7 Temperature = 30
8
9 Humidity = 66
10
11 Temperature = 30
12
13 Humidity = 66
14
15 Temperature = 30
16
17 Humidity = 66

```

Listing 9.2 "Data yang telah ditulis ke file csv"

9.1.4 Soal 4

Buatlah fungsi (file terpisah/library dengan nama NPM csv.py) untuk membaca file csv hasil arduino dan mengembalikan ke fungsi

```

1 import csv
2
3 def read(namafайл):
4     with open(namafайл, mode='r') as csv_file:
5         reader = csv.DictReader(csv_file)
6         for row in reader:
7             print(row['Ready'])
8
9 read("1174066.csv")

```

9.1.5 Ketrampilan Penanganan Error

Tuliskan peringatan error yang didapat dari mengerjakan praktik ketiga ini, dan jelaskan cara penanganan error tersebut. dan Buatlah satu fungsi yang menggunakan gunakan try except untuk menanggulangi error tersebut.

- Syntax Errors

Syntax Errors adalah kesalahan pada penulisan syntax atau kode. Solusinya adalah memperbaiki penulisan syntax atau kode

- Zero Division Error

ZeroDivisionError adalah exceptions yang terjadi saat eksekusi program menghasilkan perhitungan matematika pembagian dengan angka nol (0). Solusinya adalah tidak membagi suatu yang hasilnya nol.

- Name Error

NameError adalah exception saat kode melakukan eksekusi terhadap local name atau global name yang tidak terdefinisi atau tidak ada. Solusinya adalah memastikan variabel atau function yang akan dipanggil ada didalam program atau tidak salah mengetikannya.

- Type Error

TypeError adalah exception saat melakukan eksekusi terhadap suatu operasi atau fungsi dengan type object yang tidak sesuai. Solusinya adalah mengkoversi variabelnya sesuai dengan tipe data sesuai dengan yang akan digunakan.

```

1 import serial
2
3 def PenangananError():
4     try:
5         ser = serial.Serial('COM3', 9600)
6         print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
7     except SyntaxError:
8         print("Ada kesalahan dalam penulisan Syntax")
9     except NameError:
10        print("Variable yang dimasukkan tidak ada")
11    except TypeError:
12        print("Ada yang salah pada type data")
13    except:
14        print("Sedang terjadi sebuah kesalahan")
15
16 PenangananError()

```

9.2 Tia Nur Candida

Ketrampilan Pemrograman

9.2.1 No. 1

Buatlah fungsi (file terpisah/library dengan nama NPMrealtime.py) untuk mendapatkan data langsung dari arduino!

```

1 import serial
2
3 def getData():

```

```

4     ser = serial.Serial('COM5',9600)
5     print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
6
7 getData()

```

Listing 9.3 Fungsi untuk mendapatkan data dari Arduino.

9.2.2 No. 2

Buatlah fungsi (file terpisah/library dengan nama NPMSave.py) untuk mendapatkan data langsung dari arduino dengan looping!

```

1 import serial
2
3 def getDataLoop():
4     ser = serial.Serial('COM5',9600)
5     while (1):
6         print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
7
8 getDataLoop()

```

Listing 9.4 Fungsi untuk mendapatkan data langsung dari Arduino dengan looping.

9.2.3 No. 3

Buatlah fungsi (file terpisah/library dengan nama NPMrealtime.py) untuk mendapatkan data dari arduino dan langsung ditulis kedalam file csv!

```

1 import serial
2 import csv
3
4 def writeCsv():
5     ser = serial.Serial('COM5',9600)
6     with open('praktek.csv', mode='w') as csv_file:
7         fieldnames = ['jarak']
8         writer = csv.DictWriter(csv_file, fieldnames=fieldnames)
9
10        writer.writeheader()
11        while (1):
12            data = ser.readline().decode("utf-8").strip('\n').strip('\r')
13            writer.writerow({'jarak': data})
14
15 writeCsv()

```

Listing 9.5 Fungsi untuk mendapatkan data dari Arduino dan langsung ditulis kedalam file CSV.

9.2.4 No. 4

Buatlah fungsi (file terpisah/library dengan nama NPMcsv.py) untuk membaca file csv hasil arduino dan mengembalikan ke fungsi!

