

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 Judul Bagian Pertama	1
2 Judul Bagian Pertama	5
3 Judul Bagian Pertama	17
4 Judul Bagian Pertama	25
5 Judul Bagian Pertama	31
6 Judul Bagian Pertama	37
7 Library CSV dan Pandas	47
8 Praktek Library CSV dan Pandas	65
9 Komunikasi Perangkat Keras	79

DAFTAR ISI

Daftar Gambar	xvii
Daftar Tabel	xxi
Foreword	xxix
Kata Pengantar	xxxii
Acknowledgments	xxxiii
Acronyms	xxxv
Glossary	xxxvii
List of Symbols	xxxix
Introduction	xli
<i>Rolly Maulana Awangga, S.T., M.T.</i>	
1 Judul Bagian Pertama	1
1.1 Arjun Yuda Firwanda	1
1.1.1 Soal 1	1
1.2 Dwi Yulianingsih	1
1.2.1 Soal 1	1
	ix

1.3	Harun Ar-Rasyid	2
1.3.1	Soal 1	2
1.4	Sri Rahayu	2
1.4.1	Soal 1	2
1.5	Doli Jonviter	2
1.5.1	Soal 1	2
1.6	Rahmatul Ridha	2
1.6.1	Sejarah Python	2
1.7	Tomy Prawoto	3
1.7.1	Sejarah Python	3
2	Judul Bagian Pertama	5
2.1	Arjun Yuda Firwanda	5
2.1.1	Soal 1	5
2.2	Dwi Yulianingsih	5
2.2.1	Soal 1	5
2.3	Harun Ar-Rasyid	6
2.3.1	Soal 1	6
2.4	Sri Rahayu	6
2.4.1	Soal 1	6
2.5	Doli Jonviter	6
2.5.1	Soal 1	6
2.6	Rahmatul Ridha	6
2.6.1	Soal 1	6
2.6.2	Instalasi Anaconda	6
2.7	Tomy Prawoto	8
2.7.1	Soal 1	8
2.7.2	Instalasi Anaconda	8
3	Judul Bagian Pertama	17
3.1	Arjun Yuda Firwanda	17
3.1.1	Soal 1	17
3.2	Dwi Yulianingsih	17
3.2.1	Soal 1	17
3.3	Harun Ar-Rasyid	18
3.3.1	Soal 1	18
3.4	Sri Rahayu	18
3.4.1	Soal 1	18

3.5	Doli Jonviter	18
3.5.1	Soal 1	18
3.6	Rahmatul Ridha	18
3.6.1	Teori	18
3.7	Tomy Prawoto	21
3.7.1	Soal 1	21
3.7.2	Teori	21
4	Judul Bagian Pertama	25
4.1	Arjun Yuda Firwanda	25
4.1.1	Soal 1	25
4.2	Dwi Yulianingsih	25
4.2.1	Soal 1	25
4.3	Harun Ar-Rasyid	26
4.3.1	Soal 1	26
4.4	Sri Rahayu	26
4.4.1	Soal 1	26
4.5	Doli Jonviter	26
4.5.1	Soal 1	26
4.6	Rahmatul Ridha	26
4.6.1	praktek	26
4.6.2	Keterangan dan Penanganan eror	28
4.7	Tomy Prawoto	28
4.7.1	Soal 1	28
4.7.2	praktek	28
4.7.3	Keterangan dan Penanganan eror	30
5	Judul Bagian Pertama	31
5.1	Arjun Yuda Firwanda	31
5.1.1	Soal 1	31
5.2	Dwi Yulianingsih	31
5.2.1	Soal 1	31
5.3	Harun Ar-Rasyid	32
5.3.1	Soal 1	32
5.4	Sri Rahayu	32
5.4.1	Soal 1	32
5.5	Doli Jonviter	32
5.5.1	Soal 1	32

5.6	Rahmatul Ridha	32
5.7	Tomy Prawoto	34
5.7.1	Soal 1	34
6	Judul Bagian Pertama	37
6.1	Arjun Yuda Firwanda	37
6.1.1	Soal 1	37
6.2	Dwi Yulianingsih	37
6.2.1	Soal 1	37
6.3	Harun Ar-Rasyid	38
6.3.1	Soal 1	38
6.4	Sri Rahayu	38
6.4.1	Soal 1	38
6.5	Doli Jonviter	38
6.5.1	Soal 1	38
6.6	Rahmatul Ridha	38
6.7	Tomy Prawoto	42
6.7.1	Soal 1	42
7	Library CSV dan Pandas	47
7.1	Arjun Yuda Firwanda	47
7.1.1	Fungsi File CSV, Sejarah dan Contoh	47
7.1.2	Aplikasi apa saja yang dapat menciptakan file csv	48
7.1.3	Cara Menulis dan membaca file CSV di Exel atau Spreadsheet	48
7.1.4	Jelaskan Library CSV	48
7.1.5	Jelaskan Library Pandas	48
7.1.6	Jelaskan fungsi-fungsi yang terdapat pada library CSV	48
7.1.7	Jelaskan fungsi-fungsi yang terdapat pada library Pandas	48
7.2	Dwi Yulianingsih	48
7.2.1	Soal 1	48
7.2.2	Soal 1	48
7.2.3	Soal 2	49
7.2.4	Soal 3	49
7.2.5	Soal 4	50
7.2.6	Soal 5	50
7.2.7	Soal 6	50
7.2.8	Soal 7	50

7.2.9	bukti bebas plagiarisme	51
7.3	Harun Ar-Rasyid	51
7.3.1	Soal 1	51
7.3.2	Soal 2	52
7.3.3	Soal 3	52
7.3.4	Soal 4	52
7.3.5	Soal 5	52
7.3.6	Soal 6	52
7.3.7	Soal 7	53
7.3.8	Bukti Bebas Plagiat	53
7.4	Sri Rahayu	53
7.4.1	Soal 1	53
7.5	Doli Jonviter	54
7.5.1	Soal 1	54
7.6	Rahmatul Ridha	54
7.6.1	Soal 1	54
7.7	Tomy Prawoto	59
7.7.1	Soal 1	59

8 Praktek Library CSV dan Pandas 65

8.1	Arjun Yuda Firwanda	65
8.1.1	Soal 1	65
8.1.2	Soal 2	66
8.1.3	Soal 3	66
8.1.4	Soal 4	67
8.1.5	Soal 5	67
8.1.6	Soal 6	67
8.1.7	Soal 7	67
8.1.8	Soal 8	68
8.1.9	Soal 9	68
8.1.10	Penanganan Error	68
8.2	Dwi Yulianingsih	68
8.2.1	Soal 1	68
8.2.2	Soal 2	68
8.2.3	Soal 3	69
8.2.4	Soal 4	69
8.2.5	Soal 5	69
8.2.6	Soal 6	69

8.2.7	Soal 7	70
8.2.8	Soal 8	70
8.2.9	Soal 9	70
8.2.10	Penanganan eror	70
8.3	Harun Ar-Rasyid	70
8.3.1	Soal 1	70
8.3.2	Soal 2	71
8.3.3	Soal 3	71
8.3.4	Soal 4	71
8.3.5	Soal 5	72
8.3.6	Soal 6	72
8.3.7	Soal 7	72
8.3.8	Soal 8	72
8.3.9	Soal 9	72
8.3.10	Penanganan Error	72
8.4	Sri Rahayu	72
8.4.1	Soal 1	72
8.5	Doli Jonviter	73
8.5.1	Soal 1	73
8.5.2	Soal 8	74
8.5.3	Soal 9	74
8.6	Rahmatul Ridha	74
8.6.1	Keterampilan Pemrograman	74
8.6.2	Penanganan eror	76
8.7	Tomy Prawoto	76
8.7.1	Soal Praktek	76
9	Komunikasi Perangkat Keras	79
9.1	Rahmatul Ridha	79
9.1.1	Teori	79
9.1.2	Soal No. 2	80
9.1.3	Soal No. 3	83
9.1.4	Soal No. 4	85
9.1.5	Soal No. 5	86
9.1.6	Soal No. 6	86
9.1.7	Soal No. 7	86
9.1.8	Praktek	86
9.2	Harun Ar - Rasyid	89

9.2.1	Teori	89
9.2.2	Praktek	94
9.3	Arjun Yuda Firwanda	95
9.3.1	Teori	95
9.3.2	Praktek	98
9.4	Doli Jonviter NT Simbolon /1154016	99
9.4.1	Soal 1	99
9.4.2	Soal 2	99
9.4.3	Soal 3	100
9.4.4	Soal 4	102
9.4.5	Soal 5	103
9.4.6	Soal No. 6	103
9.4.7	Soal No. 7	103
9.4.8	Praktek	103
9.5	Dwi Yulianingsih	105
9.5.1	Teori	105
9.5.2	Praktek	107
9.6	Doli Jonviter NT Simbolon /1154016	108
9.6.1	Soal 1	109
9.6.2	Soal 2	109
9.6.3	Soal 3	110
9.6.4	Soal 4	112
9.6.5	Soal 5	112
9.6.6	Soal No. 6	113
9.6.7	Soal No. 7	113
9.6.8	Praktek	113
9.7	Tomy Prawoto	114
9.7.1	Teori	114
9.7.2	Soal No. 2	115
9.7.3	Soal No. 3	118
9.7.4	Soal No. 4	121
9.7.5	Soal No. 5	121
9.7.6	Soal No. 6	121
9.7.7	Soal No. 7	122
9.7.8	Praktek	122
	Daftar Pustaka	125
	Index	127

DAFTAR GAMBAR

2.1	Download Anaconda	7
2.2	Proses 1	7
2.3	proses 2	8
2.4	Proses 3	8
2.5	Proses 4	9
2.6	Proses 5	9
2.7	Proses 6	10
2.8	Proses 7	10
2.9	Proses 8	11
2.10	Download Anaconda	11
2.11	Proses 1	12
2.12	proses 2	12
2.13	Proses 3	13

2.14	Proses 4	13
2.15	Proses 5	14
2.16	Proses 6	14
2.17	Proses 7	15
2.18	Proses 8	15
7.1	SS Bebas Plagiarisme	51
7.2	SS Bebas Plagiarisme	53
7.3	Membuat file csv	55
7.4	Save as Type	56
7.5	Perintah ekstensi	56
7.6	Konfirmasi Penyimpanan	56
7.7	file csv yang telah disave	56
7.8	file csv pada Excel	57
7.9	Plagiarisme	59
7.10	Membuat file csv	61
7.11	Save as Type	61
7.12	Perintah ekstensi	61
7.13	Konfirmasi Penyimpanan	62
7.14	file csv yang telah disave	62
7.15	file csv pada Excel	62
9.1	Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino dan langsung ditulis kedalam file CSV.	88
9.2	Hasil dari pembacaan fungsi untuk membaca file csv hasil arduino dan mengembalikan fungsi.	88
9.3	Membuat file csv	89
9.4	Membuat file csv	90
9.5	Membuat file csv	90
9.6	Membuat file csv	90

9.7	Membuat file csv	91
9.8	Membuat file csv	91
9.9	Membuat file csv	91
9.10	Membuat file csv	92
9.11	Membuat file csv	92
9.12	Membuat file csv	92
9.13	Membuat file csv	93
9.14	Membuat file csv	94
9.15	Cek Plagiarisme	97
9.16	SS Bebas Plagiarisme	107
9.17	Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino dan langsung ditulis kedalam file CSV.	123
9.18	Hasil dari pembacaan fungsi untuk membaca file csv hasil arduino dan mengembalikan fungsi.	124

DAFTAR TABEL

Listings

src/2/1144124/1144124_Teori.py	18
src/2/1144124/1144124_Teori.py	19
src/2/1144124/1144124_Teori.py	20
src/2/1144124/1144124_Teori.py	20
src/2/1144124/1144124_Teori.py	21
src/2/1154121/1154121_Teori.py	21
src/2/1154121/1154121_Teori.py	22
src/2/1154121/1154121_Teori.py	23
src/2/1154121/1154121_Teori.py	23
src/2/1154121/1154121_Teori.py	24
src/2/1144124/1144124_Praktek.py	26
src/2/1144124/1144124_Praktek.py	26
src/2/1144124/1144124_Praktek.py	27

src/2/1144124/1144124_Praktek.py	28
src/2/1144124/1144124_Praktek.py	28
src/2/1144124/1144124_Praktek.py	28
src/2/1144124/error.py	28
src/2/1154121/1154121_Praktek.py	28
src/2/1154121/1154121_Praktek.py	28
src/2/1154121/1154121_Praktek.py	29
src/2/1154121/1154121_Praktek.py	30
src/2/1154121/1154121_Praktek.py	30
src/2/1154121/error.py	30
src/3/1144124/1144124Chapter3.py	32
src/3/1144124/1144124Chapter3.py	32
src/3/1144124/1144124Chapter3.py	33
src/3/1144124/1144124Chapter3.py	33
src/3/1144124/1144124Chapter3.py	33
src/3/1144124/1144124Chapter3.py	34
src/3/1144124/1144124Chapter3.py	34
src/3/1144124/1144124Chapter3.py	34
src/3/1144124/1154121Chapter3.py	35
src/3/1154121/1154121Chapter3.py	35
src/3/1154121/1154121Chapter3.py	35
src/3/1154121/1154121Chapter3.py	36
src/3/1154121/1154121Chapter3.py	36
src/3/1154121/1154121Chapter3.py	36
src/3/1144124/1144124Chapter3.py	38
src/3/1144124/1144124Chapter3.py	39
src/3/1144124/1144124Chapter3.py	39
src/3/1144124/1144124Chapter3.py	40
src/3/1144124/1144124Chapter3.py	40
src/3/1144124/1144124Chapter3.py	40

src/3/1144124/1144124Chapter3.py	40
src/3/1144124/1144124Chapter3.py	41
src/3/1144124/1144124Chapter3.py	41
src/3/1144124/1144124Chapter3.py	41
src/3/1144124/main.py	41
src/3/1144124/main.py	41
src/3/1144124/1144124Chapter3.py	42
src/3/1154121/1154121Chapter3.py	42
src/3/1154121/1154121Chapter3.py	43
src/3/1154121/1154121Chapter3.py	43
src/3/1154121/1154121Chapter3.py	44
src/3/1154121/1154121Chapter3.py	45
src/3/1154121/main.py	45
src/3/1154121/main.py	45
src/3/1154121/1154121Chapter3.py	46
src/4/1174009/dudul.py	49
src/4/1174027/teori/c_1174027_csv.py	52
src/4/1174027/teori/c_1174027_csv.py	52
src/4/1174027/teori/c_1174027_csv.py	53
src/4/1174027/praktek/p_1174027_pandas.py	53
7.1 Contoh penggunaan format CSV.	55
7.2 Membaca file berformat CSV list.	58
7.3 Membaca file berformat CSV dictionary.	58
7.4 Menulis file berformat CSV list.	58
7.5 Menulis file berformat CSV dictionary.	58
7.6 Membaca file berformat CSV pandas.	59
7.7 Menulis file berformat CSV pandas.	59
7.8 Contoh penggunaan format CSV.	60
7.9 Membaca file berformat CSV list.	63
7.10 Membaca file berformat CSV dictionary.	63
7.11 Menulis file berformat CSV list.	63
7.12 Menulis file berformat CSV dictionary.	64
7.13 Membaca file berformat CSV pandas.	64

7.14 Menulis file berformat CSV pandas.	64
src/4/1174008/praktek/a1174008_csv.py	65
src/4/1174008/praktek/a1174008_csv.py	66
src/4/1174008/praktek/a1174008_pandas.py	66
src/4/1174008/praktek/a1174008_pandas.py	67
src/4/1174008/praktek/main_arjun.py	68
src/4/1174008/praktek/main_arjun.py	68
src/4/1174009/praktek/d1174009_csv.py	68
src/4/1174009/praktek/d1174009_csv.py	68
src/4/1174009/praktek/d1174009_pandas.py	69
src/4/1174009/praktek/d1174009_pandas.py	70
src/4/1174009/praktek/main_dwi.py	70
src/4/1174009/praktek/main_dwi.py	70
src/4/1174009/praktek/eror.py	70
src/4/1174009/praktek/eror.py	70
src/4/1174009/praktek/eror.py	70
src/4/1174027/praktek/c_1174027_csv.py	71
src/4/1174027/praktek/c_1174027_csv.py	71
src/4/1174027/praktek/p_1174027_pandas.py	71
src/4/1174027/praktek/p_1174027_pandas.py	71
src/4/1174027/praktek/p_1174027_pandas.py	72
src/4/1174027/praktek/p_1174027_pandas.py	72
src/4/1174027/praktek/p_1174027_pandas.py	72
src/4/1174027/praktek/main_harun.py	72
src/4/1174027/praktek/main_harun.py	72
src/4/1154016/Chapter4/c_1154016_csv.py	73
src/4/1154016/Chapter4/c_1154016_csv.py	73
src/4/1154016/Chapter4/p_1154016_pandas.py	73
src/4/1154016/Chapter4/p_1154016_pandas.py	73
src/4/1154016/Chapter4/p_1154016_pandas.py	74
src/4/1154016/Chapter4/p_1154016_pandas.py	74
src/4/1154016/Chapter4/p_1154016_pandas.py	74

src/4/1154016/Chapter4/jonviter.py	74
src/4/1154016/Chapter4/jonviter.py	74
8.1 Fungsi untuk membuka file CSV dengan lib CSV mode list.	74
8.2 Fungsi untuk membuka file CSV dengan lib CSV mode dictionary.	75
8.3 Fungsi untuk membuka file CSV dengan lib Pandas mode list.	75
8.4 Fungsi untuk membuka file CSV dengan lib Pandas mode dictionary.	75
8.5 Fungsi untuk mengubah format tanggal menjadi standar dataframe.	75
8.6 Fungsi untuk mengubah index kolom.	75
8.7 Fungsi untuk mengubah atribut atau nama kolom.	75
8.8 Membuat dan mebaca file CSV menggunakan library 1144124pandas.	76
8.9 Membuat dan mmebaca file CSV menggunakan library 1144124pandas.	76
src/4/1144124/Chapter4/error.py	76
src/4/1144124/Chapter4/error.py	76
src/4/1144124/Chapter4/error.py	76
8.10 Fungsi untuk membuka file CSV dengan lib CSV mode list.	76
8.11 Fungsi untuk membuka file CSV dengan lib CSV mode dictionary.	77
8.12 Fungsi untuk membuka file CSV dengan lib Pandas mode list.	77
8.13 Fungsi untuk membuka file CSV dengan lib Pandas mode dictionary.	77
8.14 Fungsi untuk mengubah format tanggal menjadi standar dataframe.	77
8.15 Fungsi untuk mengubah index kolom.	77
8.16 Fungsi untuk mengubah atribut atau nama kolom.	78
8.17 Membuat dan mebaca file CSV menggunakan library 1154121pandas.	78
8.18 Membuat dan mmebaca file CSV menggunakan library 1154121pandas.	78
9.1 Fungsi yang menggunakan pyserial.	86
9.2 Fungsi untuk mendapatkan data dari Arduino.	86
9.3 Fungsi untuk mendapatkan data langsung dari Arduino dengan looping.	87
9.4 Fungsi untuk mendapatkan data dari Arduino dan langsung ditulis kedalam file CSV.	87
9.5 Fungsi untuk membaca file CSV hasil Arduino dan mengembalikan fungsi.	88
9.6 Fungsi untuk menanggulangi error menggunakan Try Except.	89
src/5/1174027/teori/T1174027.py	94
src/5/1174027/praktek/1174027_realtime.py	94
src/5/1174027/praktek/1174027_save.py	94
src/5/1174027/praktek/1174027_realtime.py	94
src/5/1174027/praktek/1174027_csv.py	95
src/5/1174027/praktek/1174027.py	95

src/5/1174008/teori/T1174008.py	97
src/5/1174008/praktek/1174008_realtime.py	98
src/5/1174008/praktek/1174008_save.py	98
src/5/1174008/praktek/1174008_realtime.py	98
src/5/1174008/praktek/1174008_csv.py	98
src/5/1174008/praktek/1174008.py	99
src/5/1154016/1154016_realtime.py	103
src/5/1154016/1154016_save.py	104
src/5/1154016/1154016_realtime.py	104
src/5/1154016/1154016_csv.py	104
src/5/1154016/1154016.py	104
src/5/1174009/T1174009.py	107
src/5/1174009/1174009_realtime.py	107
src/5/1174009/1174009_save.py	107
src/5/1174009/1174009_realtime.py	108
src/5/1174009/1174009_csv.py	108
src/5/1174009/1174009_error.py	108
src/5/1154016/1154016_realtime.py	113
src/5/1154016/1154016_save.py	113
src/5/1154016/1154016_realtime.py	113
src/5/1154016/1154016_csv.py	114
src/5/1154016/1154016.py	114
9.7 Fungsi yang menggunakan pyserial.	122
9.8 Fungsi untuk mendapatkan data dari Arduino.	122
9.9 Fungsi untuk mendapatkan data langsung dari Arduino dengan looping.	122
9.10 Fungsi untuk mendapatkan data dari Arduino dan langsung ditulis kedalam file CSV.	122
9.11 Fungsi untuk membaca file CSV hasil Arduino dan mengembalikan fungsi.	123
9.12 Fungsi untuk menanggulangi error menggunakan Try Except.	124

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[1].

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

BAB 1

JUDUL BAGIAN PERTAMA

1.1 Arjun Yuda Firwanda

1.1.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\n` dsb

1.2 Dwi Yulianingsih

1.2.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\n` dsb

1.3 Harun Ar-Rasyid

1.3.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\par` dsb

1.4 Sri Rahayu

1.4.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\par` dsb

1.5 Doli Jonviter

1.5.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\par` dsb

1.6 Rahmatul Ridha

1.6.1 Sejarah Python

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

Saat ini pengembangan Python terus dilakukan oleh sekumpulan pemrogram yang dikoordinir Guido dan Python Software Foundation. Python Software Foundation adalah sebuah organisasi non-profit yang dibentuk sebagai pemegang hak cipta intelektual Python sejak versi 2.1 dan dengan demikian mencegah Python dimiliki oleh perusahaan komersial. Saat ini distribusi Python sudah mencapai versi 2.7.14 dan versi 3.6.3.

1.7 Tomy Prawoto

1.7.1 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 juga didukung oleh komunitas yang besar.

Python mendukung multi paradigma pemrograman, utamanya; namun tidak dibatasi; pada pemrograman berorientasi objek, pemrograman imperatif, dan pemrograman fungsional. Salah satu fitur yang tersedia pada python adalah sebagai bahasa pemrograman dinamis yang dilengkapi dengan manajemen memori otomatis. Seperti halnya pada bahasa pemrograman dinamis lainnya, python umumnya digunakan sebagai bahasa skrip meski pada praktiknya penggunaan bahasa ini lebih luas mencakup konteks pemanfaatan yang umumnya tidak dilakukan dengan menggunakan bahasa skrip. Python dapat digunakan untuk berbagai keperluan pengembangan perangkat lunak dan dapat berjalan di berbagai platform sistem operasi.

