

# **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

---

<b>1 Library CSV dan Pandas</b>	<b>1</b>
<b>2 Praktek Library CSV dan Pandas</b>	<b>21</b>
<b>3 Library CSV dan Pandas</b>	<b>37</b>
<b>4 Praktek Library CSV dan Pandas</b>	<b>55</b>
<b>5 Library CSV dan Pandas</b>	<b>79</b>
<b>6 Praktek Library CSV dan Pandas</b>	<b>141</b>



# DAFTAR ISI

---

Daftar Gambar	xvii
Daftar Tabel	xxiii
Foreword	xxxii
Kata Pengantar	xxxiii
Acknowledgments	xxxv
Acronyms	xxxvii
Glossary	xxxix
List of Symbols	xli
Introduction	xliii
<i>Rolly Maulana Awangga, S.T., M.T.</i>	
<b>1 Library CSV dan Pandas</b>	<b>1</b>
1.1 Irvan Rizkiansyah/1174043	1
1.1.1 Soal 1	1
1.1.2 Soal 2	1
1.1.3 Soal 3	2

1.1.4	Soal 4	3
1.1.5	Soal 5	3
1.1.6	Soal 6	3
1.1.7	Soal 7	3
1.1.8	Cek Plagiarisme	3
1.2	Dwi Yulianingsih	3
1.2.1	Soal 1	3
1.3	Harun Ar-Rasyid	4
1.3.1	Soal 1	4
1.4	Sri Rahayu	4
1.4.1	Soal 1	4
1.5	Doli Jonviter	4
1.5.1	Soal 1	4
1.6	Rahmatul Ridha	4
1.6.1	Soal 1	4
1.7	Tomy Prawoto	5
1.7.1	Soal 1	5
1.8	Luthfi Muhammad Nabil/1174035	5
1.8.1	Soal 1	5
1.8.2	Soal 2	5
1.8.3	Soal 3	7
1.8.4	Soal 4	7
1.8.5	Soal 5	7
1.8.6	Soal 6	7
1.8.7	Soal 7	8
1.8.8	Cek Plagiarism	9
1.9	Hagan Rowlenstino/1174040	9
1.9.1	Soal 1	9
1.9.2	Soal 2	9
1.9.3	Soal 3	9
1.9.4	Soal 4	10
1.9.5	Soal 5	10
1.9.6	Soal 6	11
1.9.7	Soal 7	11
1.9.8	Cek Plagiarisme	11
1.10	Faisal Najib Abdullah 1174042	11
1.10.1	Pemahaman Teori	11
1.10.2	Cek Plagiarism	14

1.11	Dika Sukma Pradana 1174050	14
1.11.1	Pemahaman Teori	14
1.12	IchsanHizmanHardy/1174034	17
1.12.1	Pemahaman Teori	17
<b>2</b>	<b>Praktek Library CSV dan Pandas</b>	<b>21</b>
2.1	Luthfi Muhammad Nabil/1174035	21
2.1.1	Soal 1	21
2.1.2	Soal 2	21
2.1.3	Soal 3	22
2.1.4	Soal 4	22
2.1.5	Soal 5	22
2.1.6	Soal 6	22
2.1.7	Soal 7	22
2.1.8	Soal 8	23
2.1.9	Soal 9	23
2.1.10	Penanganan Error	23
2.2	Irvan Rizkiansyah/1174043	24
2.2.1	Soal 1	24
2.2.2	Soal 2	24
2.2.3	Soal 3	24
2.2.4	Soal 4	24
2.2.5	Soal 5	24
2.2.6	Soal 6	25
2.2.7	Soal 7	25
2.2.8	Soal 8	25
2.2.9	Soal 9	25
2.2.10	Penanganan Error	25
2.3	Hagan Rowlenstino/1174040	26
2.3.1	Soal 1	26
2.3.2	Soal 2	26
2.3.3	Soal 3	26
2.3.4	Soal 4	27
2.3.5	Soal 5	27
2.3.6	Soal 6	27
2.3.7	Soal 7	27
2.3.8	Soal 8	28
2.3.9	Soal 9	28

2.3.10	Keterampilan Penanganan Error	29
2.4	Faisal Najib Abdullah 1174042	29
2.4.1	Praktek	29
2.4.2	Ketrampilan Penanganan Error	31
2.5	Dika Sukma Pradana 1174050	31
2.5.1	Praktek	31
2.5.2	Ketrampilan Penanganan Error	33
2.6	Ichsan Hizman Hardy 1174034	34
2.6.1	Praktek	34
2.6.2	Ketrampilan Penanganan Error	35
<b>3</b>	<b>Library CSV dan Pandas</b>	<b>37</b>
3.1	Irvan Rizkiansyah/1174043	37
3.1.1	Soal 1	37
3.1.2	Soal 2	37
3.1.3	Soal 3	38
3.1.4	Soal 4	38
3.1.5	Soal 5	38
3.1.6	Soal 6	39
3.1.7	Soal 7	39
3.1.8	Plagiarisme	39
3.2	Hagan Rowlenstino/1174040	39
3.2.1	Soal 1	39
3.2.2	Soal 2	39
3.2.3	Soal 3	40
3.2.4	Soal 4	40
3.2.5	Soal 5	40
3.2.6	Soal 6	40
3.2.7	Soal 7	41
3.2.8	Cek Plagiarisme	41
3.3	Luthfi Muhammad Nabil/1174035	41
3.3.1	Soal 1	41
3.3.2	Soal 2	41
3.3.3	Soal 3	43
3.3.4	Soal 4	45
3.3.5	Soal 5	45
3.3.6	Soal 6	46
3.3.7	Soal 7	46

3.4	Faisal Najib Abdullah	46
3.4.1	Teori	46
3.5	Ichsan Hizman Hardy	48
3.5.1	Soal No. 1	48
3.5.2	Soal No. 2	48
3.5.3	Soal No. 3	49
3.5.4	Soal No. 4	50
3.5.5	Soal No. 5	50
3.5.6	Soal No. 6	50
3.5.7	Soal No. 7	50
3.6	Dika Sukma Pradana 1174050	51
3.6.1	Pemahaman Teori	51

<b>4</b>	<b>Praktek Library CSV dan Pandas</b>	<b>55</b>
4.1	Faisal Najib Abdullah 1174042	55
4.1.1	Keterampilan Pemrograman	55
4.1.2	Penanganan Error	56
4.2	Hagan Rowlenstino/1174040	57
4.2.1	Soal 1	57
4.2.2	Soal 2	58
4.2.3	Soal 3	58
4.2.4	Soal 4	59
4.2.5	Penanganan Error	59
4.3	Irvan Rizkiansyah/1174043	60
4.3.1	Nomor 1	60
4.3.2	Nomor 2	60
4.3.3	Nomor 3	60
4.3.4	Nomor 4	60
4.3.5	Penanganan Error	61
4.4	Luthfi Muhammad Nabil/1174035	61
4.4.1	Soal 1	61
4.4.2	Soal 2	61
4.4.3	Soal 3	62
4.4.4	Soal 4	62
4.4.5	Penanganan Error	63
4.5	Dika Sukma Pradana 1174050	63
4.5.1	Keterampilan Pemrograman	63
4.5.2	Penanganan Error	64

4.6	Ichsan Hizman Hardy 1174034	65
4.6.1	Keterampilan Pemrograman	65
4.6.2	Penanganan Error	67
<b>5</b>	<b>Library CSV dan Pandas</b>	<b>79</b>
5.1	Hagan Rowlenstino/1174040	79
5.1.1	Soal 1	79
5.1.2	Soal 2	79
5.1.3	Soal 3	80
5.1.4	Soal 4	80
5.1.5	Soal 5	81
5.1.6	Soal 6	82
5.1.7	Soal 7	82
5.1.8	Soal 8	82
5.1.9	Cek Plagiarisme	83
5.2	Irvan Rizkiansyah/1174043	83
5.2.1	Nomor 1	83
5.2.2	Nomor 2	84
5.2.3	Nomor 3	84
5.2.4	Nomor 4	84
5.2.5	Nomor 5	85
5.2.6	Nomor 6	86
5.2.7	Nomor 7	86
5.2.8	Nomor 8	87
5.2.9	Plagiarisme	87
5.3	Luthfi Muhammad Nabil/1174035	87
5.3.1	Soal 1	87
5.3.2	Soal 2	88
5.3.3	Soal 3	88
5.3.4	Soal 4	91
5.3.5	Soal 5	92
5.3.6	Soal 6	94
5.3.7	Soal 7	94
5.3.8	Soal 8	95
5.3.9	Plagiarisme	96
5.4	Faisal Najib Abdullah 1174042	96
5.4.1	Teori	96
5.4.2	Plagiarisme	103

5.5	Dika Sukma Pradana 1174050	103
5.5.1	Pemahaman Teori	103
5.6	Ichsan Hizman Hardy (1174034)	107
5.6.1	Teori	107
5.6.2	Praktek	112
5.6.3	Penanganan Error	115
<b>6</b>	<b>Praktek Library CSV dan Pandas</b>	<b>141</b>
6.1	Hagan Rowlenstino/1174040	141
6.1.1	Soal 1	141
6.1.2	Soal 2	142
6.1.3	Soal 3	142
6.1.4	Soal 4	143
6.1.5	Penanganan Error	143
6.2	Faisal Najib Abdullah 1174042	144
6.2.1	Praktek	144
6.2.2	Penanganan Error	146
6.3	Irvan Rizkiansyah/1174043	147
6.3.1	Nomor 1	147
6.3.2	Nomor 2	148
6.3.3	Nomor 3	149
6.3.4	Nomor 4	149
6.3.5	Penanganan Error	150
6.4	Luthfi Muhammad Nabil/1174035	151
6.4.1	Soal 1	151
6.4.2	Soal 2	151
6.4.3	Soal 3	152
6.4.4	Soal 4	152
6.4.5	Penanganan Error	153
6.4.6	Dika Sukma Pradana 1174050	154
6.4.7	Soal 1	154
6.4.8	Soal 2	154
6.4.9	Soal 3	155
6.4.10	Soal 4	156
6.4.11	Penanganan Error	156



# DAFTAR GAMBAR

---

1.1	Contoh CSV	2
1.2	Plagiarisme	4
1.3	Contoh CSV Pada Excel	6
1.4	Contoh CSV Pada Text	6
1.5	Cek Plagiarisme	9
1.6	Contoh CSV	10
1.7	Plagiarisme	12
1.8	Cek Plagiarisme	14
1.9	Contoh CSV	16
1.10	Contoh CSV	18
2.1	Contoh KeyError	23
2.2	Hasil Penanganan Error	24
2.3	fix Error	25

3.1	Plagiarisme	39
3.2	Cek Plagiarisme	41
3.3	Pop Up saat instalasi Arduino IDE	42
3.4	Hubungkan arduino dengan komputer	42
3.5	Tampilan Device Manager	43
3.6	Port dari Arduino IDE	44
3.7	Port dari Device Manager	44
3.8	Baud Rate	45
3.9	Hasil Scan Plagiarism	54
4.1	Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino.	68
4.2	Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino dengan looping.	68
4.3	Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino dan langsung ditulis kedalam file CSV.	69
4.4	Hasil dari pembacaan fungsi untuk membaca file csv hasil arduino dan mengembalikan fungsi.	70
4.5	Hasil pembacaan fungsi untuk menanggulangi error menggunakan Try Except.	70
4.6	Hasil dari Ambil Data tanpa perulangan	71
4.7	Hasil dari Ambil Data dengan perulangan	71
4.8	Menulis data dari serial ke file CSV	72
4.9	Hasil Tulis CSV dengan ambilan tanpa loop	72
4.10	Hasil dari Error	73
4.11	Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino.	73
4.12	Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino dengan looping.	73
4.13	Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino dan langsung ditulis kedalam file CSV.	74
4.14	Hasil dari pembacaan fungsi untuk membaca file csv hasil arduino dan mengembalikan fungsi.	74

4.15	Hasil pembacaan fungsi untuk menanggulangi error menggunakan Try Except.	75
4.16	Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino.	75
4.17	Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino dengan looping.	75
4.18	Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino dan langsung ditulis kedalam file CSV.	76
4.19	Hasil dari pembacaan fungsi untuk membaca file csv hasil arduino dan mengembalikan fungsi.	76
4.20	Hasil pembacaan fungsi untuk menanggulangi error menggunakan Try Except.	77
5.1	No. 5	82
5.2	No. 7	83
5.3	Cek Plagiarisme	83
5.4	Plagiarisme	87
5.5	Contoh Pengaplikasian Sumbu X dan Y	88
5.6	Contoh Grafik Garis	89
5.7	Contoh Grafik Titik	90
5.8	Contoh Grafik Batang	90
5.9	Contoh Grafik Pie	91
5.10	Contoh Pengaplikasian Label	92
5.11	Contoh Pengaplikasian Legend	92
5.12	Contoh Pengaplikasian Subplot	94
5.13	Contoh Pengaplikasian Histogram	95
5.14	Cek Plagiarisme	96
5.15	Hasil compile membuat diagram menggunakan Matplotlib.	117
5.16	Hasil compile membuat bar graph menggunakan Matplotlib.	117
5.17	Hasil compile membuat histogram menggunakan Matplotlib.	118
5.18	Hasil compile membuat scatter plot menggunakan Matplotlib.	119
5.19	Hasil compile membuat diagram menggunakan Matplotlib.	120

5.20	Hasil compile membuat Pie Plot menggunakan Matplotlib.	120
5.21	Hasil compile membuat diagram menggunakan Matplotlib.	121
5.22	Hasil compile membuat diagram menggunakan Matplotlib.	122
5.23	Hasil compile membuat subplot menggunakan Matplotlib.	123
5.24	Hasil compile membuat diagram menggunakan Matplotlib.	124
5.25	Plagiarisme	124
5.26	Scatter	125
5.27	Histogram	125
5.28	pie chart	126
5.29	bagan area	126
5.30	legend	126
5.31	label	127
5.32	Subplot	127
5.33	Histogram	127
5.34	Screenshoot bebas plagiarisme	128
5.35	Diagram Batang	128
5.36	Hasil compile membuat bar graph menggunakan Matplotlib.	129
5.37	Hasil compile membuat histogram menggunakan Matplotlib.	130
5.38	Hasil compile membuat scatter plot menggunakan Matplotlib.	131
5.39	Hasil compile membuat diagram menggunakan Matplotlib.	132
5.40	Hasil compile membuat Pie Plot menggunakan Matplotlib.	133
5.41	Hasil compile membuat diagram menggunakan Matplotlib.	134
5.42	Hasil compile membuat subplot menggunakan Matplotlib.	135
5.43	Hasil compile membuat diagram menggunakan Matplotlib.	136
5.44	Hasil compile membuat fungsi Bar Plot menggunakan Matplotlib.	137
5.45	Hasil compile membuat fungsi Scatter Plot menggunakan Matplotlib.	137
5.46	Hasil compile membuat fungsi Pie Plot menggunakan Matplotlib.	138

5.47	Hasil compile membuat fungsi Plot menggunakan Matplotlib.	139
5.48	Hasil compile membuat fungsi penanganan error.	140
6.1	Hasil compile membuat fungsi Bar Plot menggunakan Matplotlib.	158
6.2	Hasil compile membuat fungsi Scatter Plot menggunakan Matplotlib.	158
6.3	Hasil compile membuat fungsi Pie Plot menggunakan Matplotlib.	159
6.4	Hasil compile membuat fungsi Plot menggunakan Matplotlib.	160
6.5	Grafik Bar	160
6.6	Grafik Scatter	161
6.7	Grafik Pie	161
6.8	Grafik Plot	162
6.9	Hasil Kompilasi Error	162
6.10	Grafik Batang	163
6.11	Grafik Scatter	163
6.12	Grafik Pie	164
6.13	Grafik Plot	164



## DAFTAR TABEL

---



# Listings

---

src/4/1174035/Teori/chap4_1174035_teori.py	8
src/4/1174035/Teori/chap4_1174035_teori.py	8
src/4/1174042/Teori/1174042_csv.py	13
src/4/1174042/Teori/1174042_csv.py	13
src/4/1174042/Teori/1174042_csv.py	14
src/4/1174042/Teori/1174042_csv.py	14
src/4/1174035/Praktek/chap4_1174035_csv.py	21
src/4/1174035/Praktek/chap4_1174035_csv.py	21
src/4/1174035/Praktek/chap4_1174035_pandas.py	22
src/4/1174035/Praktek/chap4_1174035_main.py	23
src/4/1174035/Praktek/chap4_1174035_main2.py	23
src/4/1174035/Praktek/chap4_1174035_error.py	23
src/4/1174043/Praktek/chap4_1174043_csv.py	24
src/4/1174043/Praktek/chap4_1174043_csv.py	24

src/4/1174043/Praktek/chap4_1174043_pandas.py	24
src/4/1174043/Praktek/chap4_1174043_pandas.py	24
src/4/1174043/Praktek/chap4_1174043_pandas.py	24
src/4/1174043/Praktek/chap4_1174043_pandas.py	25
src/4/1174043/Praktek/chap4_1174043_pandas.py	25
src/4/1174043/Praktek/chap4_1174043_main.py	25
src/4/1174043/Praktek/chap4_1174043_main2.py	25
src/4/1174043/Praktek/chap4_1174043_error.py	25
src/4/1174040/Praktek/1174040_CSV1.py	26
src/4/1174040/Praktek/1174040_CSV2.py	26
src/4/1174040/Praktek/1174040_pandas1.py	26
src/4/1174040/Praktek/1174040_pandas2.py	27
src/4/1174040/Praktek/1174040_pandas3.py	27
src/4/1174040/Praktek/1174040_pandas4.py	27
src/4/1174040/Praktek/1174040_pandas5.py	27
src/4/1174040/Praktek/1174040_main.py	28
src/4/1174040/Praktek/1174040_main2.py	28
src/4/1174040/Praktek/1174040_err.py	29
2.1 Fungsi untuk membuka file CSV dengan lib CSV mode list.	29
2.2 Fungsi untuk membuka file CSV dengan lib CSV mode dictionary.	29
2.3 Fungsi untuk membuka file CSV dengan lib Pandas mode list.	30
2.4 Fungsi untuk membuka file CSV dengan lib Pandas mode dictionary.	30
2.5 Fungsi untuk mengubah format tanggal menjadi standar dataframe.	30
2.6 Fungsi untuk mengubah index kolom.	30
2.7 Fungsi untuk mengubah atribut atau nama kolom.	30
2.8 Membuat dan mebaca file CSV menggunakan library 1174006pandas.	31
2.9 Membuat dan mmebaca file CSV menggunakan library 1174006pandas.	31
src/4/1174042/Praktek/1174042_2err.py	31
2.10 Fungsi untuk membuka file CSV dengan lib CSV mode list.	31
2.11 Fungsi untuk membuka file CSV dengan lib CSV mode dictionary.	32
2.12 Fungsi untuk membuka file CSV dengan lib Pandas mode list.	32
2.13 Fungsi untuk membuka file CSV dengan lib Pandas mode dictionary.	32
2.14 Fungsi untuk mengubah format tanggal menjadi standar dataframe.	32
2.15 Fungsi untuk mengubah index kolom.	32
2.16 Fungsi untuk mengubah atribut atau nama kolom.	33
2.17 Membuat dan mebaca file CSV menggunakan library 1174006pandas.	33
2.18 Membuat dan mmebaca file CSV menggunakan library 1174006pandas.	33
src/4/1174050/Praktek/1174050_2err.py	33

src/4/1174034/praktek/1174034csv.py	34
src/4/1174034/praktek/1174034csv.py	34
src/4/1174034/praktek/1174034pandas.py	34
src/4/1174034/praktek/1174034pandas.py	34
src/4/1174034/praktek/1174034pandas.py	35
src/4/1174034/praktek/1174034pandas.py	35
src/4/1174034/praktek/1174034pandas.py	35
src/4/1174034/praktek/1174034main.py	35
src/4/1174034/praktek/1174034main.py	35
src/4/1174034/praktek/1174034main2.py	35
src/4/1174034/Praktek/1174034_2err.py	35
src/5/1174035/Teori/chap5_1174035_teori.py	46
4.1 Fungsi untuk mendapatkan data dari Arduino.	55
4.2 Fungsi untuk mendapatkan data langsung dari Arduino dengan looping.	55
4.3 Fungsi untuk mendapatkan data dari Arduino dan langsung ditulis kedalam file CSV.	56
4.4 Fungsi untuk membaca file CSV hasil Arduino dan mengembalikan fungsi.	56
4.5 Fungsi untuk menanggulangi error menggunakan Try Except.	57
src/5/1174040/Praktek/chap5_1174040_realtime.py	57
src/5/1174040/Praktek/chap5_1174040_save.py	58
src/5/1174040/Praktek/chap5_1174040_realtime.py	58
src/5/1174040/Praktek/chap5_1174040_csv.py	59
src/5/1174040/Praktek/chap5_1174040_err.py	59
src/5/1174043/Praktek/chap5_1174043_realtime.py	60
src/5/1174043/Praktek/chap5_1174043_simpan.py	60
src/5/1174043/Praktek/chap5_1174043_realtime.py	60
src/5/1174043/Praktek/chap5_1174043_csv.py	60
src/5/1174043/Praktek/chap5_1174043_error.py	61
src/5/1174035/Praktek/chap5_1174035_realtime.py	61
src/5/1174035/Praktek/chap5_1174035_save.py	61
src/5/1174035/Praktek/chap5_1174035_realtime.py	62
src/5/1174035/Praktek/chap5_1174035_save.py	62
src/5/1174035/Praktek/chap5_1174035_error.py	63
4.6 Fungsi untuk mendapatkan data dari Arduino.	63
4.7 Fungsi untuk mendapatkan data langsung dari Arduino dengan looping.	63

4.8 Fungsi untuk mendapatkan data dari Arduino dan langsung ditulis kedalam file CSV.	64
4.9 Fungsi untuk membaca file CSV hasil Arduino dan mengembalikan fungsi.	64
4.10 Fungsi untuk menanggulangi error menggunakan Try Except.	65
4.11 Fungsi untuk mendapatkan data dari Arduino.	65
4.12 Fungsi untuk mendapatkan data langsung dari Arduino dengan looping.	66
4.13 Fungsi untuk mendapatkan data dari Arduino dan langsung ditulis kedalam file CSV.	66
4.14 Fungsi untuk membaca file CSV hasil Arduino dan mengembalikan fungsi.	66
4.15 Fungsi untuk menanggulangi error menggunakan Try Except.	67
src/6/1174040/Teori/chap6_1174040_2.py	79
src/6/1174040/Teori/chap6_1174040_3.py	80
src/6/1174040/Teori/chap6_1174040_5.py	81
src/6/1174040/Teori/chap6_1174040_hist.py	82
src/6/1174043/Teori/chap6_1174043_no2.py	84
src/6/1174043/Teori/chap6_1174043_legend.py	85
src/6/1174043/Teori/chap6_1174043_subplot.py	85
src/6/1174043/Teori/chap6_1174043_histogram.py	86
src/6/1174035/Teori/chap6_1174035_Teori.py	88
src/6/1174035/Teori/chap6_1174035_Teori.py	88
src/6/1174035/Teori/chap6_1174035_Teori.py	89
src/6/1174035/Teori/chap6_1174035_Teori.py	89
src/6/1174035/Teori/chap6_1174035_Teori.py	89
src/6/1174035/Teori/chap6_1174035_Teori.py	90
src/6/1174035/Teori/chap6_1174035_Teori.py	91
src/6/1174035/Teori/chap6_1174035_Teori.py	91
src/6/1174035/Teori/chap6_1174035_Teori.py	93
src/6/1174035/Teori/chap6_1174035_Teori.py	94
src/6/1174042/1174042.py	96
src/6/1174042/1174042.py	97
src/6/1174042/1174042.py	97
src/6/1174042/1174042.py	97

5.1 Kode program membuat diagram menggunakan Matplotlib.	97
5.2 Kode program membuat bar graph menggunakan Matplotlib.	97
5.3 Kode program membuat histogram menggunakan Matplotlib.	98
5.4 Kode program membuat scatter plot menggunakan Matplotlib.	98
5.5 Kode program membuat diagram menggunakan Matplotlib.	99
5.6 Kode program membuat Pie Plot menggunakan Matplotlib.	99
5.7 Kode program membuat diagram menggunakan Matplotlib.	100
5.8 Kode program menggunakan parameter label dengan Matplotlib.	100
5.9 Kode program memanggil fungsi legend dengan Matplotlib.	100
5.10 Kode program membuat diagram menggunakan Matplotlib.	101
5.11 Kode program membuat subplot menggunakan Matplotlib.	101
5.12 Kode program membuat diagram menggunakan Matplotlib.	102
src/6/1174050/Teori/no2.py	103
src/6/1174050/Teori/scatter.py	103
src/6/1174050/Teori/histo.py	104
src/6/1174050/Teori/pie.py	104
src/6/1174050/Teori/area.py	105
src/6/1174050/Teori/6.py	105
src/6/1174050/Teori/6.py	105
src/6/1174050/Teori/6.py	106
src/6/1174034/Teori/1174034.py	107
5.13 Kode program membuat diagram menggunakan Matplotlib.	107
5.14 Kode program membuat bar graph menggunakan Matplotlib.	108
5.15 Kode program membuat histogram menggunakan Matplotlib.	108
5.16 Kode program membuat scatter plot menggunakan Matplotlib.	109
5.17 Kode program membuat diagram menggunakan Matplotlib.	109
5.18 Kode program membuat Pie Plot menggunakan Matplotlib.	110
5.19 Kode program membuat diagram menggunakan Matplotlib.	110
5.20 Kode program membuat subplot menggunakan Matplotlib.	111
5.21 Kode program membuat diagram menggunakan Matplotlib.	112
5.22 Kode program membuat fungsi Bar Plot menggunakan Matplotlib.	113
5.23 Kode program membuat fungsi Scatter Plot menggunakan Matplotlib.	113
5.24 Kode program membuat fungsi Pie Plot menggunakan Matplotlib.	114
5.25 Kode program membuat fungsi Plot menggunakan Matplotlib.	115
5.26 Kode program membuat fungsi penanganan error.	116

src/6/1174040/Praktek/chap6_1174040_bar.py	141
src/6/1174040/Praktek/chap6_1174040_scatter.py	142
src/6/1174040/Praktek/chap6_1174040_pie.py	142
src/6/1174040/Praktek/chap6_1174040_plot.py	143
src/6/1174040/Praktek/chap6_1174040_err.py	143
6.1 Kode program membuat fungsi Bar Plot menggunakan Matplotlib.	144
6.2 Kode program membuat fungsi Scatter Plot menggunakan Matplotlib.	144
6.3 Kode program membuat fungsi Pie Plot menggunakan Matplotlib.	145
6.4 Kode program membuat fungsi Plot menggunakan Matplotlib.	146
6.5 Kode program membuat fungsi penanganan error.	147
src/6/1174043/Praktek/chap6_1174043_hist.py	147
src/6/1174043/Praktek/chap6_1174043_scatter.py	148
src/6/1174043/Praktek/chap6_1174043_pie.py	149
src/6/1174043/Praktek/chap6_1174043_plot.py	149
src/6/1174043/Praktek/chap6_1174043_error.py	150
src/6/1174035/Praktek/chap6_1174035_bar.py	151
src/6/1174035/Praktek/chap6_1174035_scatter.py	151
src/6/1174035/Praktek/chap6_1174035_pie.py	152
src/6/1174035/Praktek/chap6_1174035_plot.py	152
src/6/1174035/Praktek/chap6_1174035_error.py	153
src/6/1174050/Praktek/p1174050_bar.py	154
src/6/1174050/Praktek/main.py	154
src/6/1174050/Praktek/main.py	154
src/6/1174050/Praktek/p1174050_scatter.py	154
src/6/1174050/Praktek/main.py	155
src/6/1174050/Praktek/main.py	155
src/6/1174050/Praktek/p1174050_pie.py	155
src/6/1174050/Praktek/main.py	156
src/6/1174050/Praktek/main.py	156
src/6/1174050/Praktek/p1174050_plot.py	156
src/6/1174050/Praktek/main.py	156
src/6/1174050/Praktek/main.py	156

# **FOREWORD**

---

Sepatah kata dari Kaprodi, Kabag Kemahasiswaan dan Mahasiswa



# KATA PENGANTAR

---

Buku ini diciptakan bagi yang awam dengan flask sekalipun.

R. M. AWANGGA

*Bandung, Jawa Barat*

*Februari, 2019*



## ACKNOWLEDGMENTS

---

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

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

R. M. A.



## ACRONYMS

---

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



## GLOSSARY

---

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



# SYMBOLS

---

$A$  Amplitude

$\&$  Propositional logic symbol

$a$  Filter Coefficient

$B$  Number of Beats



# INTRODUCTION

---

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

Informatics Research Center  
Bandung, Jawa Barat, Indonesia

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

Rumusnya cuman sebagai contoh aja biar keren[?].

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



# BAB 1

---

## LIBRARY CSV DAN PANDAS

---

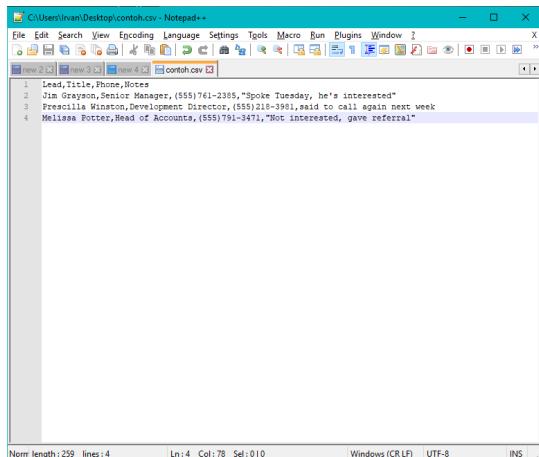
### 1.1 Irvan Rizkiansyah/1174043

#### 1.1.1 Soal 1

- Fungsi : File csv berfungsi untuk pencarian data akan menjadi lebih mudah dan cepat, dan juga mempermudah penginputan data ke dalam database secara sederhana.
- Sejarah : File csv muncul pertama kali sekitar 10 tahun sebelum Personal Computer (PC) pertama didunia yaitu sejak sekitar tahun 1972, akan tetapi sebutan file csv digunakan pertama kali pada tahun 1983.
- Contoh :  
??

#### 1.1.2 Soal 2

Ada banyak aplikasi yang dapat membuat file berformat CSV, diantaranya adalah :



**Gambar 1.1** Contoh CSV

- Notepad
- Notepad++
- Microsoft Excel
- Corel Quattro Pro
- Apache Open Office, dan masih banyak yang lainnya.

### 1.1.3 Soal 3

Cara menulis file csv menggunakan Excel :

1. Buka aplikasi Microsoft Excel kemudian buat dokumen baru
2. Tulis judul kolom untuk setiap informasi yang ingin di rekam atau catat, kemudian tulis informasi - informasi dalam kolom dengan sesuai.
3. Jika sudah selesai maka save dengan cara pilih menubar File lalu pilih Save As
4. Lalu isikan nama file tersebut dan rubah dengan memilih format file yang tersedia tersebut menjadi .csv
5. File csv sudah berhasil terbuat menggunakan Microsoft Excel

Cara membaca file csv menggunakan Excel :

1. Buka aplikasi Microsoft Excel kemudian pilih menu Open
2. Cari tempat file csv yang ingin dibuka, kemudian pilih Open
3. File csv sudah berhasil dibaca menggunakan Microsoft Excel

### 1.1.4 Soal 4

Pada file csv, tanda baca koma diartikan sebagai pembatas suatu kolom. List-directed input output didefinisikan dalam FORTRAN 77. List-directed input menggunakan tanda baca koma atau spasi sebagai pembatas, sehingga karakter yang tidak dikutip tidak dapat mengandung tanda baca koma ataupun spasi. Hal tersebut yang diadopsi oleh file csv. format csv didukung dengan library untuk banyak bahasa pemrograman, kebanyakan yang menspesifikasi pembatas field, pemisah desimal, pengkodean karakter, dan yang lainnya.

### 1.1.5 Soal 5

Pada tahun 2008, pengembangan pandas dimulai oleh AQR Capital Management. Pada akhir tahun 2009 pandas menjadi Open Sourced, dimana disupport oleh banyak komunitas atau individu di dunia untuk mengembangkan pandas. Sejak tahun 2015, pandas menjadi NumFOCUS proyek sponsor, ini juga membantu suksesnya pengembangan dari pandas itu sendiri. pandas merupakan struktur data dan data analysis tools untuk bahasa pemrograman Python, dan merupakan BSD-licensed library yang menjadikannya memiliki performa yang tinggi.

### 1.1.6 Soal 6

- Tanda baca koma : Menjadi pemisah antar kolom
- Tanda baca kutip dua : Menjadi cara untuk memasukan sebuah kalimat atau untuk memasukan karakter spasi sebagai data pada kolom informasi
- Inputan pada baris pertama akan menjadi Header, dimana akan menjadi nama sebuah kolom, dan masih banyak yang lainnya

### 1.1.7 Soal 7

Pada pandas sedikit berbeda, dimana inputan data berbentuk seperti peng-inputan pada variabel pada umumnya, hanya saja menggunakan tanda kutip satu untuk menandakan sebuah informasi pada kolom kemudian tanda kurung kotak yang didalamnya berisi informasi data dari kolom tersebut. dan lain sebagainya.