```

1 import csv
2
3 def readCsv():
4     with open('praktek.csv', mode='r') as csv_file:
5         reader = csv.DictReader(csv_file)
6         for row in reader:
7             print(row['jarak'])
8
9 readCsv()

```

Listing 9.6 Fungsi untuk membaca file CSV hasil Arduino dan mengembalikan fungsi.

9.2.5 Kode Program Praktek

9.2.6 Cek Plagiat Praktek

Ketrampilan Penanganan Error

9.2.7 No. 1

Tuliskan peringatan error yang didapat dari mengerjakan praktek kelima ini, dan jelaskan cara penanganan error tersebut. dan Buatlah satu fungsi yang menggunakan try except untuk menanggulangi error tersebut.

Peringatan error di praktek kelima ini, yaitu:

- Syntax Errors Syntax Errors adalah suatu keadaan saat kode python mengalami kesalahan penulisan. Solusinya adalah memperbaiki penulisan kode yang salah.
- Name Error NameError adalah exception yang terjadi saat kode melakukan eksekusi terhadap local name atau global name yang tidak terdefinisi. Solusinya adalah memastikan variabel atau function yang dipanggil ada atau tidak salah ketik.
- Type Error TypeError adalah exception yang akan terjadi apabila pada saat dilakukannya eksekusi terhadap suatu operasi atau fungsi dengan type object yang tidak sesuai. Solusi dari error ini adalah mengkoversi varibelnya sesuai dengan tipe data yang akan digunakan.

Fungsi yang menggunakan try except untuk menanggulangi error.

```

1 import serial
2
3 def tryExceptError():
4     try:
5         ser = serial.Serial('COM5', 9600)
6         print(sre.readline().decode("utf-8").strip('\n').strip('\r'))
7     except SyntaxError:

```

```

8     print("Kesalahan penulisan syntax")
9 except NameError:
10    print("Variable tersebut tidak ada")
11 except TypeError:
12    print("Tipe data salah")
13 except:
14    print("Terjadi sebuah kesalahan")
15
16 tryExceptError()

```

Listing 9.7 Fungsi untuk menanggulangi error menggunakan Try Except.

9.2.8 Kode Program Penanganan Error

9.2.9 Cek Plagiat Penanganan Error

9.3 Fanny Shafira Damayanti | 1174069

9.3.1 Keterampilan Pemrograman

1. Fungsi untuk mendapatkan data langsung dari arduino

```

1 import serial
2
3 def getData():
4     ser = serial.Serial('COM5',9600)
5     print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
6
7 getData()
8
9 import serial
10 import csv
11
12 def writeCsv():
13     ser = serial.Serial('COM5',9600)
14     with open('praktek.csv', mode='w') as csv_file:
15         fieldnames = ['jarak']
16         writer = csv.DictWriter(csv_file, fieldnames=fieldnames)
17
18         writer.writeheader()
19         while (1):
20             data = ser.readline().decode("utf-8").strip('\n').
21             strip('\r')
22             writer.writerow({'jarak': data})
23
24 writeCsv()

```

2. Fungsi untuk mendapatkan data langsung dari arduino dengan looping

```

1 import serial
2
3 def getDataLoop():
4     ser = serial.Serial('COM5',9600)
5     while (1):

```

```

6     print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
7
8 getDataLoop()

```

3. Fungsi untuk mendapatkan data dari arduino dan langsung ditulis kedalam file csv

```

1 import serial
2 import csv
3
4 def writeCsv():
5     ser = serial.Serial('COM5', 9600)
6     with open('praktek.csv', mode='w') as csv_file:
7         fieldnames = ['jarak']
8         writer = csv.DictWriter(csv_file, fieldnames=fieldnames)
9
10        writer.writeheader()
11        while (1):
12            data = ser.readline().decode("utf-8").strip('\n').
13            strip('\r')
14            writer.writerow({'jarak': data})
15
16 writeCsv()

```

4. Fungsi untuk membaca file csv hasil arduino dan mengembalikan ke fungsi