BAB 2

JUDUL BAGIAN PERTAMA

2.1 Arjun Yuda Firwanda

2.1.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\par` dsb

2.2 Dwi Yulianingsih

2.2.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\par` dsb

2.3 Harun Ar-Rasyid

2.3.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\par` dsb

2.4 Sri Rahayu

2.4.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\par` dsb

2.5 Doli Jonviter

2.5.1 Soal 1

Isi jawaban soal ke-1

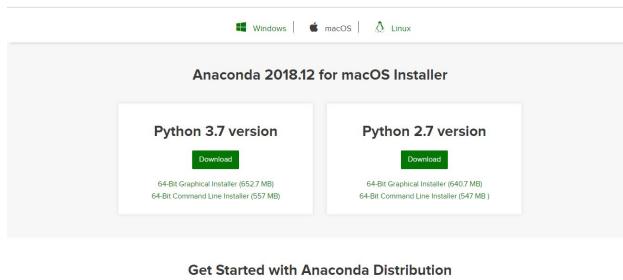
Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\par` dsb

2.6 Rahmatul Ridha

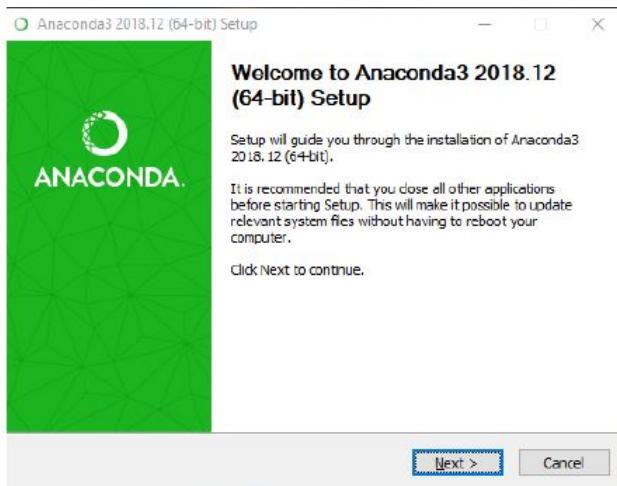
2.6.1 Soal 1

2.6.2 Instalasi Anaconda

- Instalasi Anaconda Berikut adalah langkah-langkah cara menginstal Anaconda di Windows:
 1. Download installer anaconda terbaru, seperti pada gambar 2.10. Kalian dapat memilih versi 2 atau 3, dengan versi Anaconda berapa.
 2. Setelah selesai mendownload, klik 2 kali pada installer Anaconda.
 3. Kemudian akan tampil seperti gambar 2.11, lalu klik next.
 4. Setelah itu read lisensi dan klik ‘I Agree’ seperti pada gambar 2.3.
 5. Selanjutnya ada pilihan untuk menginstallnya, yaitu ‘just me’ atau ‘all users’. Lalu klik next seperti pada gambar 2.13.
 6. Kemudian pilih okasi yang diinginkan, lalu klik next seperti pada gambar 2.5.
 7. Pilih ‘add anaconda to PATH’ atau tidak. Disini kalian memilih apakah akan mendaftarkan Anaconda sebagai default Python 3.7. Kacuali kalian



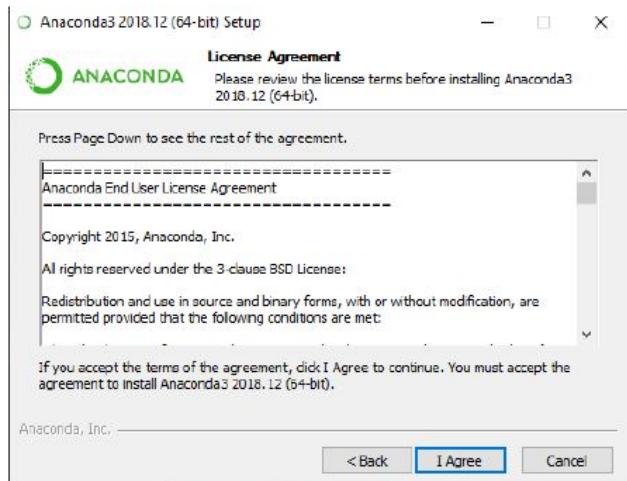
Gambar 2.1 Download Anaconda



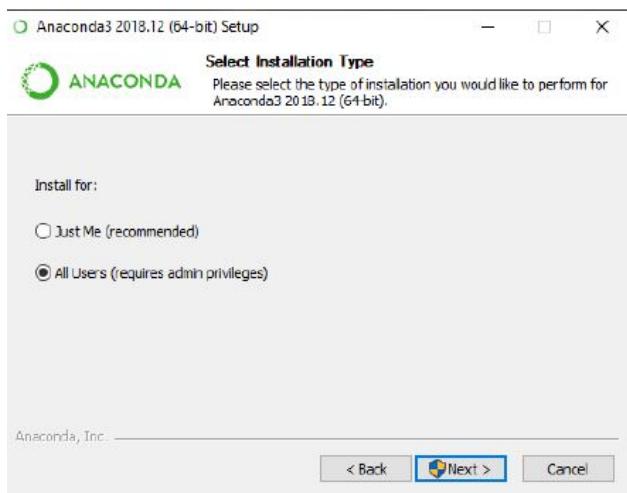
Gambar 2.2 Proses 1

berencana menginstal dan menjalankan beberapa versi Anaconda, atau beberapa versi Python, biarkan default dan biarkan kotaknya dicentang. Kemudian klik tombol Install. Jika kalian ingin melihat packages Anaconda yang sedang dipasang, klik Show Details seperti pada gambar 2.6.

8. Jika instalasi selesai, kemudian klik next seperti pada gambar 2.7.
9. Jika packages telah selesai diinstall, maka akan ada perintah untuk menginstall VS Code, lalu klik tombol Install Microsoft VS Code seperti pada gambar 2.8.
10. Setelah instalasi selesai, maka akan terlihat kotak dialog ‘Thanks for Installing Anaconda3’. Lalu klik Finish seperti pada gambar 2.9.



Gambar 2.3 proses 2



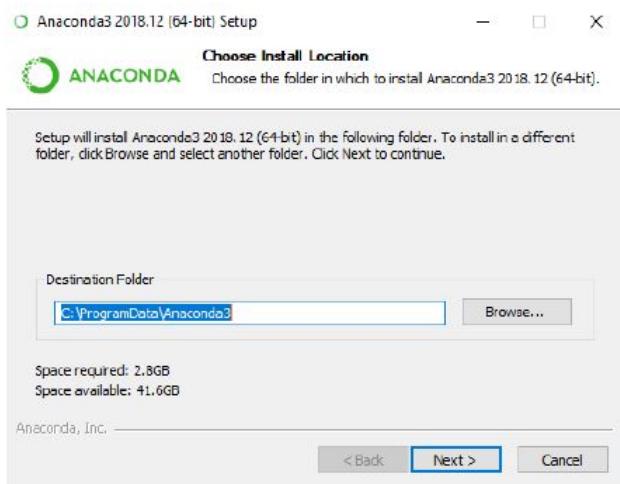
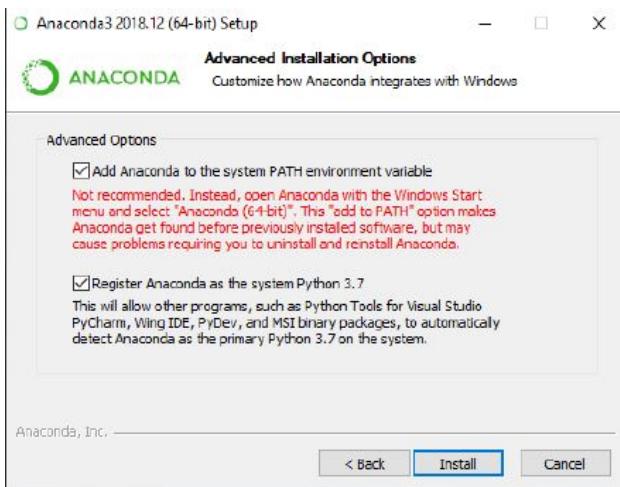
Gambar 2.4 Proses 3

2.7 Tomy Prawoto

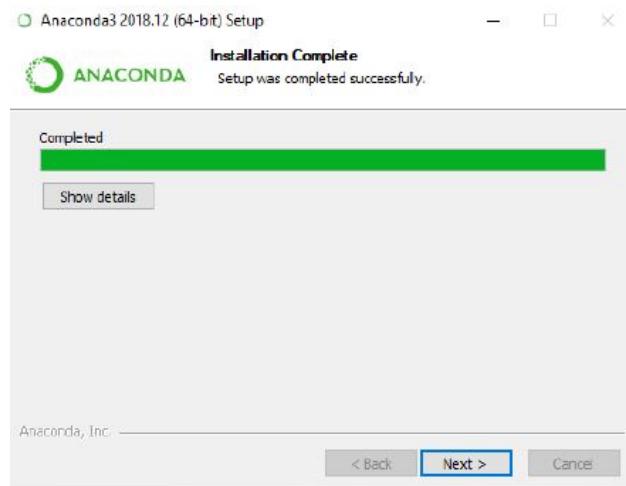
2.7.1 Soal 1

2.7.2 Instalasi Anaconda

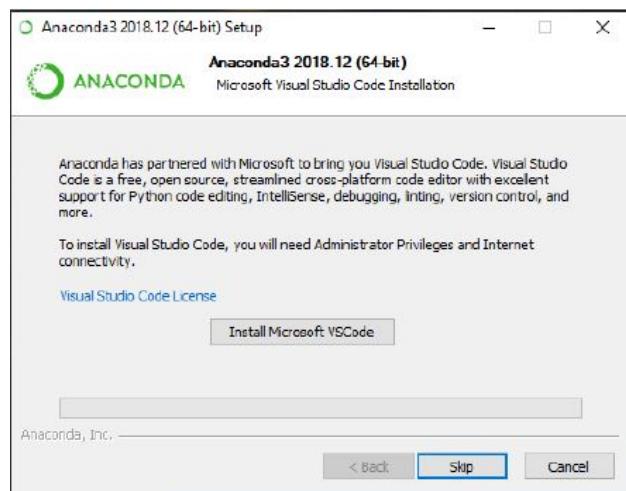
- Instalasi Anaconda Berikut adalah langkah-langkah cara menginstal Anaconda di Windows:

**Gambar 2.5** Proses 4**Gambar 2.6** Proses 5

1. Download installer anaconda terbaru, seperti pada gambar 2.10. Kalian dapat memilih versi 2 atau 3, dengan versi Anaconda berapa.
2. Setelah selesai mendownload, klik 2 kali pada installer Anaconda.
3. Kemudian akan tampil seperti gambar 2.11, lalu klik next.
4. Setelah itu read lisensi dan klik “I Agree”.
5. Selanjutnya ada pilihan untuk menginstallnya, yaitu “just me” atau “all users”. Lalu klik next.

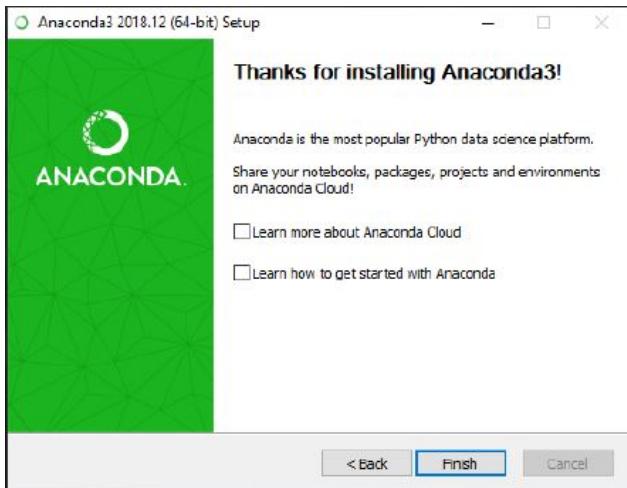


Gambar 2.7 Proses 6

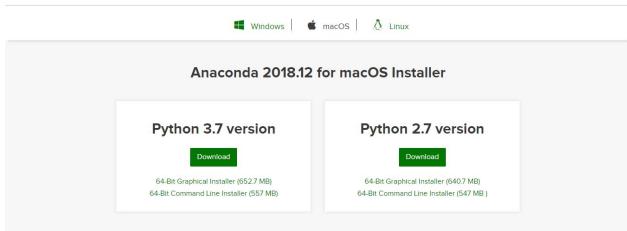


Gambar 2.8 Proses 7

6. Kemudian pilih okasi yang diinginkan, lalu klik next.
7. Pilih ‘add anaconda to PATH’ atau tidak. Disini kalian memilih apakah akan mendaftarkan Anaconda sebagai default Python 3.7. kacualu kalian berencana menginstal dan menjalankan beberapa versi Anaconda, atau beberapa versi Python, biarkan default dan biarkan kotaknya dicentang. Kemudian klik tombol Install. Jika kalian ingin melihat packages Anaconda yang sedang dipasang, klik Show Details.



Gambar 2.9 Proses 8

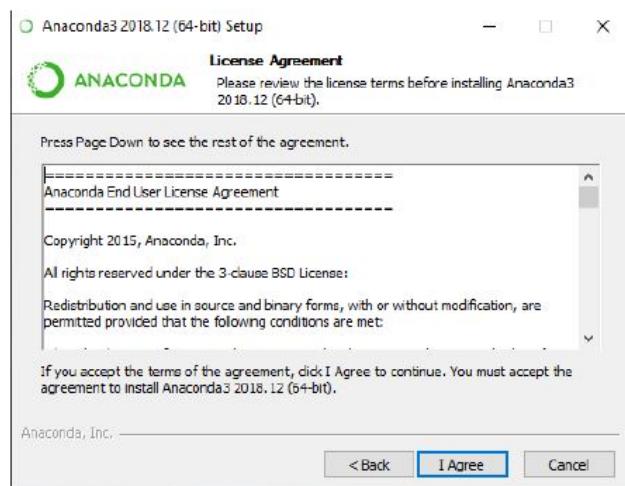


Gambar 2.10 Download Anaconda

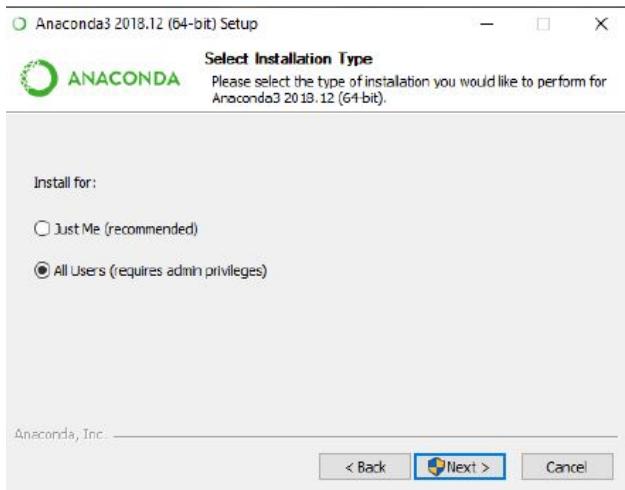
8. Jika instalasi selesai, kemudian klik next.
9. Jika packages telah selesai diinstall, maka akan ada perintah untuk menginstall VS Code, lalu klik tombol Install Microsoft VS Code.
10. Setelah instalasi selesai, maka akan terlihat kotal dialog “Thanks for Installing Anaconda3”. Lalu klik Finish.



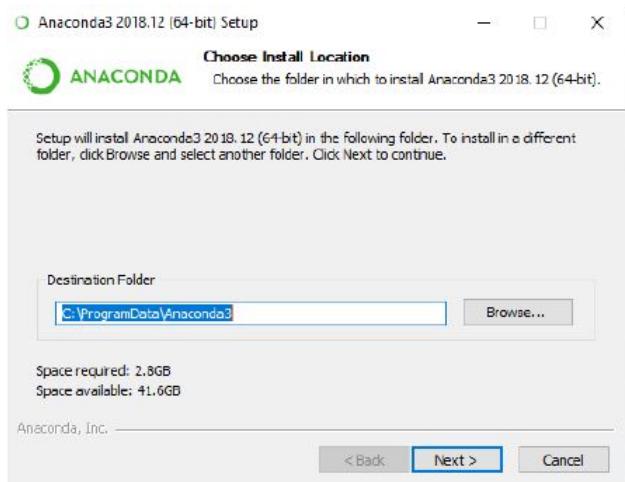
Gambar 2.11 Proses 1



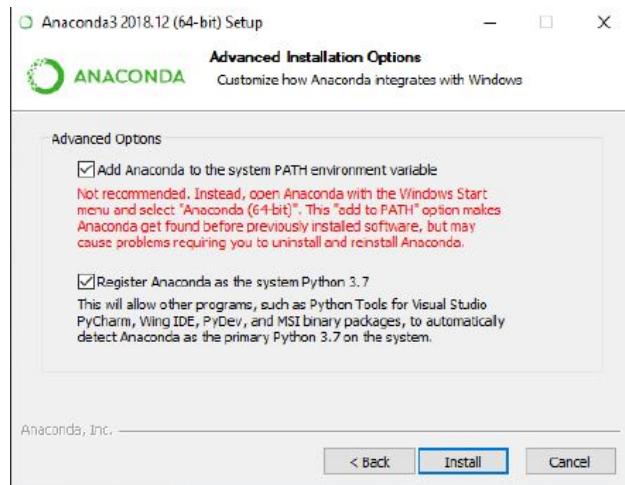
Gambar 2.12 proses 2



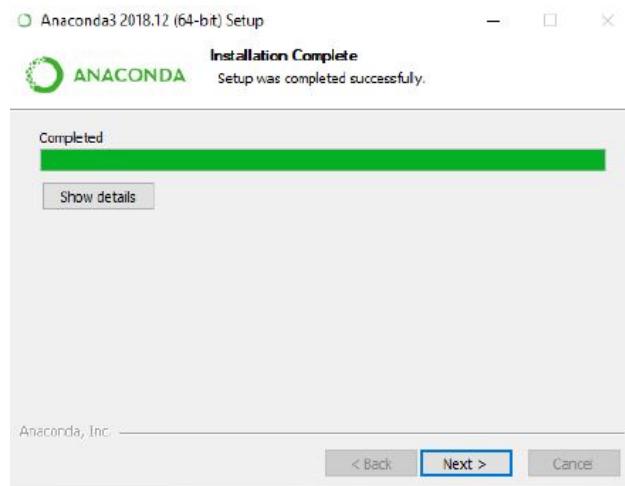
Gambar 2.13 Proses 3



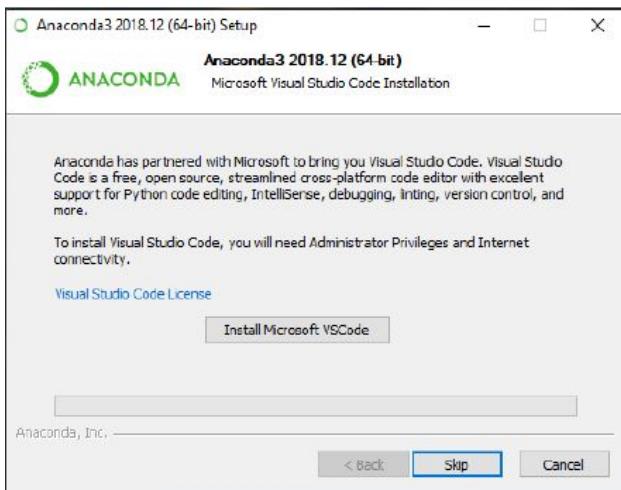
Gambar 2.14 Proses 4



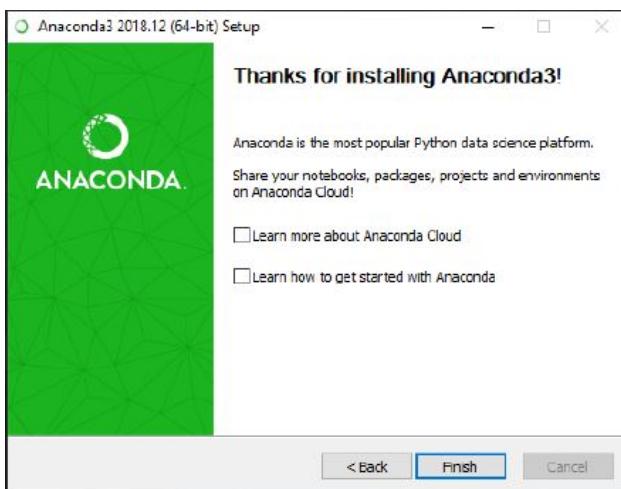
Gambar 2.15 Proses 5



Gambar 2.16 Proses 6



Gambar 2.17 Proses 7



Gambar 2.18 Proses 8

BAB 3

JUDUL BAGIAN PERTAMA

3.1 Arjun Yuda Firwanda

3.1.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\n` dsb

3.2 Dwi Yulianingsih

3.2.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\n` dsb

3.3 Harun Ar-Rasyid

3.3.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `par` dsb

3.4 Sri Rahayu

3.4.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `par` dsb

3.5 Doli Jonviter

3.5.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `par` dsb

3.6 Rahmatul Ridha

3.6.1 Teori

1. jenis-jenis variable phyton dan cara pemakaiannya. Variable merupakan tempat untuk menyimpan data, Isi dari variabel itu dapat berubah atau mutable sesuai dengan operasi yang diinginkan. Saat program dieksekusi maka variabellah yang bertugas menyimpan data. Dimana didalam phyton terdapat beberapa variable diantaranya number, boolean,string. Dalam membuat variabel Pythoncaranya adalah sebagai berikut

```
1 #variabel
2 c = 7
3 d = "Rahmatul Ridha"
4
5 print(c)
6 print(d)
7
8 #integer
9 x = 4
10 y = 6969787819
11 z = -456283
12
13 print(type(x))
14 print(type(y))
```

```
15 print(type(z))
16
17 #float
18 x = 1.115
19 y = 1.5
20 z = -35.60
21
22 print(type(x))
23 print(type(y))
24 print(type(z))
25
26 #complex
27 x = 5+5j
28 y = 5j
29 z = -5j
30
31 print(type(x))
32 print(type(y))
33 print(type(z))
34
35 #variabel string
36 a = "Hello , World!"
37 print(a[1])
38
39 b = "Hello , World!"
```

2. operator dasar aritmatika. Dimana terdapat penjumlahan,pengurangan,pembagian,perkalian nominal

```
1 print(b[2:7])
2
3 a = "Hello , World!"
4 print(a.strip())
5
6 a = "Hello , World!"
7 print(len(a))
8
9 a = "Hello , World!"
10 print(a.lower())
11
12 a = "Hello , World!"
13 print(a.upper())
14
15 #variabel boolean
16 a = 40
17 b = 300
18 if b>a:
19     print("b lebih besar dari a")
20
21 #input output
22 npm = input()
23 print(npm)
24
25 #operator
26 x = 5
27 y = 3
```

```

28
29 print(x+y)

```

3. Perulangan. Dalam phyton terdapat perulangan while dan for :

```

1
2 x = 5
3 y = 3
4
5 print(x-y)
6
7 x = 12
8 y = 3
9
10 print(x*y)
11
12 x = 8

```

4. Dimana terdapat sintak untuk memilih kondisi didalam kondisi. Untuk memilih keputusan menggunakan (kondisi if) dimana digunakan untuk mengantisipasi kondisi yang terjadi saat jalannya suatu program dan menentukan tindakan apa yang akan dilakukan sesuai dengan kondisi.

```

1 y = 2
2
3 print(x/y)
4
5 x = 5
6 y = 2
7
8 print(x%y)
9
10 #perulangan for
11 ulang = [ "Motor" , "Mobil" , "Pesawat" ]
12 for i in range(ulang):
13     print(i)
14
15 #perulangan while
16 coba = 1
17 while i < 6 :
18     print(i)
19     i += 1
20
21 #if
22 a = 40
23 b = 300
24 if b > a :
25     print("b lebih besar dari a")

```

5. Jenis-jenis sintak error pada phyton.

Syntax errors Jika dalam program terdapat kesalahan sintaks maka proses akan berhenti dan menampilkan pesan kesalahan. Runtime errors, disebut begitu

karenakesalahan tidak akan muncul sampai Anda menjalankan program tersebut.Kesalahan ini juga dikenal dengan exceptions atau pengecualian karena biasanya mengindikasikan sesuatu pengecualian yang buruk telah terjadi.

Type eror merupakan eror yang terjadi saat dilakukan eksekusi pada suatu operasi dengan type object yang tidak sesuai. ZeroDivision eror merupakan eror yang terjadi saat eksekusi program menghasilkan perhitungan matematika dengan angka 0

6. Try except cara memakai try except adalah sebagai berikut

```
1 #variabel
2 c = 7
3 d = "Rahmatul Ridha"
4
5 print(c)
6 print(d)
7
8 #integer
9 x = 4
10 y = 6969787819
```

3.7 Tomy Prawoto

3.7.1 Soal 1

Isi jawaban soal ke-1

3.7.2 Teori

1. jenis-jenis variable phyton dan cara pemakaiannya Variabel adalah lokasi memori yang dicadangkan untuk menyimpan nilai-nilai. Ini berarti bahwa ketika Anda membuat sebuah variabel Anda memesan beberapa ruang di memori. Variabel menyimpan data yang dilakukan selama program dieksekusi, yang natinya isi dari variabel tersebut dapat diubah oleh operasi - operasi tertentu pada program yang menggunakan variabel. Dalam membuat variabel Python-caranya adalah sebagai berikut

```
1 #variabel
2 c = 7
3 d = "Tomy Prawoto"
4
5 print(c)
6 print(d)
7
8 #integer
9 x = 4
10 y = 6969787819
11 z = -456283
12
```

```

13 print(type(x))
14 print(type(y))
15 print(type(z))
16
17 #float
18 x = 1.115
19 y = 1.5
20 z = -35.60
21
22 print(type(x))
23 print(type(y))
24 print(type(z))
25
26 #complex
27 x = 5+5j
28 y = 5j
29 z = -5j
30
31 print(type(x))
32 print(type(y))
33 print(type(z))
34
35 #variabel string
36 a = "Hello , World!"
37 print(a[1])
38
39 b = "Hello , World!"
```

2. operator dasar aritmatika dimana terdapat penjumlahan,pengurangan,pembagian,perkalian nominal

```

1 print(b[2:7])
2
3 a = "Hello , World!"
4 print(a.strip())
5
6 a = "Hello , World!"
7 print(len(a))
8
9 a = "Hello , World!"
10 print(a.lower())
11
12 a = "Hello , World!"
13 print(a.upper())
14
15 #variabel boolean
16 a = 40
17 b = 300
18 if b>a:
19     print("b lebih besar dari a")
20
21 #input output
22 npm = input()
23 print(npm)
24
25 #operator
```

```

26 x = 5
27 y = 3
28
29 print(x+y)

```

3. Perulangan dalam phyton terdapat perulangan while dan for

```

1
2 x = 5
3 y = 3
4
5 print(x-y)
6
7 x = 12
8 y = 3
9
10 print(x*y)
11
12 x = 8

```

4. Dimana terdapat sintak untuk meilih kondisi didalam kondisi Untuk memilih keputusan menggunakan (kondisi if) dimana digunakan untuk mengantisipasi kondisi yang terjadi saat jalannya suatu program dan menentukan tindakan apa yang akan dilakukan sesuai dengan kondisi.

```

1 y = 2
2
3 print(x/y)
4
5 x = 5
6 y = 2
7
8 print(x%y)
9
10 #perulangan for
11 ulang = ["Motor", "Mobil", "Pesawat"]
12 for i in range(ulang):
13     print(i)
14
15 #perulangan while
16 coba = 1
17 while i < 6 :
18     print(i)
19     i += 1
20
21 #if
22 a = 40
23 b = 300
24 if b > a :
25     print("b lebih besar dari a")

```

5. Jenis-jenis sintak error pada phyton Syntax errors Jika dalam program terdapat kesalahan sintaks maka proses akan berhenti dan menampilkan pesan kesalahan. Runtime errors, disebut begitu karenakesalahan tidak akan muncul sampai

Anda menjalankan program tersebut. Kesalahan ini juga dikenal dengan exceptions atau pengecualian karena biasanya mengindikasikan sesuatu pengecualian yang buruk telah terjadi.