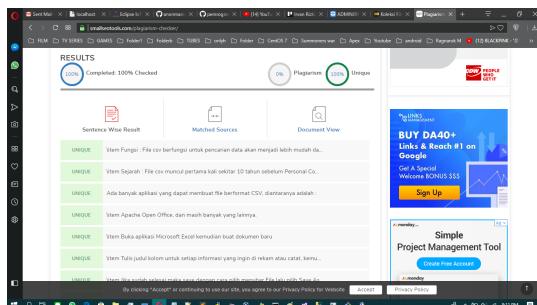
### 1.1.8 Cek Plagiarisme

??

## 1.2 Dwi Yulianingsih

### 1.2.1 Soal 1

Isi jawaban soal ke-1



**Gambar 1.2 Plagiarisme**

Kalau mau dibikin paragrap **cukup enter aja**, tidak usah pakai `par` 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 Soal 1

Isi jawaban soal ke-1

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

## 1.7 Tomy Prawoto

### 1.7.1 Soal 1

Isi jawaban soal ke-1

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

## 1.8 Luthfi Muhammad Nabil/1174035

### 1.8.1 Soal 1

Fungsi, Sejarah, dan Contoh file CSV :

- Fungsi : File CSV (Comma Separated Values) adalah tipe file khusus yang menyimpan informasi dengan metode dipisahkan dengan koma. File CSV berfungsi untuk menjadi perantara untuk beberapa aplikasi yang memiliki basis data saat mengirim data. CSV dapat dibuka di berbagai text editor yang ada. Dengan bentuk filenya yang dinamis memungkinkan file CSV dapat dimanipulasi dan dapat menyimpan informasi dengan skala besar.
- Sejarah : CSV sudah digunakan sejak tahun 1972 yang dapat dikompilasi pada bahasa pemrograman IBM Fortran. Saat itu, data yang dipisahkan oleh koma jika isinya memiliki spasi maka harus diberi tanda petik di awal dan akhir isi dari data tersebut. Nama CSV baru mulai digunakan pada tahun 1983. Pada panduan dari Osborne Executive Computer mendokumentasikan kutipan yang membolehkan isi karakter memiliki koma. Pada tahun 2005 dengan RFC4180, CSV didefinisikan sebagai MIME Content Type. lalu pada tahun 2013, definisi dari RFC4180 dipecahkan oleh rekomendasi dari W3C. Pada tahun 2014, IETF mempublikasi RFC7111 yang mendeskripsikan pecahan Uniform Resource Identifier(URI) ke dokumen CSV. RFC7111 menjelaskan bagaimana baris, kolom dapat dipilih dalam dokumen CSV menggunakan indeks posisi. Pada Tahun 2015, W3C mempublikasikan draft rekomendasi untuk CSV-metadata standards yang dimulai dengan rekomendasi pada bulan Desember dengan tahun yang sama.
- Contoh File CSV
  - CSV pada Excel 1.3
  - CSV pada Text Editor 1.4

### 1.8.2 Soal 2

Aplikasi Yang dapat membuat file CSV : Berikut file yang dapat membuat file CSV

- Spreadsheet : Spreadsheet merupakan aplikasi yang dapat membuat CSV hanya dengan memasukan data sesuai baris dan kolom yang diinginkan. Contoh spreadsheet seperti Google Spreadsheet, Microsoft Excel, dan aplikasi lainnya.

kolom1	kolom2	kolom3
isi1	isi2	isi3
isi4	isi5	isi6

**Gambar 1.3** Contoh CSV Pada Excel

```
kolom1,kolom2,kolom3
isi1,isi2,isi3
isi4,isi5,isi6
```

**Gambar 1.4** Contoh CSV Pada Text

- Bahasa Pemrograman : Bahasa pemrograman merupakan media yang dapat untuk membuat aplikasi yang dapat membuat file CSV khusus untuk bahasa pemrograman yang support dengan pembuatan file CSV. Seperti Python, C Sharp, dan lain sebagainya.
- Text Editor : Text editor juga dapat membuat file CSV, untuk membuat dengan Text Editor cukup dengan membuat file sesuai format CSV dan save file tersebut dengan ekstensi .CSV.

### 1.8.3 Soal 3

Menulis dan Membaca file CSV : Berikut cara menulis dan membaca file CSV :

- Menulis :
  1. Buka file CSV dengan spreadsheet
  2. Klik Cell yang mau diisi
  3. Masukan data yang mau diisi pada cell tersebut
  4. Lalu save file dengan format .CSV
- Membaca :
  1. Buka file CSV dengan spreadsheet

### 1.8.4 Soal 4

Sejarah Library CSV Python : Library CSV pada python merupakan library yang paling umum untuk import export data pada spreadsheet dan basis data dengan format sesuai dengan standarisasi RFC4180. Seiring dengan lahirnya bahasa pemrograman python, library mulai dibuat dan dikembangkan sampai akhirnya pada tahun 2003, pembuatnya Kevin Altis dan lainnya telah merilis versi final untuk library Python CSV.

### 1.8.5 Soal 5

Sejarah Library Pandas Python : Pandas (Python Data Analysis Library) adalah library open source yang digunakan untuk melakukan data manajemen dan data analysis. Pandas diciptakan pada tahun 2008 oleh Wes McKinney dan diperbaharui oleh Sien Chang pada tahun 2010. Inspirasi dari pembuatan pandas muncul pada komunitas yang membutuhkan library khusus untuk analisis data.

### 1.8.6 Soal 6

Fungsi - fungsi yang terdapat di library CSV :

- `csv.reader(csvfile, dialect='excel', **fmparams)`

Untuk mengembalikan object reader yang akan mengambil setiap line pada csv yang diambil. Data setiap baris diambil saat next() dipanggil. Berikut contohnya :

```

1 #CSV Reader
2 import csv
3 with open('cobain.csv', newline='') as csvfile:
4     spamreader = csv.reader(csvfile, delimiter=',', quotechar=',')
5     for row in spamreader:
6         print(','.join(row))

```

- `csv.writer(csvfile, dialect='excel', **fmtparams)`

Mengembalikan file pembuat object untuk dapat mengonversi data pada python ke file CSV yang akan dibuat. Berikut contoh penggunaan `csv.writer` :

```

1 #CSV Writer
2 import csv
3 with open('cobain.csv', 'w', newline='') as csvfile:
4     spamwriter = csv.writer(csvfile, delimiter=',',
5                             quotechar=',', quoting=csv.
      QUOTE_MINIMAL)
6     spamwriter.writerow(['Eh coba'] * 5 + [' Coba'])
7     spamwriter.writerow(['Isi', 'Isi2', 'Isi3'])

```

- `csv.register_dialect(name[, dialect[, **fmtparams]])`

Mengasosiasikan dialek dengan nama, nama yang dimasukkan harus berupa karakter.

- `csv.unregister_dialect(name)`

Menghapus asosiasi dialek dengan nama pada registry dialek.

- `csv.get_dialect(name)`

Mengambil dialek yang telah diasosiasikan dengan nama.

- `csv.list_dialects()`

Mengembalikan dialek yang telah diregistrasi.

- `csv.field_size_limit([new_limit])`

Mengembalikan maksimal kolom data yang diperbolehkan oleh pembaca.

### 1.8.7 Soal 7

Fungsi - fungsi yang terdapat di library Pandas :

- `pandas.read_csv(filepath_or_buffer[, sep, ...])`

Untuk membaca file CSV dan menyimpannya ke DataFrame

- `pandas.read_excel(io[, sheet_name, header, names, ...])`

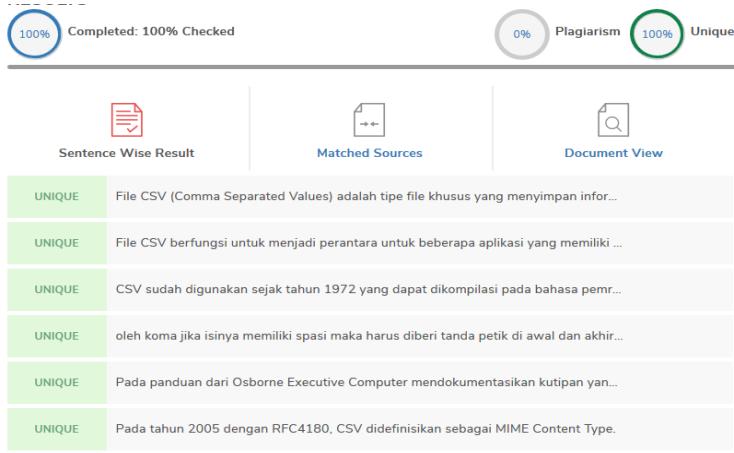
Membaca file excel dan menyimpannya ke DataFrame

- `to_csv([path, index, sep, na_rep, ...])`

Untuk membuat file CSV dari data yang ada

### 1.8.8 Cek Plagiarism

Berikut pengecekan plagiarism yang dilakukan pada website smallseotools.com :



**Gambar 1.5** Cek Plagiarisme

## 1.9 Hagan Rowlenstino/1174040

### 1.9.1 Soal 1

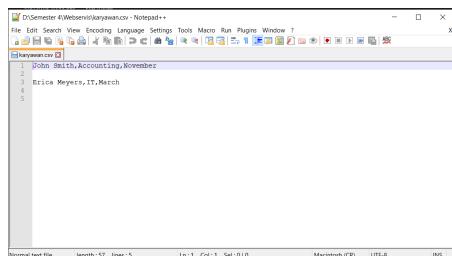
format file csv dapat menyimpan data dalam jumlah yang sangat besar juga diprgunakan untuk export dan import untuk spreadsheet ataupun database. Singkatan CSV pertamakali di pakai pada tahun 1983, dimana value yang dipisahkan dengan koma lebih mudah untuk diketik daripada data yang sejajar dengan kolom yang tetap. contohnya seperti gambar dibawah ini.

### 1.9.2 Soal 2

Ms.Excel , NotePad, notepad++, sublime, dan texteditor lainnya

### 1.9.3 Soal 3

caranya adalah :



**Gambar 1.6** Contoh CSV

- untuk write :
    1. Download template csv
    2. Buka browser lalu menuju ke Google Sheet
    3. Tekan tombol merah di pojok kanan bawah
    4. Lalu pilih upload file untuk mengupload template yang sudah di download sebelumnya
    5. Edit sesuai yang diinginkan
    6. Setelah selesai, lakukan eksport ke CSV dengan cara klik file lalu download as setelah itu pilih CSV
  - untuk read :
    1. buka Ms.Excel
    2. pilih Data lalu Get External Data dan pilih From Text
    3. lalu pilih file csv nya
    4. pilih Delimited lalu Next
    5. checklist di box Tab dan Comma
    6. lalu klik finish

#### 1.9.4 Soal 4

Library umum dalam CSV yang gunanya untuk import dan export data di dalam database yang terstandarisasi RFC4180 yang berisikan fungsi-fungsi dan kelas yang akan dipakai dalam penggerjaan file CSV.

### 1.9.5 Soal 5

Pandas diciptakan pada tahun 2008 oleh Wes McKinney dan diperbaharui pada tahun 2010 oleh Sien Chang. yang fungsinya untuk melakukan analisa data seperti import dan export data.

### 1.9.6 Soal 6

Fungsi - fungsi library csv adalah :

- `csv.reader(csvfile, dialect='excel', **fmtparams)`  
: digunakan untuk membaca line di csv
- `csv.writer(csvfile, dialect='excel', **fmtparams)`  
: untuk menulis line di csv
- `csv.register_dialect(name[, dialect[, **fmtparams]])`  
: untuk asosiasikan dialect dengan name, dimana name harus string
- `csv.unregister_dialect(name)`  
: menghapus dialect yang terasosiasi dengan name
- `csv.get_dialect(name)`  
: mengembalikan hasil dialect yang terasosiasi dengan name
- `csv.list_dialects()`  
: menampilkan semua dialect yang ada
- `csv.field_size_limit([new_limit])`  
: menampilkan field maksimal ayng di berikan oleh pembubat parse.

### 1.9.7 Soal 7

Fungsi - fungsi yang terdapat di library Pandas :

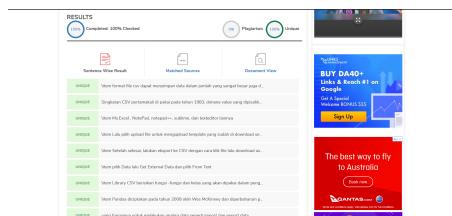
- `pandas.read_csv(filepath_or_buffer[, sep, ...])`  
: Untuk membaca file CSV
- `pandas.read_excel(io[, sheet_name, header, names, ...])`  
: Membaca file excel
- `to_csv([path, index, sep, na_rep, ...])`  
: Untuk me write ke dalam file csv

### 1.9.8 Cek Plagiarisme

## 1.10 Faisal Najib Abdullah 1174042

### 1.10.1 Pemahaman Teori

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



### **Gambar 1.7 Plagiarisme**

File CSV Nilai Berbasiskoma adalah tipe file khusus yang dapat Anda buat atau edit di Excel. File CSV menyimpan informasi yang dipisahkan oleh koma, bukan menyimpan informasi dalam kolom. Saat teks dan angka disimpan dalam file CSV, mudah untuk memindahkannya dari satu program ke program lain.

File CSV dibuat oleh program yang menangani sejumlah data yang besar. CSV merupakan 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 mengimportnya ke dalam spreadsheet untuk menganalisis data, menghasilkan grafik untuk presentasi, atau menyiapkan laporan untuk publikasi.

Contohnya, Anda dapat mengekspor kontak dari Google ke dalam file CSV, kemudian mengimpornya ke Outlook.

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

- Microsoft Excel 2013
  - Microsoft Works
  - Corel Quattro Pro
  - Apache OpenOffice
  - LibreOffice
  - Microsoft Notepad
  - Intuit Quicken 2015
  - GenScriber

## Pada Mac OS

- Microsoft Excel 2011
  - Planamesa NeoOffice
  - Apache OpenOffice
  - LibreOffice
  - GenScriber

Pada Linux

- Apache OpenOffice
- LibreOffice
- GenScriber

3. Jelaskan bagaimana cara menulis dan membaca file csv di excel atau spreadsheet?
  - Cara menulis file csv harus berupa baris dan kolom atau bisa juga di sebut berupa tabel.
  - Untuk membacanya file csv dipisahkannya menggunakan koma atau titik koma.
4. Jelaskan sejarah library csv? Library csv menyediakan fungsionalitas untuk membaca dan menulis ke file CSV. Dirancang untuk bekerja di luar kotak dengan file CSV yang dihasilkan Excel, memudahkan untuk bekerja dengan berbagai format CSV. Library csv berisi objek dan kode lain untuk membaca, menulis, dan memproses data ke file CSV.
5. Jelaskan sejarah library pandas? panda adalah pustaka Python open-source yang menyediakan alat analisis data kinerja tinggi dan struktur data yang mudah digunakan. panda tersedia untuk semua instalasi Python, tetapi itu adalah bagian penting dari distribusi Anaconda dan bekerja sangat baik di notebook Jupyter untuk berbagi data, kode, hasil analisis, visualisasi, dan teks naratif.
6. Jelaskan fungsi-fungsi yang terdapat di library csv? Terdapat 2 fungsi yang bisa digunakan oleh library csv Pertama, fungsi membaca file csv. fungsi ini bisa menggunakan list dan dictionary Dengan list :

```

1 def bacacsv():
2     with open('1174042.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'Column names are {" ".join(row)}')
8                 line_count += 1
9             else:
10                 print(f'\t NPM : {row[0]} Nama : {row[1]} Kelas :
11 {row[2]}.')
12                 line_count += 1

```

Dengan dictionary :

```

1 def bacadictionary():
2     with open('1174042.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'Column names are {" ".join(row)}')
8                 line_count += 1

```

```

9         print(f'\t NPM : {row["npm"]}\t Nama : {row["nama"]}\t
10        Kelas : {row["kelas"]}\t')
11        line_count += 1

```

Kedua, fungsi menulis file csv.

```

1 def tulis():
2     with open('test-tulis.csv', mode='w') as employee_file:
3         employee_writer = csv.writer(employee_file, delimiter=',',
4                                         quotechar='"', quoting=csv.QUOTE_MINIMAL)
5
6         employee_writer.writerow(['Ucok', 'Tukang', 'November'])

```

7. Jelaskan fungsi-fungsi yang terdapat di library pandas. Hampir sama dengan library csv, tp library pandas penulisannya lebih sederhana dan terlihat lebih rapih dari pada library csv.

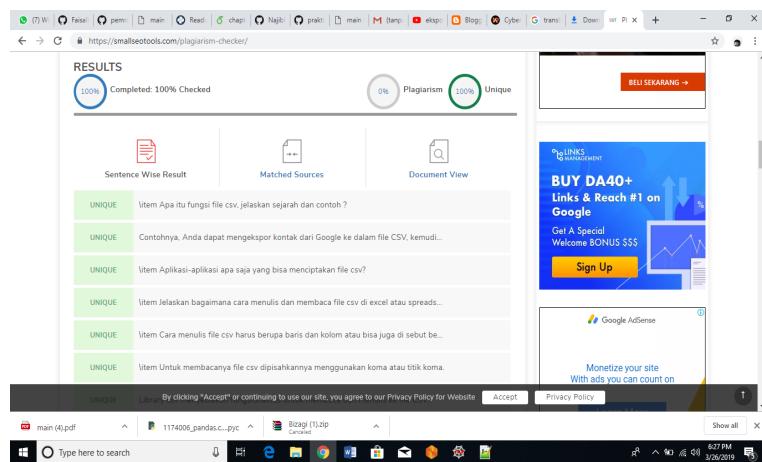
```

1 def bacapandas():
2     df = pandas.read_csv('1174042.csv')

```

## 1.10.2 Cek Plagiarism

Berikut pengecekan plagiarism yang dilakukan pada website smallseotools.com :



Gambar 1.8 Cek Plagiarisme

## 1.11 Dika Sukma Pradana 1174050

### 1.11.1 Pemahaman Teori

1. Definisi, Sejarah, dan Contoh

- Definisi

Dalam komputasi, file CSV (Comma-separated values) adalah file teks terbatas yang menggunakan koma untuk memisahkan nilai. File CSV menyimpan data tabular (angka dan teks) dalam teks biasa. Setiap baris file adalah catatan data. Setiap catatan terdiri dari satu atau lebih bidang, dipisahkan dengan koma. Penggunaan koma sebagai pemisah bidang adalah sumber nama untuk format file ini.

- Sejarah

Nama Comma-separated values dan singkatan CSV digunakan pada tahun 1983. Manual untuk komputer Osborne Executive, yang membundel spreadsheet SuperCalc, mendokumentasikan konvensi kutipan CSV yang memungkinkan string mengandung koma yang disematkan, tetapi manual tersebut tidak menentukan konvensi untuk menanamkan tanda kutip dalam string yang dikutip.

Daftar nilai yang dipisahkan dengan 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 ditinjau satu kolom dari lokasi yang dituju.

File yang dipisahkan koma digunakan untuk pertukaran informasi basis data antara mesin dari dua arsitektur yang berbeda. Karakter teks-polos dari file CSV sebagian besar menghindari ketidakcocokan seperti urutan byte dan ukuran kata. File-file ini sebagian besar dapat dibaca oleh manusia, sehingga lebih mudah untuk mengatasinya tanpa adanya dokumentasi atau komunikasi yang sempurna.

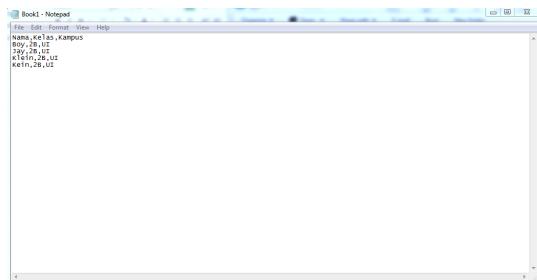
Inisiatif standardisasi utama - mentransformasikan definisi fuzzy de facto menjadi definisi yang lebih tepat dan de jure - adalah pada tahun 2005, dengan RFC4180, mendefinisikan CSV sebagai Tipe Konten MIME. Kemudian, pada 2013, beberapa kekurangan RFC4180 ditangani oleh rekomendasi W3C.

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

- Contoh : ??

## 2. Cara menulis dan membaca file csv di excel

- Ada banyak aplikasi yang dapat membuat file berformat CSV, diantaranya adalah :
  - Microsoft Excel
  - Corel Quattro Pro



**Gambar 1.9** Contoh CSV

- Apache Open Office
  - CSVed
  - CSVstar
  - CSVpad
  - Dan masih banyak lagi.
- Cara menulis file CSV di Excel :
- (a) Buat dokumen baru di Excel
  - (b) Tambahkan judul kolom untuk setiap potongan informasi yang ingin dicatat
  - (c) Pilih File
  - (d) Save As
  - (e) Gunakan kotak menurun untuk memilih format CSV
  - (f) Beri nama pada file
  - (g) Simpan
- Cara membaca file csv menggunakan Excel :
- (a) Buka aplikasi Microsoft Excel kemudian pilih menu Open
  - (b) Cari tempat file csv yang ingin dibuka
  - (c) Kemudian pilih Open
  - (d) File csv sudah berhasil dibaca menggunakan Microsoft Excel

### 3. Sejarah library csv

Nilai yang dipisahkan oleh koma adalah format data yang memberi tanggal lebih awal pada komputer pribadi lebih dari satu dekade: kompiler IBM Fortran di bawah OS atau 360 mendukungnya pada tahun 1972. Input atau output daftar-diarahkan 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.

### 4. Sejarah library pandas

Pengembang Wes McKinney mulai mengerjakan panda pada 2008 ketika di AQR Capital Management karena kebutuhan akan alat kinerja tinggi yang fleksibel untuk melakukan analisis kuantitatif pada data keuangan. Sebelum meninggalkan AQR, dia bisa meyakinkan manajemen untuk mengizinkannya membuka sumber perpustakaan. Pegawai AQR lainnya, Chang She, bergabung dengan upaya ini pada 2012 sebagai kontributor utama kedua ke perpustakaan. Pada 2015, panda ditandatangani sebagai proyek NumFOCUS yang disponsori secara fiskal, sebuah badan amal nirlaba 501 di Amerika Serikat.

## 5. Fungsi - fungsi csv

- Membaca file

```
import csv
```

```
with open('employee_birthday.txt') as csv_file:
    csv_reader = csv.reader(csv_file, delimiter=',')
```

- Menulis file

```
import csv
```

```
with open('employee_file.csv', mode='w') as employee_file:
    employee_writer = csv.writer(employee_file, delimiter=',')
```

## 6. Fungsi - fungsi pandas

- Membaca file

```
import pandas
df = pandas.read_csv('hrdata.csv')
```

- Menulis file

```
import pandas
df = pandas.read_csv('hrdata.csv',
index_col='Employee',
parse_dates=['Hired'],
header=0,
names=['Employee', 'Hired', 'Salary', 'Sick Days'])
df.to_csv('hrdata_modified.csv')
```

## 1.12 IchsanHizmanHardy/1174034

### 1.12.1 Pemahaman Teori

1. ▪ Apa saja fungsi dari File csvFile csv berfungsi untuk pencarian data akan menjadi lebih mudah dan cepat, dan juga mempermudah penginputan data ke dalam database secara sederhana. File csv akan menyimpan informasi

data yang dipisahkan dengan koma atau tanda titik koma, dimana artinya file csv tidak menyimpan data dalam bentuk kolom.

- Sejarahnya, File csv muncul pertama kali sekitar 10 tahun sebelum Personal Computer (PC) pertama didunia yaitu sejak sekitar tahun 1972, akan tetapi sebutan file csv digunakan pertama kali pada tahun 1983.
- Contoh :

The screenshot shows a Notepad window titled "1174034.csv - Notepad". The menu bar includes File, Edit, Format, View, and Help. The content of the file is as follows:

```
File Edit Format View Help
hama, prodi, kelas
Ichsan, D4 TI, 2B
Alit, D4 TI, 2B
Dika, D4 TI, 2B
Najib, D4 TI, 2B
```

Gambar 1.10 Contoh CSV

- ??
2. Ada banyak aplikasi yang dapat membuat file berformat CSV, diantaranya adalah :
  - Notepad
  - Notepad++
  - Microsoft Excel, dan masih banyak yang lainnya.
  3. Cara menulis file csv menggunakan Excel :
    - (a) Buka aplikasi Microsoft Excel kemudian buat dokumen baru
    - (b) Tulis judul kolom untuk setiap informasi yang ingin di rekam atau catat, kemudian tulis informasi - informasi dalam kolom dengan sesuai.
    - (c) Jika sudah selesai maka save dengan cara pilih menubar File lalu pilih Save As
    - (d) Lalu isikan nama file tersebut dan rubah dengan memilih format file yang tersedia tersebut menjadi .csv
    - (e) File csv sudah berhasil terbuat menggunakan Microsoft Excel
  4. Cara membaca file csv menggunakan Excel :
    - (a) Buka aplikasi Microsoft Excel kemudian pilih menu Open
    - (b) Cari tempat file csv yang ingin dibuka, kemudian pilih Open

- (c) File csv sudah berhasil dibaca menggunakan Microsoft Excel
5. Pada file csv, tanda baca koma diartikan sebagai pembatas suatu kolom. List-directed input output didefinisikan dalam FORTRAN 77. List-directed input menggunakan tanda baca koma atau spasi sebagai pembatas, sehingga karakter yang tidak dikutip tidak dapat mengandung tanda baca koma ataupun spasi. Hal tersebut yang diadopsi oleh file csv. format csv didukung dengan library untuk banyak bahasa pemrograman, kebanyakan yang menspesifikasi pembatas field, pemisah desimal, pengkodean karakter, dan yang lainnya.
  6. Pada tahun 2008, pengembangan pandas dimulai oleh AQR Capital Management. Pada akhir tahun 2009 pandas menjadi Open Sourced, dimana disupport oleh banyak komunitas atau individu di dunia untuk mengembangkan pandas. Sejak tahun 2015, pandas menjadi NumFOCUS proyek sponsor, ini juga membantu suksesnya pengembangan dari pandas itu sendiri. pandas merupakan struktur data dan data analysis tools untuk bahasa pemrograman Python, dan merupakan BSD-licensed library yang menjadikannya memiliki performa yang tinggi.
  7.
    - Tanda baca koma : Menjadi pemisah antar kolom
    - Tanda baca kutip dua : Menjadi cara untuk memasukan sebuah kalimat atau untuk memasukan karakter spasi sebagai data pada kolom informasi
    - Inputan pada baris pertama akan menjadi Header, dimana akan menjadi nama sebuah kolom, dan masih banyak yang lainnya
  8. Pada pandas sedikit berbeda, dimana inputan data berbentuk seperti peng-inputan pada variabel pada umumnya, hanya saja menggunakan tanda kutip satu untuk menandakan sebuah informasi pada kolom kemudian tanda kurung kotak yang didalamnya berisi informasi data dari kolom tersebut. dan lain sebagainya.



## BAB 2

---

# PRAKTEK LIBRARY CSV DAN PANDAS

---

### 2.1 Luthfi Muhammad Nabil/1174035

#### 2.1.1 Soal 1

Buatlah fungsi pada file chap4\_1174035.csv.py untuk membuka file csv dengan lib csv mode list :

```
1 def CSVModeList():
2     with open('chap4_1174035.csv', 'r') as f:
3         reader = csv.reader(f)
4         your_list = list(reader)
5     return your_list
```

#### 2.1.2 Soal 2

Buatlah fungsi pada file chap4\_1174035.csv.py untuk membuka file csv dengan lib csv mode dictionary :

```
1 def CSVModeDict():
2     with open('chap4_1174035.csv', newline='') as csvfile:
```

```

3     reader = csv.DictReader(csvfile)
4     for row in reader:
5         print(row['NPM'], row['Nama'], row['Jurusan'])

```

### 2.1.3 Soal 3

Buatlah fungsi pada file chap4\_1174035\_pandas.py untuk membuka file csv dengan lib pandas mode list :

```

1 def PandasModeList():
2     print("Pandas Mode List : ")
3     df = pd.read_csv('chap4_1174035_csv.csv', delimiter=',')
4     print(df)

```

### 2.1.4 Soal 4

Buatlah fungsi pada file chap4\_1174035\_pandas.py untuk membuka file csv dengan lib pandas mode dictionary :

```

1 def PandasModeDict():
2     print('Pandas Mode Dictionary : ')
3     df = pd.read_csv('chap4_1174035_csv.csv', delimiter=',')
4     res = df.to_dict(orient='records')
5     print(res)

```

### 2.1.5 Soal 5

Buatlah fungsi baru di chap4\_1174035\_pandas.py untuk mengubah format tanggal menjadi standard dataframe :

```

1 def PandasDateToStandardDT():
2     df = pd.read_csv('chap4_1174035_csv.csv', parse_dates=[ 'ttl' ])
3     print(df)

```

### 2.1.6 Soal 6

Buatlah fungsi baru di chap4\_1174035\_pandas.py untuk mengubah index kolom :

```

1 def PandasIndexCol():
2     df = pd.read_csv('chap4_1174035_csv.csv', index_col='Nama')
3     print(df)

```

### 2.1.7 Soal 7

Buatlah fungsi baru di chap4\_1174035\_pandas.py untuk mengubah atribut atau nama kolom :

```

1 def PandasNameAttribute():
2     df = pd.read_csv('chap4_1174035_csv.csv', header=0, names=['Nomor
3         Mahasiswa', 'Nama', 'Prodi', 'Tanggal Lahir'])
4     print(df)

```

## 2.1.8 Soal 8

Buatlah program chap4\_1174035\_main.py yang menggunakan library chap4\_1174035\_csv.py yang membuat dan membaca file CSV :

```
1 import chap4_1174035_csv as libcsv
2
3 #R/W
4 libcsv.WriteCSV(libcsv.CSVModeList())
```

## 2.1.9 Soal 9

Buatlah program chap4\_1174035\_main2.py yang menggunakan library chap4\_1174035\_csv.py yang membuat dan membaca file CSV :

```
1 import chap4_1174035_pandas as libpandas
2
3 #R/W
4 libpandas.PandasWrite()
```

## 2.1.10 Penanganan Error

Error yang didapat : KeyError Deskripsi : Error saat kunci ada yang salah atau tidak ada di dalam file CSV Penanganan : Menggunakan KeyError seperti pada line

```
D:\Coba\chap4>python chap4_1174035_error.py
Traceback (most recent call last):
  File "chap4_1174035_error.py", line 8, in <module>
    ErrorCoba()
  File "chap4_1174035_error.py", line 7, in ErrorCoba
    print(row['NP'], row['Nama'], row['Jurusan'])
KeyError: 'NP'
```

Gambar 2.1 Contoh KeyError

berikut :

```
1 def ErrorCoba():
2     import csv
3     try:
4         with open('chap4_1174035_csv.csv', newline='') as csvfile:
5             reader = csv.DictReader(csvfile)
6             for row in reader:
7                 print(row['NP'], row['Nama'], row['Jurusan'])
8     except KeyError :
9         print("Kunci ada yang salah")
10 ErrorCoba()
```

```
D:\Coba\chap4>python chap4_1174035_error.py
Kunci ada yang salah
```

**Gambar 2.2** Hasil Penanganan Error

## 2.2 Irvan Rizkiansyah/1174043

### 2.2.1 Soal 1

```
1 def nomor1():
2     with open('chap4_1174043_csv.csv', mode='r') as csvfile:
3         readCSV = csv.reader(csvfile, delimiter=',')
4         for row in readCSV:
5             print(row)
```

### 2.2.2 Soal 2

```
1 def nomor2():
2     with open('chap4_1174043_csv.csv', mode='r') as csvfile:
3         readCSV = csv.DictReader(csvfile)
4         for row in readCSV:
5             print(row)
```

### 2.2.3 Soal 3

```
1 def nomor3():
2     df = pd.read_csv('chap4_1174043_csv.csv')
3     print(df)
```

### 2.2.4 Soal 4

```
1 def nomor4():
2     df = pd.read_csv('chap4_1174043_csv.csv', usecols=['nomor'])
3     result = df.to_dict(orient='records')
4     print(result)
```

### 2.2.5 Soal 5

```
1 def nomor5():
2     df = pd.read_csv('chap4_1174043_csv.csv', parse_dates=[1])
3     print(df)
```

## 2.2.6 Soal 6

```

1 def nomor6():
2     df = pd.read_csv('chap4_1174043_csv.csv')
3     df.rename(columns={'nomor': 'No.'}, inplace=True)
4     print(df)

```

## 2.2.7 Soal 7

```

1 def nomor7():
2     df = pd.read_csv('chap4_1174043_csv.csv')
3     df.rename(columns={'nomor': 'No.'}, inplace=True)
4     print(df)

```

## 2.2.8 Soal 8

```

1 import chap4_1174043_csv as fungsi_csv
2
3 fungsi_csv.nomor1()
4 fungsi_csv.nomor2()
5 fungsi_csv.WriteCSV()

```

## 2.2.9 Soal 9

```

1 import chap4_1174043_pandas as pandaslib
2
3 pandaslib.nomor3()
4 pandaslib.nomor4()
5 pandaslib.nomor5()
6 pandaslib.nomor6()
7 pandaslib.nomor7()

```

## 2.2.10 Penanganan Error

```

1 def ErrorCoba():
2     import csv
3     try:
4         with open('chap4_1174043_csv.csv', newline='') as csvfile:
5             reader = csv.DictReader(csvfile)
6             for row in reader:
7                 print(row['nomor'], row['poin'], row['tg1'])
8     except KeyError:
9         print("Terdapat ERROR")
10 ErrorCoba()

```

```

In [26]: runfile('C:/Users/Irvan/Desktop/Kuliah/Semester 4/Pemrograman 3 (Web Service)/
Tugas 4/chap4_1174043_error.py', wdir='C:/Users/Irvan/Desktop/Kuliah/Semester 4/
Pemrograman 3 (Web Service)/Tugas 4')
Terdapat ERROR

```

**Gambar 2.3 fix Error**

## 2.3 Hagan Rowlenstino/1174040

### 2.3.1 Soal 1

Buatlah fungsi untuk membuka file csv dengan lib csv mode list :

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Mon Mar 25 09:10:54 2019
4
5 @author: Haganz
6 """
7
8 import csv
9
10 with open('1174040_csv.csv') as csv_file:
11     csv_reader = csv.reader(csv_file, delimiter=',')
12     for row in csv_reader:
13         print(row)

```

### 2.3.2 Soal 2

Buatlah fungsi untuk membuka file csv dengan lib csv mode dictionary :

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Mon Mar 25 09:10:54 2019
4
5 @author: Haganz
6 """
7
8 import csv
9
10 with open('1174040_csv.csv', mode='r') as csv_file:
11     csv_reader = csv.DictReader(csv_file)
12     for row in csv_reader:
13         print(row)

```

### 2.3.3 Soal 3

Buatlah fungsi untuk membuka file csv dengan lib pandas mode list :

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Tue Mar 26 09:02:39 2019
4
5 @author: Haganz
6 """
7
8 import pandas
9 df = pandas.read_csv('1174040_csvpandas.csv')
10 print(df)

```

### 2.3.4 Soal 4

Buatlah fungsi untuk membuka file csv dengan lib pandas mode dictionary :

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Tue Mar 26 09:06:49 2019
4
5 @author: Haganz
6 """
7
8
9 import pandas
10 dt = pandas.read_csv('1174040_csvpandas.csv').T.to_dict()
11 print(dt)

```

### 2.3.5 Soal 5

Buatlah fungsi untuk mengubah format tanggal menjadi standard dataframe :

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Tue Mar 26 09:36:01 2019
4
5 @author: Haganz
6 """
7
8 import pandas
9 df = pandas.read_csv('1174040_csvpandas.csv', index_col='Name',
10                      parse_dates=['Hire Date'])
10 print(df)

```

### 2.3.6 Soal 6

Buatlah fungsi untuk mengubah index kolom :

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Tue Mar 26 15:04:46 2019
4
5 @author: Haganz
6 """
7
8 import pandas
9 df = pandas.read_csv('1174040_csvpandas.csv', index_col='Hire Date')
10 print(df)

```

### 2.3.7 Soal 7

Buatlah fungsi untuk mengubah atribut atau nama kolom :

```

1 # -*- coding: utf-8 -*-
2 """

```

```

3 Created on Tue Mar 26 09:40:39 2019
4
5 @author: Haganz
6 """
7
8 import pandas
9 df = pandas.read_csv('1174040_csvpandas.csv',
10                      index_col='Employee',
11                      parse_dates=['Hired'],
12                      header=0,
13                      names=['Employee', 'Hired', 'Salary', 'Hari Sakit'])
14 print(df)

```

### 2.3.8 Soal 8

Disini saya telah membuat file CSV bernama 1174040\_csv.csv untuk di tampilkan, dan sebelum me write ke dalam file csv, terlebih dahulu buat file 1174040\_writecsv.csv :

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Tue Mar 26 12:25:57 2019
4
5 @author: Haganz
6 """
7
8 ## Read
9 import CSV_1174040
10 x = CSV_1174040
11 file = '1174040_csv.csv'
12 hasil = x.Readcsv(file)
13
14 ## Write
15 fil = '1174040_writecsv.csv'
16 result = x.Writecsv(fil)

```

### 2.3.9 Soal 9

Disini saya telah membuat file CSV bernama 1174040\_csvpandas.csv untuk ditampilkan, dan sebelum me write ke dalam file csv, telebih dahulu buat file 1174040\_writepandas.csv :

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Tue Mar 26 13:05:18 2019
4
5 @author: Haganz
6 """
7
8 import pandas_1174040
9 ##read
10 x = pandas_1174040
11 file = '1174040_csvpandas.csv'

```

```

12 hasil = x.Readpandas(file)
13
14 ##write
15 fil = '1174040_writepandas.csv'
16 result = x.Writepandas(fil)

```

### 2.3.10 Keterampilan Penanganan Error

error disini adalah FileNotFoundError : Dimana error yang terjadi karena kesalahan dalam penginputan nama file yang akan di read.