```

1 import csv
2
3 def readCsv():
4     with open('praktek.csv', mode='r') as csv_file:
5         reader = csv.DictReader(csv_file)
6         for row in reader:
7             print(row['jarak'])
8
9 readCsv()

```

9.3.2 Keterampilan Penanganan Error

1. Peringatan error yang ada di praktek kali ini, dan fungsi try except untuk menanggulangi error tersebut

- Syntax Errors Syntax Errors adalah suatu keadaan saat kode python mengalami kesalahan penulisan. Solusinya adalah memperbaiki penulisan kode yang salah.
- Name Error NameError adalah exception yang terjadi saat kode melakukan eksekusi terhadap local name atau global name yang tidak terdefinisi. Solusinya adalah memastikan variabel atau function yang dipanggil ada atau tidak salah ketik.

- Type Error TypeError adalah exception yang akan terjadi apabila pada saat dilakukannya eksekusi terhadap suatu operasi atau fungsi dengan type object yang tidak sesuai. Solusi dari error ini adalah mengkoversi varibelnya sesuai dengan tipe data yang akan digunakan.

```

1 import serial
2
3 def tryExceptError():
4     try:
5         ser = serial.Serial('COM5', 9600)
6         print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
7     except SyntaxError:
8         print("Kesalahan penulisan syntax")
9     except NameError:
10        print("Variable tersebut tidak ada")
11    except TypeError:
12        print("Tipe data salah")
13    except:
14        print("Terjadi sebuah kesalahan")
15
16 tryExceptError()

```

9.4 Aulyardha Anindita | 1174054

9.4.1 Keterampilan Pemrograman

1. Jawaban Soal No. 1

```

1 import serial
2
3 def getData():
4     ser = serial.Serial('COM5', 9600)
5     print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
6
7 getData()

```

2. Jawaban Soal No. 2

```

1 import serial
2
3 def getDataLoop():
4     ser = serial.Serial('COM5', 9600)
5     while (1):
6         print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
7
8 getDataLoop()

```

3. Jawaban Soal No. 3

```

1 import serial
2 import csv
3
4 def writeCsv():
5     ser = serial.Serial('COM5', 9600)
6     with open('praktek.csv', mode='w') as csv_file:
7         fieldnames = ['jarak']
8         writer = csv.DictWriter(csv_file, fieldnames=fieldnames)
9
10        writer.writeheader()
11        while (1):
12            data = ser.readline().decode("utf-8").strip('\n').
13            strip('\r')
14            writer.writerow({'jarak': data})
15
16 writeCsv()

```

4. Jawaban Soal No. 4

```

1 import csv
2
3 def readCsv():
4     with open('praktek.csv', mode='r') as csv_file:
5         reader = csv.DictReader(csv_file)
6         for row in reader:
7             print(row['jarak'])
8
9 readCsv()

```

9.4.2 Keterampilan Penanganan Error

Peringatan error di praktek keempat ini, yaitu:

- Syntax Errors Syntax Errors adalah keadaan dimana pada kode python mngalami kesalahan dalam penulisan. Untuk mengatasinya yaitu dengan memperbaiki penulisan kode yang salah
- Name Error NameError adalah suatu keadaan atau exception yang terjadi ketika kode melakukan eksekusi terhadap local name atau global name yang tidak terdefinisi. Untuk mengatasinya yaitu dengan memastikan variabel atau function yang dipanggil ada atau tidak salah ketik.
- Type Error TypeError adalah suatu keadaan atau exception yang akan terjadi apabila pada saat dilakukannya eksekusi terhadap suatu operasi atau fungsi dengan type object yang tidak sesuai. Untuk mengatasinya yaitu dengan mengoversi varibelnya sesuai dengan tipe data yang akan digunakan.

Fungsi yang menggunakan try except