Type eror merupakan eror yang terjadi saat dilakukan eksekusi pada suatu operasi dengan type object yang tidak sesuai. ZeroDivision eror merupakan eror yang terjadi saat eksekusi program menghasilkan perhitungan matematika dengan angka 0

6. Try except cara memakai try except adalah sebagai berikut

```
1 #variabel
2 c = 7
3 d = "Tomy Prawoto"
4
5 print(c)
6 print(d)
7
8 #integer
9 x = 4
10 y = 6969787819
```

BAB 4

JUDUL BAGIAN PERTAMA

4.1 Arjun Yuda Firwanda

4.1.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\n` dsb

4.2 Dwi Yulianingsih

4.2.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\n` dsb

4.3 Harun Ar-Rasyid

4.3.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\par` dsb

4.4 Sri Rahayu

4.4.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\par` dsb

4.5 Doli Jonviter

4.5.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\par` dsb

4.6 Rahmatul Ridha

4.6.1 praktek

1. Jawaban soal no 1

```

1 print(1144124%3)
2
3 #Nomor 1
4
5 print("###  ###  ###  ###  ###  ###  ###  #####  ###  ###")
6 print("###  ###  ###  ###  ###  ###  ###  #####  ###  ###")
7 print("###  ###  ###  ###  ###  ###  ###      ###  ###  ###")
8 print("###  ###  ###  ###  ###  ###  ###      ###  ###  ###")
9 print("###  ###  #####  #####  ###  #####  #####  #####")
10 print("###  ###          ###          ###  ###  #####  #####  ###")
    )
```

2. Jawaban soal no 2

```

1 print("###  ###          ###          ###  ###  #####  #####  ###")
    )
```

```
1 print("### ### #### ## ## #### ## ##")
2 print("### ### #### ## ## ##### ##### ## ##")
3 print("### ### #### ## ## ##### ##### ## ##")
4
5
6 #Nomor 2
```

3. Jawaban soal no 3

```
1
2 npm = input("Masukkan NPM :")
3 hitung = 0
4 while(hitung <= 51):
5     print("Hallo, " + str(npm) + "Apa kabar?")
6     hitung = hitung + 1
```

4. Jawaban soal no 4

```
1
2
3 #Nomor 3
```

5. Jawaban soal no 5

```
1
2 npm = input("Masukkan NPM :")
3 hitung = 0
4 while(hitung <= 6):
5     print("Hello, " + str(npm[4:7]) + " Apa kabar?")
6     hitung = hitung + 1
7
8 #Nomor 4
9 npm = input("Masukkan NPM : ")
10 print("Hello, " + str(npm[-3]) + " Apa kabar?")
```

6. Jawaban soal no 6

```
1 #Nomor 5
2 a = 1
```

7. Jawaban soal no 7

```
1 b = 1
2 c = 7
```

8. Jawaban soal no 8

```
1 d = 4
2 e = 0
3 f = 5
4 g = 1
5 h = a+b+c+d+e+f+g
6 i = a*b*c*d*e*f*g
7
8 print("Hasil no 5")
```

9. Jawaban soal no 9

```

1 print("NPM anda")
2 print(a, b, c, d, e, f, g, h)

```

10. Jawaban soal no 10

```

1
2 #Nomor 6

```

11. Jawaban soal no 11

```

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

```

4.6.2 Keterangan dan Penanganan eror

```

1     c = a + b
2     print(c)
3 except TypeError :
4     print("Perbedaan Tipe Data")

```

4.7 Tomy Prawoto**4.7.1 Soal 1****4.7.2 praktek**

1. Jawaban soal no 1

```

1 print(1154121%3)
2
3 #Nomor 1
4
5 print("###    ###    #####    ###    ###    ###    #####    ###")
6 print("###    ###    ###    ###    ###    #####    ###")
7 print("###    ###    ###    ###    ###    ###    ###")
8 print("###    ###    ###    ###    ###    ###    ###")
9 print("###    ###    #####    ###    ###    #####    ###")
10 print("###    ###    ###    ###    ###    #####    ###")

```

2. Jawaban soal no 2

```

1 print("###    ###    ###    ###    ###    ###    ###")
2 print("###    ###    ###    ###    ###    ###    ###")
3 print("###    ###    ###    ###    ###    #####    ###")
4 print("###    ###    #####    ###    ###    #####    ###")
5
6 #Nomor 2

```

3. Jawaban soal no 3

```

1 npm = input("Masukkan NPM :")
2 hitung = 0
3 while(hitung <= 21):
4     print("Hallo, " + str(npm) + " Apa kabar?")
5     hitung = hitung +1
6

```

4. Jawaban soal no 4

```

1
2
3 #Nomor 3

```

5. Jawaban soal no 5

```

1
2 npm = input("Masukkan NPM :")
3 hitung = 0
4 while(hitung <= 6):
5     print("Hello, " + str(npm[4:7]) + " Apa kabar?")
6     hitung = hitung +1
7
8 #Nomor 4
9 npm = input("Masukkan NPM : ")
10 print("Hello, " + str(npm[-3]) + " Apa kabar?")

```

6. Jawaban soal no 6

```

1 #Nomor 5
2 a = 1

```

7. Jawaban soal no 7

```

1 b = 1
2 c = 5

```

8. Jawaban soal no 8

```

1 d = 4
2 e = 1
3 f = 2
4 g = 1
5 h = a+b+c+d+e+f+g
6 i = a*b*c*d*e*f*g
7
8 print("Hasil no 5")

```

9. Jawaban soal no 9

```

1 print("NPM anda")
2 print(a, b, c, d, e, f, g, h)

```

10. Jawaban soal no 10

```
1  
2 #Nomor 6
```

11. Jawaban soal no 11

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

4.7.3 Keterangan dan Penanganan eror

```
1     c = a + b  
2     print(c)  
3 except TypeError :  
4     print("Perbedaan Tipe Data")
```

BAB 5

JUDUL BAGIAN PERTAMA

5.1 Arjun Yuda Firwanda

5.1.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\n` dsb

5.2 Dwi Yulianingsih

5.2.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\n` dsb

5.3 Harun Ar-Rasyid

5.3.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\par` dsb

5.4 Sri Rahayu

5.4.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\par` dsb

5.5 Doli Jonviter

5.5.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\par` dsb

5.6 Rahmatul Ridha

5.6.0.1 Pemahaman Teori

1. Apa itu fungsi, inputan fungsi dan kembalian fungsi dengan contoh kode program lainnya.

Fungsi Fungsi adalah bagian dari program yang dapat digunakan ulang. Hal ini bisa dicapai dengan memberi nama pada blok statemen, kemudian nama ini dapat dipanggil di manapun dalam program. Kita telah menggunakan beberapa fungsi builtin seperti `range`. Fungsi dalam Python didefinisikan menggunakan kata kunci `def`. Setelah `def` ada nama pengenal fungsi diikuti dengan parameter yang diapit oleh tanda kurung dan diakhiri dengan tanda titik dua `:`. Baris berikutnya berupa blok fungsi yang akan dijalankan jika fungsi dipanggil.

```
1 def uji():
2     print("Tugas 3")
```

Fungsi dapat membaca parameter, parameter adalah nilai yang disediakan kepada fungsi, dimana nilai ini akan menentukan output yang akan dihasilkan fungsi.

```
1 uji()
2
3 def uji_param(nama):
4     print("Nama saya :" + str(nama))
```

Statemen return digunakan untuk keluar dari fungsi. Kita juga dapat men-spesifikasikan nilai kembalian.

```

1 uji_param(input("Masukan Nama Kamu : "))
2
3 def uji_return(a,b):
4     r = a + b
5     return r
6
7 a = 10
8 b = 50
9 c = uji_return(a,b)
10 print(c)
```

2. Apa itu paket dan cara pemanggilan paket atau library dengan contoh kode program lainnya.

Untuk memudahkan dalam pemanggilan fungsi yang di butuhkan, agar dapat dipanggil berulang. Cara pemanggilannya

```

1 #from fungsi_rahma import *
2 #print(penulisan(int(input("Masukan NPM kamu : "))))
3
4 #class Employee:
5 #    'Common base class for all employees'
6 #    empCount = 0
7
8 #    def __init__(self, name, salary):
9 #        self.name = name
10 #       self.salary = salary
11 #       Employee.empCount += 1
12
13 #    def displayCount(self):
14 #        print ("Total Employee %d" % Employee.empCount)
```

3. Jelaskan Apa itu kelas, apa itu objek, apa itu atribut, apa itu method dan contoh kode program lainnya masing-masing. kelas merupakan sebuah blueprint yang mepresentasikan objek. objek adalah hasil cetakan dadri sebuah kelas. method adalah suatu upaya yang digunakan oleh object.

```

1 #    def displayEmployee(self):
2 #        print ("Name : ", self.name, ", Salary: ", self.salary)
```

4. Jelaskan cara pemanggilan library kelas dari instansiasi dan pemakaianya dengan contoh program lainnya. Cara Pemanggilanya :

pertama import terlebih dahulu filenya.

kemudian buat variabel untuk menampung datanya.

setelah itu panggil nama classnya dan panggil methodnya.

Gunakan perintah print untuk menampilkan hasilnya.

```

1 #This would create first object of Employee class
2 #empl = Employee("Zara", 2000)
3 #This would create second object of Employee class
4 #emp2 = Employee("Manni", 5000)
5 #empl.displayEmployee()
6 #emp2.displayEmployee()
7 #print ("Total Employee %d" % Employee.empCount)

```

5. Jelaskan dengan contoh pemakaian paket dengan perintah from kalkulator import Penambahan disertai dengan contoh kode lainnya. Penggunaan paket from namafile import, itu berfungsi untuk memanggil file dan fungsinya :

```

1 #import belajar
2 #a = 100
3 #b = 50

```

6. Jelaskan dengan contoh kodennya, pemakaian paket fungsi apabila file library ada di dalam folder. Pemakaian paket adalah perkumpulan fungsi-fungsi. contoh kodennya adalah sebagai berikut :
7. Jelaskan dengan contoh kodennya, pemakaian paket kelas apabila file library ada di dalam folder.

```

1 #def penanganan_error(a,b):
2 #     try :
3 #         c = a+b
4 #         print(c)
5 #     except TypeError:
6 #         print("We Are Different")

```

5.7 Tomy Prawoto

5.7.1 Soal 1

Isi jawaban soal ke-1

1. Apa itu fungsi, inputan fungsi dan kembalian fungsi dengan contoh kode program lainnya.

Dalam bahasa pemrograman Python, struktur data yang paling dasar adalah urutan atau lists. Setiap elemen-elemen berurutan akan diberi nomor posisi atau indeksnya. Indeks pertama dalam list adalah nol, indeks kedua adalah satu dan seterusnya. Python memiliki enam jenis urutan built-in, namun yang paling umum adalah list dan tuple. Ada beberapa hal yang dapat Anda lakukan dengan semua jenis list. Operasi ini meliputi pengindeksan, pengiris, penambahan, perbanyak, dan pengecekan keanggotaan. Selain itu, Python memiliki fungsi built-in untuk menemukan panjang list dan untuk menemukan elemen terbesar dan terkecilnya.

```

1 def uji():
2     print("Tugas 3")

```

Fungsi dapat membaca parameter, parameter adalah nilai yang disediakan kepada fungsi, dimana nilai ini akan menentukan output yang akan dihasilkan fungsi.

```

1 uji()
2
3 def uji_param(nama):
4     print("Nama saya :" + str(nama))

```

Statemen return digunakan untuk keluar dari fungsi. Kita juga dapat menspesifikasi nilai kembalian.

```

1 uji_param(input("Masukan Nama Kamu : "))
2
3 def uji_return(a,b):
4     r = a + b
5     return r
6
7 a = 10
8 b = 50
9 c = uji_return(a,b)
10 print(c)

```

2. Apa itu paket dan cara pemanggilan paket atau library dengan contoh kode program lainnya.

Untuk memudahkan dalam pemanggilan fungsi yang dibutuhkan, agar dapat dipanggil berulang. Cara pemanggilannya

```

1 #from fungsi_rahma import *
2 #print(penulisan(int(input("Masukan NPM kamu : "))))
3
4 #class Employee:
5 #    'Common base class for all employees'
6 #    empCount = 0
7
8 #    def __init__(self, name, salary):
9 #        self.name = name
10 #       self.salary = salary
11 #       Employee.empCount += 1
12
13 #    def displayCount(self):
14 #        print ("Total Employee %d" % Employee.empCount)

```

3. Jelaskan Apa itu kelas, apa itu objek, apa itu atribut, apa itu method dan contoh kode program lainnya masing-masing. kelas merupakan sebuah blueprint yang mewakili objek. objek adalah hasil cetakan dari sebuah kelas. method adalah suatu upaya yang digunakan oleh object.

```

1 #    def displayEmployee(self):
2 #        print ("Name : ", self.name, ", Salary: ", self.salary)

```

4. Jelaskan cara pemanggilan library kelas dari instansiasi dan pemakaiannya dengan contoh program lainnya. Cara Pemanggilannya :

pertama import terlebih dahulu filenya.
 kemudian buat variabel untuk menampung datanya.
 setelah itu panggil nama classnya dan panggil methodnya.
 Gunakan perintah print untuk menampilkan hasilnya.

```

1 #This would create first object of Employee class
2 #emp1 = Employee("Zara", 2000)
3 #This would create second object of Employee class
4 #emp2 = Employee("Manni", 5000)
5 #emp1.displayEmployee()
6 #emp2.displayEmployee()
7 #print ("Total Employee %d" % Employee.empCount)
```

5. Jelaskan dengan contoh pemakaian paket dengan perintah from kalkulator import Penambahan disertai dengan contoh kode lainnya. Penggunaan paket from namafile import, itu berfungsi untuk memanggil file dan fungsinya :

```

1 #import belajar
2 #a = 100
3 #b = 50
```

6. Jelaskan dengan contoh kodennya, pemakaian paket fungsi apabila file library ada di dalam folder. Pemakaian paket adalah perkumpulan fungsi-fungsi. contoh kodennya adalah sebagai berikut :

7. Jelaskan dengan contoh kodennya, pemakaian paket kelas apabila file library ada di dalam folder.

```

1 #def penanganan_error(a,b):
2 #     try :
3 #         c = a+b
4 #         print(c)
5 #     except TypeError:
6 #         print("We Are Different")
```

BAB 6

JUDUL BAGIAN PERTAMA

6.1 Arjun Yuda Firwanda

6.1.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\n` dsb

6.2 Dwi Yulianingsih

6.2.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\n` dsb

6.3 Harun Ar-Rasyid

6.3.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `par` dsb

6.4 Sri Rahayu

6.4.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `par` dsb

6.5 Doli Jonviter

6.5.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `par` dsb

6.6 Rahmatul Ridha

6.6.0.1 Ketrampilan Pemrograman

- Buatlah fungsi dengan inputan variabel NPM, dan melakukan print luaran huruf yang dirangkai dari tanda bintang, pagar atau plus dari NPM kita. Tanda bintang untuk NPM mod 3=0, tanda pagar untuk NPM mod 3 =1, tanda plus untuk NPM mod3=2.

```

1 #No 1
2 def printNPM(npm):
3
4     npm = list(str(npm))
5
6     angka1 = {"0": " ##### ", "1": " ## ", "2": " ##### ", "3": "
7 ##### ", "4": " #### ", "5": " ##### ", "6": " ##### ", "7": "
8 ##### ", "8": " ##### ", "9": " ##### "}
9     angka2 = {"0": "### ##", "1": "## ##", "2": "# ##", "3": "
# ##", "4": "## ##", "5": "## ##", "6": "## ##",
"7": "## ##", "8": "## ##", "9": "## ##"}
8     angka3 = {"0": "## ##", "1": "## ##", "2": "## ##", "3": "
## ##", "4": "## ##", "5": "## ##", "6": "## ##",
"7": "## ##", "8": "## ##", "9": "## ##"}
9     angka4 = {"0": "## ##", "1": "## ##", "2": "## ##", "3": "
## ##", "4": "## ##", "5": "## ##", "6": "## ##",
"7": "## ##", "8": "## ##", "9": "## ##"} 
```

```

10     angka5 = {"0": "### ##", "1": " ##", "2": " #### ", "3": "
11         ## ##", "4": " ##", "5": "# ##", "6": "# ## ##",
12         "7": " ##", "8": "## ##", "9": " ## ##"}
13     angka6 = {"0": "##### ", "1": "##", "2": "##### #####", "3": "
14         ##### ", "4": " ##", "5": "##### ", "6": "##### ##",
15         "7": " ##", "8": "##### ", "9": "##### "}
16
17     hasil1 = []
18     hasil2 = []
19     hasil3 = []
20     hasil4 = []
21     hasil5 = []
22     hasil6 = []
23
24     for x in npm:
25         hasil1.append(angka1[x])
26         hasil2.append(angka2[x])
27         hasil3.append(angka3[x])
28         hasil4.append(angka4[x])
29         hasil5.append(angka5[x])
30         hasil6.append(angka6[x])
31
32     print(*hasil1, sep=' ')
33     print(*hasil2, sep=' ')
34     print(*hasil3, sep=' ')
35     print(*hasil4, sep=' ')
36     print(*hasil5, sep=' ')
37     print(*hasil6, sep=' ')
38
39 printNPM(input("Masukan NPM anda: "))

```

2. Buatlah fungsi dengan inputan variabel berupa NPM. kemudian dengan menggunakan perulangan mengeluarkan print output sebanyak dua digit belakang NPM.

```

1 #No 2
2 def perulangan(npm):
3     hitung = 0
4     while(hitung < 24):
5         print("Halo, "+str(npm)+" apa kabar?")
6         hitung = hitung +1
7
8 perulangan(int(input("Masukan NPM : ")))

```

3. Buatlah fungsi dengan dengan input variabel string bernama NPM dan beriluaran output dengan perulangan berupa tiga karakter belakang dari NPM sebanyak penjumlahan tiga digit tersebut.

```

1 #No 3
2 def printNPMTigaDijit(npm):
3     ulang = 1
4     sampai = list(map(int, npm[4:7]))
5     sampai = sum(sampai)
6     while(ulang <= sampai):
7         print("Halo, "+str(npm[-3:])+ " apa kabar?")

```

```

8     ulang += 1
9
10 printNPMTigaDijit(input("Masukan NPM anda: "))

```

4. Buatlah fungsi hello word dengan input variabel string bernama NPM dan beri luaran output berupa digit ketiga dari belakang dari variabel NPM menggunakan akses langsung manipulasi string pada baris ketiga dari variabel NPM.

```

1 #No 4
2 def perulangan_3_digit_terakhir(npm):
3     npm = str(npm)
4     bil = npm[-3]
5     print("Halo, "+bil+" apa kabar?")
6
7 perulangan_3_digit_terakhir(int(input("Masukan NPM : ")))

```

5. buat fungsi program dengan input variabel NPM dan melakukan print nomor npm satu persatu kebawah.

```

1 #No 5
2 def down(npm):
3     for i in npm:
4         print(i)
5
6 down(input("Masukan NPM : "))

```

6. Buatlah fungsi dengan inputan variabel NPM, didalamnya melakukan penjumlahan dari seluruh dijitel NPM tersebut, wajib menggunakan perulangan dan atau kondisi.

```

1 #No 6
2 def penjumlahan(npm):
3     jumlah = 0
4     for i in npm:
5         jumlah += int(i)
6     print(str(jumlah)+" Adalah hasil penjumlahan")
7
8 penjumlahan(input("Masukan NPM : "))

```

7. Buatlah fungsi dengan inputan variabel NPM, didalamnya melakukan perkalian dari seluruh dijitel NPM tersebut, wajib menggunakan perulangan dan atau kondisi.

```

1 #No 7
2 def perkalian(npm):
3     jumlah = 0
4     for i in npm:
5         jumlah *= int(i)
6     print(str(jumlah)+" Adalah hasil perkalian")
7
8 perkalian(input("Masukan NPM : "))

```

8. Buatlah fungsi dengan inputan variabel NPM, Lakukan print NPM anda tapi hanya dijitel genap saja. wajib menggunakan perulangan dan atau kondisi.

9. Buatlah fungsi dengan inputan variabel NPM, Lakukan print NPM anda tapi hanya dijcit ganjil saja. wajib menggunakan perulangan dan atau kondisi.

```

1 #No 8
2 def genap():
3     npm = [1,1,4,4,1,2,4]
4     for i in npm:
5         if (i % 2) == 0:
6             print("Bilangan Genapnya : "+str(i))
7 genap()

```

10. Buatlah fungsi dengan inputan variabel NPM, Lakukan print NPM anda tapi hanya dikit yang termasuk bilangan prima saja. wajib menggunakan perulangan dan atau kondisi.

```

1 #No 9
2 def ganjil():
3     npm = [1,1,4,4,1,2,4]
4     for i in npm:
5         if (i%2)==1:
6             print("Bilangan Ganjilnya : "+str(i))
7 ganjil()

```

11. Buatlah satu library yang berisi fungsi-fungsi dari nomor diatas dengan nama file epi.py dan berikan contoh cara pemanggilannya pada file main.py.

```

1 lib = __import__('3lib')
2 npm = "1144124"
3
4 hasil1 = lib.jawabanNo1(npm)
5 hasil2 = lib.jawabanNo2(npm)
6 hasil3 = lib.jawabanNo3(npm)
7 hasil4 = lib.jawabanNo4(npm)
8 hasil5 = lib.jawabanNo5(npm)
9 hasil6 = lib.jawabanNo6(npm)
10 hasil7 = lib.jawabanNo7(npm)
11 hasil8 = lib.jawabanNo8(npm)
12 hasil9 = lib.jawabanNo9(npm)
13 hasil10 = lib.jawabanNo10(npm)
14 print()

```

12. Buatlah satu library class dengan nama gile kelas3lib.py yang merupakan modifikasi dari fungsi-fungsi nomor diatas dan berikan contoh cara pemanggilannya pada file mainm.py.

```

1 from kelas3lib import kelas3lib
2
3 npm = "1144124"
4
5 kel3lib = kelas3lib(npm)
6
7 kel3lib.jawabanNo1()
8 kel3lib.jawabanNo2()
9 kel3lib.jawabanNo3()

```

```

10 kel3lib.jawabanNo4()
11 kel3lib.jawabanNo5()
12 kel3lib.jawabanNo6()
13 kel3lib.jawabanNo7()
14 kel3lib.jawabanNo8()
15 kel3lib.jawabanNo9()
16 kel3lib.jawabanNo10()

```

6.6.0.2 Ketrampilan Penanganan Error Error yang di dapat dari mengerjakan tugas ini adalah type error, cara menaggulaginya dengan cara mengecheck kembali codingannya kemudian run kembali aplikasinya. Berikut contoh Penggunaan fungsi try dan exception :

```

1 #No 10
2 def prima(npm):
3     npm = str(npm)
4     bil = npm[2]
5     num = int(bil)
6     if num > 1:
7         for i in range(2,num):
8             if (num%i)==0:
9                 print("Bukan Bilangan Prima")
10                break
11            else:
12                print("Bilangan Primanya :" + str(num))
13        else:
14            print("Tidak Ada Bilangan Prima")
15 prima(int(input("Masukan NPM : ")))

```

6.7 Tomy Prawoto

6.7.1 Soal 1

- Buatlah fungsi dengan inputan variabel NPM, dan melakukan print luaran huruf yang dirangkai dari tanda bintang, pagar atau plus dari NPM kita. Tanda bintang untuk NPM mod 3=0, tanda pagar untuk NPM mod 3=1, tanda plus untuk NPM mod3=2.

```

1 #No 1
2 def printNPM(npm):
3
4     npm = list(str(npm))
5
6     angka1 = {"0": "# ##### ", "1": "#", "2": "##### ", "3": "# ##### ",
7               "4": "##", "5": "##### ", "6": "##### ", "7": "##### ",
8               "8": "##### ", "9": "##### "}
7     angka2 = {"0": "### ##", "1": "###", "2": "### ##", "3": "# ##",
8               "4": "## ##", "5": "## ##", "6": "## ##", "7": "## ##",
9               "8": "## ##", "9": "## ##"}
8     angka3 = {"0": "## ##", "1": "# ##", "2": "## ##", "3": "## ##",
9               "4": "# ##", "5": "## ##", "6": "## ##", "7": "## ##",
10              "8": "## ##", "9": "## ##"}