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Tue Mar 26 13:19:17 2019
4
5 @author: Haganz
6 """
7 import pandas
8 try :
9     df = pandas.read_csv('1174040_csvapandas.csv', index_col='Hire
Date')
10    print(df)
11 except FileNotFoundError:
12     print("Nama file nya salah , yang benar adalah 1174040_csvpandas.
txt")

```

## 2.4 Faisal Najib Abdullah 1174042

### 2.4.1 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('1174042teori.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 2.1** Fungsi untuk membuka file CSV dengan lib CSV mode list.

- 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('1174042teori.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 2.2** 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('1174042 teori.csv')
4     print(df)
```

**Listing 2.3** 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('1174042 teori.csv')
4     print(df)
```

**Listing 2.4** 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('1174042 teori.csv')
4     dt = pandas.DataFrame.from_dict(df)
5     print(dt)
```

**Listing 2.5** 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('1174042 teori.csv', parse_dates=['tanggal lahir'])
4     print(df)
```

**Listing 2.6** 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('1174042 teori.csv')
4     df.index = ['Row_1', 'Row_2']
5     print(df)
```

**Listing 2.7** 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__('1174042csv')
2
3 lib.bukaModeDictCsv()
4
5 lib.tulisCsv()

```

**Listing 2.8** Membuat dan membaca file CSV menggunakan library 1174006pandas.

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

```

1 lib = __import__('1174042pandas')
2
3 lib.bukaModeListPandas()
4 lib.bukaModeDictPandas()
5
6 lib.tulisCsvPandas()

```

**Listing 2.9** Membuat dan membaca file CSV menggunakan library 1174006pandas.

## 2.4.2 Ketrampilan Penanganan Error

Error yang 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 a = 1
2 b = 10
3 c = "7"
4
5 try :
6     d = a + c
7 except TypeError:
8     print("ini teh beda atuhh type datanya mana bisa bersatu")

```

## 2.5 Dika Sukma Pradana 1174050

### 2.5.1 Praktek

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

```

1 #Jawaban No. 1
2 with open('1174050teori.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:

```

**Listing 2.10** 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             line_count += 1
2         else:
3             print(f'\t{row[0]} dengan nama {row[1]} kelas {row[2]}.')
4             line_count += 1
5             print(f'Processed {line_count} lines.')

```

**Listing 2.11** 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('1174050 teori.csv')
4     print(df)

```

**Listing 2.12** 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('1174050 teori.csv')
4     print(df)

```

**Listing 2.13** 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('1174050 teori.csv')
4     dt = pandas.DataFrame.from_dict(df)
5     print(dt)

```

**Listing 2.14** 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('1174050 teori.csv', parse_dates=['tanggal lahir'])
4     print(df)

```

**Listing 2.15** 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('1174050teori.csv')
4     df.index = ['Row_1', 'Row_2']
5     print(df)

```

**Listing 2.16** 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__('1174050csv')
2
3 lib.bukaModeDictCsv()
4
5 lib.tulisCsv()

```

**Listing 2.17** Membuat dan mebaca file CSV menggunakan library 1174006pandas.

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

```

1 lib = __import__('1174050pandas')
2
3 lib.bukaModeListPandas()
4 lib.bukaModeDictPandas()
5
6 lib.tulisCsvPandas()

```

**Listing 2.18** Membuat dan mmebaca file CSV menggunakan library 1174006pandas.

## 2.5.2 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 # -*- coding: utf-8 -*-
2 """
3 Created on Tue Mar 26 19:29:31 2019
4
5 @author:    TOSHIBA
6 """
7
8 def ErrorGais():
9     import csv
10    try:
11        with open('1174050csv.py', newline='') as csvfile:
12            reader = csv.DictReader(csvfile)
13            for row in reader:
14                print(row['npm'], row['nama'], row['key'])
15    except KeyError:
16        print("ERRROOOOORRRRRRR")
17
18 ErrorGais()

```

## 2.6 Ichsan Hizman Hardy 1174034

### 2.6.1 Praktek

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

```

1 #Jawaban No. 1
2 with open('1174034 teori.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'Kolom nya adalah {", ".join(row)}')
8             line_count += 1
9         else:
10            print(f'\t{row[0]} kerja di {row[1]} lahir pada
11 bulan {row[2]}.')
12            line_count += 1

```

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

```

1
2 #Jawaban No. 2
3 def bukaModeDictCsv():
4     with open('1174034 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'])
8
9 def tulisCsv():
10    with open('teori6.csv', mode='w') as csv_file:
11        fieldnames = ['npm', 'nama', 'kelas', 'tanggal lahir']
12        writer = csv.DictWriter(csv_file, fieldnames=fieldnames)

```

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

```

1
2 import pandas
3
4 #Jawaban No. 3

```

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

```

1 df = pandas.read_csv('1174034 teori.csv')
2 print(df)
3
4 #Jawaban No. 4

```

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

```

1 df = pandas.read_csv('1174034teori.csv')
2 dt = pandas.DataFrame.from_dict(df)
3 print(dt)

```

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

```

1 #Jawaban No. 5
2 def ubahFormatTanggal():
3     df = pandas.read_csv('1174034teori.csv', parse_dates=['
4         tanggal lahir'])

```

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

```

1
2 #Jawaban No. 6
3 def ubahIndexKolom():
4     df = pandas.read_csv('1174034teori.csv')
5     df.index = ['Row_1', 'Row_2']

```

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

```

1 lib = __import__('1174034csv')

```

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

```

1
2 lib.tulisCsv()

```

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

```

1
2 lib.tulisCsvPandas()

```

## 2.6.2 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 # -*- coding: utf-8 -*-
2 """
3 Created on Tue Mar 26 19:51:35 2019
4
5 @author: Ichsan Hizman
6 """
7
8 a = 1
9 b = 10
10 c = "7"

```

```
11
12 try :
13     d = a + c
14 except TypeError:
15     print("type datanya berbedhaaa mana bisa bersatu")
```

## BAB 3

---

# LIBRARY CSV DAN PANDAS

---

### 3.1 Irvan Rizkiansyah/1174043

#### 3.1.1 Soal 1

- Device Manager pada windows berfungsi untuk menampilkan segala macam perangkat hardware yang ter-inisialisasi oleh windows itu sendiri, dan juga berguna dalam mengelola perangkat hardware yang terpasang dan terdeteksi oleh windows.
- Direktori /dev pada linux pun berguna seperti device manager pada windows, dimana pada direktori /dev ini tersimpan konfigurasi perangkat device ataupun hardware yang terdeteksi oleh sistem.

#### 3.1.2 Soal 2

Langkah - langkah instalasi driver arduino :

1. Download terlebih dahulu driver arduino yang di sediakan di website arduino

2. Kemudian hubungkan arduino ke PC, dan PC akan mendeteksi adanya perangkat yang terhubung namun tidak terbaca perangkat tersebut perangkat apa.
3. Kemudian buka Device Manager
4. Lalu pilih dan klik kanan pada perangkat arduino yg terdeteksi
5. Kemudian pilih install driver atau update driver
6. Nanti akan muncul pilihan mencari driver secara online atau mencari driver offline, pilih yang offline
7. Cari file installer driver arduino yang sudah di download sebelumnya, kemudian pilih file installer driver arduino tersebut. dan akan langsung memulai penginstalan
8. Tunggu hingga selesai dan pencet OK, maka driver arduino sudah ter-install

### 3.1.3 Soal 3

Di perlukan melakukan instalasi aplikasi arduino IDE untuk melihat dan mengecek baud rate dan port arduino kita yang terpasang, dengan memilih menu Tools kemudian pilih Serial monitor untuk melihat baud rate dan port dimana arduino yang terpasang.

### 3.1.4 Soal 4

Pada tahun 2002 PySerial pertama kalinya di luncurkan, dimana PySerial ini digunakan untuk berkomunikasi dengan mikrokontroller. Kemudian Library PySerial dikembangkan hingga sekarang.

### 3.1.5 Soal 5

Fungsi - fungsi yang ada pada PySerial :

- open ()

Berfungsi untuk membuka dan memberikan akses port serial yang terhubung

- close ()

Berfungsi untuk memberhentikan akses port serial yang terhubung

- flush ()

Berfungsi untuk menghapus seluruh data yang ditampilkan, dan masih banyak yang lainnya.

### 3.1.6 Soal 6

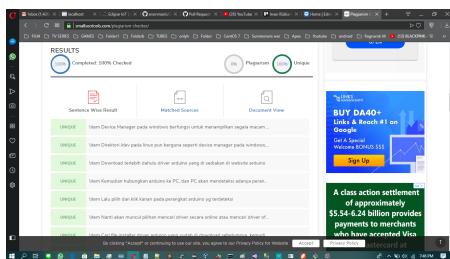
Dengan perulangan kita dapat melihat perkembangan dari mikrokontroller yang sedang berjalan dari awal hingga akhir, apakah ada hal yang aneh, dengan perulangan semua data terbaca. Akan tetapi ada saat dimana perulangan tidak dibutuhkan, misalnya seperti kita hanya membutuhkan data hasil mikrokontroller yang berjalan pada saat tertentu saja, maka tanpa perulangan pun berguna.

### 3.1.7 Soal 7

Dalam membuat fungsi pada PySerial kita hanya perlua menginisialisasikan pembuatan fungsi seperti pada biasanya seperti :

```
def nama_fungsi()
```

### 3.1.8 Plagiarisme



Gambar 3.1 Plagiarisme

## 3.2 Hagan Rowlenstino/1174040

### 3.2.1 Soal 1

- Device Manager : Seperti namanya sendiri, device manager berfungsi untuk menampilkan dan mengelola semua hardware yang terinstall ataupun dapat di instalasi ke dalam windows.
- folder /dev : Di dalam sistem operasi Linux, perangkat yang tehubung akan dianggap sebagai file. di dalam folder /dev inilah file - file tersebut berada.

### 3.2.2 Soal 2

Langkah - langkah instalasi driver arduino :

1. download file driver arduino terlebih dahulu dan masukkan ke dalam directory yang diinginkan

2. hubungkan arduino uno anda ke pc anda dengan kabel USB yang tersedia
3. lalu windows akan memunculkan pop up yang memberitahu bahwa ingin menginstall dirver, tapi nanti tidak akan menemukan drivernya
4. buka Device Manager
5. cari unknown device di dalam Device Manager di dalam tab other device
6. klik kanan pada unknown device tersebut lalu pilih update driver software
7. pilih browse my computer for driver software lalu masukkan directory dimana anda menyimpan driver arduino yang telah anda download tadi
8. setelah itu klik install dan tunggu hingga proses selesai
9. arduino pun sudah terbaca di pc anda

### 3.2.3 Soal 3

Untuk melihat atau membaca baudrate dan port kita hanya perlu menginstall Arduino IDE, setelah itu buka menu serial monitor yang berada di tab tools. Dari sana akan terlihat baik baudrate dan port yang sedang digunakan oleh arduin anda.

### 3.2.4 Soal 4

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

### 3.2.5 Soal 5

- `stop()`  
: untuk menghentikan pembacaan program
- `serial.to_bytes(sequence)`  
: berfungsi untuk mengubah sequence ke dalam bytes agar dapat dikirim ke dalam arduino.
- `close()`  
: untuk menutup port dan menghentikan pembacaan program

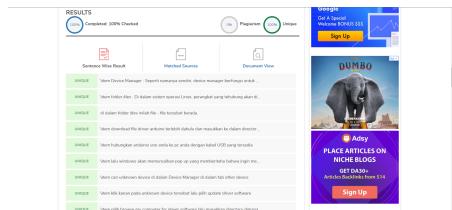
### 3.2.6 Soal 6

Dengan menggunakan pengulangan kita dapat mengambil data berkali - kali tanpa harus mengeksekusi file python tersebut berulang - ulang. Tanpa perulangan juga penting karena dapat digunakan di saat tertentu seperti jika ingin mengukur suhu ruangan yang hanya dilakukan pada saat tertentu tidak terus menerus.

### 3.2.7 Soal 7

Untuk membuat fungsi yang menggunakan pyserial kita hanya perlu untuk menginisialisasi pembubatan funsi dengan menggunakan def namafungsi() : lalu masukkan pyserial tersebut dengan indentasi. atau cukup dengan menggunakan fungsi while loop degan menggunakan while true:

### 3.2.8 Cek Plagiarisme



**Gambar 3.2 Cek Plagiarisme**

cek plagiarism hagan

## 3.3 Luthfi Muhammad Nabil/1174035

### 3.3.1 Soal 1

Apa itu fungsi device manager di windows dan /dev di linux :

- Device Manager merupakan panel kontrol pada sistem operasi windows. Pengguna dapat mengontrol dan melihat perangkat yang telah terhubung dengan komputer dengan device manager. Untuk setiap perangkat, pengguna dapat :
  - Memperbolehkan perangkat untuk beroperasi
  - Menginformasikan sistem operasi untuk melakukan aksi pada perangkat yang tidak berfungsi
  - Menginstall driver untuk perangkat yang terhubung
  - Melihat informasi dari perangkat
- /dev merupakan lokasi dari file untuk perangkat. /dev berfungsi untuk menampung data - data sebuah perangkat yang terhubung pada komputer. Perangkat yang dapat terhubung diantaranya perangkat penyimpanan data dan perangkat pengiriman data.

### 3.3.2 Soal 2

Jelaskan langkah - langkah instalasi driver dari arduino :

Untuk instalasi driver, biasanya akan langsung terinstall jika sudah menginstall arduino IDE Seperti contoh meminta instalasi untuk arduino pada gambar 3.3. Jika



**Gambar 3.3** Pop Up saat instalasi Arduino IDE

memang tidak ditemukan, maka ikuti langkah berikut : Yang dibutuhkan :

- Arduino Driver
- Arduino Jenis Apapun

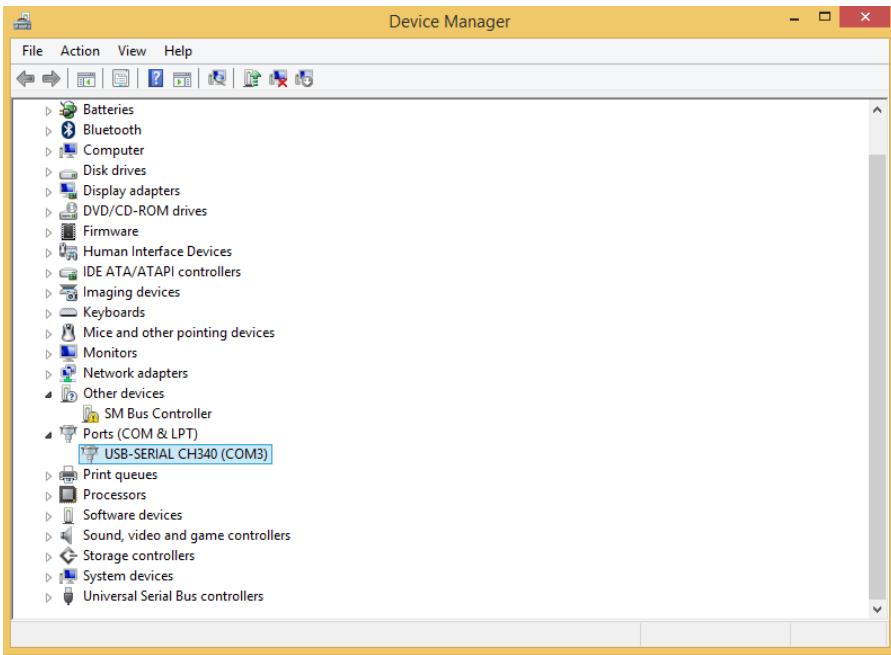
Cara Instalasi :

1. Hubungkan arduino ke komputer (untuk arduino uno dapat menggunakan kabel type B)



**Gambar 3.4** Hubungkan arduino dengan komputer

2. Lalu setelah muncul popup instalasi driver akan terinstal secara otomatis.
3. Jika instalasi otomatis gagal, buka device manager.



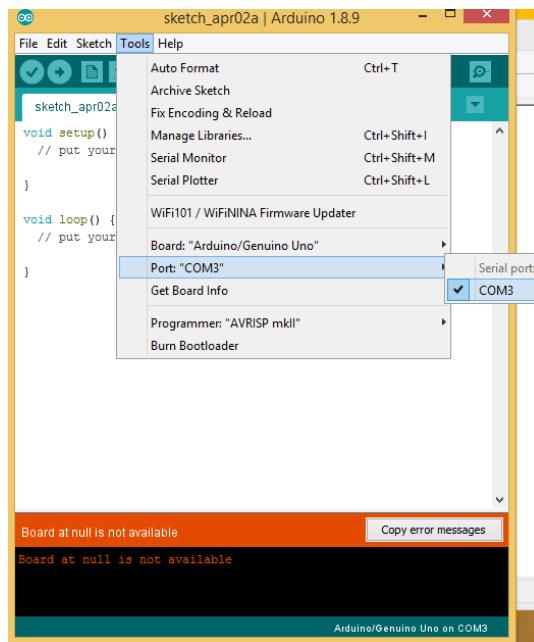
**Gambar 3.5** Tampilan Device Manager

4. Setelah device manager terbuka, cari perangkat lainnya/tidak diketahui (other device).
5. Klik kanan perangkat yang tidak diketahui lalu pilih update driver.
6. Pilih Cari driver (Browse my computer for driver software).
7. Pilih lokasi driver ke driver yang telah didownload lalu klik next/ok.
8. Setelah muncul pop up instalasi, tekan install.
9. Instalasi Sukses

### 3.3.3 Soal 3

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

- Port : Untuk membaca port, dapat membuka file/lokasi berikut :

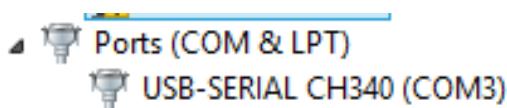


**Gambar 3.6** Port dari Arduino IDE

Arduino IDE :

1. Buka Arduino IDE
2. Lalu masuk ke Tools->Ports
3. Akan muncul port serial untuk arduino seperti pada gambar 3.6

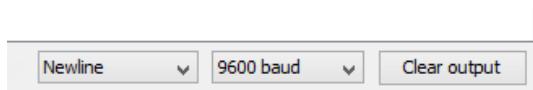
Device Manager :



**Gambar 3.7** Port dari Device Manager

1. Buka Device Manager pada windows
2. Cari Perangkat dengan nama Arduino
3. Lalu lihat port berapa yang dipakai oleh Arduino Tersebut seperti pada gambar 3.7

- Baudrate : Untuk melihat baudrate, dapat buka serial monitor dan cek baudrate dan tampilan fix pada tampilan serial monitor. Untuk setting default biasanya diantara 9600 dan 115200.



**Gambar 3.8** Baud Rate

### 3.3.4 Soal 4

Sejarah PySerial :

PySerial dibuat secara public pertama kali pada tahun 2002 dengan rilisan versi pertama 1.0. Pada saat itu PySerial sudah dapat membaca line serial dan menghapus semua isi dari serial tetapi belum dapat mengkonversi isi serial menjadi tipe data lain. PySerial juga sudah dapat melakukan Exception untuk proses yang terdapat error. Lalu pada tahun 2003 dirilis versi yang dapat mengkonversi data ke bilangan bulat, support terhadap python versi 2.2+ dan getter setter. Mulai pada tahun 2008, diberlakukan fitur iterasi untuk dapat mengambil seluruh data yang muncul pada serial yang diusulkan oleh Bernhard Bender. Kemudian pada tahun 2011, mulai dapat melihat daftar port yang terhubung dengan serial. Sampai akhirnya telah muncul versi 3.4 pada tahun 2017 yang sudah memperbaiki bug - bug yang ada.

### 3.3.5 Soal 5

Jelaskan fungsi - fungsi yang ada di pyserial.

- `open()`

Fungsi untuk membuka port serial yang terhubung

- `close()`

Fungsi untuk menutup port serial yang terhubung

- `read(size=1)`

Fungsi untuk menentukan ukuran serial yang dapat dibaca

- `read\_until(expected=LF, size=None)`

Fungsi untuk membaca serial sampai sequence yang ditentukan sudah dapat.

- `write(data)`

Fungsi untuk menulis data ke perangkat yang terhubung.

- `flush()`

Fungsi untuk menghapus seluruh data yang ditampilkan di serial.

- `readline(size=-1)`

Untuk membaca setiap line pada tampilan yang ada pada serial

- `writelines(lines)`

Sama halnya dengan `write` yaitu untuk menulis data ke perangkat yang terhubung.

### 3.3.6 Soal 6

Jelaskan mengapa butuh perulangan dalam membaca serial :

Karena perulangan digunakan untuk membaca seluruh data pada serial yang ada setiap baris. Perulangan digunakan agar data dapat muncul secara terus menerus atau realtime. Dalam konsep berjalannya sebuah microcontroller, proses yang ada akan dijalankan secara terus menerus sampai proses meminta atau processor meminta untuk memberhentikan proses tersebut sehingga jika tidak memakai perulangan, maka data yang diambil hanyalah data dimana sintaks untuk meminta data itu dipanggil.

### 3.3.7 Soal 7

Jelaskan cara membuat fungsi yang menggunakan pyserial :

Untuk membuat fungsi yang menggunakan pyserial, cukup dengan menuliskan fungsi dari pyserial dan menggunakannya dalam fungsi yang dibuat. Seperti contoh fungsi yang menggunakan PySerial :

```
1 #Fungsi Reader
2 def ReadSerial(line):
3     ser.open()
4     a=ser.readline(size=-1)
5     ser.writelines(line)
6     return a
```

## 3.4 Faisal Najib Abdullah

### 3.4.1 Teori

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

(a) Device Manager : Seperti namanya sendiri, device manager berfungsi untuk menampilkan dan mengelola semua hardware yang terinstall ataupun dapat di instalasi ke dalam windows.

(b) folder /dev : Di dalam sistem operasi Linux, perangkat yang terhubung akan dianggap sebagai file. di dalam folder /dev inilah file - file tersebut berada.

2. Jelaskan langkah-langkah instalasi driver dari arduino

Langkah - langkah instalasi driver arduino :

- (a) Pertama download file driver arduino terlebih dahulu dan masukkan ke dalam directory yang diinginkan.
  - (b) Hubungkan arduino uno anda ke pc anda dengan kabel USB yang tersedia.
  - (c) Kemudian windows akan memunculkan pop up yang memberitahu bahwa ingin menginstall dirver, tapi nanti tidak akan menemukan drivernya.
  - (d) Buka Device Manager dan cari unknown device di dalam Device Manager di dalam tab other device.
  - (e) Klik kanan pada unknown device tersebut lalu pilih update driver software
  - (f) Pilih browse my computer for driver software lalu masukkan directory di mana anda menyimpan driver arduino yang telah anda download tadi
  - (g) Setelah itu klik install dan tunggu hingga proses selesai
  - (h) Arduino pun sudah terbaca di pc anda
3. Jelaskan bagaimana cara membaca baudrate dan port dari komputer yang sudah terinstall driver
- Untuk melihat atau membaca baudrate dan port kita hanya perlu menginstall Arduino IDE, setelah itu buka menu serial monitor yang berada di tab tools. Dari sana akan terlihat baik baudrate dan port yang sedang digunakan oleh arduin anda.
4. Jelaskan sejarah library pyserial
- PySerial merupakan sebuah library yang digunakan untuk komunikasi ke port serial terutama untuk mikrokontroller. PySerial pertama kali diluncurkan pada tahun 2002 yang makin berkembang dalam setiap versinya hingga tahun 2017 lalu.
5. Jelaskan fungsi-fungsi apa saja yang dipakai dari library pyserial

- `serial.to_bytes(sequence)`  
: berfungsi untuk mengubah sequence ke dalam bytes agar dapat dikirim ke dalam arduino.
- `stop()`  
: untuk menghentikan pembacaan program
- `open()`  
: Fungsi untuk membuka port serial yang terhubung
- `close()`  
: Fungsi untuk menutup port serial yang terhubung
- `read(size=1)`  
: Fungsi untuk menentukan ukuran serial yang dapat dibaca

- `close()`

: untuk menutup port dan menghentikan pembacaan program

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

Dengan menggunakan pengulangan kita dapat mengambil data berkali - kali tanpa harus mengeksekusi file python tersebut berulang - ulang. Tanpa perulangan juga penting karena dapat digunakan di saat saat tertentu seperti jika ingin mengukur suhu ruangan yang hanya dilakukan pada saat saat tertentu tidak terus menerus.

7. Jelaskan bagaimana cara membuat fungsi yang menggunakan pyserial

Untuk membuat fungsi yang menggunakan pyserial kita hanya perlu untuk menginisialisasi pembubatan funsi dengan menggunakan def namafungsi(): lalu masukkan pyserial tersebut dengan indentasi. atau cukup dengan menggunakan fungsi while loop dengan menggunakan while true:

### 3.5 Ichsan Hizman Hardy

## Teori

### 3.5.1 Soal No. 1

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

Fungsi device manager antara lain :

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 /dev berisi file device, baik device blok maupun device karakter. Di dalamnya setidaknya ada file biner yang bernama MAKEDEV untuk membuat device secara manual.

### 3.5.2 Soal No. 2

Jelaskan langkah-langkah instalasi driver dari arduino!

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

1. Hubungkan sistem minimun Arduino Uno ke komputer dengan kabel USB type B (kabel Printer).
2. Lalu pada bagian kanan didesktop PC anda, akan muncul popup “Installing device driver software”.
3. Sistem operasi Windows tidak menyediakan driver untuk Arduino Uno.
4. Buka Device Manager, caranya pada bagian Search Program and Files lalu ketikkan “device manager” (tanpa tanda petik). Kemudian bagian Control Panel akan muncul halaman Device Manager, selanjutnya klik untuk menjalankan.
5. Cari yang bernama Unknown device yang berada pada bagian Other device, biasanya ada tanda seru berwarna kuning, itu disebabkan karena penginstallan tidak berjalan dengan sempurna.
6. Klik kanan pada “Unknown device” kemudian pilih Update Driver Software.
7. Pilih Browse my computer for driver software.
8. Arahkan lokasi folder ke folder ..arduino-1.0.5 drivers. Pastikan check-box lalu centang include subfolders. Klik Next untuk melanjutkan instalasi driver.
9. Kemudian lanjutkan dengan mengklik Install pada tampilan Windows Security.
10. Jika instalasi driver 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.

### **3.5.3 Soal No. 3**

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

#### **Membaca Port dari Komputer**

1. Hubungkan modul TX-RX serial dengan komputer melalui serial port menggunakan DB9 cable extension.
2. Buka Hyper Terminal dengan menekan start kemudian All programs lalu Accessories kemudian Communications lalu Hyper Terminal.
3. Ketik nama untuk Connection Description, misal coba, kemudian tekan OK.

4. Pada Connect to, pilihlah COM port yang dipakai di Connect using, kemudian tekan OK.
5. Masukkan nilai-nilai port settingnya, sesuai dengan DCE-nya. Kemudian tekan OK.

### 3.5.4 Soal No. 4

Jelaskan sejarah library pyserial!

PySerial adalah library/modul Python siap-pakai dan gratis yang dibuat untuk memudahkan kita dalam membuat program komunikasi data serial RS232 dalam bahasa Python. Jika modul USB-2REL dapat kita kontrol dengan mudah menggunakan Python dan PyUSB (lihat pembahasannya di sini dan di sini), maka modul SER-2REL juga dapat kita kontrol dengan mudah menggunakan Python dengan bantuan modul PySerial.

### 3.5.5 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.

### 3.5.6 Soal No. 6

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

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

### 3.5.7 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.

## 3.6 Dika Sukma Pradana 1174050

### 3.6.1 Pemahaman Teori

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

Device manager merupakan perangkat lunak untuk menampilkan seluruh perangkat keras yang di-inisialisasi atau dikenali oleh sistem operasi Windows. Device Manager membantu dalam mengelola atau me-manage semua perangkat keras yang 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 :

- Menunjukkan status mengenai suatu perangkat keras.
- Menunjukkan informasi detail mengenai suatu perangkat keras.
- Mengelola driver perangkat keras.
- Menonaktifkan dan mengaktifkan perangkat keras.
- Mengidentifikasi konflik antar perangkat keras.
- Memberitahukan terjadinya masalah pada perangkat keras.

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

2. Jelaskan langkah-langkah instalasi driver dari arduino!

- (a) Hubungkan sistem minimun Arduino Uno ke komputer dengan kabel USB type B (kabel Printer).
- (b) Lalu pada bagian kanan didesktop PC anda, akan muncul popup Installing device driver software seperti pada gambar dibawah ini.
- (c) Windows tidak menyediakan driver arduino, jadi kita harus menginstal secara manual.
- (d) Buka Device Manager, klik untuk menjalankan.
- (e) Cari Unknown device pada bagian Other device, bila terdapat tanda seru berarti peroses penginstalan belum berhasil sepenuhnya.
- (f) Klik kanan pada Unknown device lalu pilih Update Driver Software.

- (g) Pilih Browse my computer for driver software.
- (h) Arahkan lokasi folder ke folder ..arduino1.0.5 drivers. Pastikan checkbox lalu centang include subfolders. Klik Next untuk melanjutkan instalasi driver.
- (i) Kemudian lanjutkan dengan mengklik Install pada tampilan Windows Security.
- (j) Jika instalasi driver berhasil maka akan muncul Windows has successfully updated your driver software.
- (k) Perhatikan dan ingat nama COM Arduino Uno, karena nama COM ini yang akan digunakan untuk meng-upload program nantinya.
3. Jelaskan bagaimana cara membaca baudrate dan port dari komputer yang sudah terinstall driver!
- Membaca Baudrate dari Komputer
    - (a) Pertama buka Start. Cari Device Manager, lalu klik.
    - (b) Kemudian pilih Ports (COM & LPT).
    - (c) Klik dua kali pada COM yang terhubung.
    - (d) Pilih tab Port Settings, lalu lihat di Bit per second.
  - Membaca Port dari Komputer
    - (a) Pertama buka Start. Cari Device Manager, lalu klik.
    - (b) Kemudian pilih Ports (COM & LPT).
    - (c) Port dari Arduino telah terbaca oleh PC.
4. Jelaskan sejarah library pyserial!

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

5. Jelaskan fungsi-fungsi apa saja yang dipakai dari library pyserial! Fungsi-fungsi yang dipakai dari library PySerial, yaitu:
- `serial`  
: fungsi ini untuk membuka port serial.
  - `write(data)`  
: fungsi ini menulis data lewat port serial.
  - `readline()`

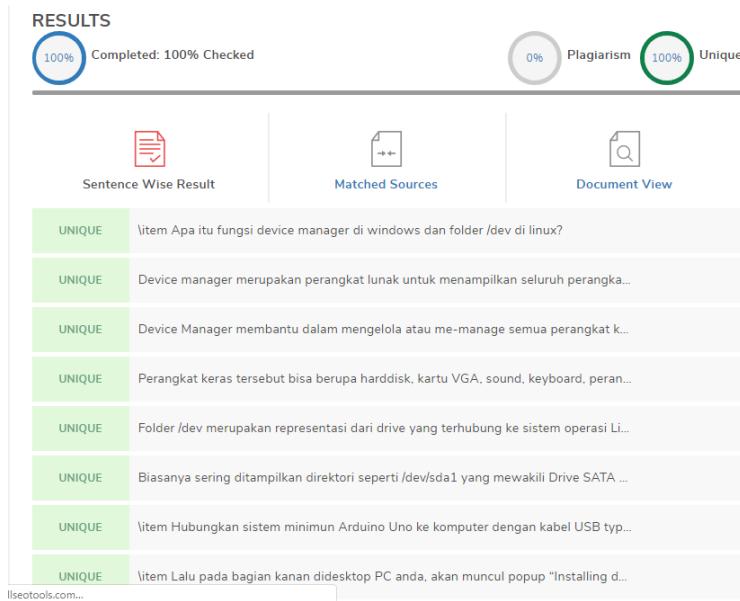
- : fungsi ini membaca sebuah string dari port serial.
  - `read(size)`  
fungsi ini untuk membaca jumlah byte dari port serial.
  - `close()`  
: fungsi ini untuk menutup port serial.
6. Jelaskan kenapa butuh perulangan dan tidak butuh perulangan dalam membaca serial!
- Perulangan dalam bahasa pemrograman 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). Perulangan for biasanya digunakan untuk mengulangi kode yang sudah dipastikan banyak perulangannya. Sedangkan while untuk perulangan yang ditentukan dan belum pasti banyaknya. 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.
7. Jelaskan bagaimana cara membuat fungsi yang menggunakan pyserial!