```

1 import serial
2
3 def tryExceptError():

```

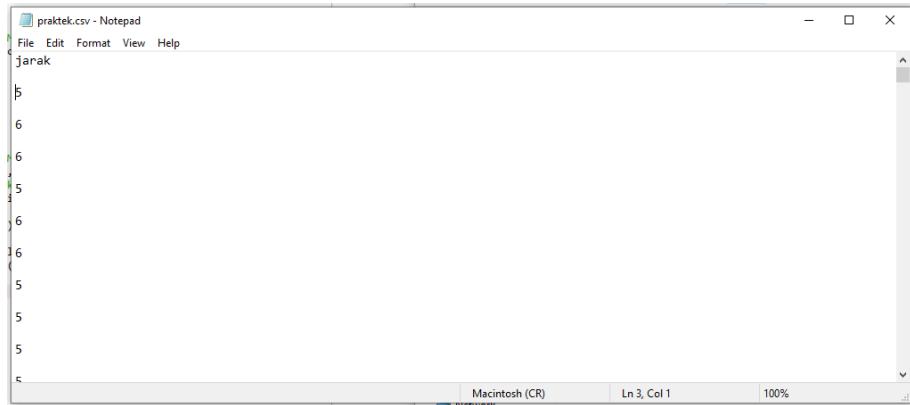
```
4     try :
5         ser = serial.Serial('COM5',9600)
6         print(sre.readline().decode("utf-8").strip('\n').strip('\r'))
7     except SyntaxError:
8         print("Kesalahan penulisan syntax")
9     except NameError:
10        print("Variable tersebut tidak ada")
11    except TypeError:
12        print("Tipe data salah")
13    except:
14        print("Terjadi sebuah kesalahan")
15
16 tryExceptError()
```

```
IPython console
Console 1/A X
In [6]: runfile('E:/Kuliah/Semester 4/Pemrograman 3/FixGit/src/5/1174006/Praktek/1174006realtime.py', wdir='E:/Kuliah/Semester 4/Pemrograman 3/FixGit/src/5/1174006/Praktek')
241
```

Gambar 9.3 Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino.

```
IPython console
Console 1/A x
In [7]: runfile('E:/Kuliah/Semester 4/Pemrograman 3/FixGit/src/5/1174006/Praktek/1174006save.py', wdir='E:/Kuliah/Semester 4/Pemrograman 3/FixGit/src/5/1174006/Praktek')
241
198
240
199
239
200
239
200
239
200
239
200
239
```

Gambar 9.4 Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino dengan looping.



Gambar 9.5 Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino dan langsung ditulis kedalam file CSV.

The screenshot shows the IPython console interface. The title bar says "IPython console". A tab labeled "Console 1/A" is active. Below it, the code cell content is displayed:

```
In [12]: runfile('E:/Kuliah/Semester 4/Pemrograman 3/FixGit/src/5/1174006/Praktek/1174006csv.py', wdir='E:/Kuliah/Semester 4/Pemrograman 3/FixGit/src/5/1174006/Praktek')
5
6
6
6
5
5
5
5
5
5
5
```

Gambar 9.6 Hasil dari pembacaan fungsi untuk membaca file csv hasil arduino dan mengembalikan fungsi.

The screenshot shows three code cells in a Jupyter Notebook. The first cell contains the following code:

```
1 import serial
2
3 def getdata():
4     ser = serial.Serial('COM5',9600)
5     print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
6
7 getdata()
```

The second cell contains:

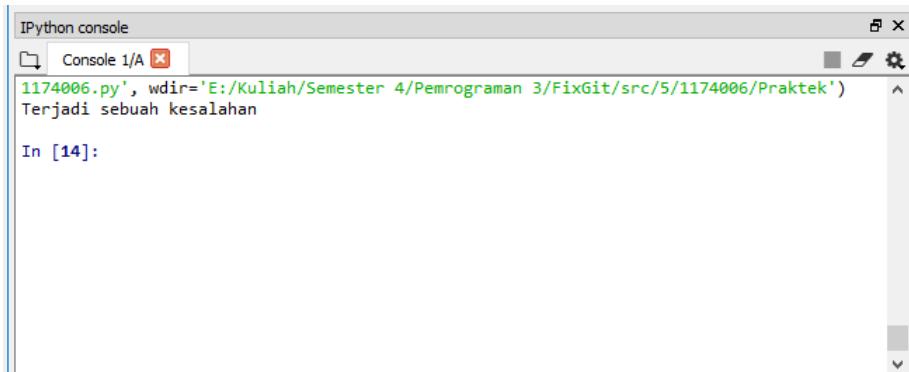
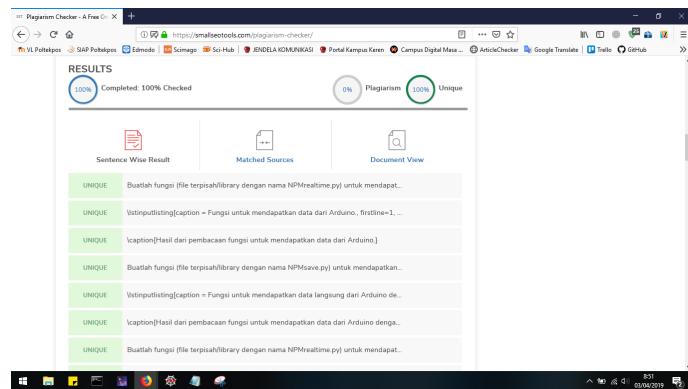
```
1 import serial
2
3 def writeCSV():
4     ser = serial.Serial('COM5',9600)
5     with open('praktek.csv', mode='w') as csv_file:
6         fieldnames = ['jarak']
7         writer = csv.DictWriter(csv_file, fieldnames=fieldnames)
8
9         writer.writeheader()
10        writer.writerow({'jarak': 0})
11        data = ser.readline().decode("utf-8").strip("\n").strip("\r")
12        writer.writerow({'jarak': data})
13    writeCSV()
```

The third cell contains:

```
1 import serial
2
3 def getDataLoop():
4     ser = serial.Serial('COM5',9600)
5     while (1):
6         print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
7
8 getDataLoop()
9
```

The fourth cell contains:

```
1 import csv
2
3 def readCsv():
4     with open('praktek.csv', mode='r') as csv_file:
5         reader = csv.DictReader(csv_file)
6         for row in reader:
7             print(row['jarak'])
8
9 readCsv()
10
```



Gambar 9.7 Hasil pembacaan fungsi untuk menanggulangi error menggunakan Try Except.

The Jupyter Notebook code cell contains the following Python code:

```
1 import serial
2
3 def tryExceptError():
4     try:
5         ser = serial.Serial('COM5',9600)
6         print(sre.readline().decode('utf-8').strip('\n').strip('\r'))
7     except SyntaxError:
8         print("ada kesalahan penulisan syntax")
9     except NameError:
10        print("Variable tersebut tidak ada")
11    except TypeError:
12        print("tipe data salah")
13    except:
14        print("Terjadi sebuah kesalahan")
15
16 tryExceptError()
17
```

The variable explorer shows the following variables:

Name	Type	Size	Value
(no variables)			

The IPython console output is as follows:

```
1174006.py', wdir='E:/Kuliah/Semester 4/Pemrograman 3/FixGit/src/5/1174006/Praktek')
Terjadi sebuah kesalahan
```

RESULTS

100% Completed: 100% Checked

0% Plagiarism 100% Unique

Sentence Wise Result

Matched Sources

Document View

UNIQUE	Tuliskan peringatan error yang didapat dari mengerjakan praktik kelima ini, dan jel...
UNIQUE	dan Buatlah satu fungsi yang menggunakan try except untuk menanggulangi error ...
UNIQUE	Syntax Errors adalah suatu keadaan saat kode python mengalami kesalahan penuli...
UNIQUE	NameError adalah exception yang terjadi saat kode melakukan eksekusi terhadap l...
UNIQUE	Solusinya adalah memastikan variabel atau function yang dipanggil ada atau tidak ...
UNIQUE	TypeError adalah exception yang akan terjadi apabila pada saat dilakukannya eksek...
UNIQUE	Solusi dari error ini adalah mengkoversi variabelnya sesuai dengan tipe data yang ak...