```

```

9     angka4 = {"0": "### ##", "1": "## ##", "2": "## ##", "3": "## ##",
10    "4": "#####", "5": "## ##", "6": "## ##", "7": "## ##",
11    "8": "## ##", "9": "#####"}
12    angka5 = {"0": "### ##", "1": "## ##", "2": "## ##", "3": "## ##",
13    "4": "## ##", "5": "## ##", "6": "## ##", "7": "## ##",
14    "8": "## ##", "9": "## ##"}
15    angka6 = {"0": "#### ", "1": "## ##", "2": "#####", "3": "## ##",
16    "4": "## ##", "5": "## ##", "6": "## ##", "7": "## ##",
17    "8": "## ##", "9": "## ##"}
18    hasil1 = []
19
20    for x in npm:
21        hasil1.append(angka1[x])
22        hasil2.append(angka2[x])
23        hasil3.append(angka3[x])
24        hasil4.append(angka4[x])
25        hasil5.append(angka5[x])
26        hasil6.append(angka6[x])
27
28        print(*hasil1, sep=' ')
29        print(*hasil2, sep=' ')
30        print(*hasil3, sep=' ')
31        print(*hasil4, sep=' ')
32        print(*hasil5, sep=' ')
33        print(*hasil6, sep=' ')
34
35 printNPM(input("Masukan NPM anda: "))

```

2. Buatlah fungsi dengan inputan variabel berupa NPM. kemudian dengan menggunakan perulangan mengeluarkan print output sebanyak dua digit belakang NPM.

```

1 #No 2
2 def perulangan(npm):
3     hitung = 0
4     while(hitung < 21):
5         print("Halo, "+str(npm)+" apa kabar?")
6         hitung = hitung + 1
7
8 perulangan(int(input("Masukan NPM : ")))

```

3. Buatlah fungsi dengan dengan input variabel string bernama NPM dan beriluaran output dengan perulangan berupa tiga karakter belakang dari NPM sebanyak penjumlahan tiga digit tersebut.

```

1 #No 3
2 def printNPMTigaDijit(npm):
3     ulang = 1
4     sampai = list(map(int, npm[4:7]))

```

```

5     sampai = sum(sampai)
6     while(ulang <= sampai):
7         print("Halo, "+str(npm[-3:])+ " apa kabar?")
8         ulang += 1
9
10 printNPMTigaDijit(input("Masukan NPM anda: "))

```

4. Buatlah fungsi hello word dengan input variabel string bernama NPM dan beri luaran output berupa digit ketiga dari belakang dari variabel NPM menggunakan akses langsung manipulasi string pada baris ketiga dari variabel NPM.

```

1 #No 4
2 def perulangan_3_digit_terakhir(npm):
3     npm = str(npm)
4     bil = npm[-3]
5     print("Halo, "+bil+" apa kabar?")
6
7 perulangan_3_digit_terakhir(int(input("Masukan NPM : ")))

```

5. buat fungsi program dengan input variabel NPM dan melakukan print nomor npm satu persatu kebawah.

```

1 #No 5
2 def down(npm):
3     for i in npm:
4         print (i)
5
6 down(input("Masukan NPM : "))

```

6. Buatlah fungsi dengan inputan variabel NPM, didalamnya melakukan penjumlahan dari seluruh dijut NPM tersebut, wajib menggunakan perulangan dan atau kondisi.

```

1 #No 6
2 def penjumlahan(npm):
3     jumlah = 0
4     for i in npm:
5         jumlah += int(i)
6     print(str(jumlah)+" Adalah hasil penjumlahan")
7
8 penjumlahan(input("Masukan NPM : "))

```

7. Buatlah fungsi dengan inputan variabel NPM, didalamnya melakukan perkalian dari seluruh dijut NPM tersebut, wajib menggunakan perulangan dan atau kondisi.

```

1 #No 7
2 def perkalian(npm):
3     jumlah = 0
4     for i in npm:
5         jumlah *= int(i)
6     print(str(jumlah)+" Adalah hasil perkalian")
7
8 perkalian(input("Masukan NPM : "))

```

8. Buatlah fungsi dengan inputan variabel NPM, Lakukan print NPM anda tapi hanya dijít genap saja. wajib menggunakan perulangan dan atau kondisi.
9. Buatlah fungsi dengan inputan variabel NPM, Lakukan print NPM anda tapi hanya dijít ganjil saja. wajib menggunakan perulangan dan atau kondisi.

```
1 #No 8
2 def genap():
3     npm = [1,1,5,4,1,2,1]
4     for i in npm:
5         if (i % 2) == 0:
6             print("Bilangan Genapnya : "+str(i))
7 genap()
```

10. Buatlah fungsi dengan inputan variabel NPM, Lakukan print NPM anda tapi hanya dikit yang termasuk bilangan prima saja. wajib menggunakan perulangan dan atau kondisi.

```
1 #No 9
2 def ganjil():
3     npm = [1,1,5,4,1,2,1]
4     for i in npm:
5         if (i%2)==1:
6             print("Bilangan Ganjilnya : "+str(i))
7 ganjil()
```

11. Buatlah satu library yang berisi fungsi-fungsi dari nomor diatas dengan nama file epi.py dan berikan contoh cara pemanggilannya pada file main.py.

```
1 lib = __import__('3lib')
2 npm = "1154121"
3
4 hasil1 = lib.jawabanNo1(npm)
5 hasil2 = lib.jawabanNo2(npm)
6 hasil3 = lib.jawabanNo3(npm)
7 hasil4 = lib.jawabanNo4(npm)
8 hasil5 = lib.jawabanNo5(npm)
9 hasil6 = lib.jawabanNo6(npm)
10 hasil7 = lib.jawabanNo7(npm)
11 hasil8 = lib.jawabanNo8(npm)
12 hasil9 = lib.jawabanNo9(npm)
13 hasil10 = lib.jawabanNo10(npm)
14 print()
```

12. Buatlah satu library class dengan nama gile kelas3lib.py yang merupakan modifikasi dari fungsi-fungsi nomor diatas dan berikan contoh cara pemanggilannya pada file mainn.py.

```
1 from kelas3lib import kelas3lib
2
3 npm = "1154121"
4
5 kelas3lib = kelas3lib(npm)
```

```
6
7 kel3lib.jawabanNo1()
8 kel3lib.jawabanNo2()
9 kel3lib.jawabanNo3()
10 kel3lib.jawabanNo4()
11 kel3lib.jawabanNo5()
12 kel3lib.jawabanNo6()
13 kel3lib.jawabanNo7()
14 kel3lib.jawabanNo8()
15 kel3lib.jawabanNo9()
16 kel3lib.jawabanNo10()
```

6.7.1.1 Ketrampilan Penanganan Error Error yang di dapat dari mengerjakan tugas ini adalah type error, cara menaggulaginya dengan cara mengecheck kembali codingannya kemudian run kembali aplikasinya. Berikut contoh Penggunaan fungsi try dan exception :

```
1 #No 10
2 def prima(npm):
3     npm = str(npm)
4     bil = npm[2]
5     num = int(bil)
6     if num > 1:
7         for i in range(2,num):
8             if (num%i)==0:
9                 print("Bukan Bilangan Prima")
10                break
11            else:
12                print("Bilangan Primanya :" + str(num))
13        else:
14            print("Tidak Ada Bilangan Prima")
15 prima(int(input("Masukan NPM : ")))
```

BAB 7

LIBRARY CSV DAN PANDAS

7.1 Arjun Yuda Firwanda

7.1.1 Fungsi File CSV, Sejarah dan Contoh

- Fungsi CSV (Comma Separated Values) merupakan format file dalam bahasa pemrograman python. CSV adalah file yang berextensi.
- File CSV merupakan file khusus yang dapat menyimpan informasi di dalam kolom. CSV memudahkan untuk memindahkan dari satu program ke program yang lain. Ketika teks dan angka disimpan dalam file CSV, mudah untuk memindahkannya dari satu program ke program lainnya.
- Contoh CSV, Microsoft Excel menggunakan format biner atau Binary Interchange (BIF). Microsoft merilis office system 2007 dengan format xml. Microsoft Excel juga mendukung format CSV, Dbase File (DBF), Symbolic Link (SYLK), Format Interchange Data (DIF).

7.1.2 Aplikasi apa saja yang dapat menciptakan file csv

- Text Editor seperti Notepad++, Sublime, Visual Studio Code, Atom.
- Program Spreedsheet seperti, Microsoft Exel, Google Spreadshare, LibreOffice.

7.1.3 Cara Menulis dan membaca file CSV di Exel atau Spreadsheet

Cara menulisnya paling atassebagai headernya, untuk mepermudah membedakan data. Baris kedua dan seterusnya itu untuk data itu sendiri. Setelah dibuat kemudian di save as dan pilih format CSV. Dan untuk membuka file yang telah dibuat cukup double klik.

7.1.4 Jelaskan Library CSV

Library CSV dibuat untuk memudahkan mengolah data dan mempermudah untuk melakukan export dan import file csv.

7.1.5 Jelaskan Library Pandas

Library Pandas dibuat agar bahasa pemograman python bisa bersaing R dan matlab, yang digunakan untuk mengolah banyak data , keperluan big data, data mining data science.

7.1.6 Jelaskan fungsi-fungsi yang terdapat pada library CSV

- Membaca File, fungsi pembacaan file output yang berupa list sebagai hasilnya.
- Menulis File, fungsi menulis file pada csv utnuk menyederhanakan contoh data mahasiswa yang terdiri field yaitu nama, npm, kelas. Dan menyimpan hasilnya dengan format datamhs.csv. Kolom atas sebagai headernya, dan kolom kedua dan seterusnya sebagai datanya.

7.1.7 Jelaskan fungsi-fungsi yang terdapat pada library Pandas

Fungsi pada library pandas juga hampir sama dengan library csv. Perbedaanya ialah library pandas penulisannya lebih sederhana dan lebih rapih.

7.2 Dwi Yulianingsih

7.2.1 Soal 1

7.2.2 Soal 1

CSV (Comma Separated Value) adalah format basis data sederhana yang dimana setiap record yang ada dipisahkan dengan tanda koma (,) atau titik

koma (;). Format data file csv dapat diolah dengan berbagai text editor dengan mudah. Anda tidak perlu (dan Anda tidak akan) membuat pengurai CSV Anda sendiri dari awal. Ada beberapa perpustakaan yang dapat diterima yang dapat Anda gunakan. Pustaka csv Python akan berfungsi untuk sebagian besar kasus. Jika pekerjaan Anda memerlukan banyak data atau analisis numerik, panda library juga memiliki kemampuan penguraian CSV, yang seharusnya menangani sisanya. Dalam bahasa pemrograman Python telah disediakan modul csv yang khusus untuk mengolah data berformat csv. Untuk memanipulasi data csv dengan python tentunya yang pertama dilakukan adalah mengimport modul csv dengan perintah import csv. File CSV biasanya dibuat oleh program yang menangani sejumlah besar data. Mereka adalah cara yang nyaman untuk mengekspor data dari spreadsheet dan basis data serta mengimpor atau menggunakan dalam program lain. Misalnya, Anda dapat mengekspor hasil program penambangan data ke file CSV dan kemudian mengimpornya ke dalam spreadsheet untuk menganalisis data, menghasilkan grafik untuk presentasi, atau menyiapkan laporan untuk publikasi. Contohnya adalah sebagai berikut :

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

7.2.3 Soal 2

Ada beberapa aplikasi yang dapat menciptakan file dengan format csv diantaranya google sheet, number di MacOS dan microsoft excel.

7.2.4 Soal 3

Cara membuat file csv di excel cukup mudah yaitu :

- Buat foldernya
- Pilih save as
- pilih file dengan format csv

Cara membaca file di csv :

- Klik data - get external data - form text

- Akan muncul Text Import Wizard, arahkan pada file csv yang ingin anda buka lalu Open.
- Setelah File terbuka, akan muncul Text Import Wizard.
- Pilih Delimited, Kemudian Next (Di sini, bisa juga menentukan baris awal yang akan di import)
- Centrang pada Tab dan Comma (Atau sesuai pengaturan File Anda) lalu Next.
- Atur Format data pada tiap kolom yang tampil dan klik Finish

7.2.5 Soal 4

CSV muncul untuk memudahkan data science dan analis karena dinilai terdapat banyak kemudahan yang didapat. CSV dapat dimaksimalkan jika dipadu dengan python karena python adalah bahasa pemrograman yang support ke banyak library termasuk csv. Maka karena itulah perpaduan python dan csv seringkali digunakan oleh perusahaan-perusahaan besar dalam mengolah datanya.

7.2.6 Soal 5

Pandas merupakan tool yang dapat digunakan sebagai alat analisis data dan struktur untuk bahasa pemrograman Python. Pandas dapat mengolah data dengan mudah, salah satu fitur yang ada dalam pandas adalah Dataframe. Fitur dataframe dapat membaca sebuah file dan menjadikannya tabble, juga dapat mengolah suatu data dengan menggunakan operasi seperti join, group by dan teknik lainnya yang terdapat pada SQL. Dalam hal ini pandas tidak jauh beda dengan csv yaitu memiliki keunggulan dalam pengolahan data-data besar dan dapat disupport dengan baik dengan python walaupun mengimport data dalam jumlah banyak.

7.2.7 Soal 6

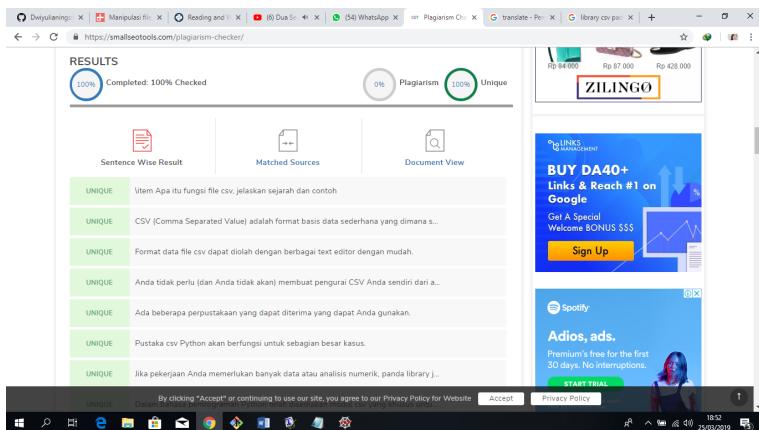
Library csv mempunyai keunggulan dibandingkan format data lainnya adalah soal kompatibilitas. File csv dapat digunakan, diolah, dieksport/impor, dan di-modifikasi menggunakan berbagai macam perangkat lunak dan bahasa pemrograman. Pada library csv mempunyai fungsi import dan eksport data yang baik dan bisa digunakan dalam jumlah besar.

7.2.8 Soal 7

pandas menyediakan beragam fungsi operasi untuk mengolah data. Contoh jika menggunakan series bisa mencari nilai max, min, dan mean secara langsung,

bahkan juga bisa melakukan operasi perpangkatan pada nilai Series secara langsung. Pandas dapat mengolah suatu data dan mengolahnya seperti join, distinct, group by, agregasi, dan teknik seperti pada SQL. Hanya saja dilakukan pada tabel yang dimuat dari file ke RAM.

7.2.9 bukti bebas plagiarisme



Gambar 7.1 SS Bebas Plagiarisme

7.3 Harun Ar-Rasyid

7.3.1 Soal 1

File CSV (Nilai Terbatas Koma) adalah jenis file khusus yang dapat Anda buat atau edit di Excel. File CSV menyimpan informasi yang dipisahkan oleh koma, tidak menyimpan informasi dalam kolom. Ketika teks dan angka disimpan dalam file CSV, mudah untuk memindahkannya dari satu program ke program lainnya. Dari rilis pertama, Excel menggunakan format file biner yang disebut Binary Interchange File Format (BIFF) sebagai format file utamanya. Ini berubah ketika Microsoft merilis Office System 2007 yang memperkenalkan Office Open XML sebagai format file utamanya. Office Open XML adalah file kontainer berbasis XML yang mirip dengan XML Spreadsheets (XMLSS), yang diperkenalkan di Excel 2002. File versi XML tidak bisa menyimpan makro VBA. Meskipun mendukung format XML baru, Excel 2007 masih mendukung format lama yang masih berbasis BIFF tradisional. Selain itu Microsoft Excel juga mendukung format Comma Separated Values (CSV), DBase File (DBF), SYMbolic LinK (SYLK), Format Interchange Data (DIF) dan banyak format lainnya, termasuk format lembar kerja 1-2 Lotus - 3 (WKS, WK1, WK2, dll.) Dan Quattro Pro.

7.3.2 Soal 2

- Texteditor Seperti notepad,visual studio code,atom,sublime dan lain sebagainya
- Program Spreadsheet Seperti excell,google spreadshare,LibreOfficecalc

7.3.3 Soal 3

Untuk menulisnya untuk yang paling atas itu kita buat headernya,untuk mempermudah membedakan datanya,dan untuk baris kedua dan seterusnya itu untuk data itu sendiri. dan setelah di buat kalian save as kemudian pilih format CSV. dan untuk membukanya cukup di double clik file tersebut.

7.3.4 Soal 4

library csv dibuat untuk permudah mengolah data. Dan mempermudah untuk melakukan export dan import file csv itu sendiri

7.3.5 Soal 5

library pandas dibuat agar bahasa pemrograman python bisa bersaing R dan matlab, yang digunakan untuk mengolah banyak data , keperluan big data, data mining data science dan sebagainya.

7.3.6 Soal 6

Terdapat 2 fungsi yang bisa digunakan oleh library csv Pertama,fungsi membaca file csv. fungsi ini bisa menggunakan list dan dictionary Dengan list :

```

1   with open('1174027.csv') 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' {", ".join(row)}')
7               line_count += 1
8           else:
9               print(f'\t Orang Ini Memiliki NPM {row[0]} Bernama \
{row[1]} Berada Dikelas {row[2]}.')
10              line_count += 1

```

Dengan dictionary :

```

1       csv_reader = csv.DictReader(csv_file)
2       line_count = 0
3       for row in csv_reader:
4           if line_count == 0:
5               print(f' {", ".join(row)}')
6               line_count += 1
7           print(f'\t Orang Ini Memiliki NPM : {row["name"]} Bernama \
: {row["department"]} Berada Dikelas : {row["birthday month"]}.')
8
9

```

```

8     line_count += 1
9
10 def nulis():

```

Kedua,fungsi menulis file csv.

```

1     employee_writer.writerow(['Ucok', 'Tukang', 'November'])
2     employee_writer.writerow(['Udin', 'Mandor', 'April'])
3
4
5 def bacalistpandas():

```

7.3.7 Soal 7

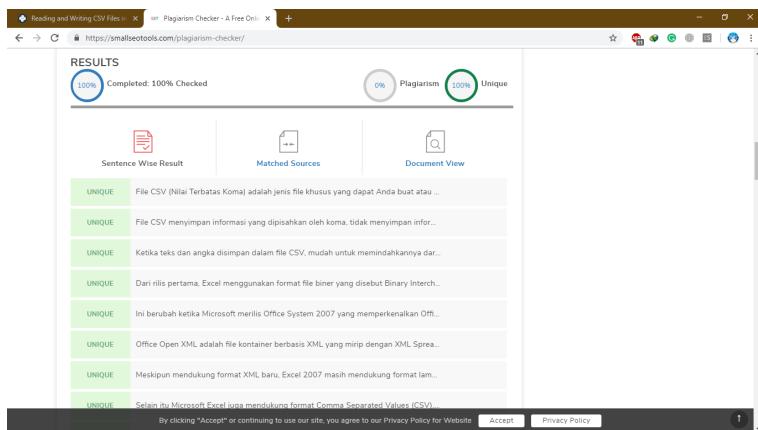
Hampir sama dengan library csv, tp library pandas penulisannya lebih sederhana dan terlihat lebih rapih dari pada library csv.

```

1 df = pandas.read_csv('1174027.csv')
2 print(df)

```

7.3.8 Bukti Bebas Plagiat



Gambar 7.2 SS Bebas Plagiarisme

7.4 Sri Rahayu

7.4.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap cukup enter aja, tidak usah pakai par dsb

7.5 Doli Jonviter

7.5.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai par dsb

7.6 Rahmatul Ridha

7.6.1 Soal 1

Kerjakan soal berikut ini, masing-masing bernilai 5 untuk hari pertama. Praktek teori penunjang yang dikerjakan dengan deadline besok jam 4 pagi :

1. Apa itu fungsi file csv, jelaskan sejarah dan contohnya.

- Apa itu Fungsi file csv

Format file csv *Comma Separated Values* yaitu suatu format data pada basis data dimana setiap record yang dapat dipisahkan dengan menggunakan tanda koma (‘,’) atau juga bisa dengan menggunakan titik koma (‘;’) sebagai tanda pemisah antara datu elemen dengan elemen yang lainnya. Selain bahasa programnya yang sederhana, format ini juga dapat dibuka dengan menggunakan berbagai *text-editor* seperti Notepad, Wordpad, dan MS Excel.

File CSV (nilai berbatas koma) merupakan tipa file khusus yang dapat dibuat atau diedit dengan menggunakan excel. File csv menyimpan informasi yang dapat dipisah oleh koma (,), bukan untuk menyimpan informasi dalam kolom. Saat teks dan angka yang disimpan dalam file csv, dapat memudahkan untuk memindahkannya dari satu program ke program yang lainnya.

- Sejarah CSV

Nilai yang dipisahkan oleh koma adalah format data yang memberi tanggal lebih awal pada komputer pribadi lebih dari satu dekade: kompiler IBM Fortran (level H extended) di bawah OS / 360 mendukungnya pada tahun 1972. Input / output yang diarahkan oleh daftar (“bentuk bebas”) didefinisikan dalam FORTRAN 77, disetujui pada tahun 1978. Input yang diarahkan dafatar menggunakan koma atau spasi untuk pembatas, sehingga string karakter yang tidak dikutip tidak dapat mengandung koma atau spasi.

Nama ”nilai yang dipisahkan koma” dan singkatan ”CSV” digunakan pada tahun 1983. Manual untuk komputer Osborne Executive, yang menggabungkan SuperCalc spreadsheet, mendokumentasikan konvensi kutipan CSV yang memungkinkan string berisi koma yang disematkan, tetapi manual tersebut tidak menentukan konvensi untuk menyematkan tanda kutip dalam string yang dikutip. Daftar nilai yang dipisahkan koma lebih mudah untuk diketik (misalnya ke dalam kartu berlubang) daripada data yang selaras dengan

kolom tetap dan cenderung menghasilkan hasil yang salah jika suatu nilai dilubangi satu kolom dari lokasi yang dituju.

Pada 2014 IETF menerbitkan RFC7111 yang menjelaskan aplikasi fragmen URI ke dokumen CSV. RFC7111 menentukan bagaimana rentang baris, kolom, dan sel dapat dipilih dari dokumen CSV menggunakan indeks posisi. Pada 2015 W3C, dalam upaya meningkatkan CSV dengan semantik formal, mempublikasikan draft rekomendasi pertama untuk standar metadata CSV, yang dimulai sebagai rekomendasi pada bulan Desember tahun yang sama.

- Contohnya

```
1 npm , nama , kelas  
2 1144124 , Rahmatul Ridha , D4TI5A  
3 1144003 , Khalid Ahmad Khadafi , D4TI5A
```

Listing 7.1 Contoh penggunaan format CSV.

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

- Text editor (Notepad, Wordpad, dan lain-lain)
- Spreadsheet (Microsoft Excel)

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

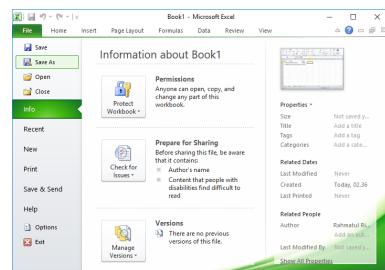
Menulis File CSV

- (a) Buat dokumen baru diexcel.
- (b) Tambahkan judul kolom untuk setiap potongan informasi yang ingin dicatat, contohnya npm, nama, kelas. Lalu ketikkan informasi dalam kolom yang sesuai.
- (c) Setelah selesai dibuat, file excel yang telah dibuat akan terlihat seperti 7.10

A	B	C
npm	nama	kelas
1144124	Rahmatul Ridha	D4TI5A
1144003	Khalid Ahmad Khadafi	D4TI5A

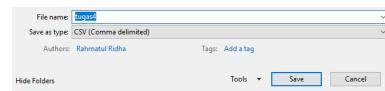
Gambar 7.3 Membuat file csv

- (d) Kemudian isi kolom ‘File name’ dengan nama file anda dan kolom ‘Save as type’ pilih yang berekstensi .csv 7.11.



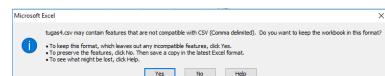
Gambar 7.4 Save as Type

- (e) Kemudian file yang Anda telah terbuat tadi tersimpan dengan ekstensi .csv. Untuk melihat isi filenya tinggal klik dua kali pada file tersebut 7.12.



Gambar 7.5 Perintah ekstensi

- (f) Lalu tinggal klik ‘Yes’7.13.