```
import serial

def baca():
    ser = serial.Serial("COM4", 9600)
    baca = ser.readline()
    print(baca)

baca()
```

8. Plagiarism



**Gambar 3.9** Hasil Scan Plagiarism

## BAB 4

---

# PRAKTEK LIBRARY CSV DAN PANDAS

---

### 4.1 Faisal Najib Abdullah 1174042

#### 4.1.1 Keterampilan Pemrograman

1. Buatlah fungsi file terpisah dengan nama NPM realtime.py untuk mendapatkan data langsung dari arduino

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

**Listing 4.1** Fungsi untuk mendapatkan data dari Arduino.

2. Buatlah fungsi file terpisah dengan nama NPM save.py untuk mendapatkan data langsung dari arduino dengan looping

```
1 import serial  
2
```

```

3 def getDataLoop():
4     ser = serial.Serial('COM3',9600)
5     while (1):
6         print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
7
8 getDataLoop()

```

**Listing 4.2** Fungsi untuk mendapatkan data langsung dari Arduino dengan looping.

- Buatlah fungsi file terpisah dengan nama NPM realtime.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('COM3',9600)
6     with open('speedometer.csv', mode='w') as csv_file:
7         fieldnames = ['speed']
8         writer = csv.DictWriter(csv_file, fieldnames=fieldnames)
9
10        writer.writeheader()
11        while (1):
12            data = ser.readline().decode("utf-8").strip('\n').
13            strip('\r')
14            writer.writerow({'speed': data})
15
16 writeCsv()

```

**Listing 4.3** Fungsi untuk mendapatkan data dari Arduino dan langsung ditulis kedalam file CSV.

- Buatlah fungsi file terpisah dengan nama NPM csv.py untuk membaca file csv hasil arduino dan mengembalikan ke fungsi

```

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

```

**Listing 4.4** Fungsi untuk membaca file CSV hasil Arduino dan mengembalikan fungsi.

#### 4.1.2 Penanganan Error

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

Peringatan error di praktek kelima ini, yaitu:

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

Fungsi yang menggunakan try except untuk menanggulangi error.

```

1 import serial
2
3 def tryExceptError():
4     try:
5         ser = serial.Serial('COM3', 9600)
6         print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
7     except SyntaxError:
8         print("Kesalahan penulisan syntax")
9     except NameError:
10        print("Variable tersebut tidak ada")
11    except TypeError:
12        print("Tipe data salah")
13    except:
14        print("Terjadi sebuah kesalahan")
15
16 tryExceptError()

```

**Listing 4.5** Fungsi untuk menanggulangi error menggunakan Try Except.

## 4.2 Hagan Rowlenstino/1174040

### 4.2.1 Soal 1

membuat fungsi untuk mendapatkan data langsung dari arduino. disini saya menggunakan arduino dengan servo. disini saya akan menginputkan data berupa angka untuk menambahkan berapa lama servo kembali dari letak yang ditentukan ke letak awal.

```

1 import serial
2 import time
3 import csv
4

```

```

5 def Nomor1() :
6     arduino = serial.Serial('COM7', 9600)
7     a = input()
8     conv = str.encode(str(a))
9     time.sleep(2)
10    print(conv)
11    arduino.write(conv)
12    reachedPos = str(arduino.readline())
13    print(reachedPos)

```

#### 4.2.2 Soal 2

membuat fungsi untuk mendapatkan data langsung dari arduino dengan looping. disini saya menggunakan arduino dengan servo. disini saya akan menginputkan data berupa angka untuk menambahkan berapa lama servo kembali dari letak yang ditentukan ke letak awal.

```

1 import serial
2 import time
3
4 arduino = serial.Serial('COM7', 9600)
5 def Nomor2() :
6     while True:
7         a = input()
8         conv = str.encode(str(a))
9         time.sleep(2)
10        print(conv)
11        arduino.write(conv)
12        reachedPos = str(arduino.readline())
13        print(reachedPos)
14
15 x = Nomor2()

```

#### 4.2.3 Soal 3

membuat fungsi untuk mendapatkan data langsung dari arduino lalu me write data tersebut ke dalam file csv. disini saya menggunakan arduino dengan servo. disini saya akan menginputkan data berupa angka untuk menambahkan berapa lama servo kembali dari letak yang ditentukan ke letak awal. Nilai inputan tersebutlah yang saya write ke dalam file 1174040\_realtime2.csv

```

1 def Nomor3() :
2     arduino = serial.Serial('COM7', 9600)
3     a = input()
4     conv = str.encode(str(a))
5     time.sleep(2)
6     print(conv)
7     arduino.write(conv)
8     reachedPos = str(arduino.readline())
9     print(reachedPos)
10    with open('1174040_realtime2.csv', mode='w') as file_tulis:
11        employee_writer = csv.writer(file_tulis, delimiter=',', quotechar
12        = '"', quoting=csv.QUOTE_MINIMAL)

```

```
12 employee_writer.writerow([a])
```

#### 4.2.4 Soal 4

fungsi untuk membaca csv yang brisikan data dari arduino tadi :

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Tue Mar 26 12:19:17 2019
4
5 @author: Haganz
6 """
7
8 import csv
9
10 def Readcsv(file) :
11     with open(file) as file_csv:
12         csv_reader = csv.reader(file_csv , delimiter=',')
13         for row in csv_reader:
14             b = print(row)
15     return(b)
16 x = Readcsv(input())
```

#### 4.2.5 Penanganan Error

Disini saya mendapatkan Error yaitu FileNotFoundError dimana ada kesalahan saat menginputkan nama file yang akan digunakan, untuk penanganannya yaitu dengan cara menginputkan nama file yang benar

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Tue Mar 26 12:19:17 2019
4
5 @author: Haganz
6 """
7
8 import csv
9
10 def Readcsv(file) :
11     with open(file) as file_csv:
12         csv_reader = csv.reader(file_csv , delimiter=',')
13         for row in csv_reader:
14             b = print(row)
15     return(b)
16 try :
17     x = Readcsv(input())
18 except FileNotFoundError:
19     print("Nama file nya salah , yang benar adalah 1174040_realtime2.csv")
```

## 4.3 Irvan Rizkiansyah/1174043

### 4.3.1 Nomor 1

```

1 def No1() :
2     arduino = serial.Serial('COM6', 9600)
3     x = input()
4     c = str.encode(str(x))
5     time.sleep(2)
6     print(c)
7     arduino.write(c)
8     s = str(arduino.readline())
9     print(s)

```

### 4.3.2 Nomor 2

```

1 import time
2 import serial
3
4 arduino = serial.Serial('COM6', 9600)
5 def No2() :
6     while True:
7         x = input()
8         c = str.encode(str(x))
9         time.sleep(2)
10        print(c)
11        arduino.write(c)
12        s = str(arduino.readline())
13        print(s)
14
15 test = No2()

```

### 4.3.3 Nomor 3

```

1 def No3() :
2     arduino = serial.Serial('COM6', 9600)
3     x = input()
4     c = str.encode(str(x))
5     time.sleep(2)
6     print(c)
7     arduino.write(c)
8     s = str(arduino.readline())
9     print(s)
10    with open('csvFile_1174043.csv', mode='w') as file_tulis:
11        employee_writer = csv.writer(file_tulis, delimiter=',', quotechar
12        ='"', quoting=csv.QUOTE_MINIMAL)
13        employee_writer.writerow([x])

```

### 4.3.4 Nomor 4

```

1 import csv
2
3 def Bacacsv(file) :
4     with open(file) as csv_file:

```

```

5     baca = csv.reader(csv_file , delimiter=',')
6     for row in baca:
7         b = print(row)
8     return(b)
9 x = Bacacsv(input())

```

### 4.3.5 Penanganan Error

```

1 import csv
2
3 def Bacacsv(file) :
4     with open(file) as csv_file:
5         baca = csv.reader(csv_file , delimiter=',')
6         for row in baca:
7             x = print(row)
8     return(x)
9 try :
10     y = Bacacsv(input())
11 except FileNotFoundError:
12     print("Nama File CSV SALAH !")

```

## 4.4 Luthfi Muhammad Nabil/1174035

### 4.4.1 Soal 1

Membuat fungsi untuk mengambil data langsung dari arduino.

```

1 def BacaSerial():
2     import serial
3     ser = serial.Serial('COM4' , 9600, timeout=2)
4     x = ser.readline()
5     return x

```

### 4.4.2 Soal 2

Membuat fungsi untuk mengambil data langsung dari arduino dengan looping.

```

1 def BacaSerialLoop():
2     import serial
3     ser = serial.Serial('COM4' , 9600, timeout=2)
4     lis = []
5     while True:
6         x = ser.readline()
7         if x:
8             print(x)
9             lis.append(x)
10        else:
11            break
12    return lis

```

#### 4.4.3 Soal 3

Membuat fungsi untuk mengambil data langsung dari arduino lalu me write data tersebut ke dalam file csv.

```

1 #Simpan Dengan Loop
2 def BacaSimpan():
3     import csv
4     with open('chap5_1174035_csv_CSVWrite.csv', mode='w') as mhsFile:
5         mhs = csv.writer(mhsFile, delimiter=',', quotechar='''', quoting=
6             csv.QUOTE_MINIMAL)
7         lis = []
8         lis.append(str(BacaSerial()))
9         import serial
10        ser = serial.Serial('COM4', 9600, timeout=2)
11        x = ser.readline()
12        mhs.writerow(lis)

13 #SimpanTanpaLoop
14 def BacaLoop():
15     import csv
16     with open('chap5_1174035_csv_CSVWrite.csv', mode='w') as mhsFile:
17         mhs = csv.writer(mhsFile, delimiter=',', quotechar='''', quoting=
18             csv.QUOTE_MINIMAL)
19         import serial
20         ser = serial.Serial('COM4', 9600, timeout=2)
21         lis = []
22         while True:
23             x = ser.readline()
24             if x:
25                 print(x)
26                 lis.append(x)
27             else:
28                 break
29             for x in lis:
30                 exlis = []
31                 exlis.append(x)
32                 mhs.writerow(exlis)
```

#### 4.4.4 Soal 4

Fungsi untuk membaca csv yang berisikan data dari isi serial arduino yang dibaca sebelumnya :

```

1 def BacaSerialLoop():
2     import serial
3     ser = serial.Serial('COM4', 9600, timeout=2)
4     lis = []
5     while True:
6         x = ser.readline()
7         if x:
8             print(x)
9             lis.append(x)
10            else:
11                break
```

```
12     return lis
```

#### 4.4.5 Penanganan Error

Error yang didapat : Serial Exception Definisi : Serial Exception terjadi saat koneksi antara serial dan komputer terjadi masalah atau tidak dapat berkomunikasi. Untuk menangani error tersebut dapat menggunakan Try Except.

Pemakaian :

```
1 def BacaSerialLoop():
2     import serial
3     try:
4         ser = serial.Serial('COM4', 9600, timeout=2)
5         lis = []
6         while True:
7             x = ser.readline()
8             if x:
9                 print(x)
10                lis.append(x)
11            else:
12                break
13        return lis
14    except serial.SerialException:
15        print("Arduino Belum Terhubung")
```

Hasil (Saat Error) dapat dilihat pada gambar 4.10

### 4.5 Dika Sukma Pradana 1174050

#### 4.5.1 Keterampilan Pemrograman

- Buatlah fungsi file terpisah dengan nama NPM realtime.py untuk mendapatkan data langsung dari arduino

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

**Listing 4.6** Fungsi untuk mendapatkan data dari Arduino.

- Buatlah fungsi file terpisah dengan nama NPM save.py untuk mendapatkan data langsung dari arduino dengan looping

```
1 import serial
2
3 def getDataLoop():
4     ser = serial.Serial('COM3', 9600)
5     while (1):
```

```

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

```

**Listing 4.7** Fungsi untuk mendapatkan data langsung dari Arduino dengan looping.

- Buatlah fungsi file terpisah dengan nama NPM realtime.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('COM3',9600)
6     with open('gpstrackerr.csv', mode='w') as csv_file:
7         fieldnames = ['Coordinate']
8         writer = csv.DictWriter(csv_file, fieldnames=fieldnames)
9
10        writer.writeheader()
11        while (1):
12            data = ser.readline().decode("utf-8").strip('\n').
13            strip('\r')
14            writer.writerow({'Coordinate': data})
15
15 writeCsv()

```

**Listing 4.8** Fungsi untuk mendapatkan data dari Arduino dan langsung ditulis kedalam file CSV.

- Buatlah fungsi file terpisah dengan nama NPM csv.py untuk membaca file csv hasil arduino dan mengembalikan ke fungsi

```

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

```

**Listing 4.9** Fungsi untuk membaca file CSV hasil Arduino dan mengembalikan fungsi.

#### 4.5.2 Penanganan Error

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

Peringatan error di praktek kelima ini, yaitu:

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

Fungsi yang menggunakan try except untuk menanggulangi error.

```

1 import serial
2
3 def tryExceptError():
4     try :
5         ser = serial.Serial('COM3', 9600)
6         print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
7     except SyntaxError:
8         print("Syntax Error")
9     except NameError:
10        print("Tidak ada variabel tersebut")
11    except TypeError:
12        print("Tipe data salah")
13    except:
14        print("Terjadi sebuah kesalahan")
15
16 tryExceptError()

```

**Listing 4.10** Fungsi untuk menanggulangi error menggunakan Try Except.

## 4.6 Ichsan Hizman Hardy 1174034

### 4.6.1 Keterampilan Pemrograman

1. Buatlah fungsi file terpisah dengan nama NPM realtime.py untuk mendapatkan data langsung dari arduino

```

1 import serial
2
3 def getData():
4     ser = serial.Serial('COM9', 9600)
5     print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
6
7 getData()

```

**Listing 4.11** Fungsi untuk mendapatkan data dari Arduino.

2. Buatlah fungsi file terpisah dengan nama NPM save.py untuk mendapatkan data langsung dari arduino dengan looping

```

1 import serial
2
3 def getDataLoop():
4     ser = serial.Serial('COM9',9600)
5     while (1):
6         print(ser.readline().decode("utf-8").strip('\n').strip('\r'))
7
8 getDataLoop()

```

**Listing 4.12** Fungsi untuk mendapatkan data langsung dari Arduino dengan looping.

3. Buatlah fungsi file terpisah dengan nama NPM realtime.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('COM9',9600)
6     with open('gpstrackerr.csv', mode='w') as csv_file:
7         fieldnames = ['Coordinate']
8         writer = csv.DictWriter(csv_file, fieldnames=fieldnames)
9
10        writer.writeheader()
11        while (1):
12            data = ser.readline().decode("utf-8").strip('\n').
13            strip('\r')
14            writer.writerow({'Coordinate': data})
15
16 writeCsv()

```

**Listing 4.13** Fungsi untuk mendapatkan data dari Arduino dan langsung ditulis kedalam file CSV.

4. Buatlah fungsi file terpisah dengan nama NPM csv.py untuk membaca file csv hasil arduino dan mengembalikan ke fungsi

```

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

```

**Listing 4.14** Fungsi untuk membaca file CSV hasil Arduino dan mengembalikan fungsi.

#### 4.6.2 Penanganan Error

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

Peringatan error di praktek kelima ini, yaitu:

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

Fungsi yang menggunakan try except untuk menanggulangi error.

```
1 import serial
2
3 def tryExceptError():
4     try:
5         ser = serial.Serial('COM9', 9600)
6         print(sre.readline().decode("utf-8").strip('\n').strip('\r'))
7     except SyntaxError:
8         print("Syntax Error")
9     except NameError:
10        print("Tidak ada variabel tersebut")
11    except TypeError:
12        print("Tipe data salah")
13    except:
14        print("Terjadi sebuah kesalahan")
15
16 tryExceptError()
```

**Listing 4.15** Fungsi untuk menanggulangi error menggunakan Try Except.

```
D:\NAJIB\SEMESTER _4_NAJIB\Pemrograman 3\Py4>python 1174042_realtime.py  
sp01,0
```

**Gambar 4.1** Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino.

```
D:\NAJIB\SEMESTER _4_NAJIB\Pemrograman 3\Py4>python 1174042_save.py  
sp01,0  
sp01,1  
sp01,1  
sp01,33  
sp01,22  
sp01,23
```

**Gambar 4.2** Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino dengan looping.

A1	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1 speed														
2														
3 sp01,0														
4														
5 sp01,0														
6														
7 sp01,0														
8														
9 sp01,0														
10														
11 sp01,4														
12														
13 sp01,32														
14														
15 sp01,35														
16														
17 sp01,33														

**Gambar 4.3** Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino dan langsung dituliskan kedalam file CSV.

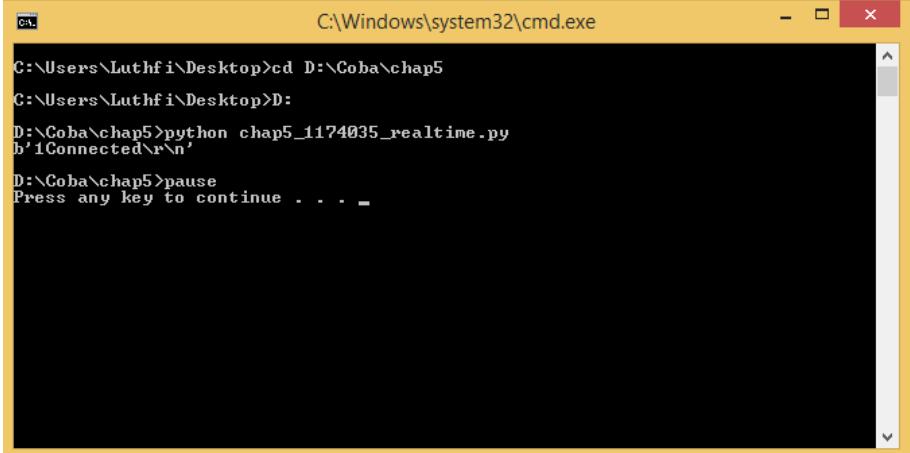
```
C:\Windows\system32\cmd.exe

D:\NAJIB\SEMESTER _4_NAJIB\Pemrograman 3\Py4>python 1174042_csv.py
sp01,0
sp01,0
sp01,0
sp01,0
sp01,4
sp01,32
sp01,35
sp01,33
sp01,0
sp01,0
sp01,0
sp01,4
sp01,31
sp01,31
sp01,32
sp01,32
sp01,31
sp01,32
sp01,32
sp01,21
sp01,21
sp01,32
sp01,21
sp01,32
sp01,32
sp01,33
sp01,34
sp01,35
```

**Gambar 4.4** Hasil dari pembacaan fungsi untuk membaca file csv hasil arduino dan mengembalikan fungsi.

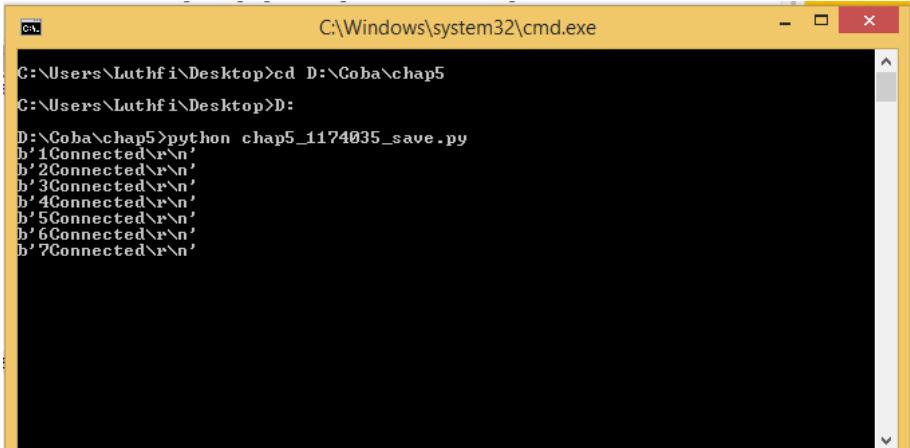
```
D:\NAJIB\SEMESTER _4_NAJIB\Pemrograman 3\Py4>python 1174042.py
Variable tersebut tidak ada
```

**Gambar 4.5** Hasil pembacaan fungsi untuk menanggulangi error menggunakan Try Except.



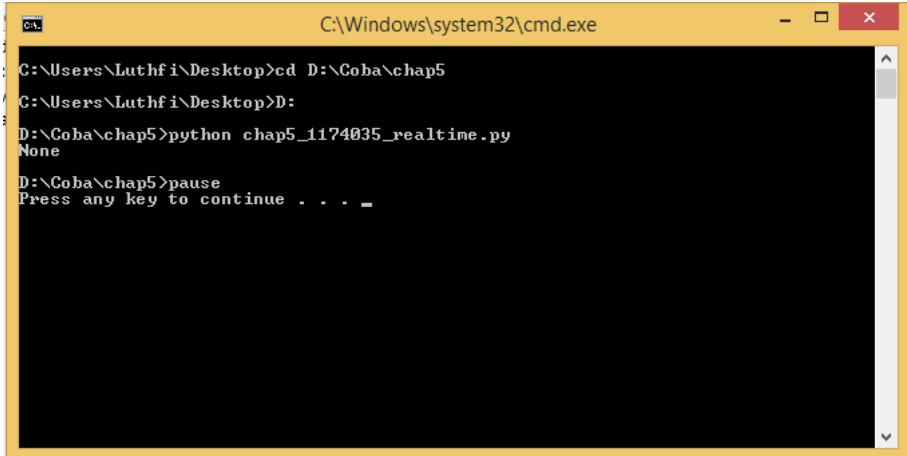
```
C:\Windows\system32\cmd.exe
C:\Users\Luthfi\Desktop>cd D:\Coba\chap5
C:\Users\Luthfi\Desktop>D:
D:\Coba\chap5>python chap5_1174035_realtime.py
b'1Connected\r\n'
D:\Coba\chap5>pause
Press any key to continue . . .
```

**Gambar 4.6** Hasil dari Ambil Data tanpa perulangan



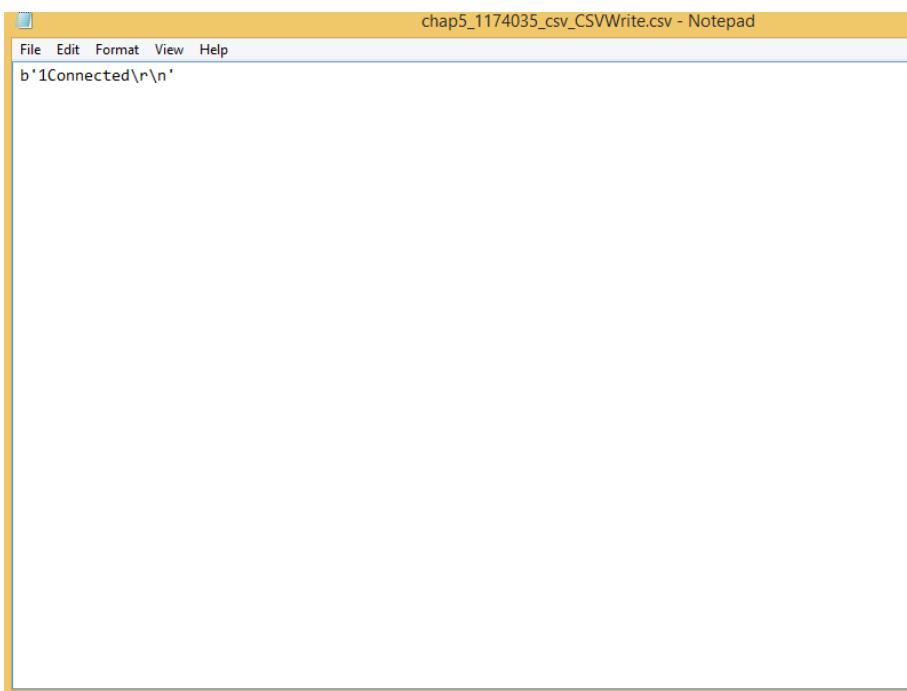
```
C:\Windows\system32\cmd.exe
C:\Users\Luthfi\Desktop>cd D:\Coba\chap5
C:\Users\Luthfi\Desktop>D:
D:\Coba\chap5>python chap5_1174035_save.py
b'1Connected\r\n'
b'2Connected\r\n'
b'3Connected\r\n'
b'4Connected\r\n'
b'5Connected\r\n'
b'6Connected\r\n'
b'7Connected\r\n'
```

**Gambar 4.7** Hasil dari Ambil Data dengan perulangan

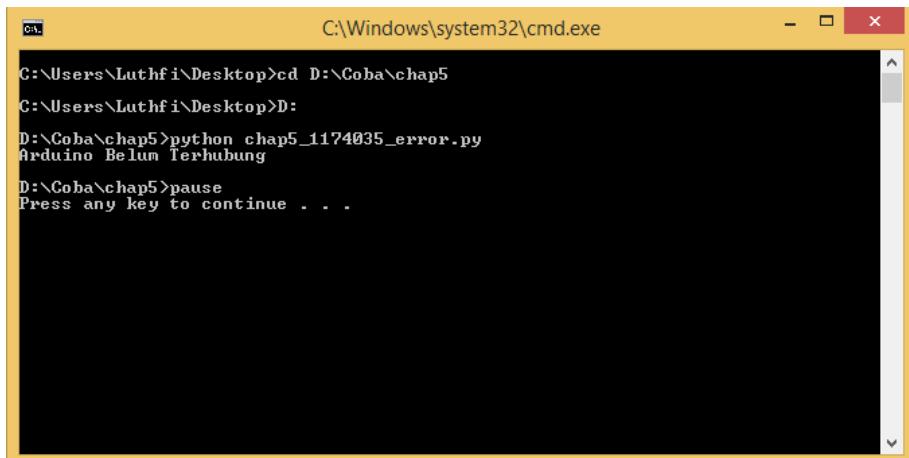


```
C:\Windows\system32\cmd.exe
C:\Users\Luthfi\Desktop>cd D:\Coba\chap5
C:\Users\Luthfi\Desktop>D:
D:\Coba\chap5>python chap5_1174035_realtimetime.py
None
D:\Coba\chap5>pause
Press any key to continue . . .
```

**Gambar 4.8** Menulis data dari serial ke file CSV



**Gambar 4.9** Hasil Tulis CSV dengan ambilan tanpa loop



C:\Windows\system32\cmd.exe

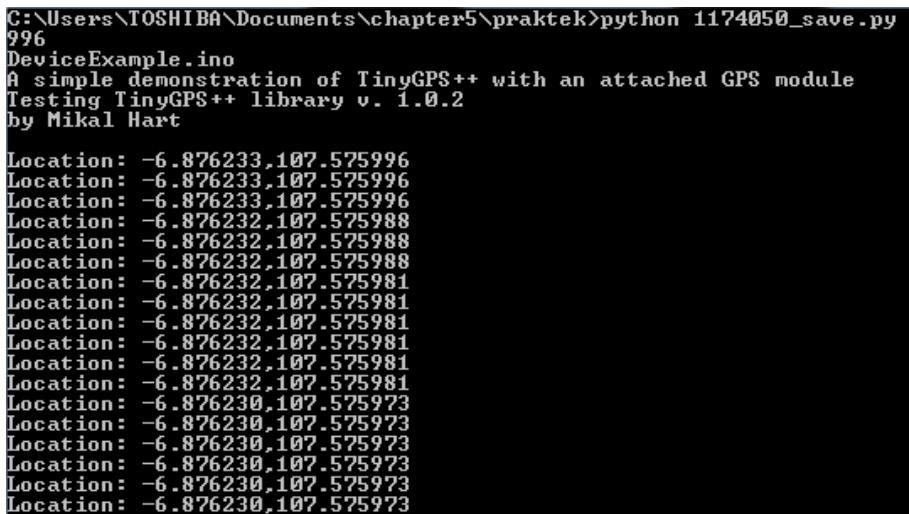
```
C:\Users\Luthfi\Desktop>cd D:\Coba\chap5
C:\Users\Luthfi\Desktop>D:
D:\Coba\chap5>python chap5_1174035_error.py
Arduino Belum Terhubung
D:\Coba\chap5>pause
Press any key to continue . . .
```

Gambar 4.10 Hasil dari Error



```
C:\Users\TOSHIBA\Documents\chapter5\praktek>python 1174050_realtimetime.py
DeviceExample.ino
```

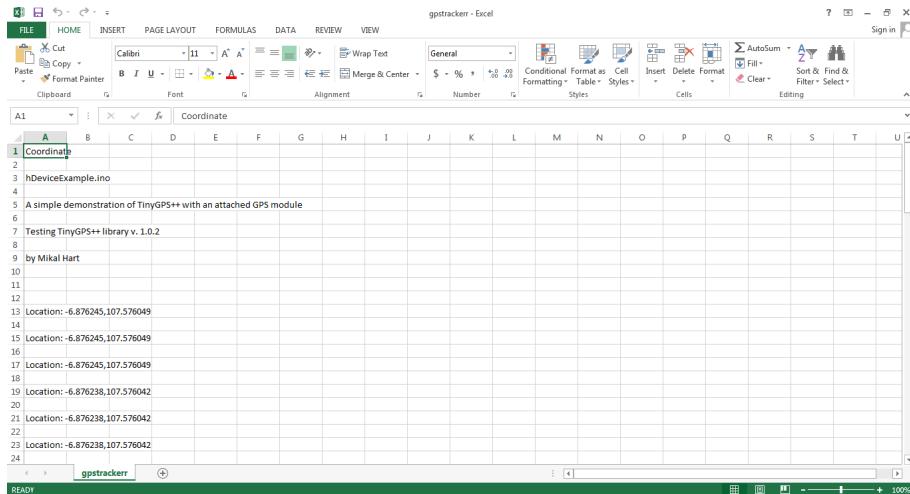
Gambar 4.11 Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino.



```
C:\Users\TOSHIBA\Documents\chapter5\praktek>python 1174050_save.py
996
DeviceExample.ino
A simple demonstration of TinyGPS++ with an attached GPS module
Testing TinyGPS++ library v. 1.0.2
by Mikal Hart

Location: -6.876233,107.575996
Location: -6.876233,107.575996
Location: -6.876233,107.575996
Location: -6.876232,107.575988
Location: -6.876232,107.575988
Location: -6.876232,107.575988
Location: -6.876232,107.575981
Location: -6.876232,107.575981
Location: -6.876232,107.575981
Location: -6.876232,107.575981
Location: -6.876232,107.575981
Location: -6.876230,107.575973
Location: -6.876230,107.575973
Location: -6.876230,107.575973
Location: -6.876230,107.575973
Location: -6.876230,107.575973
Location: -6.876230,107.575973
```

Gambar 4.12 Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino dengan looping.



**Gambar 4.13** Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino dan langsung ditulis kedalam file CSV.

```
C:\Users\TOSHIBA\Documents\chapter5\praktek>python 1174050_csv.py
hDeviceExample.ino
A simple demonstration of TinyGPS++ with an attached GPS module
Testing TinyGPS++ library v. 1.0.2
by Mikal Hart

Location: -6.876245,107.576049
Location: -6.876245,107.576049
Location: -6.876245,107.576049
Location: -6.876238,107.576042
Location: -6.876238,107.576042
Location: -6.876238,107.576042
Location: -6.876248,107.576057
Location: -6.876248,107.576057
Location: -6.876248,107.576057
Location: -6.876252,107.576057
Location: -6.876252,107.576057
Location: -6.876252,107.576057
Location: -6.876250,107.576057
Location: -6.876250,107.576057
Location: -6.876250,107.576057
Location: -6.876250,107.576057
Location: -6.876246,107.576049
```

**Gambar 4.14** Hasil dari pembacaan fungsi untuk membaca file csv hasil arduino dan mengembalikan fungsi.

```
C:\Users\TOSHIBA\Documents\chapter5\praktek>python 1174050.py
Tidak ada variabel tersebut
```

**Gambar 4.15** Hasil pembacaan fungsi untuk menanggulangi error menggunakan Try Except.

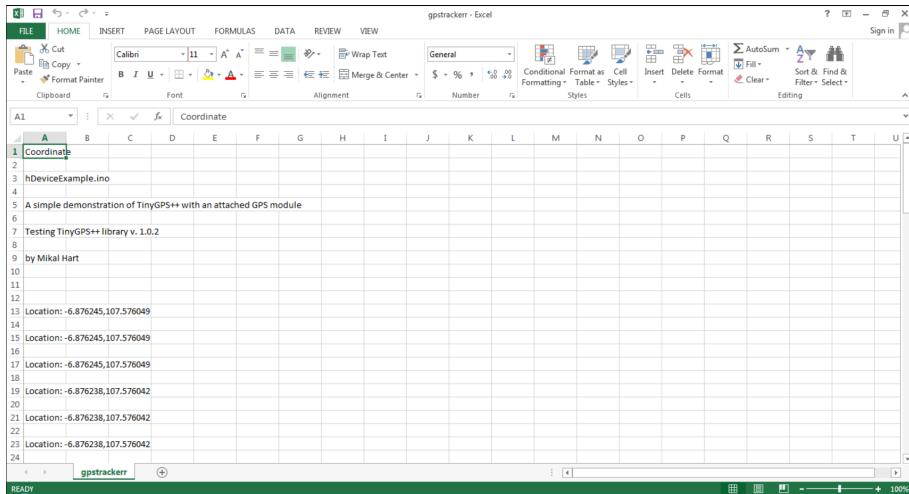
```
C:\Users\TOSHIBA\Documents\chapter5\praktek>python 1174050_realtime.py
DeviceExample.ino
```

**Gambar 4.16** Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino.

```
C:\Users\TOSHIBA\Documents\chapter5\praktek>python 1174050_save.py
996
DeviceExample.ino
A simple demonstration of TinyGPS++ with an attached GPS module
Testing TinyGPS++ library v. 1.0.2
by Mikal Hart

Location: -6.876233,107.575996
Location: -6.876233,107.575996
Location: -6.876233,107.575996
Location: -6.876232,107.575988
Location: -6.876232,107.575988
Location: -6.876232,107.575988
Location: -6.876232,107.575981
Location: -6.876232,107.575981
Location: -6.876232,107.575981
Location: -6.876232,107.575981
Location: -6.876232,107.575981
Location: -6.876232,107.575981
Location: -6.876230,107.575973
Location: -6.876230,107.575973
Location: -6.876230,107.575973
Location: -6.876230,107.575973
Location: -6.876230,107.575973
Location: -6.876230,107.575973
```

**Gambar 4.17** Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino dengan looping.



**Gambar 4.18** Hasil dari pembacaan fungsi untuk mendapatkan data dari Arduino dan langsung dituliskan kedalam file CSV.

```
C:\Users\TOSHIBA\Documents\chapter5\praktek>python 1174050_csv.py
hDeviceExample.ino
A simple demonstration of TinyGPS++ with an attached GPS module
Testing TinyGPS++ library v. 1.0.2
by Mikal Hart

Location: -6.876245,107.576049
Location: -6.876245,107.576049
Location: -6.876245,107.576049
Location: -6.876238,107.576042
Location: -6.876238,107.576042
Location: -6.876238,107.576042
Location: -6.876248,107.576057
Location: -6.876248,107.576057
Location: -6.876248,107.576057
Location: -6.876252,107.576057
Location: -6.876252,107.576057
Location: -6.876252,107.576057
Location: -6.876250,107.576057
Location: -6.876250,107.576057
Location: -6.876250,107.576057
Location: -6.876250,107.576057
Location: -6.876250,107.576057
Location: -6.876246,107.576049

```

**Gambar 4.19** Hasil dari pembacaan fungsi untuk membaca file csv hasil arduino dan mengembalikan fungsi.

```
C:\Users\TOSHIBA\Documents\chapter5\praktek>python 1174050.py
Tidak ada variabel tersebut
```

**Gambar 4.20** Hasil pembacaan fungsi untuk menanggulangi error menggunakan Try Except.



## BAB 5

---

# LIBRARY CSV DAN PANDAS

---

### 5.1 Hagan Rowlenstino/1174040

#### 5.1.1 Soal 1

library matplotlib adalah sebuah library untuk memplotting 2 Dimensi yang outputnya adalah gambar publikasi yang bermutu di banyak format hardocpy serta lingkungan interaktif berbagai platform

#### 5.1.2 Soal 2

untuk membuat sumbu x dan y :

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Tue Apr  9 03:43:07 2019
4
5 @author: Haganz
6 """
7
8 import matplotlib.pyplot as plt
9 x = (4,8,13,17,20)
```

```

10 y = (54, 67, 98, 78, 45)
11 plt.plot(x,y)
12 plt.show()

```

### 5.1.3 Soal 3

- PLOT Garis :

```

1 import matplotlib.pyplot as plt
2 x = (4,8,13,17,20)
3 y = (54, 67, 98, 78, 45)
4 #PLOT Garis
5 plt.plot(x,y)
6 plt.show()

```

- Plot Sebaran

```

1 x = (4,8,13,17,20)
2 y = (54, 67, 98, 78, 45)
3 plt.scatter(x,y)
4 plt.show()

```

- Plot Batang / Histogram

```

1 b
2     =[2,4,6,5,42,543,5,3,73,64,42,97,63,76,63,8,73,97,23,45,56,89,45,3,23,2
3 num_bins = 6
4 n, bins, patches = plt.hist(b, num_bins, facecolor = 'green')
5 plt.show()

```

- Pie

```

1 labels = 'Jalan Jalan', 'Main', 'Makan', 'Belajar'
2 sizes = [15, 30, 45, 10]
3 explode = (0, 0.1, 0, 0)
4
5 fig1, ax1 = plt.subplots()
6 ax1.pie(sizes, explode=explode, labels=labels, autopct='%1.0f%%',
7 shadow=True, startangle=90)
8 ax1.axis('equal')
9
10 plt.show()

```

### 5.1.4 Soal 4

- Legend : Penjelasan garis beserta contoh dari garis yang dijelaskan tersebut untuk membuat legend dapat menggunakan sintaks berikut :

```
legend('legend grafik1','...','legend grafikN','Nilai Pos')
```

- Label : Memberikan penamaan untuk skala(sumbu) yang kita buat. Untuk menambahkan label, kita dapat menggunakan sintaks berikut :

```
xlabel('teks horizontal axis')
```

```
ylabel('teks vertikal axis')
```

### 5.1.5 Soal 5

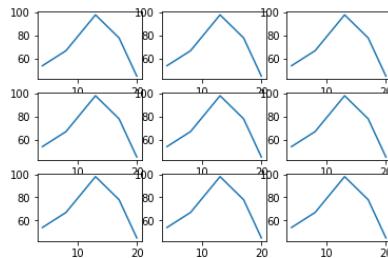
Subplot pada matplotlib berfungsi untuk membuat banyak plot grafik dalam satu figure saja. dimana subplot dapat kita definisikan sebagai berikut :

```
subplot(m,n,i)
```

Dimana m adalah tinggi nya, n adalah lebar nya dan i adalah urutan penempatannya. Untuk membuat 9 subplot dapat dilihat dari contoh dibawah ini :

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Tue Apr  9 03:43:07 2019
4
5 @author: Haganz
6 """
7
8 import matplotlib.pyplot as plt
9
10 plt.subplot(331)
11 x = (4,8,13,17,22)
12 y = (54, 67, 98, 78, 45)
13 plt.plot([4,8,13,17,20],[54, 67, 98, 78, 45])
14 plt.subplot(332)
15
16 plt.plot([4,8,13,17,20],[54, 67, 98, 78, 45])
17 plt.subplot(333)
18
19 plt.plot([4,8,13,17,20],[54, 67, 98, 78, 45])
20 plt.subplot(334)
21
22 plt.plot([4,8,13,17,20],[54, 67, 98, 78, 45])
23 plt.subplot(335)
24
25 plt.plot([4,8,13,17,20],[54, 67, 98, 78, 45])
26 plt.subplot(336)
27
28 plt.plot([4,8,13,17,20],[54, 67, 98, 78, 45])
29 plt.subplot(337)
30
31 plt.plot([4,8,13,17,20],[54, 67, 98, 78, 45])
32 plt.subplot(338)
33
34 plt.plot([4,8,13,17,20],[54, 67, 98, 78, 45])
35 plt.subplot(339)
```

```
37 plt.plot([4,8,13,17,20],[54, 67, 98, 78, 45])
38 plt.show()
```



**Gambar 5.1** No. 5

### 5.1.6 Soal 6

color yang dapat digunakan adalah red(r),blue(b),green(g),cyan(c),magenta(m),yellow(y),black

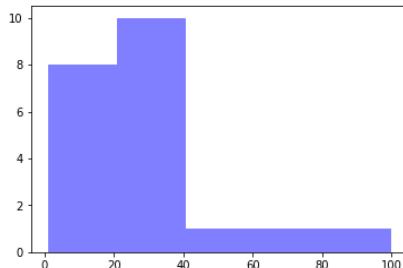
### 5.1.7 Soal 7

fungsi Hist digunakan untuk membuat histogram yang berfungsi untuk menampilkan frekuensi data dengan menggunakan grafik batang

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Tue Apr  9 09:25:22 2019
4
5 @author: Haganz
6 """
7
8 import numpy as np
9 import matplotlib.mlab as mlab
10 import matplotlib.pyplot as plt
11
12 x = [21,22,23,4,5,6,77,8,9,10,31,32,33,34,35,36,37,18,49,100,1]
13 num_bins = 5
14 n, bins, patches = plt.hist(x, num_bins, facecolor='blue', alpha=0.5)
15 plt.show()
```

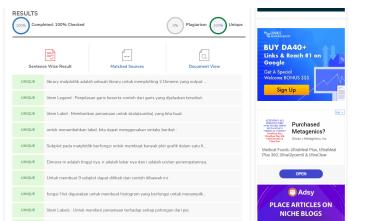
### 5.1.8 Soal 8

- Labels : Untuk memberi penamaan terhadap setiap potongan dari pie
- colors : Untuk memberikan warna spesifik terhadap potongan potongan pie, jika tidak diberi warna maka dia akan menggambil warna dari pie yang telah berjalan

**Gambar 5.2** No. 7

- startangle : Untuk menetapkan dari sudut mana grafik tersebut akan dimulai
- shadow : untuk memberikan efek bayangan di bawah pie maupun potongannya
- explode : untuk menentukan seberapa jauh pemisahkan potongan pie dari potongan - potongan lainnya
- autopct : untuk memberikan nilai atau skalar numeric pada label didalam potongan pie . seperti mengubah dari 10 menjadi 10.0

### 5.1.9 Cek Plagiarisme

**Gambar 5.3** Cek Plagiarisme

## 5.2 Irvan Rizkiansyah/1174043

### 5.2.1 Nomor 1

matplotlib merupakan library plotting 2D Python dan berfungsi untuk membuat plot, histogram, grafik batang, scatterplot, dan masih banyak yang lainnya hanya dengan beberapa baris code saja.

### 5.2.2 Nomor 2

Pembuatan sumbu X dan Y di matplotlib, mirip dengan membuat sebuah variabel yang mempunyai array. Kemudian variabel yang dibuat yang nanti-nya akan menjadi sumbu X dan Y dan akan dipanggil menggunakan fungsi plot.

```
1 import matplotlib.pyplot as plt  
2  
3 x = [4,8,13,20,23]  
4 y = [54,67,99,78,80]  
5  
6 plt.plot(x,y)  
7 plt.show()
```

### 5.2.3 Nomor 3

Perbedaan fungsi yang ada pada matplotlib :

- Fungsi scatter atau plot sebaran merupakan sebuah grafik yang menunjukkan hubungan dari dua set data, seperti halnya hubungan dari tinggi badan dan berat badan. Cara menggunakannya dengan memanggil fungsi scatter yang terdapat pada matplotlib

```
matplotlib.pyplot.scatter(x,y)
```

- Fungsi histogram merupakan sebuah grafik yang akan menampilkan frekuensi data menggunakan sebuah batang. Cara menggunakannya dengan memanggil fungsi hist yang terdapat pada matplotlib

```
matplotlib.pyplot.hist(x,y)
```

- Fungsi plot garis atau plot sinus merupakan sebuah grafik yang menunjukkan frekuensi data menggunakan garis. Cara menggunakannya dengan memanggil fungsi plot yang terdapat pada matplotlib

```
matplotlib.pyplot.plot(x,y)
```

Dan masih banyak yang lainnya.

### 5.2.4 Nomor 4

Fungsi legend merupakan dimana akan menamai sebuah bar pada grafik untuk membedakan bar tersebut dengan bar yang lainnya, dimana jika terdapat banyak bar pada grafik tersebut. Sedangkan fungsi label untuk menamai sumbu x dan sumbu y. Cara menggunakannya hanya tinggal memanggil fungsi legend dan fungsi label yang terdapat pada matplotlib dan memberikan nama untuk legend dan label tersebut. Contohnya :

```
1 import matplotlib.pyplot as plt
2
3 x = [4,8,13,20,23]
4 y = [54,67,99,78,80]
5 y2 = [60,80,69,65,95]
6
7 plt.plot(x,y, label='contoh')
8 plt.plot(x,y2, label='contoh2')
9 plt.xlabel('label1')
10 plt.ylabel('label2')
11 plt.legend()
12 plt.show()
```

## 5.2.5 Nomor 5

Fungsi subplot adalah untuk memanggil banyak grafik hanya dengan sekali panggil saja. Cara kerja dari fungsi subplot ini sendiri cukup simpel, dimana cukup memanggil fungsi subplot yang terdapat pada matplotlib

```
matplotlib.pyplot.subplot(nbaris, nkolom, index)
```

Dibawah ini merupakan ilustrasi dan parameter apa saja yang digunakan jika ingin menggambar plot dengan 9 sub plot :

```
1 import matplotlib.pyplot as plt
2
3 x = [4,8,13,20,23]
4 y = [54,67,99,78,80]
5 y2 = [60,80,69,65,95]
6
7 plt.plot(x,y, label='contoh')
8 plt.plot(x,y2, label='contoh2')
9 plt.xlabel('label1')
10 plt.ylabel('label2')
11 plt.legend()
12
13 plt.subplot(3,3,1)
14 plt.title('subplot(3,3,1)')
15 plt.plot(x,y)
16
17 plt.subplot(3,3,2)
18 plt.title('subplot(3,3,2)')
19 plt.plot(x,y)
20
21 plt.subplot(3,3,3)
22 plt.title('subplot(3,3,3)')
23 plt.plot(x,y)
24
25 plt.subplot(3,3,4)
26 plt.title('subplot(3,3,4)')
27 plt.plot(x,y)
28
29 plt.subplot(3,3,5)
30 plt.title('subplot(3,3,5)')
```

```

31 plt.plot(x,y)
32
33 plt.subplot(3,3,6)
34 plt.title('subplot(3,3,6)')
35 plt.plot(x,y)
36
37 plt.subplot(3,3,7)
38 plt.title('subplot(3,3,7)')
39 plt.plot(x,y)
40
41 plt.subplot(3,3,8)
42 plt.title('subplot(3,3,8)')
43 plt.plot(x,y)
44
45 plt.subplot(3,3,9)
46 plt.title('subplot(3,3,9)')
47 plt.plot(x,y)

```

### 5.2.6 Nomor 6

Matplotlib mengenal beberapa format untuk color diantaranya :

- RGB atau RGBA contohnya (0.1, 0.2, 0.5)
- hex RGB atau RGBA contohnya '#0F0F0F'
- Representasi string dari nilai float contohnya '0.5'
- atau hanya seperti kata awal dari warna tersebut yang ingin digunakan contohnya warna Cyan maka hanya perlu menggunakan 'c', warna Green maka hanya perlu menggunakan 'g' dan masih banyak yang lainnya seperti 'b', 'r', 'm', dan yang lainnya
- X11/CSS4 nama warna
- nama dari xkcd color survey contohnya 'xkcd:sky blue'
- Tableau Color dari 'T10' palette kategorikal contohnya 'tab:olive'

### 5.2.7 Nomor 7

Fungsi hist merupakan fungsi untuk menggunakan grafik bar atau batang, dimana frekuensi setiap elemen data yang ada pada daftar ditunjukkan dengan grafik histogram. Angka yang dikelompokkan dalam bentuk rentang tertentu yang sudah ditentukan disebut bins.

```

1 import matplotlib.pyplot as plt
2 x =
3     [2,4,6,5,42,543,5,3,73,64,42,97,63,76,63,8,73,97,23,45,56,89,45,3,23,2,5,78
4
5 num_bins = 6
6 n, bins, patches = plt.hist(x, num_bins, facecolor = 'green')
7 plt.show()

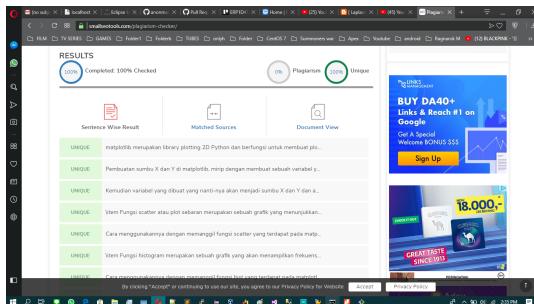
```

## 5.2.8 Nomor 8

Parameter yang terdapat pada fungsi pie :

- labels : dimana berfungsi untuk memberi nama untuk setiap slice yang terdapat pada grafik pie
- colors : dimana berfungsi untuk memberikan warna untuk setiap slice yang terdapat pada gradik pie
- startangle : dimana berfungsi jika tidak ada, maka awalan putaran dari grafik pie berlawanan dari arah jarum jam dari titik sumbu x
- shadow : dimana berfungsi untuk memberikan efek grafis bayangan pada grafik pie
- explode : dimana berfungsi untuk menentukan fraksi dari jari-jari untuk mengimbangi setiap slice menggunakan sumbu x
- autopct : string atau fungsi yang digunakan untuk label pada slice dengan nilai numerik, maka label akan ditempatkan didalam slice.

## 5.2.9 Plagiarisme



Gambar 5.4 Plagiarisme

## 5.3 Luthfi Muhammad Nabil/1174035

### 5.3.1 Soal 1

Apa itu fungsi library matplotlib

Library matplotlib adalah library bahasa pemrograman python untuk menampilkan grafik - grafik sehingga data dapat divisualisasikan dengan lebih baik. Isi dari grafik tersebut dapat menggunakan data sesuai kebutuhan. Matplotlib sudah digunakan banyak aplikasi data sains untuk menampilkan visualisasi data yang berupa grafik.

### 5.3.2 Soal 2

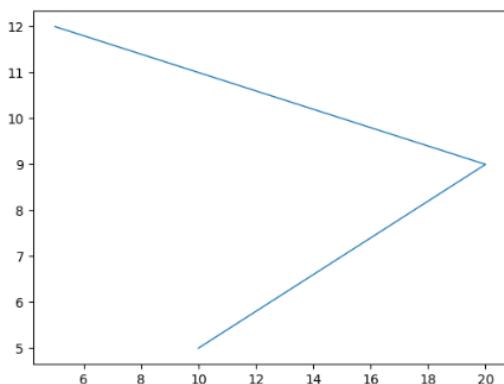
Jelaskan langkah-langkah membuat sumbu X dan Y di matplotlib

Pada library matplotlib, hanya membutuhkan dua variabel atau dua kumpulan data yang memiliki posisi sama. Isi dari data tersebut menggunakan nilai angka dan isinya dapat berupa integer atau float. Berikut contoh dari koding menampilkan data dengan sumbu X dan Y :

```

1 import matplotlib.pyplot as plt
2 x = [10,20,5]
3 y = [5,9,12]
4 plt.plot(x,y, linewidth=1.0)
5 plt.show()
    
```

Berikut hasilnya terdapat pada gambar 5.5



**Gambar 5.5** Contoh Pengaplikasian Sumbu X dan Y

### 5.3.3 Soal 3

Jelaskan bagaimana perbedaan fungsi dan cara pakai untuk berbagai jenis plot di matplotlib

- Grafik Garis

Untuk plot pada diagram garis, dibutuhkan data nilai angka yang dapat berupa satu atau dua kumpulan data. Berikut pengaplikasian untuk diagram garis :

```

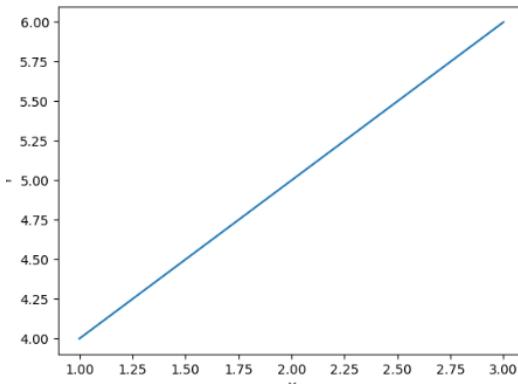
1 import matplotlib.pyplot as plt
2 plt.plot([1,2,3,4])
3 plt.ylabel('Angka')
4 plt.show()
    
```

Atau dapat juga mengubah jalur garis dengan 2 kumpulan angka seperti berikut :

```

1 import matplotlib.pyplot as plt
2 plt.plot([1,2,3,4], [3,2,1,5])
3 plt.ylabel('Angka')
4 plt.show()

```



**Gambar 5.6** Contoh Grafik Garis

#### ▪ Grafik Titik

Untuk plot pada diagram titik, dapat melakukan langkah seperti berikut :

```

1 from matplotlib import style
2 import matplotlib.pyplot as plt
3
4 plt.plot([1,2,3,4], [1,4,9,16], 'ro')
5 plt.axis([0,6,0,20])
6 plt.show()

```

Terlihat perbedaannya pada penambahan parameter ketiga dengan isinya yaitu 'ro'

#### ▪ Grafik Batang

Untuk plot diagram batang diperlukan koding baru menggunakan plt.bar(). Contoh pengaplikasian dapat diaplikasikan sebagai berikut :

```

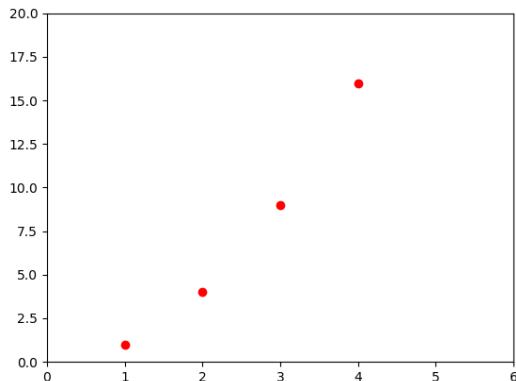
1 import matplotlib.pyplot as plt
2 plt.bar([1,2,3,4], [1,2,4,2])
3 plt.show()

```

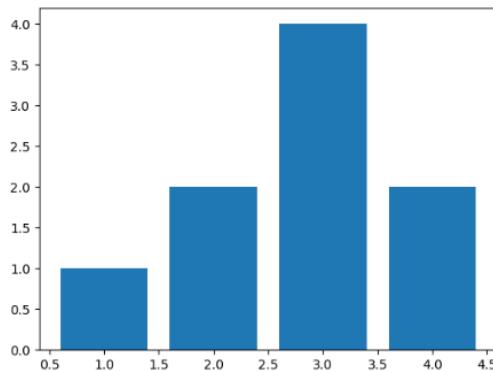
Hasil dapat dilihat pada gambar 5.8

#### ▪ Grafik Lingkaran

Untuk plot diagram batang perlu menggunakan subplot. Berikut contoh pengaplikasiannya :



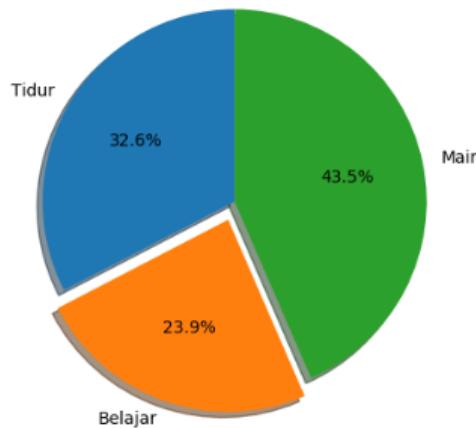
**Gambar 5.7** Contoh Grafik Titik



**Gambar 5.8** Contoh Grafik Batang

```
1 import matplotlib.pyplot as plt
2 labels = 'Tidur', 'Belajar', 'Main',
3 sizes = [15, 11, 20]
4 explode = (0, 0.1, 0)
5
6 fig1, ax1 = plt.subplots()
7 ax1.pie(sizes, explode=explode, labels=labels, autopct='%1.1f%%',
8         shadow=True, startangle=90)
9 ax1.axis('equal')
10 plt.show()
```

Hasil dapat dilihat pada gambar 5.9



**Gambar 5.9** Contoh Grafik Pie

#### 5.3.4 Soal 4

Jelaskan bagaimana cara menggunakan legend dan label serta kaitannya dengan fungsi tersebut

- **Label**

Untuk menggunakan label, hanya perlu menambahkan sintaks plt.xlabel() atau plt.ylabel(). Label berfungsi untuk memberikan label pada setiap garis atau label untuk gambar yang ada. Berikut contoh untuk menambahkan label pada matplotlib :

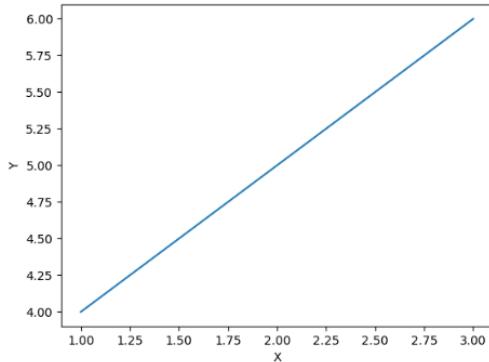
```
1 import matplotlib.pyplot as plt
2 xxx =[1,2,3]
3 yyy =[4,5,6]
4 plt.plot(xxx,yyy)
5 plt.xlabel("X")
6 plt.ylabel("Y")
7 plt.show()
```

Pada sintaks berikut, akan menambahkan label pada sumbu x dengan tulisan 'X', begitu juga dengan sumbu y dengan label 'Y'. Berikut hasilnya pada gambar 5.10

- **Legend**

Untuk menggunakan legend, perlu mengadakan label saat menginisiasi sebuah plot. Berikut contoh penggunaan legend :

```
1 import matplotlib.pyplot as plt
2 x = [5,8,10]
3 y = [12,16,6]
```

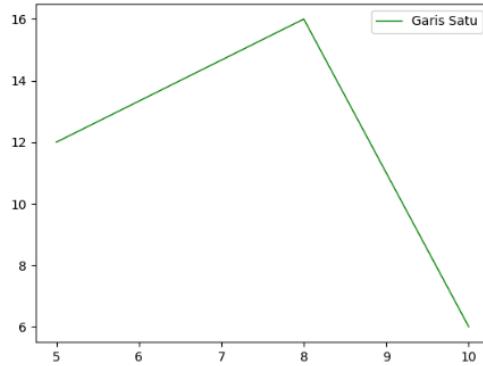


**Gambar 5.10** Contoh Pengaplikasian Label

```

4 plt.plot(x,y, 'g', label="Garis Satu", linewidth=1)
5 plt.legend()
6 plt.show()
```

Pada sintaks plt.plot(), terdapat parameter untuk menginput label yang isinya 'Garis Satu' untuk mengatur isi atau judul dari label tersebut. Lalu plt.legend() adalah sintaks untuk menunjukan legend yang akan dimunculkan. Berikut contoh dari pengaplikasiannya pada gambar 5.11



**Gambar 5.11** Contoh Pengaplikasian Legend

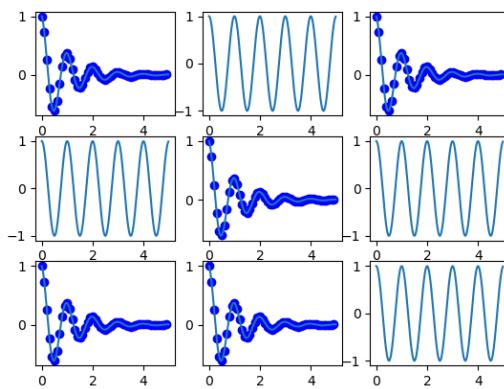
### 5.3.5 Soal 5

Jelaskan apa fungsi dari subplot di matplotlib, dan bagaimana cara kerja dari fungsi subplot

Subplot adalah fungsi untuk menampilkan beberapa diagram pada satu aplikasi atau sintaks. Untuk menginisiasi subplot, dapat menggunakan fungsi plt.subplot() dan memasukan ukuran subplot di dalamnya (Contoh : plt.subplot(331)). Pada fungsi plt.subplot(331), angka pertama pada contoh merupakan jumlah baris yang dapat dipakai dan angka kedua merupakan jumlah kolom yang dapat dipakai sedangkan angka ketiga merupakan di baris dan kolom berapa grafik akan disimpan. Posisi grafik dimulai dari kiri atas dan berpindah ke kanan dan jika sudah mencapai batas kolom maka akan berpindah ke baris selanjutnya yang dimulai dari kiri kembali. Contoh penggunaan subplot :

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 def f(t):
5     return np.exp(-t) * np.cos(2*np.pi*t)
6 t1 = np.arange(0.0, 5.0, 0.1)
7 t2 = np.arange(0.0, 5.0, 0.02)
8 #Baris 1 Kolom 1
9 plt.subplot(331)
10 plt.plot(t1, f(t1), 'bo', t2, f(t2))
11
12 #Baris 1 Kolom 2
13 plt.subplot(332)
14 plt.plot(t2, np.cos(2*np.pi*t2))
15
16 #Baris 1 Kolom 3
17 plt.subplot(333)
18 plt.plot(t1, f(t1), 'bo', t2, f(t2))
19
20 #Baris 2 Kolom 1
21 plt.subplot(334)
22 plt.plot(t2, np.cos(2*np.pi*t2))
23
24 #Baris 2 Kolom 2
25 plt.subplot(335)
26 plt.plot(t1, f(t1), 'bo', t2, f(t2))
27
28 #Baris 2 Kolom 3
29 plt.subplot(336)
30 plt.plot(t2, np.cos(2*np.pi*t2))
31
32 #Baris 3 Kolom 1
33 plt.subplot(337)
34 plt.plot(t1, f(t1), 'bo', t2, f(t2))
35
36 #Baris 3 Kolom 2
37 plt.subplot(338)
38 plt.plot(t1, f(t1), 'bo', t2, f(t2))
39
40 #Baris 3 Kolom 3
41 plt.subplot(339)
42 plt.plot(t2, np.cos(2*np.pi*t2))
43 plt.show()
```

Berikut hasil yang didapat pada coding berikut ditampilkan pada gambar 5.12



**Gambar 5.12** Contoh Pengaplikasian Subplot

### 5.3.6 Soal 6

Sebutkan semua parameter color yang bisa digunakan

- b : Untuk memberikan warna biru
- g : Untuk memberikan warna hijau
- r : Untuk memberikan warna merah
- c : Untuk memberikan warna biru muda
- m : Untuk memberikan warna pink
- y : Untuk memberikan warna kuning
- k : Untuk memberikan warna hitam
- w : Untuk memberikan warna putih

### 5.3.7 Soal 7

Jelaskan bagaimana cara kerja dari fungsi hist

Fungsi hist pada library matplotlib yaitu menhistogramkan kumpulan data yang akan ditampilkan biasanya berupa grafik batang. Grafik akan dimasukkan beberapa data dan diambil frekuensi dari data tersebut. Berikut contoh coding dari histogram :

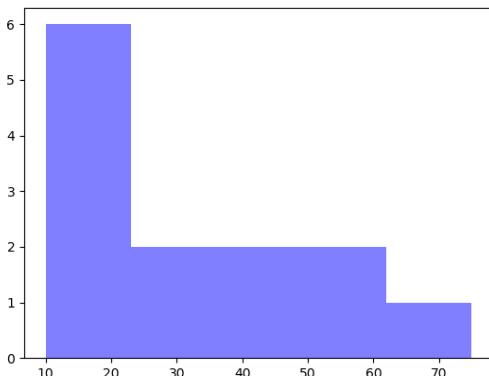
```

1 import numpy as np
2 import matplotlib.mlab as mlab
3 import matplotlib.pyplot as plt
4
5 x = [10, 10, 10, 10, 40, 20, 15, 25, 40, 75, 60, 50, 30]

```

```
6 num_bins = 5
7 n, bins, patches = plt.hist(x, num_bins, facecolor='blue', alpha
8 = 0.5)
plt.show()
```

Berikut hasil dari histogram yang ada pada gambar 5.13



**Gambar 5.13** Contoh Pengaplikasian Histogram

### 5.3.8 Soal 8

Jelaskan lebih mendalam tentang parameter dari fungsi pie.