Gambar 7.6 Konfirmasi Penyimpanan

Melihat File CSV di Excel atau Spreadsheet

- (a) Pertama klik dua kali pada file yang yang berekstensi CSV7.14.



Gambar 7.7 file csv yang telah disave

- (b) Kemudian file akan terbuka secara otomatis di aplikasi Excel atau spreadsheet7.15.

	A	B	C	D	E	F	G	H	I
1	npm	name	keles						
2	1144124	Rahmatul Ridha	DAT1SA						
3	1144003	Khalid Ahmed Khadef	DAT1SA						
4									
5									
6									
7									
8									
9									
10									
11									
12									
13									
14									

Gambar 7.8 file csv pada Excel

4. Jelaskan sejarah library csv.

Format yang disebut CSV *Comma Separated Values* adalah format impor dan ekspor paling umum untuk spreadsheet dan basis data. Format CSV digunakan selama bertahun-tahun sebelum upaya untuk menggambarkan format dengan cara standar di RFC 4180. Kurangnya standar yang didefinisikan dengan baik berarti bahwa perbedaan halus sering ada dalam data yang diproduksi dan dikonsumsi oleh aplikasi yang berbeda. Perbedaan-perbedaan ini dapat membuatnya menjengkelkan untuk memproses file CSV dari berbagai sumber.

Namun, sementara pembatas dan mengutip karakter bervariasi, format keseluruhan cukup mirip sehingga dimungkinkan untuk menulis satu modul yang dapat secara efisien memanipulasi data seperti itu, menyembunyikan detail membaca dan menulis data dari programmer. Modul csv mengimplementasikan kelas untuk membaca dan menulis data tabular dalam format CSV.

Hal ini memungkinkan programmer untuk mengatakan, "tulis data ini dalam format yang disukai oleh Excel," atau "baca data dari file ini 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 mereka sendiri.

5. Jelaskan sejarah library Pandas.

Pandas merupakan toolkit yang powerfull sebagai alat untuk analisis data dan struktur pada bahasa pemrograman Python. Dengan menggunakan pandas kita dapat mengolah data dengan mudah, salah satunya yaitu fiturnya adalah Dataframe. Dengan adanya fitur dataframe kita dapat membaca sebuah file dan menjadikannya tabel, kita juga dapat mengolah suatu data dengan menggunakan operasi seperti join, distinct, group by, agregasi, dan teknik lainnya yang terdapat pada SQL. Saat ini banyak format file yang dapat dibaca menggunakan Pandas, seperti file .txt, .csv, .tsv dan lainnya.

6. Jelaskan fungsi-fungsi yang terdapat diliibrary csv.

(a) reader

Fungsi ini digunakan untuk membaca isi file berformat CSV dari list.

```
1 #Membaca File CSV dengan Fungsi reader dengan library CSV
2 import csv
3
4 with open('teori.csv') as csv_file:
5     csv_reader = csv.reader(csv_file, delimiter=',')
6     for row in csv_reader:
7         print(row[0], row[1], row[2])
```

Listing 7.2 Membaca file berformat CSV list.

(b) DictReader

Fungsi ini digunakan untuk membaca isi file berformat CSV dari dictionary.

```
1 #Membaca File CSV dengan Fungsi DictReader dengan library CSV
2 import csv
3
4 with open('teori.csv', mode='r') as csv_file:
5     csv_reader = csv.DictReader(csv_file)
6     for row in csv_reader:
7         print(row['npm'], row['nama'], row['kelas'])
```

Listing 7.3 Membaca file berformat CSV dictionary.

(c) write

Fungsi ini digunakan untuk menulis file berformat CSV dari list.

```
1 #Menulis File CSV dengan Fungsi writer dengan library CSV
2 import csv
3
4 with open('teori2.csv', mode='w') as csv_file:
5     csv_writer = csv.writer(csv_file, delimiter=',', quotechar
6     ="'", quoting=csv.QUOTE_MINIMAL)
7     csv_writer.writerow(['npm', 'nama', 'kelas'])
8     csv_writer.writerow(['1144124', 'Anggreini Kharisma', 'D4TI2B'])
9     csv_writer.writerow(['1144016', 'Dino Maulana Putra', 'D4TI2C'])
```

Listing 7.4 Menulis file berformat CSV list.

(d) DictWrite

Fungsi ini digunakan untuk menulis file berformat CSV dari dictionary.

```
1 #Menulis File CSV dengan Fungsi DictWriter dengan library CSV
2 import csv
3
4 with open('teori3.csv', mode='w') as csv_file:
5     fieldnames = ['npm', 'nama', 'kelas']
6     writer = csv.DictWriter(csv_file, fieldnames=fieldnames)
7
8     writer.writeheader()
9     writer.writerow({'npm': '1144002', 'nama': 'Anne Marie', 'kelas
10 : 'D4TI2A'})
11     writer.writerow({'npm': '1144065', 'nama': 'Lisa', 'kelas
12 : 'D4TI2A'})
```

Listing 7.5 Menulis file berformat CSV dictionary.

7. Jelaskan fungsi-fungsi yang terdapat di library pandas.

(a) `read_csv`

Fungsi ini digunakan untuk membaca isi file berformat CSV

```
1 #Membaca File CSV dengan Fungsi read_csv dengan Library Pandas
2 import pandas
3
4 df = pandas.read_csv('teori.csv')
5 print(df)
```

Listing 7.6 Membaca file berformat CSV pandas.

(b) `to_csv`

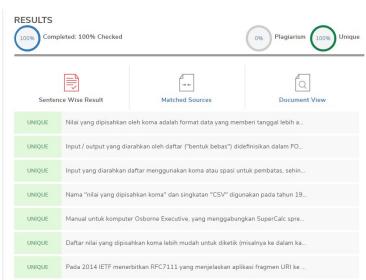
Fungsi ini digunakan untuk menulis file berformat CSV

```
1 #Menulis File CSV dengan Fungsi to_csv dengan Library Pandas
2 import pandas
3
4 df = pandas.read_csv('teori.csv')
5 df.to_csv('teori4.csv')
```

Listing 7.7 Menulis file berformat CSV pandas.

8. Cek plagiarisme

Berikut adalah cek plagiarisme pada teorinya pada 7.9



Gambar 7.9 Plagiarisme

7.7 Tomy Prawoto

7.7.1 Soal 1

Isi jawaban soal ke-1

1. Apa itu fungsi file csv, jelaskan sejarah dan contohnya.

- Apa itu Fungsi file csv

Format file csv *Comma Separated Values* yaitu suatu format data pada basis data dimana setiap record yang dapat dipisahkan dengan menggunakan tanda koma (',') atau juga bisa dengan menggunakan titik koma (';') sebagai tanda pemisah antara datu elemen dengan elemen yang lainnya. Selain bahasa programnya yang sederhana, format ini juga dapat dibuka dengan menggunakan berbagai *text-editor* seperti Notepad, Wordpad, dan MS Excel.

File CSV (nilai berbatas koma) merupakan tipa file khusus yang dapat dibuat atau diedit dengan menggunakan excel. File csv menyimpan informasi yang dapat dipisah oleh koma (,), bukan untuk menyimpan informasi dalam kolom. Saat teks dan angka yang disimpan dalam file csv, dapat memudahkan untuk memindahkannya dari satu program ke program yang lainnya.

▪ Sejarah CSV

Nilai yang dipisahkan oleh koma adalah format data yang memberi tanggal lebih awal pada komputer pribadi lebih dari satu dekade: kompiler IBM Fortran (level H extended) di bawah OS / 360 mendukungnya pada tahun 1972. Input / output yang diarahkan oleh daftar ("bentuk bebas") didefinisikan dalam FORTRAN 77, disetujui pada tahun 1978. Input yang diarahkan daftar menggunakan koma atau spasi untuk pembatas, sehingga string karakter yang tidak dikutip tidak dapat mengandung koma atau spasi.

Nama "nilai yang dipisahkan koma" dan singkatan "CSV" digunakan pada tahun 1983. Manual untuk komputer Osborne Executive, yang menggabungkan SuperCalc spreadsheet, mendokumentasikan konvensi kutipan CSV yang memungkinkan string berisi koma yang disematkan, tetapi manual tersebut tidak menentukan konvensi untuk menyematkan tanda kutip dalam string yang dikutip. Daftar nilai yang dipisahkan koma lebih mudah untuk diketik (misalnya ke dalam kartu berlubang) daripada data yang selaras dengan kolom tetap dan cenderung menghasilkan hasil yang salah jika suatu nilai dilubangi satu kolom dari lokasi yang dituju.

Pada 2014 IETF menerbitkan RFC7111 yang menjelaskan aplikasi fragmen URI ke dokumen CSV. RFC7111 menentukan bagaimana rentang baris, kolom, dan sel dapat dipilih dari dokumen CSV menggunakan indeks posisi. Pada 2015 W3C, dalam upaya meningkatkan CSV dengan semantik formal, mempublikasikan draft rekomendasi pertama untuk standar metadata CSV, yang dimulai sebagai rekomendasi pada bulan Desember tahun yang sama.

▪ Contohnya

```
1 npm , nama , kelas  
2 1154016 , Doli Jonviter , D4TI4C  
3 1154116 , Benedictus , D4TI4C
```

Listing 7.8 Contoh penggunaan format CSV.

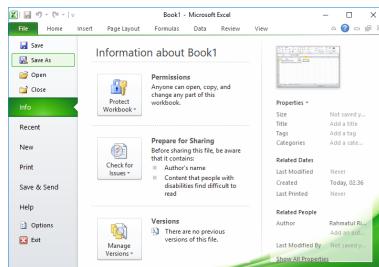
2. Aplikasi-aplikasi apa saja yang bisa menciptakan file csv ?

- Text editor (Notepat, Wordpad, dan lain-lain)
 - Spreadsheet (Microsoft Excel)
3. Jelaskan bagaimana cara menulis dan membaca file csv diexcel atau spreadsheet.
- Menulis File CSV**
- Buat dokumen baru diexcel.
 - Tambahkan judul kolom untuk setiap potongan informasi yang ingin dicatat, contohnya npm, nama, kelas. Lalu ketikkan informasi dalam kolom yang sesuai.
 - Setelah selesai dibuat, file excel yang telah dibuat akan terlihat seperti 7.10

NPM	Nama	Kelas
1154016	Doli jonviter	D4TI4C
1154116	Benedictus	D4TI4C

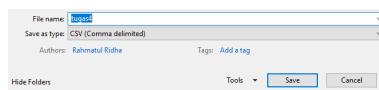
Gambar 7.10 Membuat file csv

- Kemudian isi kolom ‘File name’ dengan nama file anda dan kolom ‘Save as type’ pilih yang berekstensi .csv 7.11.



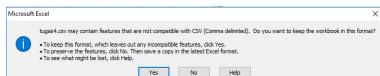
Gambar 7.11 Save as Type

- Kemudian file yang Anda telah terbuat tadi tersimpan dengan ekstensi .csv. Untuk melihat isi filenya tinggal klik dua kali pada file tersebut 7.12.



Gambar 7.12 Perintah ekstensi

- Lalu tinggal klik ‘Yes’ 7.13.



Gambar 7.13 Konfirmasi Penyimpanan

Melihat File CSV di Excel atau Spreadsheet

- (a) Pertama klik dua kali pada file yang berekstensi CSV7.14.

Gambar 7.14 file csv yang telah disave

- (b) Kemudian file akan terbuka secara otomatis di aplikasi Excel atau spreadsheet7.15.

A	B	C	D	E	F	G	H	I
1	NPM	Nama	Kelas					
2	11540116 Dolly Jonviter		D4TI4C					
3	11541116 Benedictus		D4TI4C					
4								
5								
6								
7								
8								
9								
10								
11								
12								
13								

Gambar 7.15 file csv pada Excel

4. Jelaskan sejarah library csv.

Format yang disebut CSV *Comma Separated Values* adalah format impor dan ekspor paling umum untuk spreadsheet dan basis data. Format CSV digunakan selama bertahun-tahun sebelum upaya untuk menggambarkan format dengan cara standar di RFC 4180. Kurangnya standar yang didefinisikan dengan baik berarti bahwa perbedaan halus sering ada dalam data yang diproduksi dan dikonsumsi oleh aplikasi yang berbeda. Perbedaan-perbedaan ini dapat membuatnya menjengkelkan untuk memproses file CSV dari berbagai sumber.

Namun, sementara pembatas dan mengutip karakter bervariasi, format keseluruhan cukup mirip sehingga dimungkinkan untuk menulis satu modul yang dapat secara efisien memanipulasi data seperti itu, menyembunyikan detail membaca dan menulis data dari programmer. Modul csv mengimplementasikan kelas untuk membaca dan menulis data tabular dalam format CSV.

Hal ini memungkinkan programmer untuk mengatakan, "tulis data ini dalam format yang disukai oleh Excel," atau "baca data dari file ini 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 mereka sendiri.

5. Jelaskan sejarah library Pandas.

Pandas merupakan toolkit yang powerfull sebagai alat untuk analisis data dan struktur pada bahasa pemrograman Python. Dengan menggunakan pandas kita dapat mengolah data dengan mudah, salah satunya yaitu fiturnya adalah Dataframe. Dengan adanya fitur dataframe kita dapat membaca sebuah file dan menjadikannya tabel, kita juga dapat mengolah suatu data dengan menggunakan operasi seperti join, distinct, group by, agregasi, dan teknik lainnya yang terdapat pada SQL. Saat ini banyak format file yang dapat dibaca menggunakan Pandas, seperti file .txt, .csv, .tsv dan lainnya.

6. Jelaskan fungsi-fungsi yang terdapat diliibrary csv.

(a) reader

Fungsi ini digunakan untuk membaca isi file berformat CSV dari list.

```
1 #Membaca File CSV dengan Fungsi reader dengan library CSV
2 import csv
3
4 with open('teori.csv') as csv_file:
5     csv_reader = csv.reader(csv_file, delimiter=',')
6     for row in csv_reader:
7         print(row[0], row[1], row[2])
```

Listing 7.9 Membaca file berformat CSV list.

(b) DictReader

Fungsi ini digunakan untuk membaca isi file berformat CSV dari dictionary.

```
1 #Membaca File CSV dengan Fungsi DictReader dengan library CSV
2 import csv
3
4 with open('teori.csv', mode='r') as csv_file:
5     csv_reader = csv.DictReader(csv_file)
6     for row in csv_reader:
7         print(row['npm'], row['nama'], row['kelas'])
```

Listing 7.10 Membaca file berformat CSV dictionary.

(c) write

Fungsi ini digunakan untuk menulis file berformat CSV dari list.

```
1 #Menulis File CSV dengan Fungsi writer dengan library CSV
2 import csv
3
4 with open('teori2.csv', mode='w') as csv_file:
5     csv_writer = csv.writer(csv_file, delimiter=',', quotechar="\"", quoting=csv.QUOTE_MINIMAL)
```

```

6     csv_writer.writerow(['npm', 'nama', 'kelas'])
7     csv_writer.writerow(['1154016', 'Doli Jonviter', 'D4TI4C']
8     )
9     csv_writer.writerow(['1154116', 'Benedictus', 'D4TI4C'])

```

Listing 7.11 Menulis file berformat CSV list.

(d) DictWrite

Fungsi ini digunakan untuk menulis file berformat CSV dari dictionary.

```

1 #Menulis File CSV dengan Fungsi DictWriter dengan library CSV
2 import csv
3
4 with open('teori3.csv', mode='w') as csv_file:
5     fieldnames = ['npm', 'nama', 'kelas']
6     writer = csv.DictWriter(csv_file, fieldnames=fieldnames)
7
8     writer.writeheader()
9     writer.writerow({'npm': '1154125', 'nama': 'Yoshi Nababan',
10                      'kelas': 'D4TI4A'})
11    writer.writerow({'npm': '1154123', 'nama': 'Simon', 'kelas':
12                      'D4TI4A'})

```

Listing 7.12 Menulis file berformat CSV dictionary.

7. Jelaskan fungsi-fungsi yang terdapat di library pandas.

(a) read_csv

Fungsi ini digunakan untuk membaca isi file berformat CSV

```

1 #Membaca File CSV dengan Fungsi read_csv dengan Library Pandas
2 import pandas
3
4 df = pandas.read_csv('teori.csv')
5 print(df)

```

Listing 7.13 Membaca file berformat CSV pandas.

(b) to_csv

Fungsi ini digunakan untuk menulis file berformat CSV

```

1 #Menulis File CSV dengan Fungsi to_csv dengan Library Pandas
2 import pandas
3
4 df = pandas.read_csv('teori.csv')
5 df.to_csv('teori4.csv')

```

Listing 7.14 Menulis file berformat CSV pandas.

BAB 8

PRAKTEK LIBRARY CSV DAN PANDAS

8.1 Arjun Yuda Firwanda

8.1.1 Soal 1

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

```
1 #nol
2 import csv
3
4 with open('cobadulu.txt') as csv_file:
5     csv_reader = csv.reader(csv_file, delimiter=',')
6     line_count = 0
7     for row in csv_reader:
8         if line_count == 0:
9             print(f'Kolom nya adalah {", ".join(row)}')
10            line_count += 1
11        else:
12            print(f'\t{row[0]} kerja di {row[1]} lahir pada bulan
13            {row[2]}.')
14            line_count += 1
15            print(f'Processed {line_count} lines.')
```

8.1.2 Soal 2

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

```

1 #no2
2 import csv
3
4 with open('cobadulu.txt', mode='r') as csv_file:
5     csv_reader = csv.DictReader(csv_file)
6     line_count = 0
7     for row in csv_reader:
8         if line_count == 0:
9             print(f'kolom nya adalah {", ".join(row)}')
10            line_count += 1
11            print(f'\t{row["nama"]} kerja di {row["kerjaan"]} department ,
12            dan lahir pada bulan {row["bulan"]}.')
13            line_count += 1
14            print(f'Processed {line_count} lines.')
15
16 def bacacsvlist():
17     with open('cobadulu.txt') as csv_file:
18         csv_reader = csv.reader(csv_file, delimiter=',')
19         line_count = 0
20         for row in csv_reader:
21             if line_count == 0:
22                 print(f' {", ".join(row)}')
23                 line_count += 1
24             else:
25                 print(f'\t Orang Ini {row[0]} \ {row[1]} lahir {row
26 [2]}.')
27                 line_count += 1
28
29 def nulis():
30     with open('test-tulis.csv', mode='w') as employee_file:
31         employee_writer = csv.writer(employee_file, delimiter=',',
32         quotechar='"', quoting=csv.QUOTE_MINIMAL)
33
34         employee_writer.writerow(['Ucok', 'Tukang', 'Desember'])
35         employee_writer.writerow(['Butet', 'Mandor', 'Januari'])

```

8.1.3 Soal 3

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

```

1 #no3
2 df = pandas.read_csv('isi.csv')
3 print(df)

```

8.1.4 Soal 4

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

```
1 #no4
2 df = pandas.read_csv('isi.csv')
3 uji = pandas.DataFrame.from_dict(df)
4 print(uji)
```

8.1.5 Soal 5

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

```
1 #no5
2 df = pandas.read_csv('isi.csv', parse_dates=['Hire Date'])
3 print(df)
```

8.1.6 Soal 6

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

```
1 #no6
2 df = pandas.read_csv('isi.csv', index_col='Name')
3 print(df)
```

8.1.7 Soal 7

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

```
1 #no7
2 df = pandas.read_csv('isi.csv',
3                      header=0,
4                      names=['Nama', 'Tgl Masuk', 'Gaji', 'Jatah Sakit'])
5 print(df)
6
7 def bacalistpandas():
8     df = pandas.read_csv('isi.csv')
9     print(df)
10
11 def write():
12     df = pandas.read_csv('isi.csv',
13                           index_col='Employee',
14                           parse_dates=['Hired'],
15                           header=0,
16                           names=['Employee', 'Hired', 'Salary', 'Sick Days'])
17     df.to_csv('a1174008_pandas_baru.csv')
```

8.1.8 Soal 8

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

```
1 import a1174008_csv
2 a1174008_csv.bacacsvlist()
3 a1174008_csv.nulis()
```

8.1.9 Soal 9

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

```
1 import a1174008_pandas
2 a1174008_pandas.bacalistpandas()
3 a1174008_pandas.write()
```

8.1.10 Penanganan Error

Dalam praktek kali ini belum menemukan error

8.2 Dwi Yulianingsih

8.2.1 Soal 1

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

```
1 with open('coba.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'Kolom nya adalah {" ".join(row)}')
7             line_count += 1
8         else:
9             print(f'\t{row[0]} kerja di {row[1]} lahir pada bulan
{row[2]}.')
10            line_count += 1
11            print(f'Processed {line_count} lines.')
```

8.2.2 Soal 2

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

```
1 #no2
2 import csv
```

```

3 with open('coba.txt', mode='r') as csv_file:
4     csv_reader = csv.DictReader(csv_file)
5     line_count = 0
6     for row in csv_reader:
7         if line_count == 0:
8             print(f'kolom nya adalah {", ".join(row)}')
9             line_count += 1
10            print(f'\t{row["nama"]} kerja di {row["kerjaan"]} department ,
11            dan lahir pada bulan {row["bulan"]}.')
12            line_count += 1
13    print(f'Processed {line_count} lines.')

```

8.2.3 Soal 3

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

```

1 import pandas
2 #no3
3 df = pandas.read_csv('isi.csv')
4 print(df)

```

8.2.4 Soal 4

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

```

1 #no4
2 df = pandas.read_csv('isi.csv')
3 uji = pandas.DataFrame.from_dict(df)
4 print(uji)

```

8.2.5 Soal 5

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

```

1 #no5
2 df = pandas.read_csv('isi.csv', parse_dates=['Hire Date'])
3 print(df)

```

8.2.6 Soal 6

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

```

1 #no6
2 df = pandas.read_csv('isi.csv', index_col='Name')
3 print(df)

```

8.2.7 Soal 7

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

```
1 #no7
2 df = pandas.read_csv('isi.csv',
3                      header=0,
4                      names=['Nama', 'Tgl Masuk', 'Gaji', 'Jatah Sakit'])
5 print(df)
```

8.2.8 Soal 8

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

```
1 import d1174009_csv
2 d1174009_csv.bacacsvlist()
3 d1174009_csv.nulis()
```

8.2.9 Soal 9

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

```
1 import d1174009_pandas
2 d1174009_pandas.bacalistpandas()
3 d1174009_pandas.write()
```

8.2.10 Penanganan eror

Ada kalanya saat kita baca file, tapi filenya belum ada. Maka biasanya akan terjadi IOError.

```
1 #IOError: [Errno 2] No such file or directory: 'file.txt'
```

maka ditangani dengan cara seperti dibawah ini :

```
1 #try :
2 #    f = open("file.txt","r")
3 #except IOError as err:
4 #    print "Terjadi kesalahan: {}".format(err)
```

maka akan muncul peringatan seperti dibawah :

```
1 #Terjadi kesalahan: [Errno 2] No such file or directory: 'file.txt'
```

8.3 Harun Ar-Rasyid

8.3.1 Soal 1

Berikut adalah pemanggilan file csv dengan library csv yang menggunakan list

```

1 def bacacsvlist():
2     with open('1174027.csv') 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' {", ".join(row)}')
8                 line_count += 1
9             else:
10                print(f'\t Orang Ini Memiliki NPM {row[0]} Bernama \
{row[1]} Berada Dikelas {row[2]}.')
11                line_count += 1

```

8.3.2 Soal 2

Berikut adalah pemanggilan file csv dengan library csv yang menggunakan dictionary

```

1 def bacacsvdictionary():
2     with open('1174027.csv', 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' {", ".join(row)}')
8                 line_count += 1
9             print(f'\t Orang Ini Memiliki NPM : {row["name"]} Bernama
: {row["department"]} Berada Dikelas : {row["birthday month"]}.')
10            line_count += 1

```

8.3.3 Soal 3

Berikut adalah pemanggilan file csv dengan library pandas yang menggunakan list

```

1 def bacalistpandas():
2     df = pandas.read_csv('1174027.csv')
3     print(df)

```

8.3.4 Soal 4

Berikut adalah pemanggilan file csv dengan library pandas yang menggunakan dictionary

```

1 def bacadictpandas():
2     df = pandas.read_csv('1174027.csv')
3     uji = pandas.DataFrame.from_dict(df)
4     print(uji)

```

8.3.5 Soal 5

Berikut penggunaan untuk merubah standar penulisan tanggal, yang mengikuti standar penulisan dari pandas.

```
1 def standartanggal():
2     df = pandas.read_csv('1174027.csv', parse_dates=['ttl'])
3     print(df)
```

8.3.6 Soal 6

Berikut merupakan pergantian index kolom

```
1 def changeindexcol():
2     df = pandas.read_csv('1174027.csv', index_col='npm')
3     print(df)
```

8.3.7 Soal 7

berikut merupakan penggunaan untuk merename atribut yang digunakan, atau merubah nama header 0

```
1 def renameatt():
2     df = pandas.read_csv('1174027.csv',
3                           header=0,
4                           names=['Nomor Induk Mahasiswa', 'Name', 'Class', 'Tanggal
Lahir'])
5     print(df)
```

8.3.8 Soal 8

```
1 import c_1174027.csv
2 c_1174027.csv.bacacsvlist()
3 c_1174027.csv.nulis()
```

8.3.9 Soal 9

```
1
2 import p_1174027-pandas
3 p_1174027-pandas.bacalistpandas()
4 p_1174027-pandas.write()
```

8.3.10 Penanganan Error

Dalam praktek kali ini alhamdulliha tidak menemukan error

8.4 Sri Rahayu

8.4.1 Soal 1

Isi jawaban soal ke-1

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `\par` dsb

8.5 Doli Jonviter

8.5.1 Soal 1

- Berikut adalah pemanggilan file csv dengan library csv yang menggunakan list

```

1 def bacacsvlist():
2     with open('1154016.csv') 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' {", ".join(row)}')
8                 line_count += 1
9             else:
10                print(f'\t NPM {row[0]} Bernama \ {row[1]}\n'
11 Belajar dikelas {row[2]}')
12                line_count += 1

```

- Berikut adalah pemanggilan file csv dengan library csv yang menggunakan dictionary

```

1 def bacacsvdictionary():
2     with open('1154016.csv', 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' {", ".join(row)}')
8                 line_count += 1
9             print(f'\t NPM : {row["name"]} Nama : {row["'
10 department"]}\nBelajar Dikelas : {row["birthday month"]}.')
11             line_count += 1

```

- Berikut adalah pemanggilan file csv dengan library pandas yang menggunakan list

```

1
2 def bacalistpandas():
3     df = pandas.read_csv('1154016.csv')

```

- Berikut adalah pemanggilan file csv dengan library pandas yang menggunakan dictionary

```

1
2 def bacadictpandas():
3     df = pandas.read_csv('1154016.csv')
4     uji = pandas.DataFrame.from_dict(df)

```

- Berikut penggunaan untuk merubah standar penulisan tanggal, yang mengikuti standar penulisan dari pandas.

```

1 def standartanggal():
2     df = pandas.read_csv('1154016.csv', parse_dates=[ 'date' ])

```

6. Berikut merupakan pergantian index kolom

```

1 def changeindexcol():
2     df = pandas.read_csv('1154016.csv', index_col='npm')

```

7. berikut merupakan penggunaan untuk merename atribut yang digunakan, atau merubah nama header 0

```

1 def renameatt():
2     df = pandas.read_csv('1154016.csv',
3                           header=0,
4                           names=['Nomor Induk Mahasiswa', 'Name', 'Class', ' '
5                           'Angkatan'])

```

8.5.2 Soal 8

```

1 import c_1154016.csv
2 c_1154016.csv.bacacsvlist()
3 c_1154016.csv.nulis()

```

8.5.3 Soal 9

```

1
2 import p_1154016_pandas
3 p_1154016_pandas.bacalistpandas()
4 p_1154016_pandas.write()

```

8.6 Rahmatul Ridha

8.6.1 Keterampilan Pemrograman

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

```

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])

```

Listing 8.1 Fungsi untuk membuka file CSV dengan lib CSV mode list.