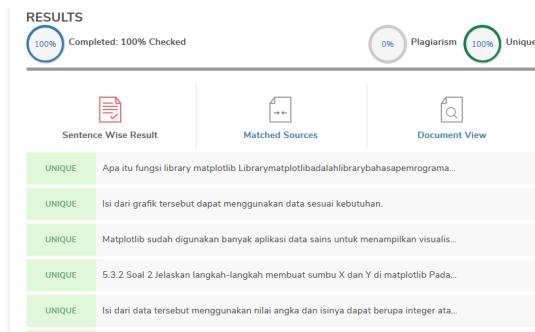
- **labels** : Isi dengan tipe data list dan tidak wajib untuk digunakan. Fungsi parameter labels untuk memberi label pada setiap pecahan data yang ada pada grafik pie yang ditampilkan.
- **colors** : Tipe data array atau sejenis dan tidak wajib untuk digunakan. Fungsi parameter colors untuk mengganti warna pada setiap pecahan yang ada. Jika tidak digunakan atau ditentukan, maka warna yang akan dipakai adalah warna yang aktif atau standar.
- **startangle** : Tipe data pecahan atau float, tidak wajib untuk digunakan. Fungsi parameter startangle adalah fungsi untuk memutar grafik agar berubah posisi dengan acuan yaitu angle awalan dari grafik pie.
- **shadow** : Bertipe data boolean dan tidak wajib digunakan. Fungsi parameter shadow digunakan untuk membuat bayangan pada bawah grafik pie yang ditampilkan.
- **explode** : Bertipe data array atau sejenis dan tidak wajib digunakan. Fungsi parameter explode adalah menentukan radius untuk mengimbangi setiap pecahan

pada grafik pie. Jika radius lebih dari 0 maka pecahan akan mulai menjauh dari pusat dan terlihat seperti keluar dari grafik lingkaran tersebut.

- autopct : Bertipe data string atau fungsi dan tidak wajib digunakan. Fungsi parameter autopct adalah memberi label pada irisan dengan labelnya berupa fungsi atau string.

### 5.3.9 Plagiarisme

Berikut hasil pengecekan plagiarisme untuk bagian teori terdapat pada gambar 5.14.



**Gambar 5.14 Cek Plagiarisme**

## 5.4 Faisal Najib Abdullah 1174042

### 5.4.1 Teori

#### 5.4.1.1 Soal No. 1

Apa itu fungsi library matplotlib?

Matplotlib merupakan salah satu library Python 2D yang dapat menghasilkan plot dengan kualitas yang tinggi dalam berbagai format dan dapat digunakan di berbagai platform. Matplotlib berfungsi sebagai pembuat grafik di berbagai platform, seperti Python dan Jupyter. Grafik yang dibuat menggunakan Matplotlib bisa dibuat dalam berbagai bentuk, seperti grafik garis, batang, lingkaran, histogram, dan sebagainya.

#### 5.4.1.2 Soal No. 2

Jelaskan langkah-langkah membuat sumbu X dan Y di matplotlib!

1. Pertama import library Matplotlib.

```
| from matplotlib import pyplot as plt
```

2. Buat variabel x yang menampung list untuk sumbu x dan variabel y yang menampung list untuk sumbu y.

```
1 x=[1,2,3]
2 y=[5,2,4]
```

3. Panggil fungsi plot dan isi parameter pertama dengan variabel x dan parameter kedua dengan variabel y.

```
1 plt.plot(x,y)
```

4. Lalu panggil plot tadi dengan memanggil fungsi show.

```
1 plt.show()
```

## Kode Program

```
1 from matplotlib import pyplot as plt
2
3 x=[1,2,3]
4 y=[5,2,4]
5
6 plt.plot(x,y)
7
8 plt.show()
```

**Listing 5.1** Kode program membuat diagram menggunakan Matplotlib.

## Hasil Compile

### 5.4.1.3 Soal No. 3

Jelaskan bagaimana perbedaan fungsi dan cara pakai untuk berbagai jenis(bar, histogram ,scatter ,line, dll) jenis plot di matplotlib!

#### 1. Bar Graph

Perbedaan bar graph dengan jenis plot yang lain adalah bar graph menggunakan bar atau batang-batang untuk membandingkan data di antara berbagai kategori.

#### Kode Program

```
1 from matplotlib import pyplot as plt
2
3 plt.bar
    ([2012.7,2013.7,2014.7,2015.7,2016.7,2017.7],[9000,9500,10000,15000,20000])
4 label="Mac",color='r',width=.3)
5 plt.bar
    ([2013,2014,2015,2016,2017,2018],[20000,25000,30000,35000,40000,45000])
6 label="Windows",color='g',width=.3)
7 plt.bar
    ([2013.3,2014.3,2015.3,2016.3,2017.3,2018.3],[2000,2500,3000,3500,4000,4500])
```

```

8 label="Linux", color='b', width=.3)
9 plt.legend()
10 plt.xlabel('Tahun')
11 plt.ylabel('Jumlah Pengguna')
12 plt.title('Pengguna Sistem Operasi Dari Tahun ke Tahun')
13 plt.show()

```

**Listing 5.2** Kode program membuat bar graph menggunakan Matplotlib.

## Hasil Compile

### 2. Histogram

Perbedaan histogram dengan jenis plot yang lain adalah histogram akan membuat plot dimana plot yang dimunculkan merupakan gabungan dari beberapa data yang telah dikelompokkan.

#### Kode Program

```

1 import matplotlib.pyplot as plt
2 orang =
3     [11,22,16,9,10,15,22,55,62,45,21,22,24,20,21,19,34,42,42,4,2,102,95,85
4 umur = [0,10,20,30,40,50,60,70,80,90,100]
5 plt.hist(orang, umur, histtype='bar', rwidth=0.8)
6 plt.xlabel('Umur')
7 plt.ylabel('Jumlah Orang')
8 plt.title('Histogram')
9 plt.show()

```

**Listing 5.3** Kode program membuat histogram menggunakan Matplotlib.

## Hasil Compile

### 3. Scatter Plot

Perbedaan scatter plot dengan jenis plot lain adalah scatter plot menampilkan data sebagai kumpulan titik, masing-masing memiliki nilai satu variabel yang menentukan posisi pada sumbu horizontal dan nilai variabel lain menentukan posisi pada sumbu vertikal.

#### Kode Program

```

1 import matplotlib.pyplot as plt
2 x = [1,1.5,2,2.5,3,3.5,3.6]
3 y = [7.5,8,8.5,9,9.5,10,10.5]
4
5 x1=[8,8.5,9,9.5,10,10.5,11]
6 y1=[3,3.5,3.7,4,4.5,5,5.2]
7
8 plt.scatter(x,y, label='pendapatan tinggi simpanan rendah',color=
9     'g')
10 plt.scatter(x1,y1,label='pendapatan rendah simpanan tinggi',color
11     ='b')
12 plt.xlabel('simpanan dalam ratusan')
13 plt.ylabel('pendapatan dalam ribuan')
14 plt.title('Scatter Plot')

```

```

13 plt.legend()
14 plt.show()
```

**Listing 5.4** Kode program membuat scatter plot menggunakan Matplotlib.

## Hasil Compile

### 4. Area Plot

Perbedaan area plot dengan jenis plot lain adalah area plot digunakan untuk melacak perubahan dari waktu ke waktu untuk dua atau lebih kelompok terkait yang membentuk satu kategori secara keseluruhan.

#### Kode Program

```

1 import matplotlib.pyplot as plt
2 hari = [1,2,3,4,5]
3
4 tidur =[7,8,6,11,7]
5 makan = [2,3,4,3,4]
6 kuliah =[7,8,7,2,5]
7 jalan_jalan = [8,5,7,8,13]
8
9 plt.plot([],[],color='c', label='Tidur', linewidth=5)
10 plt.plot([],[],color='m', label='Makan', linewidth=5)
11 plt.plot([],[],color='y', label='Kuliah', linewidth=5)
12 plt.plot([],[],color='k', label='Jalan-jalan', linewidth=5)
13
14 plt.stackplot(hari,tidur,makan,kuliah,jalan_jalan, colors=['c','m',
   'y','k'])
15
16 plt.xlabel('x')
17 plt.ylabel('y')
18 plt.title('Kegiatan Sehari-hari Najib')
19 plt.legend()
20 plt.show()
```

**Listing 5.5** Kode program membuat diagram menggunakan Matplotlib.

## Hasil Compile

### 5. Pie Plot

Perbedaan pie plot dengan jenis plot lain adalah pie plot digunakan untuk menunjukkan persentase atau data proporsional di mana setiap potongan pie mewakili kategori.

#### Kode Program

```

1 import matplotlib.pyplot as plt
2
3 potong = [7,2,2,12]
4 kegiatan = ['Tidur','Makan','Kuliah','Jalan-jalan']
5 kolom = ['c','m','y','g']
6
7 plt.pie(potong,
   labels=kegiatan,
```

```

9 colors=kolom ,
10 startangle=90,
11 shadow= True ,
12 explode=(0 ,0 ,0.2 ,0) ,
13 autopct='%.1f%%')
14
15 plt.title('Kegiatan Sehari-hari Najib')
16 plt.show()

```

**Listing 5.6** Kode program membuat Pie Plot menggunakan Matplotlib.

## Hasil Compile

### 6. Line Graph

Perbedaan line graph dengan jenis plot lain adalah line graph menampilkan diagram dalam bentuk garis.

#### Kode Program

```

1 from matplotlib import pyplot as plt
2
3 y = [4000,6000,10000,13000,14000,17000]
4 x = [2014,2015,2016,2017,2018,2019]
5 plt.plot(x,y)
6 plt.title('Pemakai Sistem Operasi Linux')
7 plt.ylabel('Tahun')
8 plt.xlabel('Jumlah Pemakai Sistem Operasi Linux pada Tahun ke tahun')
9 plt.show()

```

**Listing 5.7** Kode program membuat diagram menggunakan Matplotlib.

## Hasil Compile

#### 5.4.1.4 Soal No. 4

Jelaskan bagaimana cara menggunakan legend dan label serta kaitannya dengan fungsi tersebut!

- Untuk menggunakan legend definisikan parameter label di tiap fungsi plot. Parameter label digunakan untuk memberikan label pada line sebagai pembeda antar line.

```

1 plt.plot(x,y,'b',label='Team Najib', linewidth=1)
2 plt.plot(x2,y2,'r',label='Team Abdullah', linewidth=1)

```

**Listing 5.8** Kode program menggunakan parameter label dengan Matplotlib.

- Kemudian panggil fungsi legend.

```
1 plt.legend()
```

**Listing 5.9** Kode program memanggil fungsi legend dengan Matplotlib.

#### Kode Program

```

1 from matplotlib import pyplot as plt
2
3 x = [2014,2015,2016,2017,2018,2019]
4 y = [76,87,105,122,148,170]
5 x2 = [2014,2015,2016,2017,2018,2019]
6 y2 = [78,97,114,134,146,167]
7 plt.plot(x,y,'b',label='Team Najib', linewidth=1)
8 plt.plot(x2,y2,'r',label='Team Abdullah', linewidth=1)
9 plt.title('Najib Wars')
10 plt.ylabel('Jumlah Pendukung')
11 plt.xlabel('Tahun')
12 plt.legend()
13 plt.grid(True, color='k')
14 plt.show()

```

**Listing 5.10** Kode program membuat diagram menggunakan Matplotlib.

## Hasil Compile

### 5.4.1.5 Soal No. 5

Jelaskan apa fungsi dari subplot di matplotlib, dan bagaimana cara kerja dari fungsi subplot, sertakan ilustrasi dan gambar sendiri dan apa parameternya jika ingin menggambarkan plot dengan 9 subplot di dalamnya!

Fungsi subplot adalah untuk membuat beberapa plot di dalam satu gambar.

Cara kerja subplot, yaitu fungsi subplot memiliki parameter pertama adalah jumlah kolom, parameter kedua adalah jumlah baris, dan parameter ketiga adalah index plot keberapanya.

## Kode Program

```

1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 t = np.arange(0.0, 9.0, 1)
5 s = [1,2,3,4,5,6,7,8,9]
6
7 for i in range(1, 10):
8     plt.subplot(3,3,i)
9     plt.xticks([]), plt.yticks([])
10    plt.title('subplot(1,2,'+str(i)+')')
11    plt.plot(t,s,'-y')
12
13 plt.show()

```

**Listing 5.11** Kode program membuat subplot menggunakan Matplotlib.

## Hasil Compile

### 5.4.1.6 Soal No. 6

Sebutkan semua parameter color yang bisa digunakan (contoh: m,c,r,k,... dkk)!

- 'b' (blue)
- 'g' (green)
- 'r' (red)
- 'c' (cyan)
- 'm' (magenta)
- 'y' (yellow)
- 'k' (black)
- 'w' (white)

#### **5.4.1.7 Soal No. 7**

Jelaskan bagaimana cara kerja dari fungsi hist, sertakan ilustrasi dan gambar sendiri!

Cara kerja dari fungsi hist yaitu fungsi hist akan menerima parameter yang diberikan, kemudian fungsi hist akan dieksekusi sesuai dengan parameter yang diberikan.

#### **Kode Program**

```

1 import matplotlib.pyplot as plt
2 orang =
3     [11,22,16,9,10,15,22,55,62,45,21,22,34,42,42,4,2,102,95,85,55,110,120,70,65]
4 umur = [0,10,20,30,40,50,60,70,80,90,100]
5 plt.hist(orang, umur, histtype='bar', rwidth=0.8)
6 plt.xlabel('Umur')
7 plt.ylabel('Jumlah Orang')
8 plt.title('Histogram')
9 plt.show()
```

**Listing 5.12** Kode program membuat diagram menggunakan Matplotlib.

#### **Hasil Compile**

#### **5.4.1.8 Soal No. 8**

Jelaskan lebih mendalam tentang parameter dari fungsi pie diantaranya labels, colors, startangle, shadow, explode, autopct!

- labels : untuk memberikan label di tiap persentase.
- colors : untuk memberikan warna di tiap persentase.
- startangle : untuk memutar plot sesuai dengan derajat yang ditentukan.
- shadow : untuk memberikan bayangan pada plot.
- explode : untuk memisahkan antar tiap potongan pie pada plot.
- autopct : untuk menentukan jumlah angka dibelakang koma.

## 5.4.2 Plagiarisme

Berikut hasil pengecekan plagiarisme untuk bagian teori.

## 5.5 Dika Sukma Pradana 1174050

### 5.5.1 Pemahaman Teori

#### 1. Apa itu fungsi library matplotlib?

Matplotlib adalah perpustakaan merencanakan untuk bahasa pemrograman Python dan ekstensi numerik matematika NumPy. Ini menyediakan API berorientasi objek untuk menanamkan plot ke aplikasi menggunakan toolkit GUI untuk tujuan umum seperti Tkinter, wxPython, Qt, atau GTK+. Ada juga antarmuka "pylab" prosedural yang didasarkan pada mesin negara (seperti OpenGL), yang dirancang agar mirip dengan MATLAB, meskipun penggunaannya tidak disarankan. PiPy menggunakan Matplotlib.

Matplotlib awalnya ditulis oleh John D. Hunter, memiliki komunitas pengembangan aktif, dan didistribusikan di bawah lisensi gaya BSD. Michael Droettboom dinominasikan sebagai pengembang utama matplotlib sesaat sebelum kematian John Hunter pada Agustus 2012, dan selanjutnya bergabung dengan Thomas Caswell.

Pada 23 Juni 2017, matplotlib 2.0.x mendukung versi Python 2.7 hingga 3.6. Matplotlib 1.2 adalah versi pertama dari matplotlib untuk mendukung Python 3.x. Matplotlib 1.4 adalah versi terakhir dari Matplotlib untuk mendukung Python 2.6.

Matplotlib telah berjanji untuk tidak mendukung Python 2 hingga 2020 dengan menandatangani Pernyataan Python 3

#### 2. Jelaskan langkah-langkah membuat sumbu X dan Y di matplotlib! Pada Matplotlib untuk membuat sumbu X dan Y dapat dilakukan dengan cara yaitu :

```
1 import matplotlib.pyplot as plt
2
3 x = (1,2,3,4)
4 y = (4,5,6,7)
5
6 plt.plot(x,y)
7 plt.show()
```

#### 3. Jelaskan bagaimana perbedaan fungsi dan cara pakai untuk berbagai jenis(bar,histogram,scat-

- Sebuah plot sebaran/titik adalah sebuah grafik yang menunjukkan hubungan antara dua set data.

```
1 #Scatter
2 import matplotlib.pyplot as plt
```

```

3 x = [2,4,6,7,9,13,19,26,29,31,36,40,48,51,57,67,69,71,78,88]
4 y =
      [54,72,43,2,8,98,109,5,35,28,48,83,94,84,73,11,464,75,200,54]

5 plt.scatter(x,y)
6 plt.show()

```

- Sebuah histogram adalah grafik yang menampilkan frekuensi data menggunakan batang, dimana angka dikelompokkan dalam rentang tertentu. Dengan kata lain, frekuensi setiap elemen data di dalam daftar ditunjukkan menggunakan histogram. Angka yang dikelompokkan dalam bentuk rentang tertentu disebut bins.

```

1 #Histogram
2 import matplotlib.pyplot as plt
3 x =
      [2,4,6,5,42,543,5,3,73,64,42,97,63,76,63,8,73,97,23,45,56,89,45,3,23

4 num_bins = 6
5 n, bins, patches = plt.hist(x, num_bins, facecolor = 'm')
6 plt.show()

```

- pie chart adalah diagram yang digunakan untuk membandingkan antar bagian terhadap total. biasanya pie chart dalam bentuk persentase karena nilainya merupakan bagian-bagian yang dijumlahkan menjadi satu. sehingga bisa lihat kontribusi paling besar atau paling kecil dalam membentuk nilai. Pie chart digunakan untuk perbandingan yang sedikit. pie chart digunakan untuk membandingkan antar bagian terhadap total.

```

1 #Pie chart
2 import matplotlib.pyplot as plt
3
4 days = [1,2,3,4,5]
5
6 Turu =[7,8,6,11,7]
7 Mangan = [2,3,4,3,2]
8 Kuli =[7,8,7,2,2]
9 Ulin = [8,5,7,8,13]
10 slices = [7,2,2,13]
11 activities = ['turu','mangan','kuli','ulin']
12 cols = ['c','m','r','b']
13
14 plt.pie(slices,
15         labels=activities,
16         colors=cols,
17         startangle=90,
18         shadow= True,
19         explode=(0.1,0,0,0),
20         autopct='%1.1f%')
21
22 plt.title('Pie Plot')
23 plt.show()

```

- Bagan area benar-benar mirip dengan bagan garis, kecuali area antara sumbu x dan garis diisi dengan warna atau bayangan. Ini mewakili evolusi vari-

abel numerik mengikuti variabel numerik lainnya. Jika kamu ingin mewakili evolusi ini untuk beberapa grup dalam waktu yang bersamaan, Anda mungkin tertarik dengan bagan area bertumpuk, di mana setiap grup ditampilkan satu sama lain.

```

1 #area plot
2 import matplotlib.pyplot as plt
3 days = [1,2,3,4,5]
4
5 sleeping =[7,8,6,11,7]
6 eating = [2,3,4,3,2]
7 working =[7,8,7,2,2]
8 playing = [8,5,7,8,13]
9
10 plt.plot([],[],color='m', label='Turu', linewidth=5)
11 plt.plot([],[],color='c', label='makan', linewidth=5)
12 plt.plot([],[],color='r', label='kuli', linewidth=5)
13 plt.plot([],[],color='k', label='ulin', linewidth=5)
14
15 plt.stackplot(days, sleeping,eating,working,playing, colors=['m','c','r','k'])
16
17 plt.xlabel('x')
18 plt.ylabel('y')
19 plt.title('Lets Rock!!!')
20 plt.legend()
21 plt.show()
```

4. Jelaskan bagaimana cara menggunakan legend dan label serta kaitannya dengan fungsi tersebut!

Legenda adalah penjelasan garis dilengkapi dengan sampel garis yang dijelaskan. Untuk membuat legenda pada plot anda dapat menggunakan syntax fungsi legend yang dapat dituliskan sebagai berikut:

```
1 plt.legend()
```

Untuk menambah label pada garis sumbu pada grafik dapat menggunakan syntax fungsi xlabel dan fungsi ylabel pada MATLAB. Kedua label ditulis setelah syntax deklarasi plot.

```
1 labels=activities ,
```

5. Jelaskan apa fungsi dari subplot di matplotlib, dan bagaimana cara kerja dari fungsi subplot, sertakan ilustrasi dan gambar sendiri dan apa parameternya jika ingin menggambar plot dengan 9 subplot di dalamnya

Pada saat fungsi plot dijalankan, grafik atau gelombang akan ditampilkan dalam figure yang aktif. Untuk beberapa kasus, perlu menampilkan plot grafik atau gelombang dalam figure atau gambar yang berbeda maupun menampilkan lebih dari satu plot dalam satu gambar. Hal ini bisa terjadi jika menggabungkan beberapa plot dalam 1 figure.

```

1 #subplot
2 import matplotlib.pyplot as plt
3
4 plt.subplot(331)
5 x = (4,8,12,16,20)
6 y = (54, 58, 62, 66, 70)
7
8 plt.plot(x,y)
9 plt.subplot(332)
10
11 plt.plot(x,y)
12 plt.subplot(333)
13
14 plt.plot(x,y)
15 plt.subplot(334)
16
17 plt.plot(x,y)
18 plt.subplot(335)
19
20 plt.plot(x,y)
21 plt.subplot(336)
22
23 plt.plot(x,y)
24 plt.subplot(337)
25
26 plt.plot(x,y)
27 plt.subplot(338)
28
29 plt.plot(x,y)
30 plt.subplot(339)
31
32 plt.plot(x,y)
33 plt.show()

```

6. Sebutkan semua parameter color yang bisa digunakan!

Kode warna yang digunakan dalam python adalah sebagai berikut: R adalah warna Merah G adalah warna Hijau B adalah warna Biru M adalah warna Ungu Y adalah warna Kuning C adalah warna Biru Muda K adalah warna Hitam

7. Jelaskan bagaimana cara kerja dari fungsi hist, sertakan ilustrasi dan gambar sendiri!

pada fungsi histogram titik koordinat tidak boleh sama karena dalam diagram ini digunakan untuk mendata selisih dari hasil rentang nilai tertentu.

8. Jelaskan lebih mendalam tentang parameter dari fungsi pie diantaranya labels, colors, startangle, shadow, explode, autopct

- labels digunakan untuk memberi label/penjelasan dari bagian pie chart yang kita buat.
- colors digunakan untuk mewarnai pie chart diagram yang telah dibuat, membuat warna yang berbeda antar bagian.

- startangle digunakan untuk membuat diagram/chart mem-flip atau berbalik arah.
- explode digunakan untuk menonjolkan salah satu bagian dari pie chart.
- shadows digunakan untuk memberi bayangan pada pie chart yang kita buat.
- autopct digunakan untuk memberi persen dari bagian-bagian pie chart yang kita buat.

## 9. Bebas Plagiarisme

### 5.6 Ichsan Hizman Hardy (1174034)

#### 5.6.1 Teori

##### 5.6.1.1 Soal No. 1

Apa itu fungsi library matplotlib?

Matplotlib digunakan untuk memvisualisasikan data dengan lebih rapi dan indah. Matplotlib juga mempunyai plot untuk menampilkan gambar 2D ataupun 3D.

##### 5.6.1.2 Soal No. 2

Jelaskan langkah-langkah membuat sumbu X dan Y di matplotlib!

1. Pertama mengimport library.

```
1 from matplotlib import pyplot as plt
```

2. Selanjutnya hasilkan nilai untuk sumbu x dan sumbu y.

```
1 x=[1,2,3]
2 y=[5,2,4]
```

3. Selanjutnya buat fungsi untuk mem-plot diagram batang.

```
1 plt.plot(x,y)
```

4. Selanjutnya kita tampilkan plot nya.

```
1 plt.show()
```

### Kode Program

```
1 from matplotlib import pyplot as plt
2
3 x=[1,2,3]
4 y=[5,2,4]
5
6 plt.plot(x,y)
7
8 plt.show()
```

**Listing 5.13** Kode program membuat diagram menggunakan Matplotlib.

## Gambar yang dihasilkan

### 5.6.1.3 Soal No. 3

Jelaskan bagaimana perbedaan fungsi dan cara pakai untuk berbagai jenis(bar, histogram ,scatter ,line, dll) jenis plot di matplotlib!

#### 1. Bar Graph

Perbedaan bar graph dengan jenis plot yang lain adalah bar graph menggunakan bar atau batang-batang untuk membandingkan data di antara berbagai kategori.

#### Kode Program

```

1 from matplotlib import pyplot as plt
2
3 x=[1,2,3]
4 y=[5,2,4]
5
6 plt.plot(x,y)
7
8 plt.show()
```

**Listing 5.14** Kode program membuat bar graph menggunakan Matplotlib.

#### Hasil Compile

#### 2. Histogram

Perbedaan histogram dengan jenis plot yang lain adalah histogram akan membuat plot dimana plot yang dimunculkan merupakan gabungan dari beberapa data yang telah dikelompokkan.

#### Kode Program

```

1 import matplotlib.pyplot as plt
2 orang =
3     [11,22,16,9,10,15,22,55,62,45,21,22,34,42,42,4,2,102,95,85,55,110,120,
4
5 umur = [0,10,20,30,40,50,60,70,80,90,100]
6 plt.hist(orang, umur, histtype='bar', rwidth=0.8)
7 plt.xlabel('Umur')
8 plt.ylabel('Jumlah Orang')
9 plt.title('Histogram')
10 plt.show()
```

**Listing 5.15** Kode program membuat histogram menggunakan Matplotlib.

#### Hasil Compile

#### 3. Scatter Plot

Perbedaan scatter plot dengan jenis plot lain adalah scatter plot menampilkan data sebagai kumpulan titik, masing-masing memiliki nilai satu variabel yang menentukan posisi pada sumbu horizontal dan nilai variabel lain menentukan posisi pada sumbu vertikal.

#### Kode Program

```

1 import matplotlib.pyplot as plt
2 x = [1,1.5,2,2.5,3,3.5,3.6]
3 y = [7.5,8,8.5,9,9.5,10,10.5]
4
5 x1=[8,8.5,9,9.5,10,10.5,11]
6 y1=[3,3.5,3.7,4,4.5,5,5.2]
7
8 plt.scatter(x,y, label='pendapatan tinggi simpanan rendah',color=
  'r')
9 plt.scatter(x1,y1,label='pendapatan rendah simpanan tinggi',color
  ='g')
10 plt.xlabel('simpanan dalam ratusan')
11 plt.ylabel('pendapatan dalam ribuan')
12 plt.title('Scatter Plot')
13 plt.legend()
14 plt.show()

```

**Listing 5.16** Kode program membuat scatter plot menggunakan Matplotlib.

## Hasil Compile

### 4. Area Plot

Perbedaan area plot dengan jenis plot lain adalah area plot digunakan untuk melacak perubahan dari waktu ke waktu untuk dua atau lebih kelompok terkait yang membentuk satu kategori secara keseluruhan.

#### Kode Program

```

1 import matplotlib.pyplot as plt
2 hari = [1,2,3,4,5]
3
4 tidur =[7,8,6,11,7]
5 makan = [2,3,4,3,4]
6 kuliah =[7,8,7,2,5]
7 jalan_jalan = [8,5,7,8,13]
8
9 plt.plot([],[],color='c', label='Tidur', linewidth=5)
10 plt.plot([],[],color='m', label='Makan', linewidth=5)
11 plt.plot([],[],color='y', label='Kuliah', linewidth=5)
12 plt.plot([],[],color='k', label='Jalan-jalan', linewidth=5)
13
14 plt.stackplot(hari,tidur,makan,kuliah,jalan_jalan, colors=['c','m',
  'y','k'])
15
16 plt.xlabel('x')
17 plt.ylabel('y')
18 plt.title('Kegiatan Sehari-hari')
19 plt.legend()
20 plt.show()

```

**Listing 5.17** Kode program membuat diagram menggunakan Matplotlib.

## Hasil Compile

### 5. Pie Plot

Perbedaan pie plot dengan jenis plot lain adalah pie plot digunakan untuk menunjukkan persentase atau data proporsional di mana setiap potongan pie mewakili kategori.

### Kode Program

```

1
2
3
4
5
6
7 import matplotlib.pyplot as plt
8
9 potong = [7,2,2,12]
10 kegiatan = ['Tidur', 'Makan', 'Kuliah', 'Jalan-jalan']
11 kolom = ['c', 'm', 'y', 'g']
12
13 plt.pie(potong,
14         labels=kegiatan,
15         colors=kolom,
16         startangle=90,
17         shadow=True,
18         explode=(0,0,0.2,0),
19         autopct='%.1f%%')
20
21 plt.title('Kegiatan Sehari-hari')
22 plt.show()

```

**Listing 5.18** Kode program membuat Pie Plot menggunakan Matplotlib.

### Hasil Compile

## 6. Line Graph

Perbedaan line graph dengan jenis plot lain adalah line graph menampilkan diagram dalam bentuk garis.

### Kode Program

```

1 from matplotlib import pyplot as plt
2
3 y = [4000,6000,10000,13000,14000,17000]
4 x = [2014,2015,2016,2017,2018,2019]
5 plt.plot(x,y)
6 plt.title('Pemakai Sistem Operasi Linux')
7 plt.ylabel('Tahun')
8 plt.xlabel('Jumlah Pemakai Sistem Operasi Linux')
9 plt.show()

```

**Listing 5.19** Kode program membuat diagram menggunakan Matplotlib.

### Hasil Compile

#### 5.6.1.4 Soal No. 4

Jelaskan bagaimana cara menggunakan legend dan label serta kaitannya dengan fungsi tersebut!

**Legend** Legend adalah penjelasan garis dilengkapi dengan sampel garis yang dijelaskan. Untuk membuat legenda pada plot anda dapat menggunakan syntax fungsi legend pada MATLAB.

**Label** Untuk menambah label pada garis sumbu pada grafik dapat menggunakan syntax fungsi xlabel dan fungsi ylabel pada MATLAB. Kedua label ditulis setelah syntax deklarasi plot.

#### 5.6.1.5 Soal No. 5

Jelaskan apa fungsi dari subplot di matplotlib, dan bagaimana cara kerja dari fungsi subplot, sertakan ilustrasi dan gambar sendiri dan apa parameternya jika ingin menggabungkan plot dengan 9 subplot di dalamnya!

Fungsi subplot adalah untuk membuat beberapa plot di dalam satu gambar.

Cara kerja subplot, yaitu fungsi subplot memiliki parameter pertama adalah jumlah kolom, parameter kedua adalah jumlah baris, dan parameter ketiga adalah index plot keberapanya.

#### Kode Program

```

1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 t = np.arange(0.0, 9.0, 1)
5 s = [1,2,3,4,5,6,7,8,9]
6
7 for i in range(1, 10):
8     plt.subplot(3,3,i)
9     plt.xticks([]), plt.yticks([])
10    plt.title('subplot(1,2,'+str(i)+')')
11    plt.plot(t,s,'-y')
12
13 plt.show()
```

**Listing 5.20** Kode program membuat subplot menggunakan Matplotlib.

#### Hasil Compile

#### 5.6.1.6 Soal No. 6

Sebutkan semua parameter color yang bisa digunakan (contoh: m,c,r,k,... dkk)!