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

```

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'])

```

Listing 8.2 Fungsi untuk membuka file CSV dengan lib CSV mode dictionary.

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

```

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

```

Listing 8.3 Fungsi untuk membuka file CSV dengan lib Pandas mode list.

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

```

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

```

Listing 8.4 Fungsi untuk membuka file CSV dengan lib Pandas mode dictionary.

- Buat fungsi baru di NPMpandas.py untuk mengubah format tanggal menjadi standar dataframe.

```

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

```

Listing 8.5 Fungsi untuk mengubah format tanggal menjadi standar dataframe.

- Buat fungsi baru di NPMpandas.py untuk mengubah index kolom.

```

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

```

Listing 8.6 Fungsi untuk mengubah index kolom.

- Buat fungsi baru di NPMpandas.py untuk mengubah atribut atau nama kolom.

```

1 #Jawaban No. 6
2 def ubahIndexKolom():
3     df = pandas.read_csv('teori.csv')

```

```

4 df.index = [ 'Row_1' , 'Row_2' ]
5 print(df)

```

Listing 8.7 Fungsi untuk mengubah atribut atau nama kolom.

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

```

1 lib = __import__('1144124_csv')
2
3 csv = lib.bukaModeListCsv();

```

Listing 8.8 Membuat dan mebaca file CSV menggunakan library 1144124pandas.

- Buat program main2.py yang menggunakan library NPMpandas.py yang membuat dan membaca file csv.

```

1 lib = __import__('1144124_pandas')
2
3 pandas = lib.bukaModeListPandas();

```

Listing 8.9 Membuat dan mmebaca file CSV menggunakan library 1144124pandas.

8.6.2 Penanganan eror

Saat akan membaca file, akan tetapi file yang akan dibaca belum ada. Maka biasanya akan terjadi IOError.

```
1 #IOError: [Errno 2] No such file or directory: 'file.txt'
```

maka penanganannya dengan cara seperti dibawah ini :

```

1 #try:
2 #    f = open("file.txt","r")
3 #except IOError as err:
4 #    print "Terjadi kesalahan: {}".format(err)

```

dan akan muncul peringatan seperti dibawah :

```
1 #Terjadi kesalahan: [Errno 2] No such file or directory: 'file.txt'
```

8.7 Tomy Prawoto

8.7.1 Soal Praktek

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

```

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])

```

Listing 8.10 Fungsi untuk membuka file CSV dengan lib CSV mode list.

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

```

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'])

```

Listing 8.11 Fungsi untuk membuka file CSV dengan lib CSV mode dictionary.

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

```

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

```

Listing 8.12 Fungsi untuk membuka file CSV dengan lib Pandas mode list.

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

```

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

```

Listing 8.13 Fungsi untuk membuka file CSV dengan lib Pandas mode dictionary.

5. Buat fungsi baru di NPMpandas.py untuk mengubah format tanggal menjadi standar dataframe.

```

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

```

Listing 8.14 Fungsi untuk mengubah format tanggal menjadi standar dataframe.

6. Buat fungsi baru di NPMpandas.py untuk mengubah index kolom.

```

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

```

Listing 8.15 Fungsi untuk mengubah index kolom.

7. Buat fungsi baru di NPMpandas.py untuk mengubah atribut atau nama kolom.

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

Listing 8.16 Fungsi untuk mengubah atribut atau nama kolom.

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

```
1 lib = __import__('1144124_csv')
2
3 csv = lib.bukaModeListCsv();
```

Listing 8.17 Membuat dan membaca file CSV menggunakan library 1144124pandas.

9. Buat program main2.py yang menggunakan library NPMpandas.py yang membuat dan membaca file csv.

```
1 lib = __import__('1154121_pandas')
2
3 pandas = lib.bukaModeListPandas();
```

Listing 8.18 Membuat dan membaca file CSV menggunakan library 1154121pandas.

BAB 9

KOMUNIKASI PERANGKAT KERAS

9.1 Rahmatul Ridha

9.1.1 Teori

9.1.1.1 Soal No. 1 Apa itu fungsi device manager di windows dan folder /dev di linux?

Device manager merupakan perangkat lunak yang berfungsi untuk menampilkan seluruh perangkat keras yang telah di-inisialisasi atau dikenali oleh sistem operasi Windows. Device Manager membantu dalam mengelola atau me-manage semua perangkat keras yang terpasang 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. Mengelola driver perangkat keras.
2. Menunjukkan informasi detail mengenai suatu perangkat keras.
3. Mengidentifikasi konflik antar perangkat keras.

4. Menonaktifkan dan mengaktifkan perangkat keras.
5. Memberitahukan terjadinya masalah pada perangkat keras.
6. Menunjukkan status mengenai suatu perangkat keras.

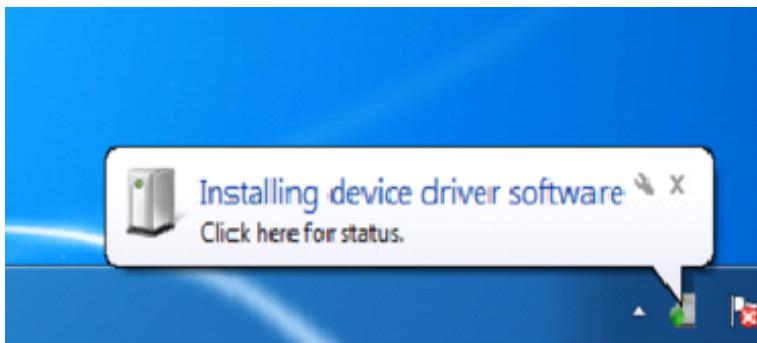
Folder /dev merupakan representasi dari drive yang terhubung sudah ke sistem operasi Linux dan oleh sistem yang dianggap sebagai file-file direktori. Biasanya sering ditampilkan direktori seperti /dev/sdal yang mewakili Drive SATA pertama dalam sistem.

9.1.2 Soal No. 2

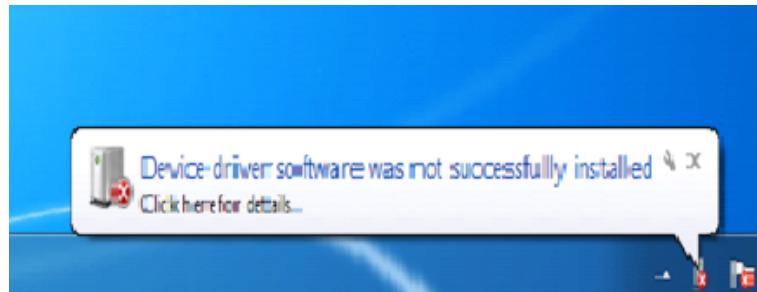
Jelaskan langkah-langkah instalasi driver dari arduino!

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

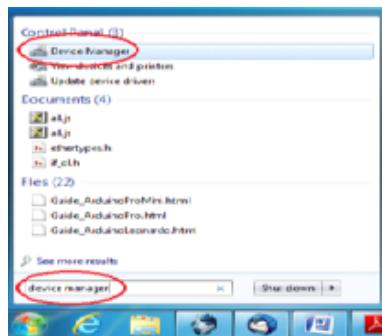
1. Pertama pastikan Arduino IDE telah terinstall.
2. Lalu hubungkan port USB Arduino Uno ke port USB PC.
3. Kemudian PC anda akan mendeteksi perangkat baru yang terpasang dan akan muncul pop seperti 3.



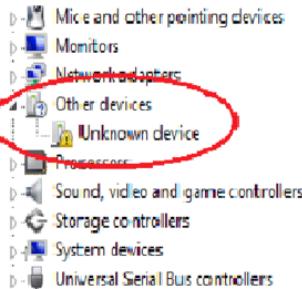
4. Karena Arduino Uno baru pertama kali terpasang, maka akan muncul pop up error seperti ini.



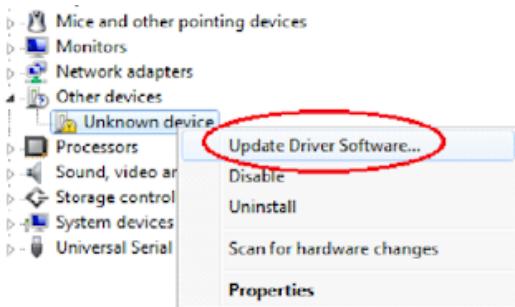
5. Buka ‘Start’ lalu cari Device Manager, kemudian klik ‘Device Manager’.



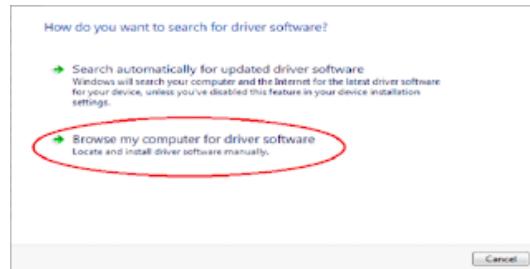
6. Setelah Device Manager terbuka, silahkan cari ‘Unknown Device’ yang berada di Other Device.



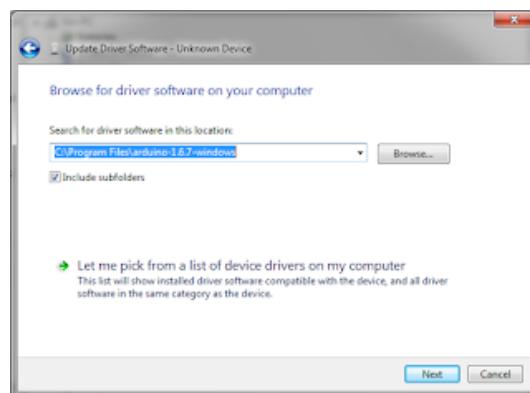
7. Kemudian klik kanan pada ‘Unknown Device’, lalu pilih ‘Update Driver Software’.



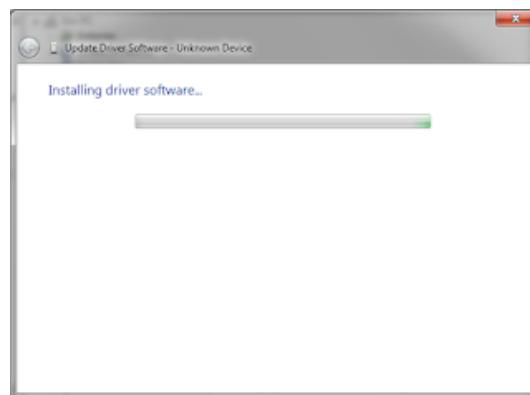
8. Setelah itu muncul window baru, lalu pilih ‘Browse my computer for driver software’.



9. Lalu cari folder yang terinstall Arduino IDE dengan mengklik browse. Kemudian klik ‘Next’.



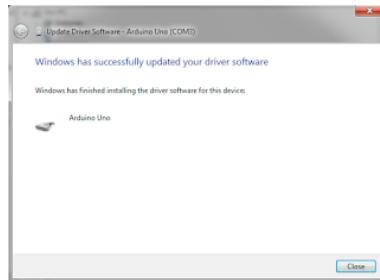
10. Windows akan mencari dan menginstall driver yang berada pada folder tersebut.



11. Setelah itu akan muncul window, lalu klik ‘Install’.



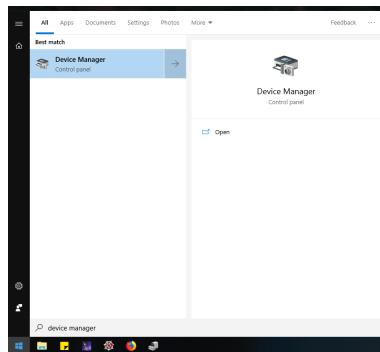
12. Jika berhasil terinstal maka akan muncul window seperti ini.



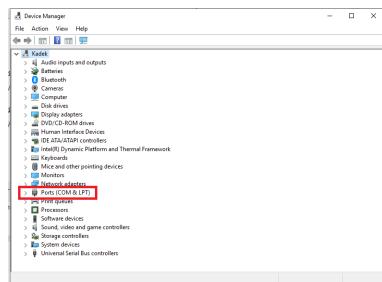
9.1.3 Soal No. 3

Jelaskan bagaimana cara membaca baudrate dan port dari komputer yang sudah terinstall driver! **Membaca Baudrate dari Komputer**

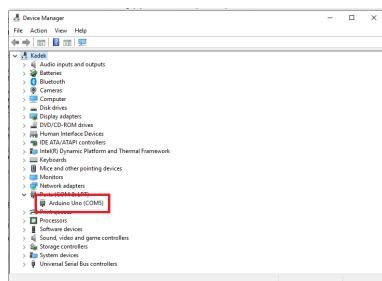
1. Pertama buka 'Start'. Cari 'Device Manager', lalu klik.



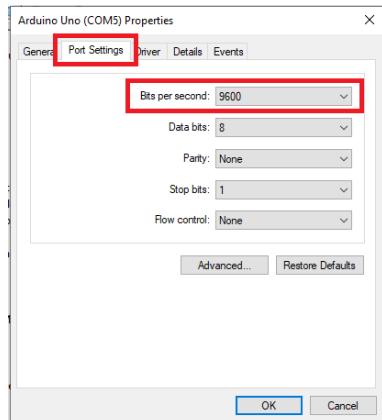
2. Kemudian pilih 'Ports (COM & LPT)'.



3. Klik dua kali pada COM yang terhubung.

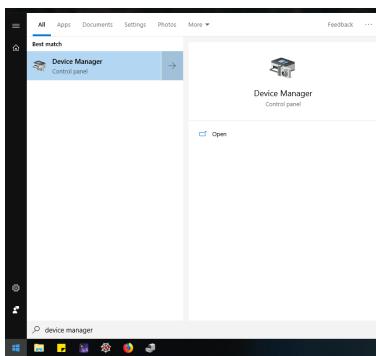


4. Pilih tab 'Port Settings', lalu lihat di 'Bit per second'.

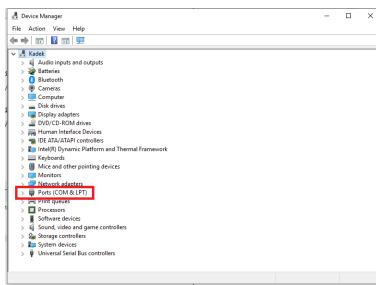


Membaca Port dari Komputer

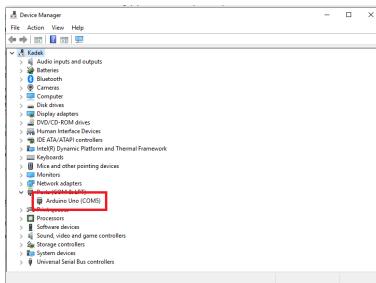
1. Pertama buka 'Start'. Cari 'Device Manager', lalu klik.



2. Kemudian pilih ‘Ports (COM & LPT)’.



3. Port dari Arduino telah terbaca oleh PC.



9.1.4 Soal No. 4

Jelaskan sejarah library pyserial!

PySerial adalah paket Python yang memfasilitasi 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 dapat digunakan oleh komputer dan perangkat keras lain seperti

mouse, keyboard, dan webcam. USB adalah singkatan dari Universal Serial Bus. USB dibangun diatas dan memperluas antarmuka komunikasi serial asli.

9.1.5 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.

9.1.6 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 akan muncul banyak. Sedangkan apabila tidak membutuhkan perulangan maka Arduino hanya akan membaca data sekali saja.

9.1.7 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 9.1 Fungsi yang menggunakan pyserial.

9.1.8 Praktek

9.1.8.1 Soal No. 1 Buatlah fungsi (file terpisah/library dengan nama NPMreal-time.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 9.2 Fungsi untuk mendapatkan data dari Arduino.

9.1.8.2 Soal 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 9.3 Fungsi untuk mendapatkan data langsung dari Arduino dengan looping.

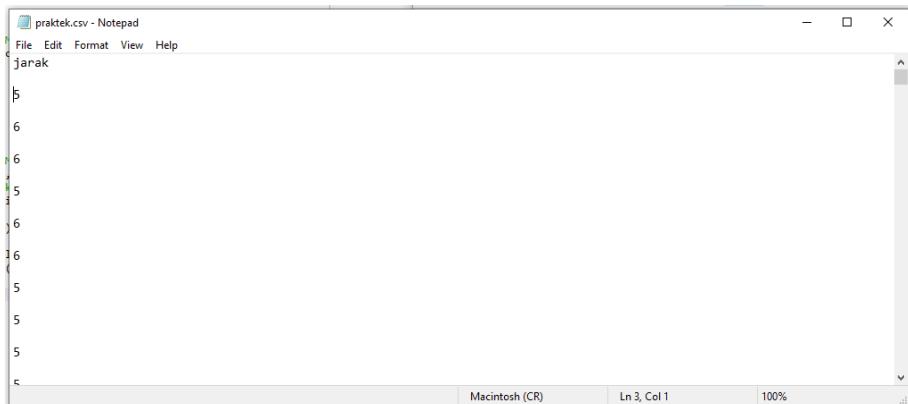
9.1.8.3 Soal No. 3 Buatlah fungsi (file terpisah/library dengan nama NPMreal-time.py) untuk mendapatkan data dari arduino dan langsung ditulis kedalam file csv!

```

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

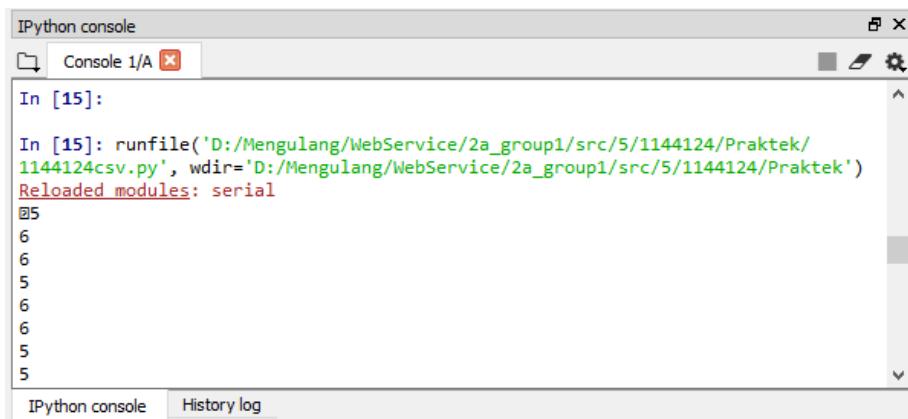


Gambar 9.1 Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino dan langsung ditulis kedalam file CSV.

9.1.8.4 Soal 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 9.5 Fungsi untuk membaca file CSV hasil Arduino dan mengembalikan fungsi.



Gambar 9.2 Hasil dari pembacaan fungsi untuk membaca file csv hasil arduino dan mengembalikan fungsi.

9.1.8.5 Penanganan Error

- Tuliskan peringatan error yang didapatkan dari mengerjakan praktek ketiga ini, dan jelaskan cara penanganan error tersebut. Dan buatlah satu fungsi yang menggunakan try except untuk menanggulangi error tersebut. 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:
10        print("Terjadi sebuah kesalahan")
11
12 tryExceptError()
```

Listing 9.6 Fungsi untuk menanggulangi error menggunakan Try Except.

9.2 Harun Ar - Rasyid

9.2.1 Teori

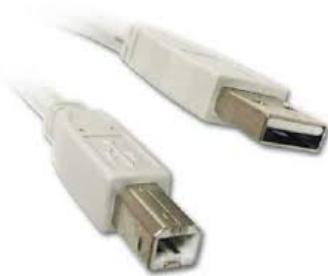
9.2.1.1 Apa itu fungsi device manager di windows dan folder /dev di linux

Fungsi device manager dan folder /dev itu berfungsi untuk mengetahui device apa saja yang telah terinstal di laptop anda serta mengetahui port yang digunakan oleh device tersebut.

9.2.1.2 Jelaskan langkah-langkah instalasi driver dari arduino

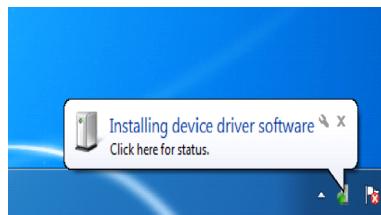
1. Cara Auto

- Pertama Hubungkan sistem minimum Arduino Uno ke komputer dengan kabel USB type B(kabel Printer)



Gambar 9.3 Membuat file csv

- Lalu pada bagian kanan didesktop PC anda, akan muncul popup “Installing device driver software” seperti pada gambar dibawah ini.



Gambar 9.4 Membuat file csv

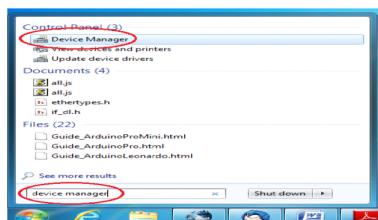
- Tunggu hingga selesai.
- Jika sudah selesai anda bisa mengecheck di device manager.



Gambar 9.5 Membuat file csv

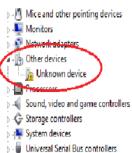
2. Cara Manual

- Penginstalan secara manual akan dilakukan jika penginstalan secara auto gagal dilakukan.
- Buka Device Manager, caranya pada bagian Search Program and Files lalu ketikkan “device manager”, perhatikan gambar dibawah ini. Pada bagian Control Panel akan muncul Device Manager, klik untuk menjalankan.



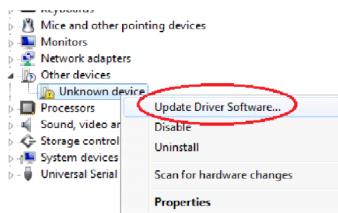
Gambar 9.6 Membuat file csv

- Cari Unknown device pada bagian Other device, biasanya terdapat tanda seru berwarna kuning, itu disebabkan karena penginstallan tidak berjalan dengan sempurna.



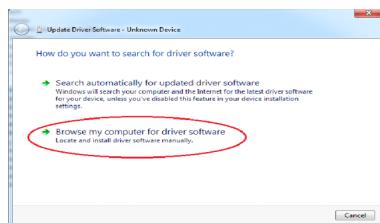
Gambar 9.7 Membuat file csv

- Klik kanan pada “Unknown device” kemudian pilih Update Driver Software.



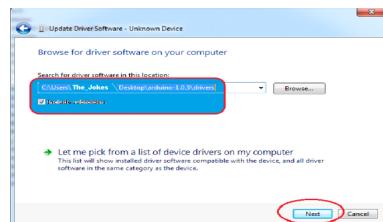
Gambar 9.8 Membuat file csv

- Pilih Browse my computer for driver software.