- 'b' (blue)
- 'g' (green)
- 'r' (red)
- 'c' (cyan)
- 'm' (magenta)
- 'y' (yellow)

- 'k' (black)
- 'w' (white)

#### **5.6.1.7 Soal No. 7**

Jelaskan bagaimana cara kerja dari fungsi hist, sertakan ilustrasi dan gambar sendiri!

Cara kerja dari fungsi hist yaitu fungsi hist akan menerima parameter yang diberikan, kemudian fungsi hist akan dieksekusi sesuai dengan parameter yang diberikan.

#### **Kode Program**

```

1 import matplotlib.pyplot as plt
2 orang =
3     [11,22,16,9,10,15,22,55,62,45,21,22,34,42,42,4,2,102,95,85,55,110,120,70,65,
4 umur = [0,10,20,30,40,50,60,70,80,90,100]
5 plt.hist(orang, umur, histtype='bar', rwidth=0.8)
6 plt.xlabel('Umur')
7 plt.ylabel('Jumlah Orang')
8 plt.title('Histogram')
9 plt.show()
```

**Listing 5.21** Kode program membuat diagram menggunakan Matplotlib.

#### **Hasil Compile**

#### **5.6.1.8 Soal No. 8**

Jelaskan lebih mendalam tentang parameter dari fungsi pie diantaranya labels, colors, startangle, shadow, explode, autopct!

- labels : untuk memberikan label di tiap persentase.
- colors : untuk memberikan warna di tiap persentase.
- startangle : untuk memutar plot sesuai dengan derajat yang ditentukan.
- shadow : untuk memberikan bayangan pada plot.
- explode : untuk memisahkan antar tiap potongan pie pada plot.
- autopct : untuk menentukan jumlah angka dibelakang koma.

#### **5.6.2 Praktek**

##### **5.6.2.1 Soal No. 1**

Buatlah librari fungsi (file terpisah/library dengan nama NPMbar.py) untuk plot dengan jumlah subplot adalah  $NPM \bmod 3 + 2$ !

#### **Kode Program**

```

1 from matplotlib import pyplot as plt
2
3 def bar():
4
5     hasil = 1174006 % 3 + 2
6
7     for i in range(1, hasil+1):
8         plt.subplot(2,2,i)
9         plt.bar
([2012.7,2013.7,2014.7,2015.7,2016.7,2017.7],[9000,9500,10000,15000,20000,25000],
10
11         label="Mac",color='b',width=.3)
12         plt.bar
([2013,2014,2015,2016,2017,2018],[20000,25000,30000,35000,40000,45000],
13
14         label="Windows",color='r',width=.3)
15         plt.bar
([2013.3,2014.3,2015.3,2016.3,2017.3,2018.3],[2000,2500,3000,3500,4000,4500],
16
17         label="Linux",color='g',width=.3)
18         plt.legend()
19         plt.xlabel('Tahun')
20         plt.ylabel('Jumlah Pengguna')
21         plt.title('Pengguna OS Dari Tahun ke Tahun')
22         plt.subplots_adjust(wspace=1, hspace=.7)
23
24 plt.show()

```

**Listing 5.22** Kode program membuat fungsi Bar Plot menggunakan Matplotlib.

## Hasil Compile

### 5.6.2.2 Soal No. 2

Buatlah librari fungsi (file terpisah/library dengan nama NPMscatter.py) untuk plot dengan jumlah subplot NPM mod 3 + 2!

## Kode Program

```

1 from matplotlib import pyplot as plt
2
3 def scatter():
4
5     hasil = 1174006 % 3 + 2
6
7     x = [1,1.5,2,2.5,3,3.5,3.6]
8     y = [7.5,8,8.5,9,9.5,10,10.5]
9
10    x1=[8,8.5,9,9.5,10,10.5,11]
11    y1=[3,3.5,3.7,4,4.5,5,5.2]
12
13    for i in range(1, hasil+1):
14        plt.subplot(2,2,i)
15        plt.scatter(x,y, label='pendapatan tinggi simpanan rendah',
16                     color='r')

```

```

16     plt.scatter(x1,y1,label='pendapatan rendah simpanan tinggi',
17     color='g')
18     plt.xlabel('simpanan dalam ratusan')
19     plt.ylabel('pendapatan dalam ribuan')
20     plt.title('Scatter Plot')
21     plt.legend()
22     plt.subplots_adjust(wspace=1.1, hspace=.7)
23
plt.show()

```

**Listing 5.23** Kode program membuat fungsi Scatter Plot menggunakan Matplotlib.

## Hasil Compile

### 5.6.2.3 Soal No. 3

Buatlah librari fungsi (file terpisah/library dengan nama NPMpie.py) untuk plot dengan jumlah subplot NPM mod 3 + 2!

## Kode Program

```

1 from matplotlib import pyplot as plt
2
3 def pie():
4
5     hasil = 1174006 % 3 + 2
6
7     potong = [7,4,2,12]
8     kegiatan = ['Tidur','Makan','Kuliah','Jalan-jalan']
9     kolom = ['c','m','y','g']
10
11    for i in range(1, hasil+1):
12        plt.subplot(2,2,i)
13        plt.pie(potong,
14                 labels=kegiatan,
15                 colors=kolom,
16                 startangle=90,
17                 shadow=True,
18                 explode=(0,0,0.2,0),
19                 autopct='%1.1f%%')
20        plt.title('Kegiatan Sehari-hari')
21        plt.subplots_adjust(hspace=.4)
22
23
plt.show()

```

**Listing 5.24** Kode program membuat fungsi Pie Plot menggunakan Matplotlib.

## Hasil Compile

### 5.6.2.4 Soal No. 4

Buatlah librari fungsi (file terpisah/library dengan nama NPMplot.py) untuk plot dengan jumlah subplot NPM mod 3 + 2

## Kode Program

```

1 from matplotlib import pyplot as plt
2
3 def plot():
4
5     hasil = 1174006 % 3 + 2
6
7     x = [2014,2015,2016,2017,2018,2019]
8     y = [76,87,105,122,148,170]
9     x2 = [2014,2015,2016,2017,2018,2019]
10    y2 = [78,97,114,134,146,167]
11
12    for i in range(1, hasil+1):
13        plt.subplot(2,2,i)
14        plt.plot(x,y,'b',label='Team Captain America', linewidth=1)
15        plt.plot(x2,y2,'r',label='Team Iron Man', linewidth=1)
16        plt.title('Civil Wars')
17        plt.ylabel('Jumlah Pendukung')
18        plt.xlabel('Tahun')
19        plt.legend()
20        plt.grid(True, color='k')
21        plt.subplots_adjust(wspace=.4, hspace=.7)
22
23    plt.show()

```

**Listing 5.25** Kode program membuat fungsi Plot menggunakan Matplotlib.

## Hasil Compile

### 5.6.3 Penanganan Error

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

Peringatan error di praktek kelima ini, yaitu:

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

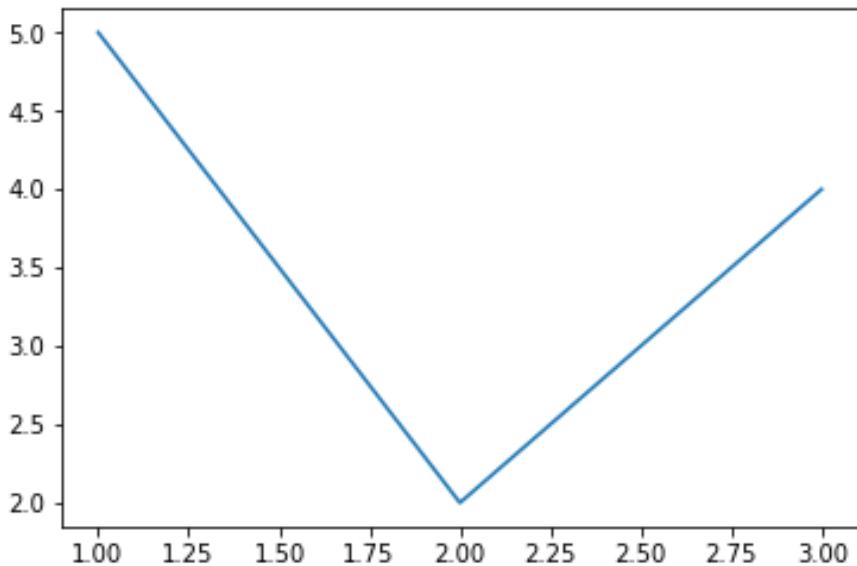
Fungsi yang menggunakan try except untuk menanggulangi error.

## Kode Program

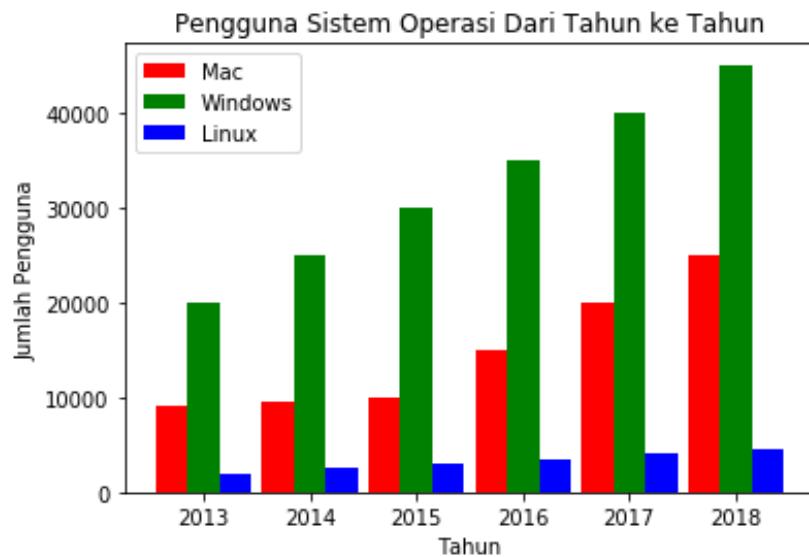
```
1 from matplotlib import pyplot as plt
2
3 def tryExceptError():
4     try:
5         a=[1,2,3]
6         y=[5,2,4]
7         plt.plot(x,y)
8         plt.show()
9     except SyntaxError:
10        print("Kesalahan penulisan syntax")
11    except NameError:
12        print("Variable tersebut tidak ada")
13    except TypeError:
14        print("Tipe data salah")
15    except:
16        print("Terjadi sebuah kesalahan")
17
18 tryExceptError()
```

**Listing 5.26** Kode program membuat fungsi penanganan error.

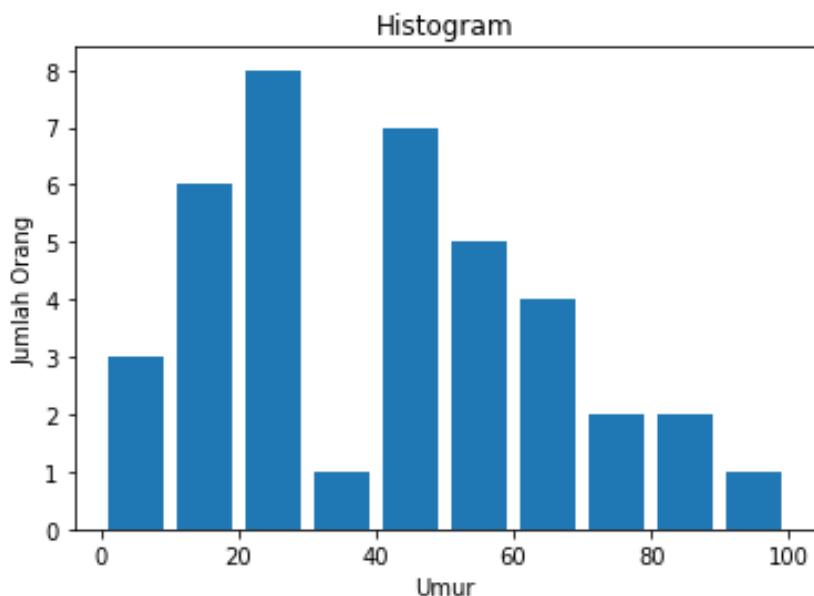
## Hasil Compile



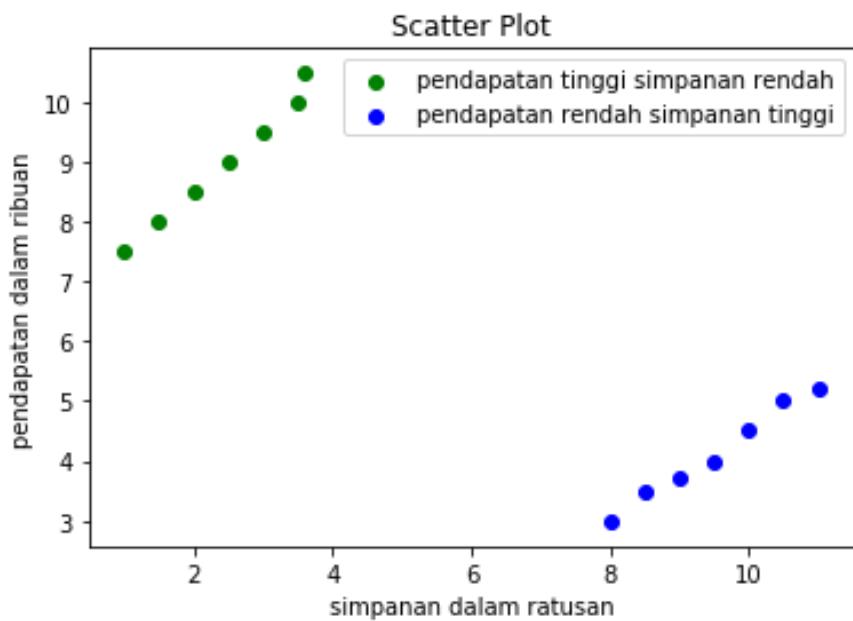
**Gambar 5.15** Hasil compile membuat diagram menggunakan Matplotlib.



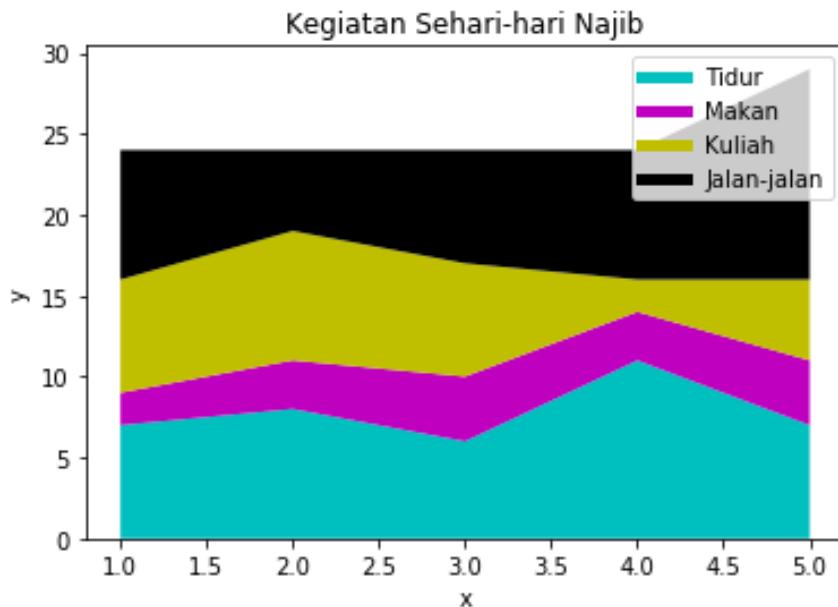
**Gambar 5.16** Hasil compile membuat bar graph menggunakan Matplotlib.



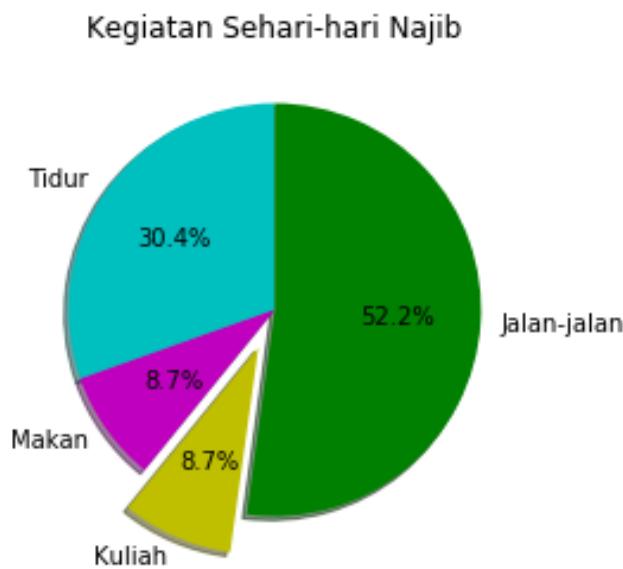
**Gambar 5.17** Hasil compile membuat histogram menggunakan Matplotlib.



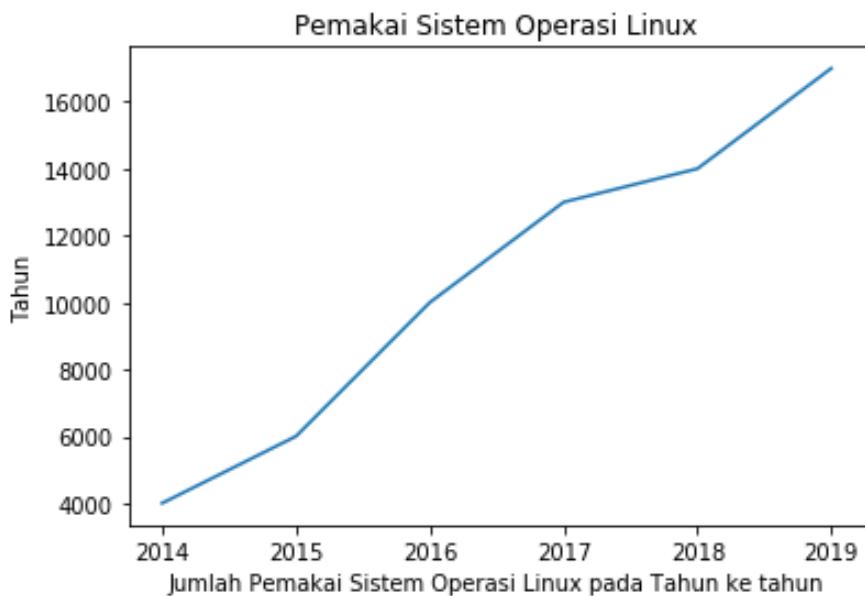
**Gambar 5.18** Hasil compile membuat scatter plot menggunakan Matplotlib.



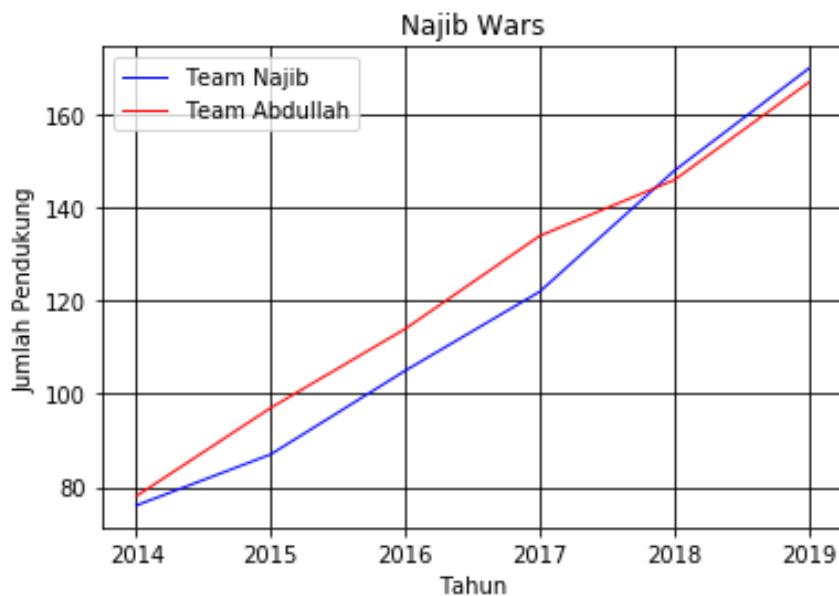
Gambar 5.19 Hasil compile membuat diagram menggunakan Matplotlib.



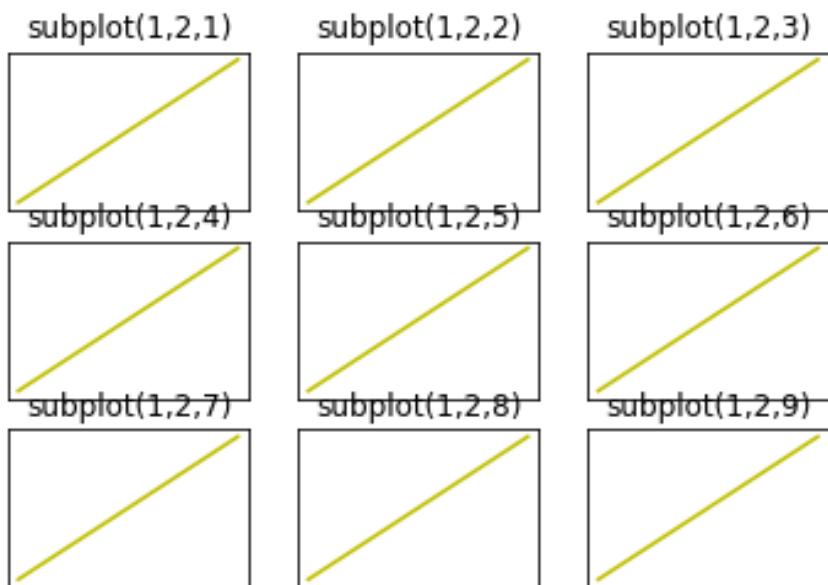
Gambar 5.20 Hasil compile membuat Pie Plot menggunakan Matplotlib.



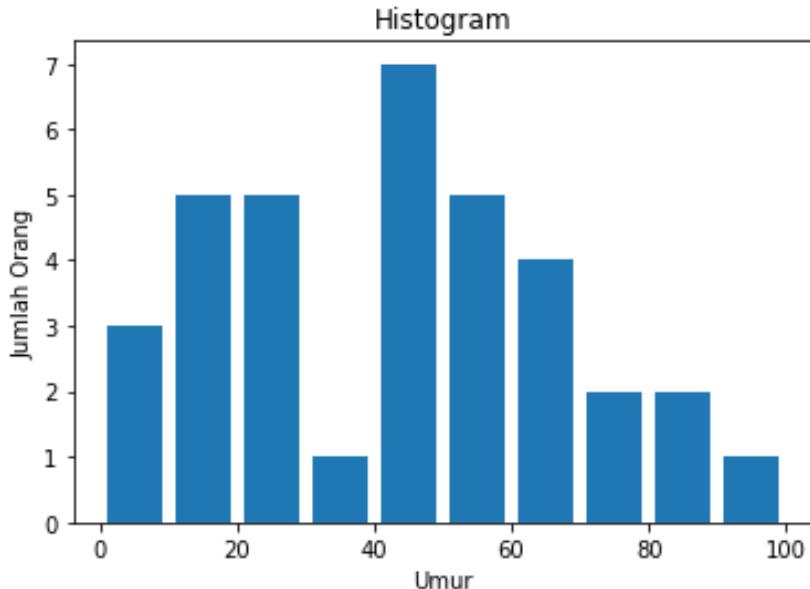
**Gambar 5.21** Hasil compile membuat diagram menggunakan Matplotlib.



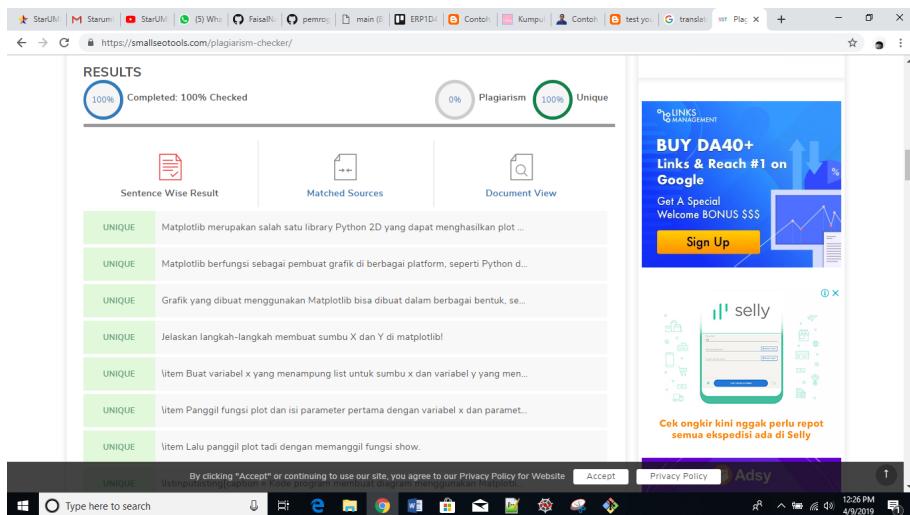
**Gambar 5.22** Hasil compile membuat diagram menggunakan Matplotlib.



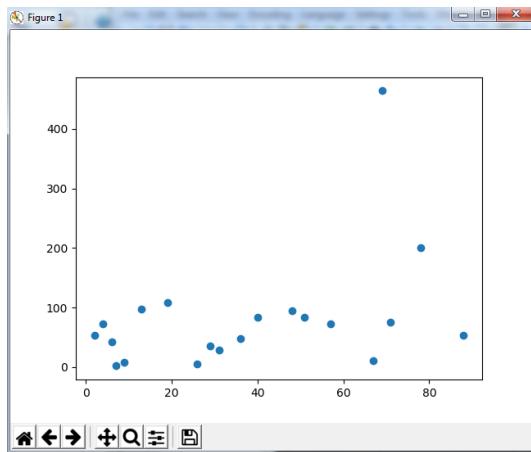
**Gambar 5.23** Hasil compile membuat subplot menggunakan Matplotlib.



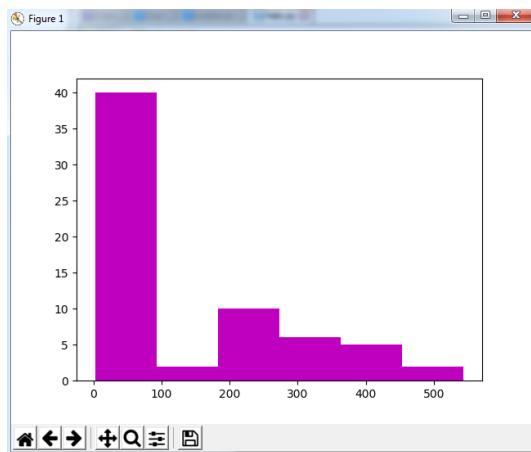
**Gambar 5.24** Hasil compile membuat diagram menggunakan Matplotlib.



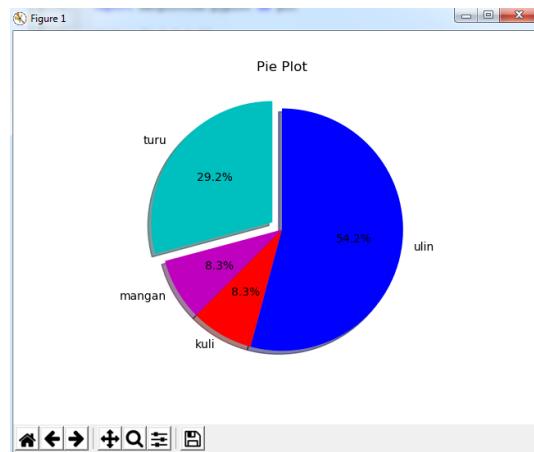
**Gambar 5.25** Plagiarisme



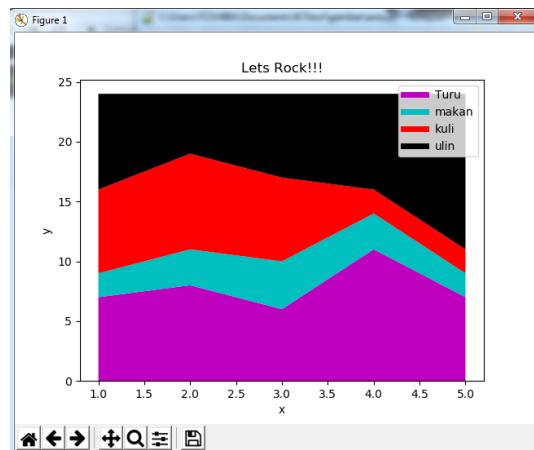
Gambar 5.26 Scatter



Gambar 5.27 Histogram



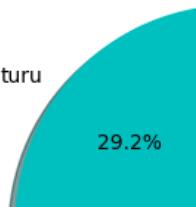
Gambar 5.28 pie chart



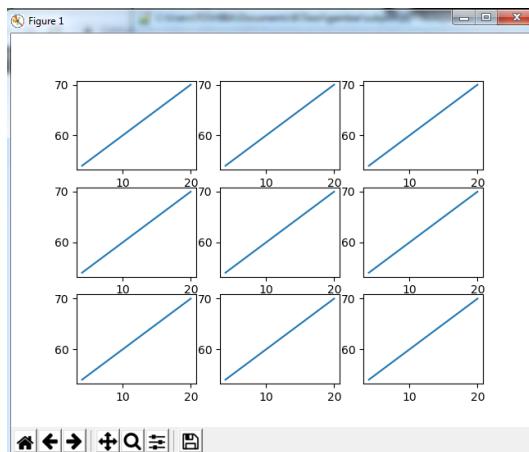
Gambar 5.29 bagan area



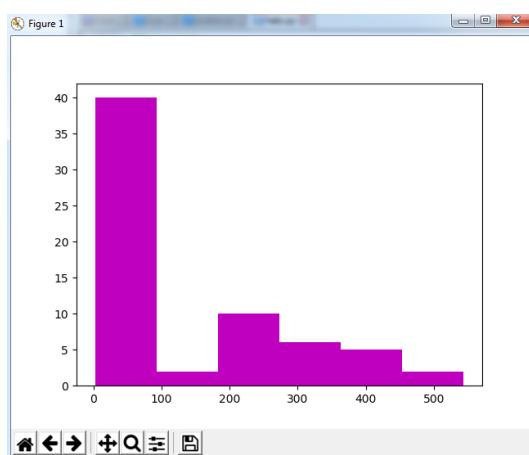
Gambar 5.30 legend



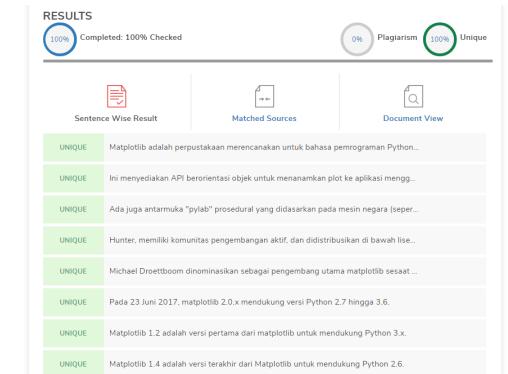
**Gambar 5.31** label



**Gambar 5.32** Subplot

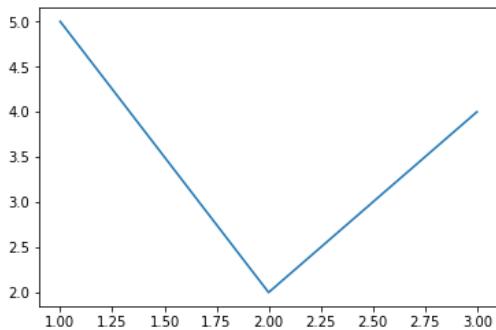


**Gambar 5.33** Histogram

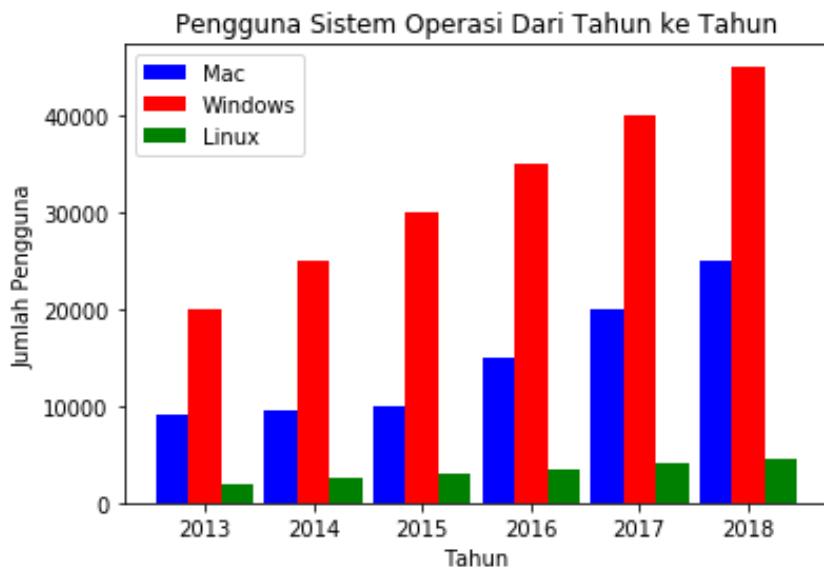


**Gambar 5.34** Screenshoot bebas plagiarisme

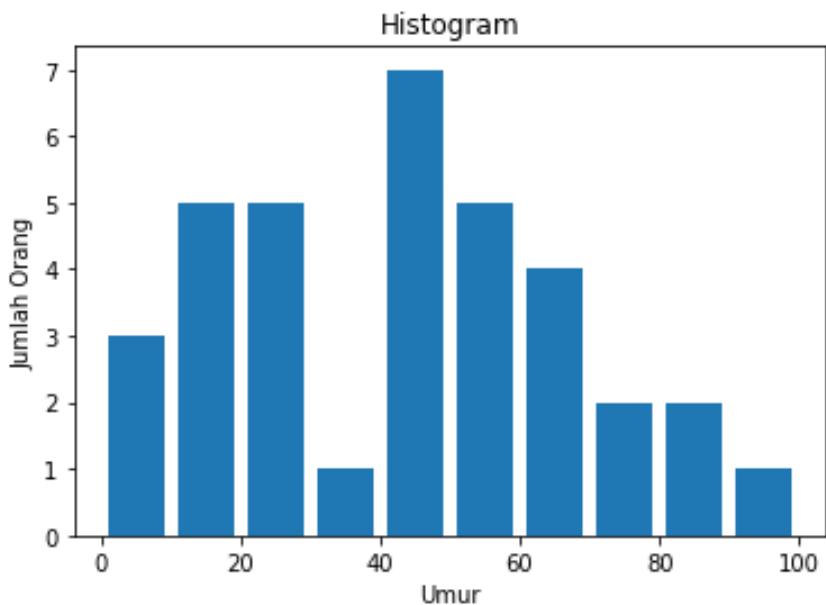
```
In [2]: runfile('D:/KULIAH/SEMESTER 4/PEMROGRAMAN 3/untukchap6/1174034.py'  
KULIAH/SEMESTER 4/PEMROGRAMAN 3/untukchap6')
```