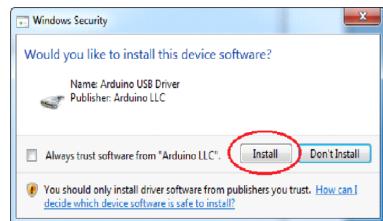
Gambar 9.9 Membuat file csv

- Arahkan lokasi folder ke folder ..arduino-1.0.5 drivers. Pastikan checkbox lalu centang include subfolders. Klik Next untuk melanjutkan instalasi driver.



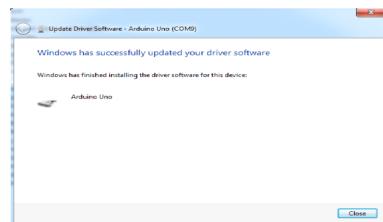
Gambar 9.10 Membuat file csv

- Kemudian lanjutkan dengan mengklik Install pada tampilan Windows Security.



Gambar 9.11 Membuat file csv

- Jika instalasi driver berhasil maka akan muncul Windows has successfully updated your driver software.



Gambar 9.12 Membuat file csv

- Perhatikan dan ingat nama COM Arduino Uno, karena nama COM ini yang akan digunakan untuk meng-upload program nantinya.



Gambar 9.13 Membuat file csv

9.2.1.3 Jelaskan bagaimana cara membaca baudrate dan port dari komputer yang sudah terinstall driver Untuk baudrate itu bisa dicek melalui arduino IDE, kemudian untuk mengecheck port bisa dilakukan dengan device manager

9.2.1.4 Jelaskan sejarah library pyserial Modul ini merangkum akses untuk port serial. Ini menyediakan backends untuk Python yang berjalan di Windows, Linux, BSD (mungkin sistem yang mendukung POSIX), Jython dan IronPython (.NET dan Mono). Modul bernama "serial" secara otomatis memilih backend yang sesuai. Antarmuka berbasis kelas yang sama pada semua platform yang didukung. Akses ke pengaturan port melalui properti Python. Dukungan untuk berbagai ukuran byte, bit stop, paritas dan kontrol aliran dengan RTS / CTS dan / atau Xon / Xoff. Bekerja dengan atau tanpa menerima batas waktu. File seperti API dengan "read" dan "write" ("readline" dll. Juga didukung). File-file dalam paket ini adalah 100 persen Python murni. Port diatur untuk transmisi biner. Tidak ada stripping byte NULL, terjemahan CR-LF dll. (Yang berkali-kali diaktifkan untuk POSIX.) Ini membuat modul ini bermanfaat secara universal. Kompatibel dengan pustaka io (Python 2.6+)

9.2.1.5 Jelaskan fungsi-fungsi apa saja yang dipakai dari library pyserial Serial – fungsi ini untuk membuka port serial Write(data) – untuk menulis data lewat port serial Readline() – untuk membaca string dari port serial Read(size) – untuk membaca jumlah byte dari port serial Close() – ini untuk menutup port serial

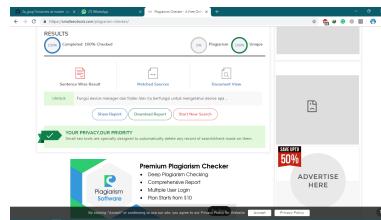
9.2.1.6 Jelaskan kenapa butuh perulangan dan tidak butuh perulangan dalam membaca serial Perulangan dalam bahasa pemrograman berfungsi menyuruh komputer melakukan sesuatu secara berulang-ulang. Terdapat dua jenis perulangan dalam bahasa pemrograman python, yaitu perulangan dengan for dan while. Perulangan for disebut counted loop (perulangan yang terhitung), sementara perulangan while disebut uncounted loop (perulangan yang tak terhitung). Perbedaannya adalah perulangan for biasanya digunakan untuk mengulangi kode yang sudah diketahui banyak perulangannya. Sementara while untuk perulangan yang memiliki syarat dan tidak tentu berapa banyak perulangannya. Perulangan diperlukan agar dapat membaca data secara berulang kali sehingga data yang muncul lebih dari satu. Sedangkan apabila tidak memakai perulangan maka data akan terbaca satu kali saja.

9.2.1.7 Jelaskan bagaimana cara membuat fungsi yang menggunakan pyserial Berikut merupakan contoh penggunaan fungsi yang menggunakan pyserial

```

1 import serial
2
3 def baca():
4     ser = serial.Serial("COM6", 115200)
5     baca = ser.readline()
6     print(baca)
7
8 baca()

```



Gambar 9.14 Membuat file csv

9.2.1.8 Scan Plagiarisme

9.2.2 Praktek

```

1 import serial
2
3 def ambildata():
4     ser = serial.Serial('COM6', 9600)
5     print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
6
7 ambildata()

```

```

1 import serial
2
3 def ulang():
4     ser = serial.Serial('COM6', 9600)
5     while(1):
6         print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
7
8 ulang()

```

```

1 import csv
2
3 def tuliscsv():
4     ser = serial.Serial('COM6', 9600)
5     with open('uji.csv', mode='w') as csv_file:

```

```

6     fieldnames = [ 'jarak' ]
7     writer = csv.DictWriter(csv_file , fieldname=fieldnames)
8
9     writer.writeheader()
10    while(1):
11        data = ser.readline().decode("utf-8").strip('\n').strip('
12        writer.writerow({ 'jarak' : data})
13
14 tuliscsv()

```

```

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

```

9.2.2.5 Penanganan Error Untuk kali ini saya menemukan Type Error, yaitu error yang menampilkan jika type data na berbeda berusaha disatukan.

```

1 import serial
2
3 def tryExceptError():
4     try:
5         ser = serial.Serial('COM6',9600)
6         print(ser.readline().decode("utf-8").strip('\n').strip('
7     except TypeError:
8         print("Terjadi ketidaksamaan type")
9
10 tryExceptError()

```

9.3 Arjun Yuda Firwanda

9.3.1 Teori

9.3.1.1 Apa itu fungsi device manager di windows dan folder /dev di linux
Fungsi device manager untuk membantu dan mengelola semua hardware yang terpasang dalam suatu windows.

9.3.1.2 Jelaskan langkah-langkah instalasi driver dari arduino

- ubungkan sistem minimun Arduino Uno ke komputer dengan kabel USB type B (kabel Printer).

- Lalu pada bagian kanan didesktop PC anda, akan muncul popup “Installing device driver software”.
- Sistem operasi Windows tidak menyediakan driver untuk Arduino Uno, lalu proses instalasinya harus dilakukan secara manual.
- Buka Device Manager, caranya pada bagian Search Program and Files lalu ketikkan “device manager”.
- Cari Unknown device pada bagian Other device, terdapat tanda seru yang berwarna kuning, itu disebabkan karena penginstallan tidak berjalan dengan sempurna.
- Klik kanan pada “Unknown device”, pilih Update Driver Software.
- Pilih Browse my computer for driver software.
- Arahkan lokasi folder ke folder arduino-1.0, drivers. Pastikan check-box lalu centang include subfolders. Klik Next untuk melanjutkan instalasi driver.
- Kemudian lanjutkan dengan mengklik 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 mengupload program nantinya.

9.3.1.3 Jelaskan bagaimana cara membaca baudrate dan port dari komputer yang sudah terinstall driver Untuk baudrate dapat bisa dicek melalui arduino IDE, kemudian untuk mengecek port bisa dilakukan dengan device manager.

9.3.1.4 Jelaskan sejarah library pyserial Modul ini merangkum akses untuk port serial. Ini menyediakan backends untuk Python yang berjalan di Windows, Linux, BSD (mungkin sistem yang mendukung POSIX), Jython dan IronPython (.NET dan Mono). Modul bernama ”serial” secara otomatis memilih backend yang sesuai. Antarmuka berbasis kelas yang sama pada semua platform yang didukung. Akses ke pengaturan port melalui properti Python. Dukungan untuk berbagai ukuran byte, bit stop, paritas dan kontrol aliran dengan RTS / CTS dan / atau Xon / Xoff. Bekerja dengan atau tanpa menerima batas waktu. File seperti API dengan ”read” dan ”write” (“readline” dll. Juga didukung). File-file dalam paket ini adalah 100 persen Python murni. Port diatur untuk transmisi biner. Tidak ada stripping byte NULL, terjemahan CR-LF dll. (Yang berkali-kali diaktifkan untuk POSIX.) Ini membuat modul ini bermanfaat secara universal. Kompatibel dengan pustaka io (Python 2.6+)

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

- Serial, fungsi ini untuk membuka port serial
- Write(data), untuk menulis data lewat port serial
- Readline, untuk membaca string dari port serial
- Read(size), untuk membaca jumlah byte dari port serial
- Close, ini untuk menutup port serial

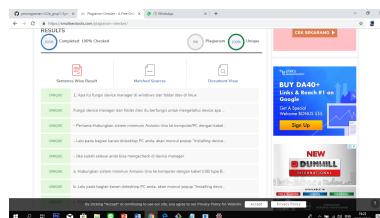
9.3.1.6 Jelaskan kenapa butuh perulangan dan tidak butuh perulangan dalam membaca serial

Perulangan dalam bahasa pemrograman berfungsi untuk menyuruh komputer melakukan sesuatu secara berulang-ulang. Terdapat dua jenis perulangan dalam bahasa pemrograman python diantaranya adalah perulangan dengan for dan while. Perulangan for atau counted loop (perulangan yang terhitung). Perulangan while atau uncounted loop (perulangan yang tak terhitung). Perbedaannya pada perulangan for biasanya digunakan untuk mengulangi kode yang sudah diketahui banyak perulangannya. Perulangan while untuk perulangan yang memiliki syarat dan tidak tentu berapa banyak perulangannya. Perulangan digunakan untuk membaca data secara berulang-ulang dan apabila tidak memakai perulangan, maka data akan terbaca satu per satu.

```

1 import serial
2
3 def baca():
4     ser = serial.Serial("COM6", 115200)
5     baca = ser.readline()
6     print(baca)
7
8 baca()

```



Gambar 9.15 Cek Plagiarisme

9.3.1.8 Cek Plagiarisme

9.3.2 Praktek

```

1 import serial
2
3 def ambildata():
4     ser = serial.Serial('COM6',9600)
5     print(ser.readline().decode("utf-8").strip('\n').strip('r'))
6
7 ambildata()

```

```

1 import serial
2
3 def ulang():
4     ser = serial.Serial('COM6',9600)
5     while(1):
6         print(ser.readline().decode("utf-8").strip('\n').strip('r'))
7
8 ulang()

```

```

1 import csv
2
3 def tuliscsv():
4     ser = serial.Serial('COM6',9600)
5     with open('uji.csv',mode='w') as csv_file:
6         fieldnames = ['jarak']
7         writer = csv.DictWriter(csv_file, fieldname=fieldnames)
8
9         writer.writeheader()
10        while(1):
11            data = ser.readline().decode("utf-8").strip('\n').strip('r')
12            writer.writerow({'jarak': data})
13
14 tuliscsv()

```

```

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

```

9.3.2.5 Penanganan Error Untuk kali ini saya menemukan Type Error, yaitu error yang menampilkan jika type data na berbeda berusaha disatukan.

```
1 import serial
2
3 def tryExceptError():
4     try:
5         ser = serial.Serial('COM6', 9600)
6         print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
7     except TypeError:
8         print("Terjadi ketidaksamaan type")
9
10 tryExceptError()
```

9.4 Doli Jonviter NT Simbolon /1154016

Pemahaman Teori

9.4.1 Soal 1

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

Device Manager dapat membantu dalam mengelola semua hardware yang terpasang dalam suatu sistem Windows. Berikut fungsi kegunaan Device Manager antara lain adalah :

1. Menunjukkan status suatu hardware.
2. Menunjukkan informasi detil suatu hardware.
3. Mengelola driver hardware
4. Disable dan Enable hardware
5. Mengidentifikasi konflik antar perangkat keras.

Folder /bin merupakan isi program biner yang harus ada apabila sistem yang dipasang dalam mode single-user, dan juga ada beberapa program penting seperti bash.

9.4.2 Soal 2

Jelaskan langkah-langkah instalasi driver dari arduino!

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

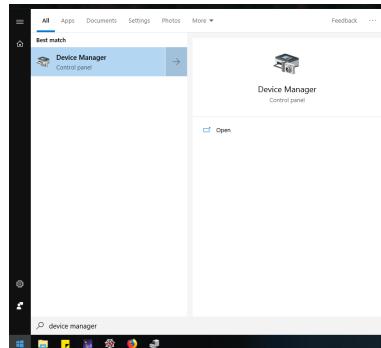
1. Langkah pertama Hubungkan sistem minimun Arduino Uno ke komputer dengan kabel USB .

2. Lalu pada bagian kanan didesktop PC , akan muncul popup “Installing device driver software” seperti pada gambar dibawah ini.
3. Kemudian jika sistem operasi Windows tidak menyediakan driver untuk Arduino Uno,maka harus melakukan instalasinya harus dilakukan secara manual.
4. Lalu Buka Device Manager, dengan cara pada bagian Search Program and Files lalu ketikkan “device manager” (tanpa tanda petik).
5. kemudian Pada bagian COntrol Panel akan muncul Device Manager, lalu klik untuk menjalankan program tersebut.
6. Setelah itu cari Unknown device pada bagian Other device, biasanya terdapat tanda seru berwarna kuning, itu disebabkan karena penginstallan gagal.
7. Klik kanan pada bagian “Unknown device” kemudian pilih Update Driver Software.
8. kemudian cari Browse my computer for driver software pada laptop anda.
9. setelah itu lakukan dengan mengklik Install pada tampilan Windows Security.
10. Jika instalasi driver pada laptop anda berhasil maka akan muncul Windows has successfully updated your driver software.
11. Perhatikan dan ingat nama COM Arduino Uno, karena nama COM ini yang akan digunakan untuk meng-upload program nantinya

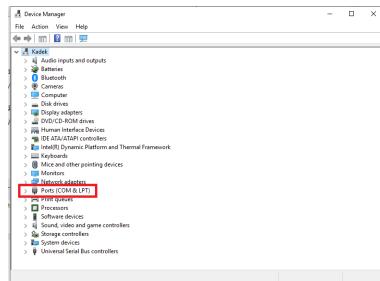
9.4.3 Soal 3

Jelaskan bagaimana cara membaca baudrate dan port dari komputer yang sudah terinstall driver! **Membaca Baudrate dari Komputer**

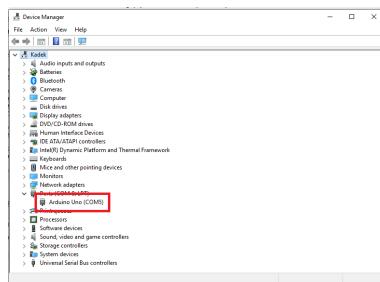
1. Pertama buka ‘Start’. Cari ‘Device Manager’, lalu klik.



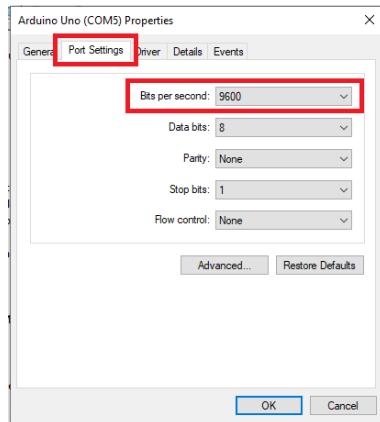
2. Kemudian pilih ‘Ports (COM & LPT)’.



3. Klik dua kali pada COM yang terhubung.

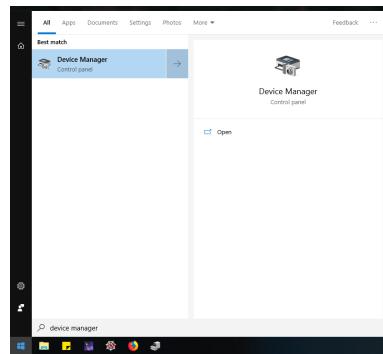


4. Pilih tab 'Port Settings', lalu lihat di 'Bit per second'.

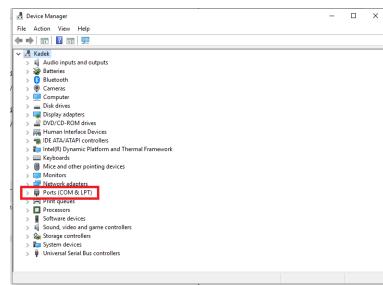


Membaca Port dari Komputer

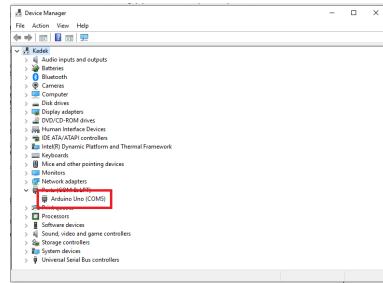
1. Pertama buka 'Start'. Cari 'Device Manager', lalu klik.



2. Kemudian pilih ‘Ports (COM & LPT)’.



3. Port dari Arduino telah terbaca oleh PC.



9.4.4 Soal 4

Jelaskan sejarah library pyserial!.

PySerial adalah paket Python yang memfasilitasi 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 dapat digunakan oleh komputer dan perangkat keras lain seperti mouse, keyboard, dan webcam. USB adalah singkatan dari Universal Serial Bus. USB dibangun diatas dan memperluas antarmuka komunikasi serial asli.

9.4.5 Soal 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.

9.4.6 Soal No. 6

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

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

9.4.7 Soal No. 7

Jelaskan bagaimana cara membuat fungsi yang menggunakan pyserial!

Fungsi yang berada pada Python, dibuat dengan nama kata kunci def kemudian diikuti dengan nama fungsinya pada python. Seperti halnya dengan blok kode yang lain, kita juga harus memberikan identasi untuk menuliskan isi fungsi.

9.4.8 Praktek

```
1 import serial
2
3 def ambildata():
4     ser = serial.Serial('COM6', 9600)
5     print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
6
7 ambildata()
```

```

1 import serial
2
3 def ulang():
4     ser = serial.Serial('COM6',9600)
5     while(1):
6         print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
7
8 ulang()

```

```

1 import csv
2
3 def tuliscsv():
4     ser = serial.Serial('COM6',9600)
5     with open('uji.csv',mode='w') as csv_file:
6         fieldnames = ['jarak']
7         writer = csv.DictWriter(csv_file, fieldnames=fieldnames)
8
9         writer.writeheader()
10        while(1):

```

```

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

```

9.4.8.5 Penanganan Error Untuk kali ini saya menemukan Type Error, yaitu error yang menampilkan jika type data na berbeda berusaha disatukan.

```

1 import serial
2
3 def tryExceptError():
4     try:
5         ser = serial.Serial('COM6',9600)
6         print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
7     except TypeError:
8         print("Terjadi ketidaksamaan type")
9
10 tryExceptError()

```

9.5 Dwi Yulianingsih

9.5.1 Teori

9.5.1.1 Soal 1 Device Manager dalam system operasi Windows, merupakan perluasan dari Microsoft Management Console. Device Manager akan menampilkan seluruh hardware yang bisa di-inisialisasi (dikenali) oleh system operasi Windows. Tampilannya ter-organisir sedemikian hingga yang akan memudahkan pengelolaan setiap hardware yang ada. Fungsi-fungsi kegunaan Device Manager antara lain sebagai berikut :

- Menunjukkan status suatu hardware
- Menunjukkan informasi detail suatu hardware
- Mengelola driver hardware
- Disable & Enable hardware
- Meng-identifikasi konflik antar hardware, dll.

Directory pada /dev berisi file device, baik device blok maupun device karakter. di dalamnya sekurang-kurangnya harus memiliki file biner MAKEDEV untuk membuat device ini secara manual.

9.5.1.2 Soal 2

- Hubungkan sistem minimun Arduino Uno ke komputer dengan kabel USB type B (kabel Printer).
- Lalu pada bagian kanan didesktop PC anda, akan muncul popup “Installing device driver software” seperti pada gambar dibawah ini.
- Sistem operasi Windows tidak menyediakan driver untuk Arduino Uno seperti yang terlihat pada gambar dibawah ini, lalu proses instalasinya harus dilakukan secara manual.
- Buka Device Manager, caranya pada bagian Search Program and Files lalu ketikkan “device manager” (tanpa tanda petik), perhatikan gambar dibawah ini. Pada bagian Control Panel akan muncul Device Manager, klik untuk menjalankan.
- Cari Unknown device pada bagian Other device, biasanya terdapat tanda seru berwarna kuning, itu disebabkan karena penginstallan tidak berjalan dengan sempurna.
- Klik kanan pada “Unknown device” kemudian pilih Update Driver Software.
- Pilih Browse my computer for driver software.

- Arahkan lokasi folder ke folder ..arduino1.0.5 drivers. Pastikan checkbox lalu centang include subfolders. Klik Next untuk melanjutkan instalasi driver.
- Kemudian lanjutkan dengan mengklik 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.

9.5.1.3 Soal 3 Hubungkan Port USB pada Arduino dengan Port USB komputer. Buka Software Arduino pada Komputer. Tuliaskan program berikut ini pada Arduino Sketch.

9.5.1.4 Soal 4 Modul ini merangkum akses untuk port serial. Ini menyediakan backends untuk Python yang berjalan di Windows, Linux, BSD (mungkin sistem yang mendukung POSIX), Jython dan IronPython (.NET dan Mono). Modul bernama "serial" secara otomatis memilih backend yang sesuai. Antarmuka berbasis kelas yang sama pada semua platform yang didukung. Akses ke pengaturan port melalui properti Python. Dukungan untuk berbagai ukuran byte, bit stop, paritas dan kontrol aliran dengan RTS / CTS dan / atau Xon / Xoff. Bekerja dengan atau tanpa menerima batas waktu. File seperti API dengan "read" dan "write" ("readline" dll. Juga didukung). File-file dalam paket ini adalah 100 persen Phyton murni. Port diatur untuk transmisi biner. Tidak ada stripping byte NULL, terjemahan CR-LF dll. (Yang berkali-kali diaktifkan untuk POSIX.) Ini membuat modul ini bermanfaat secara universal. Kompatibel dengan pustaka io (Python 2.6+).

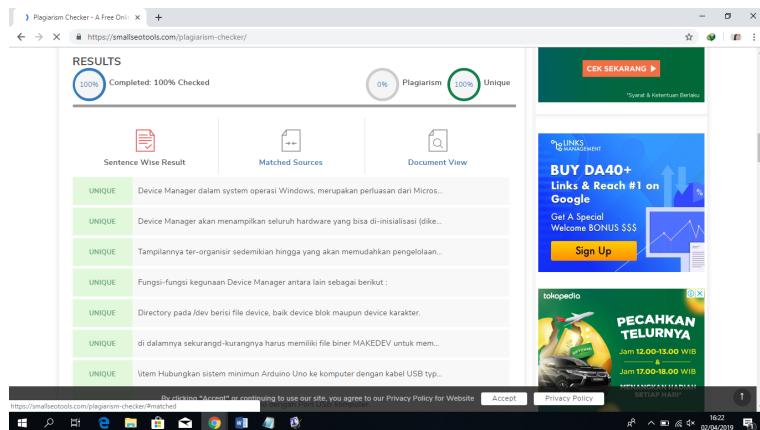
9.5.1.5 Soal 5 Fungsi-fungsi yang dipakai dari library Pyserial, diantara :

- Serial – fungsi ini untuk membuka port serial
- Write(data) – untuk menulis data lewat port serial
- Readline() – untuk membaca string dari port serial
- Read(size) – untuk membaca jumlah byte dari port serial
- Close() – ini untuk menutup port serial

9.5.1.6 Soal 6 Perulangan dalam bahasa pemrograman berfungsi menyuruh komputer melakukan sesuatu secara berulang-ulang. Terdapat dua jenis perulangan dalam bahasa pemrograman python, yaitu perulangan dengan for dan while. Perulangan for disebut counted loop (perulangan yang terhitung), sementara perulangan while disebut uncounted loop (perulangan yang tak terhitung). Perbedaannya adalah perulangan for biasanya digunakan untuk mengulangi kode yang sudah diketahui banyak perulangannya. Sementara while untuk perulangan yang memiliki syarat dan tidak tentu berapa banyak perulangannya. Perulangan diperlukan agar dapat membaca data secara berulang kali sehingga data yang muncul lebih dari satu. Sedangkan apabila tidak memakai perulangan maka data akan terbaca satu kali saja.

9.5.1.7 Soal 7 Berikut merupakan contoh penggunaan fungsi yang menggunakan perulangan

```
1 import serial
2
3 def baca():
4     ser = serial.Serial("COM6", 115200)
5     baca = ser.readline()
6     print(baca)
7
8 baca()
```



Gambar 9.16 SS Bebas Plagiarisme

9.5.1.8 Bukti bebas plagiarisme

9.5.2 Praktek

1. Soal 1

```
1 import serial
2
3 def ambildata():
4     ser = serial.Serial('COM6', 9600)
5     print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
6
7 ambildata()
```

2. Soal 2

```
1 import serial
2
3 def ulang():
4     ser = serial.Serial('COM6', 9600)
```

```

5     while(1):
6         print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
7
8 ulang()

```

3. Soal 3

```

1 import csv
2
3 def tuliscsv():
4     ser = serial.Serial('COM6',9600)
5     with open('uji.csv',mode='w') as csv_file:
6         fieldnames = ['jarak']
7         writer = csv.DictWriter(csv_file, fieldnames=fieldnames)
8
9         writer.writeheader()
10        while(1):
11            data = ser.readline().decode("utf-8").strip('\n').
12            strip('\r')
13            writer.writerow({'jarak': data})
14 tuliscsv()

```

4. Soal 4

```

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

```

9.5.2.1 Penanganan Error cara untuk menangani eror yang dapat dilakukan adalah sebagai berikut:

```

1 import serial
2
3 def tryExceptError():
4     try:
5         ser = serial.Serial('COM6',9600)
6         print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
7     except TypeError:
8         print("Terjadi ketidaksamaan type")
9
10 tryExceptError()

```

9.6 Doli Jonviter NT Simbolon /1154016

Pemahaman Teori

9.6.1 Soal 1

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

Device Manager dapat membantu dalam mengelola semua hardware yang terpasang dalam suatu sistem Windows. Berikut fungsi kegunaan Device Manager antara lain adalah :

1. Menunjukkan status suatu hardware.
2. Menunjukkan informasi detil suatu hardware.
3. Mengelola driver hardware
4. Disable dan Enable hardware
5. Mengidentifikasi konflik antar perangkat keras.

Folder /bin merupakan isi program binner yang harus ada apabila sistem yang dipasang dalam mode single-user, dan juga ada beberapa program penting seperti bash.

9.6.2 Soal 2

Jelaskan langkah-langkah instalasi driver dari arduino!

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

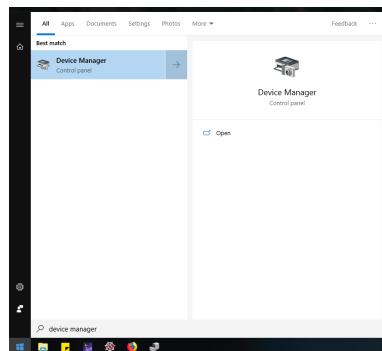
1. Langkah pertama Hubungkan sistem minimun Arduino Uno ke komputer dengan kabel USB .
2. Lalu pada bagian kanan didesktop PC , akan muncul popup “Installing device driver software” seperti pada gambar dibawah ini.
3. Kemudian jika sistem operasi Windows tidak menyediakan driver untuk Arduino Uno,maka harus melakukan instalasinya harus dilakukan secara manual.
4. Lalu Buka Device Manager, dengan cara pada bagian Search Program and Files lalu ketikkan “device manager” (tanpa tanda petik).
5. kemudian Pada bagian Control Panel akan muncul Device Manager, lalu klik untuk menjalankan program tersebut.
6. Setelah itu cari Unknown device pada bagian Other device, biasanya terdapat tanda seru berwarna kuning, itu disebabkan karena penginstallan gagal.
7. Klik kanan pada bagian “Unknown device” kemudian pilih Update Driver Software.
8. kemudian cari Browse my computer for driver software pada laptop anda.

9. setelah itu lakukan dengan mengklik Install pada tampilan Windows Security.
10. Jika instalasi driver pada laptop anda berhasil maka akan muncul Windows has successfully updated your driver software.
11. Perhatikan dan ingat nama COM Arduino Uno, karena nama COM ini yang akan digunakan untuk meng-upload program nantinya

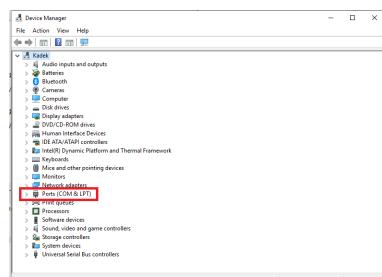
9.6.3 Soal 3

Jelaskan bagaimana cara membaca baudrate dan port dari komputer yang sudah terinstall driver! **Membaca Baudrate dari Komputer**

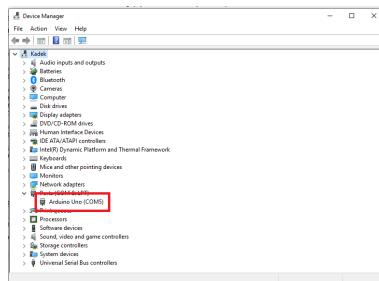
1. Pertama buka ‘Start’. Cari ‘Device Manager’, lalu klik.



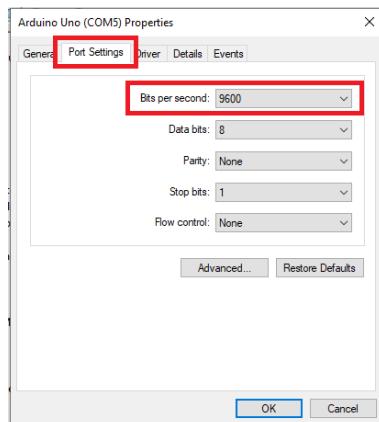
2. Kemudian pilih ‘Ports (COM & LPT)’.



3. Klik dua kali pada COM yang terhubung.

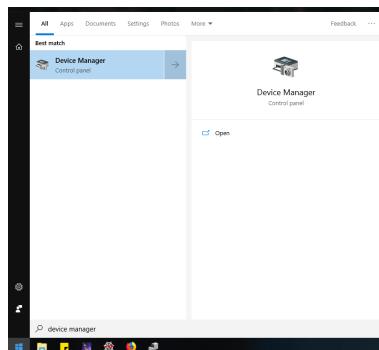


- Pilih tab ‘Port Settings’, lalu lihat di ‘Bit per second’.

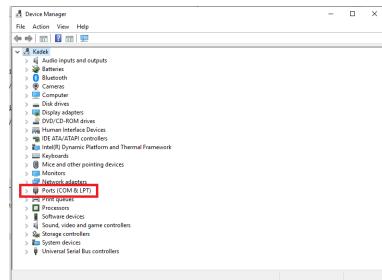


Membaca Port dari Komputer

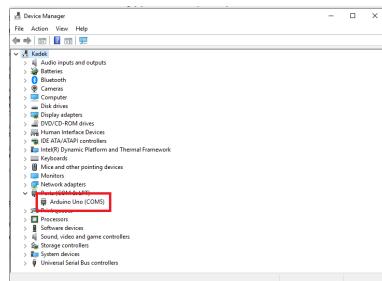
- Pertama buka ‘Start’. Cari ‘Device Manager’, lalu klik.



- Kemudian pilih ‘Ports (COM & LPT)’.



3. Port dari Arduino telah terbaca oleh PC.



9.6.4 Soal 4

Jelaskan sejarah library pyserial!.

PySerial adalah paket Python yang memfasilitasi 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 dapat digunakan oleh komputer dan perangkat keras lain seperti mouse, keyboard, dan webcam. USB adalah singkatan dari Universal Serial Bus. USB dibangun diatas dan memperluas antarmuka komunikasi serial asli.

9.6.5 Soal 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.

9.6.6 Soal No. 6

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

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

9.6.7 Soal No. 7

Jelaskan bagaimana cara membuat fungsi yang menggunakan pyserial!

Fungsi yang berada pada Python, dibuat dengan nama kata kunci def kemudian diikuti dengan nama fungsinya pada python. Seperti halnya dengan blok kode yang lain, kita juga harus memberikan identasi untuk menuliskan isi fungsi.

9.6.8 Praktek

```
1 import serial
2
3 def ambildata():
4     ser = serial.Serial('COM6',9600)
5     print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
6
7 ambildata()
```

```
1 import serial
2
3 def ulang():
4     ser = serial.Serial('COM6',9600)
5     while(1):
6         print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
7
8 ulang()
```

```
1 import csv
2
3 def tuliscsv():
4     ser = serial.Serial('COM6',9600)
5     with open('uji.csv',mode='w') as csv_file:
```

```

6     fieldnames = [ 'jarak' ]
7     writer = csv.DictWriter(csv_file , fieldname=fieldnames)
8
9     writer.writeheader()
10    while(1):

```

```

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

```

9.6.8.5 Penanganan Error Untuk kali ini saya menemukan Type Error, yaitu error yang menampilkan jika type data na berbeda berusaha disatukan.

```

1 import serial
2
3 def tryExceptError():
4     try:
5         ser = serial.Serial('COM6',9600)
6         print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
7     except TypeError:
8         print("Terjadi ketidaksamaan type")
9
10 tryExceptError()

```

9.7 Tomy Prawoto

9.7.1 Teori

9.7.1.1 Soal No. 1 Apa itu fungsi device manager di windows dan folder /dev di linux?

Device manager merupakan perangkat lunak yang berfungsi untuk menampilkan seluruh perangkat keras yang telah di-inisialisasi atau dikenali oleh sistem operasi Windows. Device Manager membantu dalam mengelola atau me-manage semua perangkat keras yang terpasang 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. Mengelola driver perangkat keras.
2. Menunjukkan informasi detail mengenai suatu perangkat keras.

3. Mengidentifikasi konflik antar perangkat keras.
4. Menonaktifkan dan mengaktifkan perangkat keras.
5. Memberitahukan terjadinya masalah pada perangkat keras.
6. Menunjukkan status mengenai suatu perangkat keras.

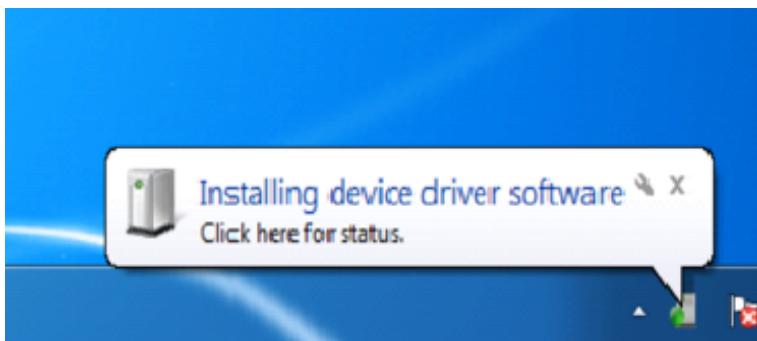
Folder /dev merupakan representasi dari drive yang terhubung sudah ke sistem operasi Linux dan oleh sistem yang dianggap sebagai file-file direktori. Biasanya sering ditampilkan direktori seperti /dev/sdal yang mewakili Drive SATA pertama dalam sistem.

9.7.2 Soal No. 2

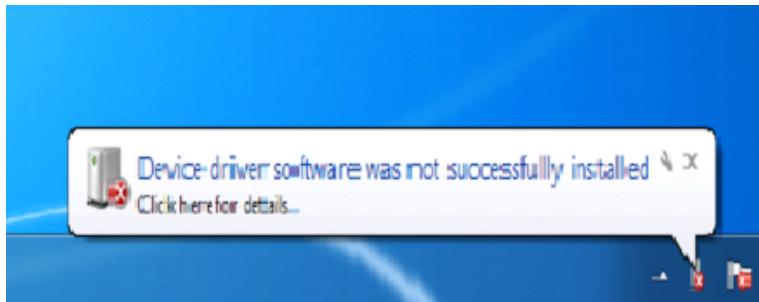
Jelaskan langkah-langkah instalasi driver dari arduino!

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

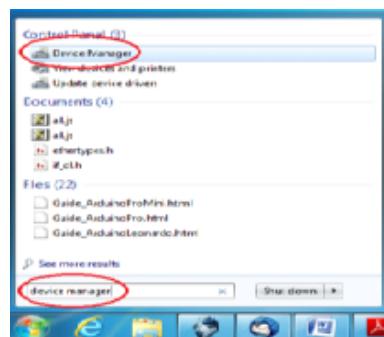
1. Pertama pastikan Arduino IDE telah terinstall.
2. Lalu hubungkan port USB Arduino Uno ke port USB PC.
3. Kemudian PC anda akan mendeteksi perangkat baru yang terpasang dan akan muncul pop seperti 3.



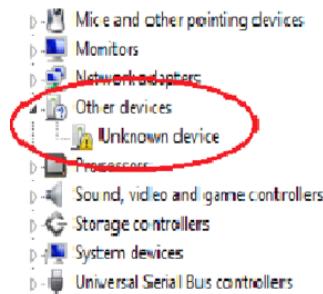
4. Karena Arduino Uno baru pertama kali terpasang, maka akan muncul pop up error seperti ini.



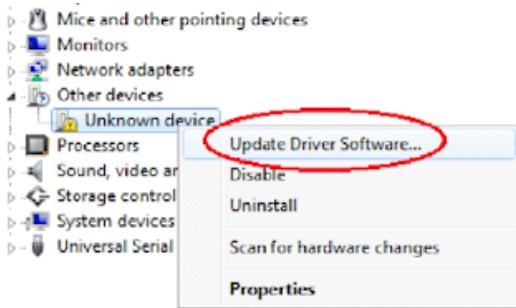
5. Buka ‘Start’ lalu cari Device Manager, kemudian klik ‘Device Manager’.



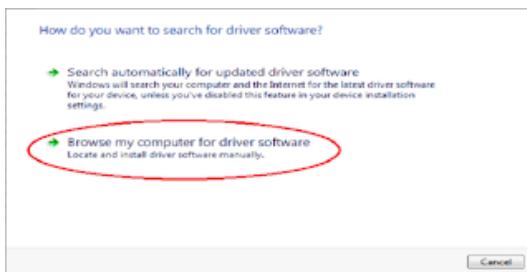
6. Setelah Device Manager terbuka, silahkan cari ‘Unknown Device’ yang berada di Other Device.



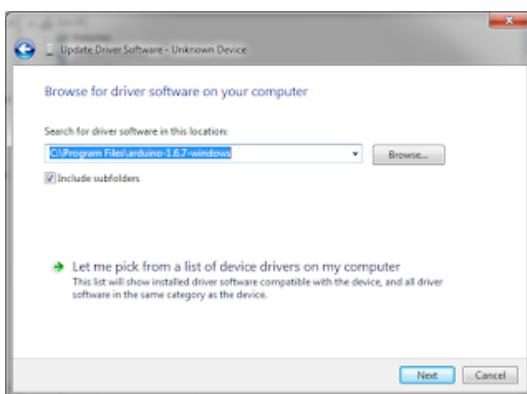
7. Kemudian klik kanan pada ‘Unknown Device’, lalu pilih ‘Update Driver Software’.



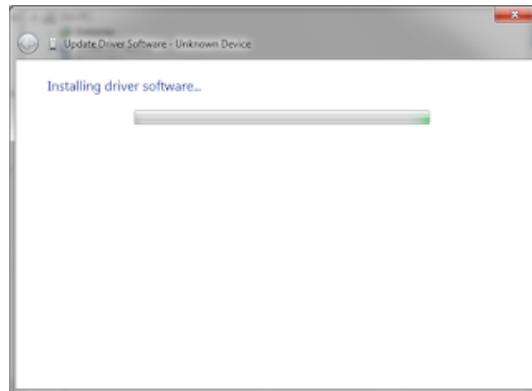
- Setelah itu muncul window baru, lalu pilih ‘Browse my computer for driver software’.



- Lalu cari folder yang terinstall Arduino IDE dengan mengklik browse. Kemudian klik ‘Next’.



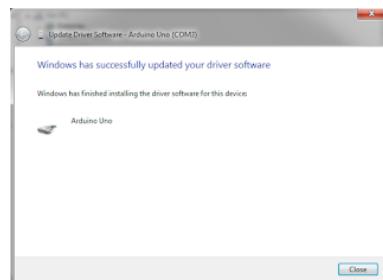
- Windows akan mencari dan menginstall driver yang berada pada folder tersebut.



11. Setelah itu akan muncul window, lalu klik ‘Install’.



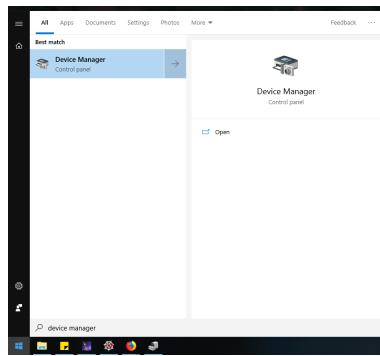
12. Jika berhasil terinstal maka akan muncul window seperti ini.



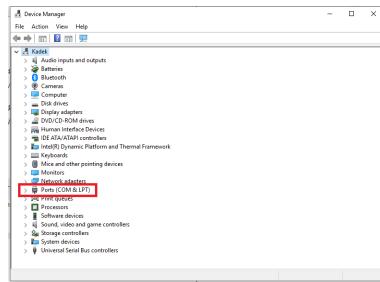
9.7.3 Soal No. 3

Jelaskan bagaimana cara membaca baudrate dan port dari komputer yang sudah terinstall driver! **Membaca Baudrate dari Komputer**

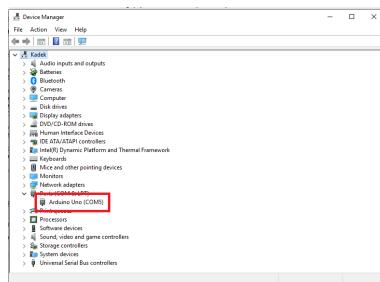
1. Pertama buka ‘Start’. Cari ‘Device Manager’, lalu klik.



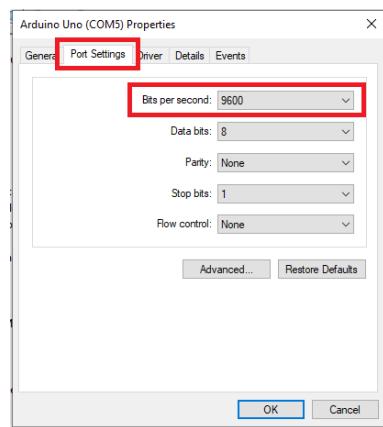
2. Kemudian pilih ‘Ports (COM & LPT)’.



3. Klik dua kali pada COM yang terhubung.

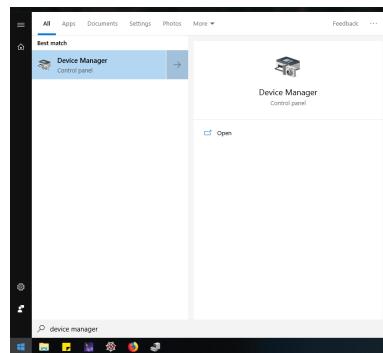


4. Pilih tab ‘Port Settings’, lalu lihat di ‘Bit per second’.

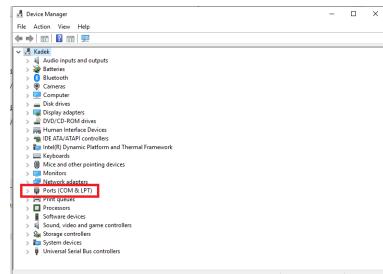


Membaca Port dari Komputer

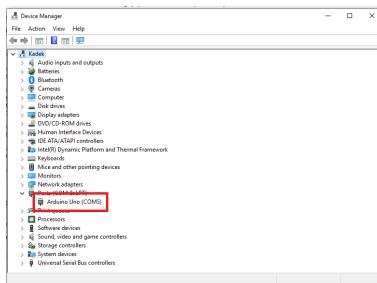
- Pertama buka 'Start'. Cari 'Device Manager', lalu klik.



- Kemudian pilih 'Ports (COM & LPT)'.



- Port dari Arduino telah terbaca oleh PC.



9.7.4 Soal No. 4

Jelaskan sejarah library pyserial!

PySerial adalah paket Python yang memfasilitasi 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 dapat digunakan oleh komputer dan perangkat keras lain seperti mouse, keyboard, dan webcam. USB adalah singkatan dari Universal Serial Bus. USB dibangun diatas dan memperluas antarmuka komunikasi serial asli.

9.7.5 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.

9.7.6 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 akan muncul banyak. Sedangkan apabila tidak membutuhkan perulangan maka Arduino hanya akan membaca data sekali saja.

9.7.7 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 9.7 Fungsi yang menggunakan pyserial.

9.7.8 Praktek

9.7.8.1 Soal No. 1 Buatlah fungsi (file terpisah/library dengan nama NPMreal-time.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 9.8 Fungsi untuk mendapatkan data dari Arduino.

9.7.8.2 Soal 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 9.9 Fungsi untuk mendapatkan data langsung dari Arduino dengan looping.

9.7.8.3 Soal No. 3 Buatlah fungsi (file terpisah/library dengan nama NPMreal-time.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)

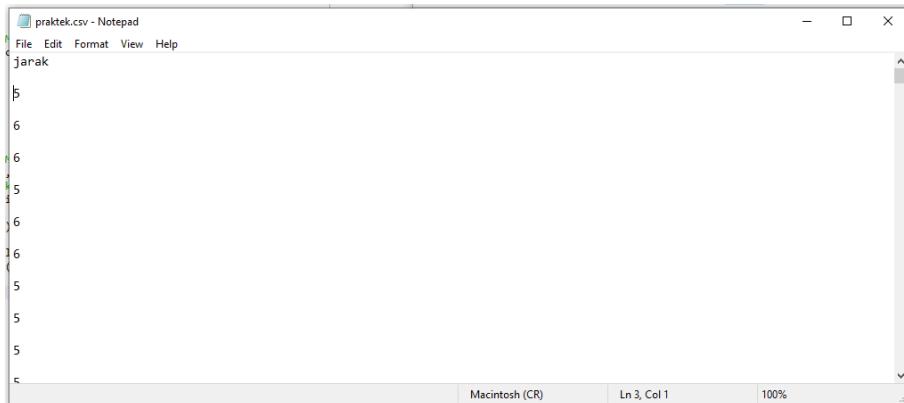
```

```

9         writer.writerow()
10        while (1):
11            data = ser.readline().decode("utf-8").strip('\n').strip(
12                '\r')
13            writer.writerow({'jarak': data})
14
15 writeCsv()

```

Listing 9.10 Fungsi untuk mendapatkan data dari Arduino dan langsung ditulis kedalam file CSV.



Gambar 9.17 Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino dan langsung ditulis kedalam file CSV.

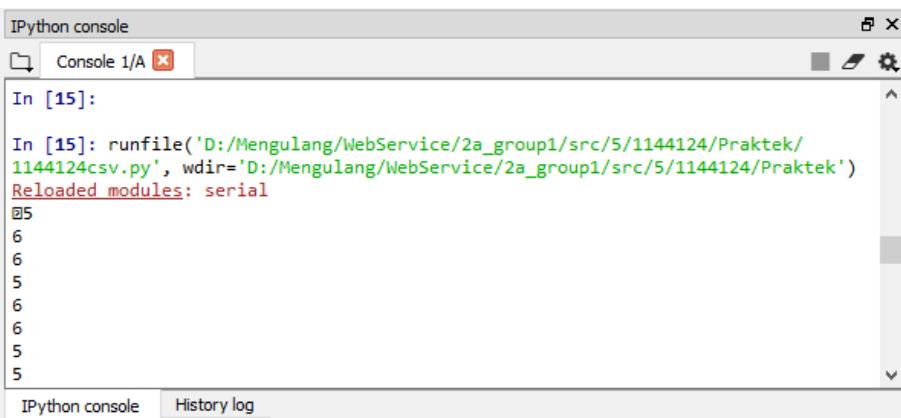
9.7.8.4 Soal No. 4 Buatlah fungsi (file terpisah/library dengan nama NPMcsv.py) untuk membaca file csv hasil arduino dan mengembalikan ke fungsi!

```

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 9.11 Fungsi untuk membaca file CSV hasil Arduino dan mengembalikan fungsi.



```
In [15]: runfile('D:/Mengulang/WebService/2a_group1/src/5/1144124/Praktek/1144124csv.py', wdir='D:/Mengulang/WebService/2a_group1/src/5/1144124/Praktek')
Reloaded modules: serial
5
6
6
5
6
5
5
5
```

The screenshot shows an IPython console window with the title "IPython console". It displays the command "In [15]" followed by the execution of a Python script named "1144124csv.py". The output shows several numerical values (5, 6, 5, 6, 5, 5) being printed sequentially. Below the console, there are two tabs: "IPython console" and "History log", with "IPython console" currently selected.

Gambar 9.18 Hasil dari pembacaan fungsi untuk membaca file csv hasil arduino dan mengembalikan fungsi.

9.7.8.5 Penanganan Error

- Tuliskan peringatan error yang didapatkan dari mengerjakan prakter ketiga ini, dan jelaskan cara penanganan error tersebut. Dan buatlah satu fungsi yang menggunakan try except untuk menanggulangi error tersebut. 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(ser.readline().decode("utf-8").strip('\n').strip('\r'))
7     except SyntaxError:
8         print("Kesalahan penulisan syntax")
9     except:
10        print("Terjadi sebuah kesalahan")
11
12 tryExceptError()
```

Listing 9.12 Fungsi untuk menanggulangi error menggunakan Try Except.

DAFTAR PUSTAKA

1. R. Awangga, “Sampeu: Servicing web map tile service over web map service to increase computation performance,” in *IOP Conference Series: Earth and Environmental Science*, vol. 145, no. 1. IOP Publishing, 2018, p. 012057.

Index

disruptif, xli
modern, xli