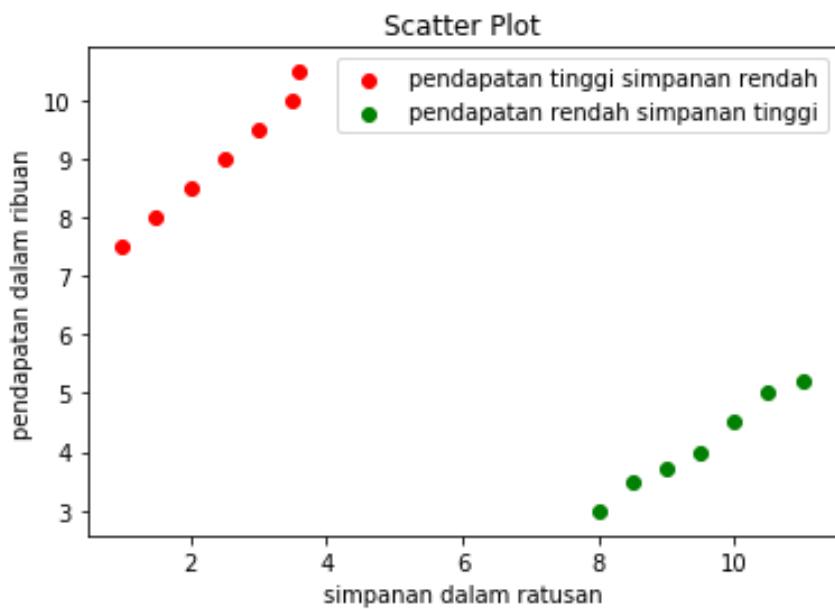
**Gambar 5.35** Diagram Batang



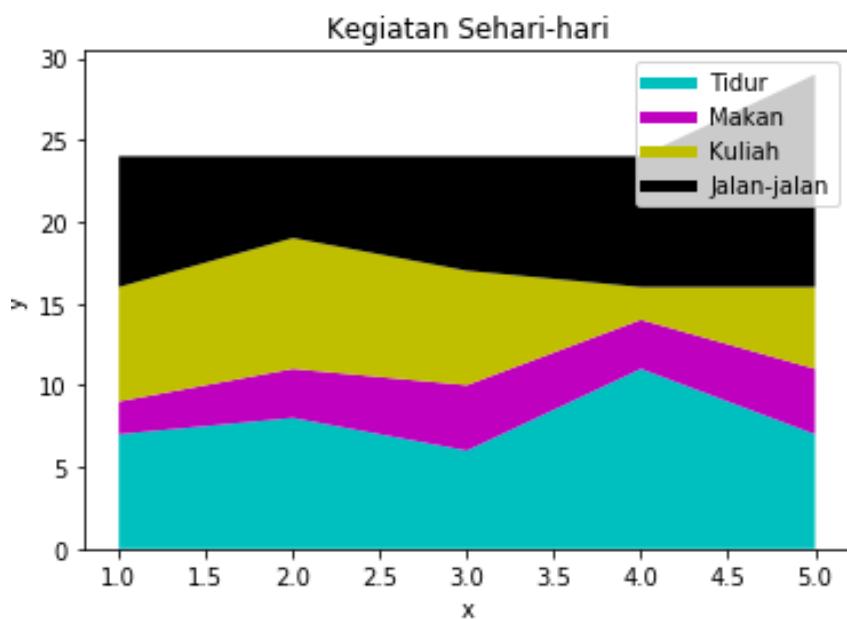
**Gambar 5.36** Hasil compile membuat bar graph menggunakan Matplotlib.



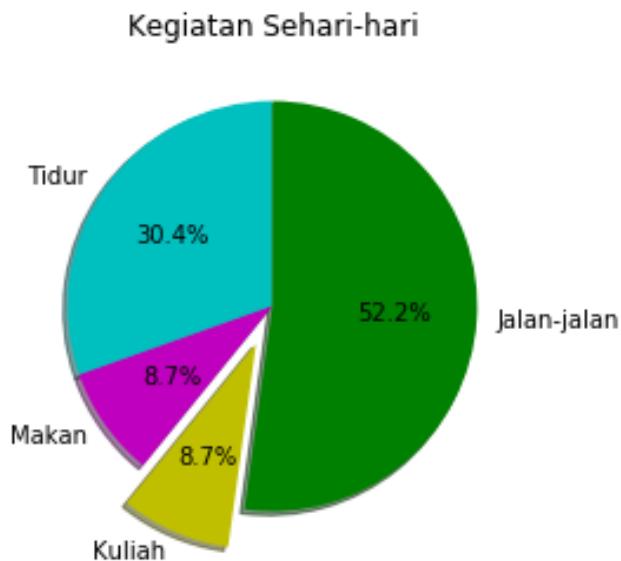
**Gambar 5.37** Hasil compile membuat histogram menggunakan Matplotlib.



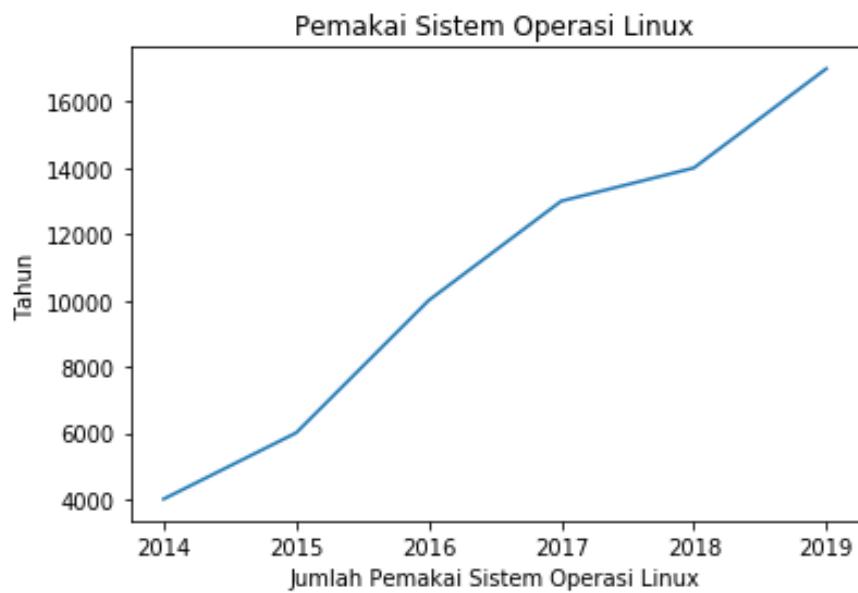
**Gambar 5.38** Hasil compile membuat scatter plot menggunakan Matplotlib.



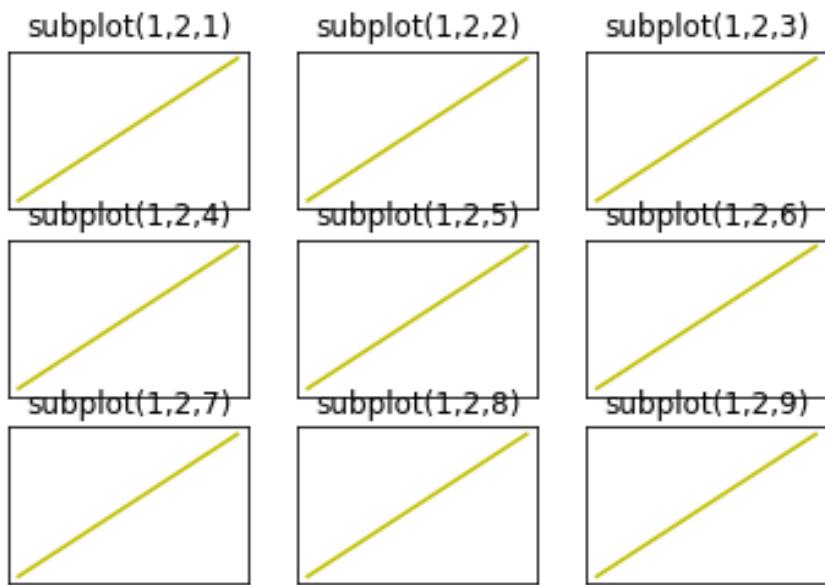
**Gambar 5.39** Hasil compile membuat diagram menggunakan Matplotlib.



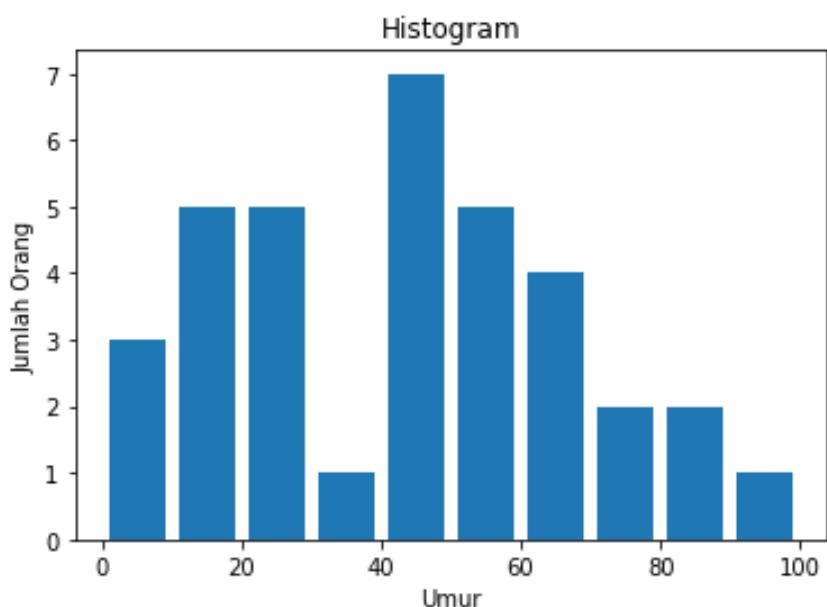
**Gambar 5.40** Hasil compile membuat Pie Plot menggunakan Matplotlib.



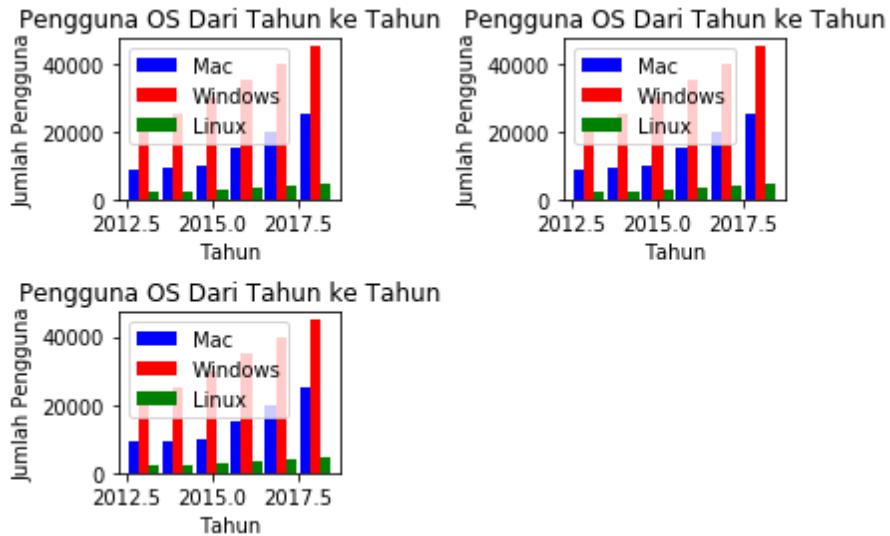
**Gambar 5.41** Hasil compile membuat diagram menggunakan Matplotlib.



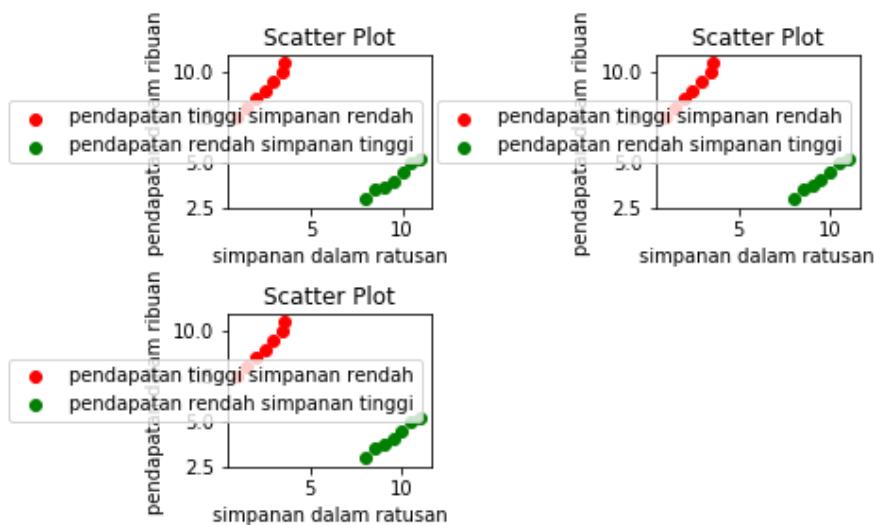
**Gambar 5.42** Hasil compile membuat subplot menggunakan Matplotlib.



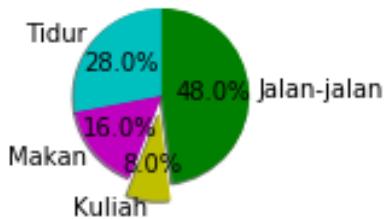
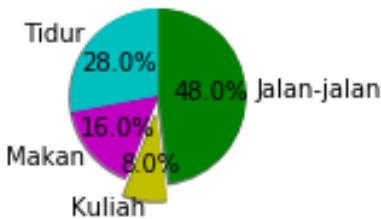
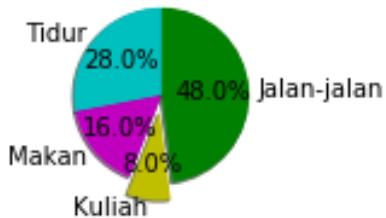
**Gambar 5.43** Hasil compile membuat diagram menggunakan Matplotlib.

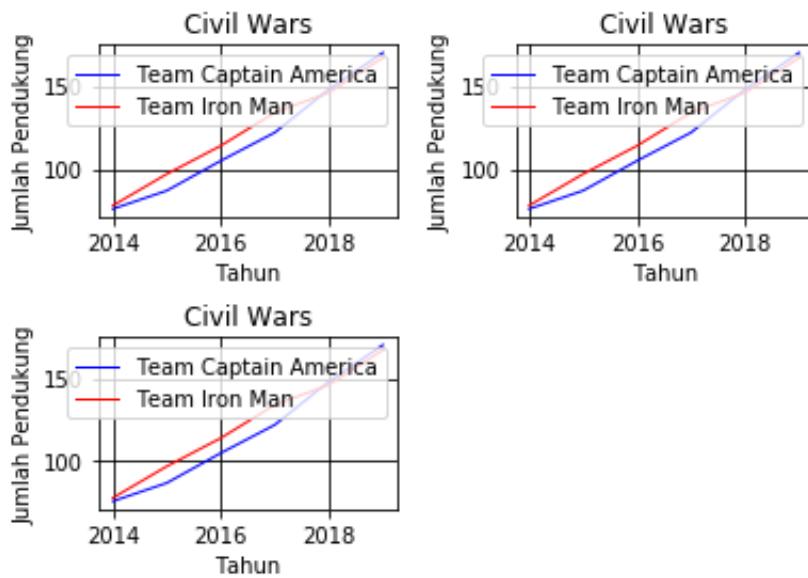


**Gambar 5.44** Hasil compile membuat fungsi Bar Plot menggunakan Matplotlib.



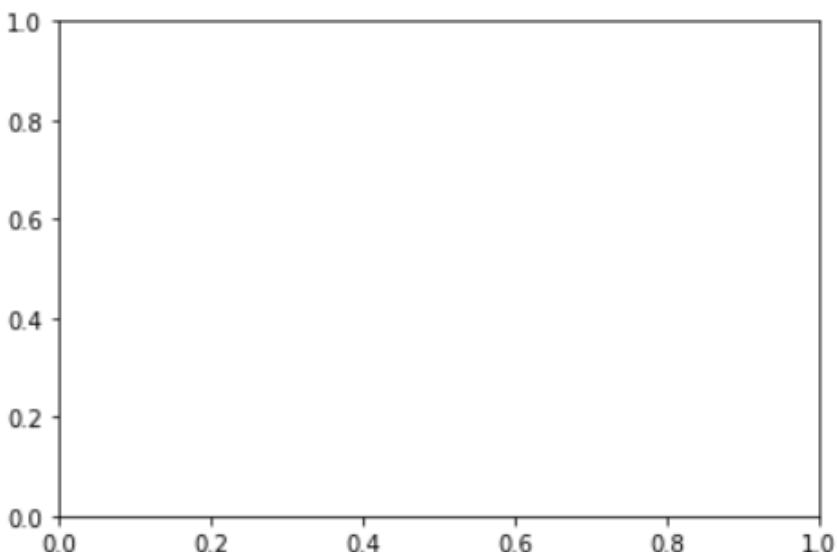
**Gambar 5.45** Hasil compile membuat fungsi Scatter Plot menggunakan Matplotlib.

**Kegiatan Sehari-hari****Kegiatan Sehari-hari****Kegiatan Sehari-hari****Gambar 5.46** Hasil compile membuat fungsi Pie Plot menggunakan Matplotlib.



Gambar 5.47 Hasil compile membuat fungsi Plot menggunakan Matplotlib.

Terjadi sebuah kesalahan



**Gambar 5.48** Hasil compile membuat fungsi penanganan error.

# BAB 6

---

## PRAKTEK LIBRARY CSV DAN PANDAS

---

### 6.1 Hagan Rowlenstino/1174040

#### 6.1.1 Soal 1

Bar :

```
1 from matplotlib import pyplot as plt
2
3 def bar():
4
5     mod = 1174040 % 3 + 2
6     x = [2012,2013,2014,2015,2016,2017]
7     y = [1000,2000,3000,4000,5000,6000]
8     for i in range(1, mod+1):
9         plt.subplot(2,3,i)
10        plt.bar(x,y)
11        plt.bar(x,y)
12        plt.xlabel('Tahun')
13        plt.ylabel('Jumlah Produksi')
14        plt.title('Jumlah Produksi Per Tahun')
15        plt.subplots_adjust(wspace=1, hspace=.7)
```

```

17
18     plt.show()

```

### 6.1.2 Soal 2

Scatter :

```

1 from matplotlib import pyplot as plt
2
3 def scatter():
4
5     mod = 1174040 % 3 + 2
6
7     x = (4,8,13,17,20)
8     y = (54, 67, 98, 78, 45)
9
10
11    for i in range(1, mod+1):
12        plt.subplot(2,3,i)
13        plt.scatter(x,y,)
14        plt.title('Scatter Plot')
15        plt.subplots_adjust(wspace=1.1, hspace=.7)
16
17    plt.show()

```

### 6.1.3 Soal 3

Pie :

```

1 from matplotlib import pyplot as plt
2
3 def pie():
4
5     mod = 1174040 % 3 + 2
6
7     x = [7,7,7,12]
8     kegiatan = ['Tidur', 'Makan', 'Nonton', 'Main Game']
9
10
11    for i in range(1, mod+1):
12        plt.subplot(2,3,i)
13        plt.pie(x,
14                 labels=kegiatan ,
15                 startangle=90,
16                 shadow= True ,
17                 explode=(0,0,0,0.1) ,
18                 autopct='%1.1f%%')
19        plt.title('Kegiatan Sehari-hari')
20        plt.subplots_adjust(hspace=.7)
21
22    plt.show()

```

### 6.1.4 Soal 4

Plot :

```
1 from matplotlib import pyplot as plt
2
3 def plot():
4
5     mod = 1174040 % 3 + 2
6
7     x = (4,8,13,17,20)
8     y = (54, 67, 98, 78, 45)
9
10    for i in range(1, mod+1):
11        plt.subplot(2,3,i)
12        plt.plot(x,y, 'b', linewidth=1)
13        plt.title('PLOT')
14    plt.subplots_adjust(wspace=.4, hspace=.7)
15
16 plt.show()
```

### 6.1.5 Penanganan Error

Error yang ditemukan yaitu ValueError. cara penanggulangan nya yatu dengan mengecek kembali panjang dan lebar subplotnya agar tidak kurang dari urutan yang kita buat.

```
1 from matplotlib import pyplot as plt
2
3 def err():
4
5     mod = 1174040 % 3 + 2
6
7     x = (4,8,13,17,20)
8     y = (54, 67, 98, 78, 45)
9     try:
10         for i in range(1, mod+1):
11             plt.subplot(1,3,i)
12             plt.plot(x,y, 'b', linewidth=1)
13             plt.title('PLOT')
14         plt.subplots_adjust(wspace=.4, hspace=.7)
15
16         plt.show()
17     except ValueError:
18         print(" Jumlah subplot tidak sama dengan perhitungan mod. panjang ataupun lebar subplot tidak boleh lebih sedikit dari urutannya")
```

## 6.2 Faisal Najib Abdullah 1174042

### 6.2.1 Praktek

#### 6.2.1.1 Soal No. 1

Buatlah librari fungsi (file terpisah/library dengan nama NPMbar.py) untuk plot dengan jumlah subplot adalah NPM mod 3 + 2!

#### Kode Program

```

1 from matplotlib import pyplot as plt
2
3 def bar():
4
5     hasil = 1174042 % 3 + 2
6
7     for i in range(1, hasil+1):
8         plt.subplot(2,2,i)
9         plt.bar
([2012.7,2013.7,2014.7,2015.7,2016.7,2017.7],[9000,9500,10000,15000,20000,25000],
10          label="Mac",color='b',width=.3)
11         plt.bar
([2013,2014,2015,2016,2017,2018],[20000,25000,30000,35000,40000,45000],
12          label="Windows",color='r',width=.3)
13         plt.bar
([2013.3,2014.3,2015.3,2016.3,2017.3,2018.3],[2000,2500,3000,3500,4000,4500],
14          label="Linux",color='g',width=.3)
15         plt.legend()
16         plt.xlabel('Tahun')
17         plt.ylabel('Jumlah Pengguna')
18         plt.title('Pengguna OS Dari Tahun ke Tahun')
19         plt.subplots_adjust(wspace=1, hspace=.7)
20
21     plt.show()
```

**Listing 6.1** Kode program membuat fungsi Bar Plot menggunakan Matplotlib.

#### Hasil Compile

#### 6.2.1.2 Soal No. 2

Buatlah librari fungsi (file terpisah/library dengan nama NPMscatter.py) untuk plot dengan jumlah subplot NPM mod 3 + 2!

#### Kode Program

```

1 from matplotlib import pyplot as plt
2
3 def scatter():
4
5     hasil = 1174042 % 3 + 2
```

```

6      x = [1,1.5,2,2.5,3,3.5,3.6]
7      y = [7.5,8,8.5,9,9.5,10,10.5]
8
9
10     x1=[8,8.5,9,9.5,10,10.5,11]
11     y1=[3,3.5,3.7,4,4.5,5,5.2]
12
13     for i in range(1, hasil+1):
14         plt.subplot(2,2,i)
15         plt.scatter(x,y, label='pendapatan tinggi simpanan rendah',
16 color='r')
17         plt.scatter(x1,y1,label='pendapatan rendah simpanan tinggi',
18 color='g')
19         plt.xlabel('simpanan dalam ratusan')
20         plt.ylabel('pendapatan dalam ribuan')
21         plt.title('Scatter Plot')
22         plt.legend()
23         plt.subplots_adjust(wspace=1.1, hspace=.7)

24     plt.show()

```

**Listing 6.2** Kode program membuat fungsi Scatter Plot menggunakan Matplotlib.

## Hasil Compile

### 6.2.1.3 Soal No. 3

Buatlah librari fungsi (file terpisah/library dengan nama NPMpie.py) untuk plot dengan jumlah subplot NPM mod 3 + 2!

## Kode Program

```

1 from matplotlib import pyplot as plt
2
3 def pie():
4
5     hasil = 1174042 % 3 + 2
6
7     potong = [7,4,2,12]
8     kegiatan = ['Tidur','Makan','Kuliah','Jalan-jalan']
9     kolom = ['c','m','y','g']
10
11    for i in range(1, hasil+1):
12        plt.subplot(2,2,i)
13        plt.pie(potong,
14                 labels=kegiatan,
15                 colors=kolom,
16                 startangle=90,
17                 shadow=True,
18                 explode=(0,0,0.2,0),
19                 autopct='%1.1f%%')
20        plt.title('Kegiatan Sehari-hari')
21        plt.subplots_adjust(hspace=.4)
22

```

```
23 plt.show()
```

**Listing 6.3** Kode program membuat fungsi Pie Plot menggunakan Matplotlib.

## Hasil Compile

### 6.2.1.4 Soal No. 4

Buatlah librari fungsi (file terpisah/library dengan nama NPMplot.py) untuk plot dengan jumlah subplot NPM mod 3 + 2

## Kode Program

```
1 from matplotlib import pyplot as plt
2
3 def plot():
4
5     hasil = 1174042 % 3 + 2
6
7     x = [2014,2015,2016,2017,2018,2019]
8     y = [76,87,105,122,148,170]
9     x2 = [2014,2015,2016,2017,2018,2019]
10    y2 = [78,97,114,134,146,167]
11
12    for i in range(1, hasil+1):
13        plt.subplot(2,2,i)
14        plt.plot(x,y,'b',label='Team Najib', linewidth=1)
15        plt.plot(x2,y2,'r',label='Team Abdullah', linewidth=1)
16        plt.title('Najib Wars')
17        plt.ylabel('Jumlah Pendukung')
18        plt.xlabel('Tahun')
19        plt.legend()
20        plt.grid(True, color='k')
21        plt.subplots_adjust(wspace=.4, hspace=.7)
22
23    plt.show()
```

**Listing 6.4** Kode program membuat fungsi Plot menggunakan Matplotlib.

## Hasil Compile

### 6.2.2 Penanganan Error

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

Peringatan error di praktek kelima ini, yaitu:

- Syntax Errors Syntax Errors adalah suatu keadaan saat kode python mengalami kesalahan penulisan. Solusinya adalah memperbaiki penulisan kode yang salah.

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

Fungsi yang menggunakan try except untuk menanggulangi error.

### Kode Program

```

1 from matplotlib import pyplot as plt
2
3 def tryExceptError():
4     try:
5         a=[1,2,3]
6         y=[5,2,4]
7         plt.plot(x,y)
8         plt.show()
9     except SyntaxError:
10        print("Kesalahan penulisan syntax")
11    except NameError:
12        print("Variable tersebut tidak ada")
13    except TypeError:
14        print("Tipe data salah")
15    except:
16        print("Terjadi sebuah kesalahan")
17
18 tryExceptError()
```

**Listing 6.5** Kode program membuat fungsi penanganan error.

### Hasil Compile

## 6.3 Irvan Rizkiansyah/1174043

### 6.3.1 Nomor 1

Histogram :

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Wed Apr 10 13:48:08 2019
4
5 @author: Irvan
6 """
```

```

8 import matplotlib.pyplot as plt
9
10 print('Jumlah Subplot Sesuai NPM mod 3 + 2 =', 1174043%3+2)
11
12 x =
13     [2,4,6,5,42,543,5,3,73,64,42,97,63,76,63,8,73,97,23,45,56,89,45,3,23,2,5,78
14 num_bins = 6
15 n, bins, patches = plt.hist(x, num_bins, facecolor = 'green')
16
17 plt.subplot(2,2,1)
18 plt.title('subplot(2,2,1)')
19 plt.hist(x, num_bins, facecolor = 'green')
20
21 plt.subplot(2,2,2)
22 plt.title('subplot(2,2,2)')
23 plt.hist(x, num_bins, facecolor = 'green')
24
25 plt.subplot(2,2,3)
26 plt.title('subplot(2,2,3)')
27 plt.hist(x, num_bins, facecolor = 'green')
28
29 plt.subplot(2,2,4)
30 plt.title('subplot(2,2,4)')
31 plt.hist(x, num_bins, facecolor = 'green')

```

### 6.3.2 Nomor 2

Scatter :

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Wed Apr 10 13:57:31 2019
4
5 @author: Irvan
6 """
7
8 import matplotlib.pyplot as plt
9
10 print('Jumlah Subplot Sesuai NPM mod 3 + 2 =', 1174043%3+2)
11
12 x = [4,8,13,20,23]
13 y = [54,67,99,78,80]
14
15 plt.subplot(2,2,1)
16 plt.title('subplot(2,2,1)')
17 plt.scatter(x,y)
18
19 plt.subplot(2,2,2)
20 plt.title('subplot(2,2,2)')
21 plt.scatter(x,y)
22
23 plt.subplot(2,2,3)
24 plt.title('subplot(2,2,3)')
25 plt.scatter(x,y)

```

```
26
27 plt.subplot(2,2,4)
28 plt.title('subplot(2,2,4)')
29 plt.scatter(x,y)
```

### 6.3.3 Nomor 3

Pie :

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Wed Apr 10 14:01:09 2019
4
5 @author: Irvan
6 """
7
8 import matplotlib.pyplot as plt
9
10 print('Jumlah Subplot Sesuai NPM mod 3 + 2 =', 1174043%3+2)
11
12 labels = 'test1', 'test2', 'test3', 'test4'
13 sizes = [25,35,20,20]
14
15 plt.subplot(2,2,1)
16 plt.title('subplot(2,2,1)')
17 plt.pie(sizes, labels=labels, autopct='%1.1f%%', shadow=True,
18         startangle=140)
19 plt.axis('equal')
20
21 plt.subplot(2,2,2)
22 plt.title('subplot(2,2,1)')
23 plt.pie(sizes, labels=labels, autopct='%1.1f%%', shadow=True,
24         startangle=140)
25 plt.axis('equal')
26
27 plt.subplot(2,2,3)
28 plt.title('subplot(2,2,1)')
29 plt.pie(sizes, labels=labels, autopct='%1.1f%%', shadow=True,
30         startangle=140)
31 plt.axis('equal')
32
33 plt.subplot(2,2,4)
34 plt.title('subplot(2,2,1)')
35 plt.pie(sizes, labels=labels, autopct='%1.1f%%', shadow=True,
36         startangle=140)
37 plt.axis('equal')
```

### 6.3.4 Nomor 4

Plot :

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Wed Apr 10 14:08:15 2019
```

```

4
5 @author: Irvan
6 """
7
8 import matplotlib.pyplot as plt
9
10 print('Jumlah Subplot Sesuai NPM mod 3 + 2 =', 1174043%3+2)
11
12 x = [4,8,13,20,23]
13 y = [54,67,99,78,80]
14
15 plt.subplot(2,2,1)
16 plt.title('subplot(3,3,1)')
17 plt.plot(x,y)
18
19 plt.subplot(2,2,2)
20 plt.title('subplot(3,3,2)')
21 plt.plot(x,y)
22
23 plt.subplot(2,2,3)
24 plt.title('subplot(3,3,3)')
25 plt.plot(x,y)
26
27 plt.subplot(2,2,4)
28 plt.title('subplot(3,3,4)')
29 plt.plot(x,y)

```

### 6.3.5 Penanganan Error

```

1 from matplotlib import pyplot as plt
2
3 def err():
4
5     mod = 1174040 % 3 + 2
6
7     x = (4,8,13,17,20)
8     y = (54, 67, 98, 78, 45)
9     try:
10         for i in range(1, mod+1):
11             plt.subplot(1,3,i)
12             plt.plot(x,y,'b', linewidth=1)
13             plt.title('PLOT')
14             plt.subplots_adjust(wspace=.4, hspace=.7)
15
16         plt.show()
17     except ValueError:
18         print(" Jumlah subplot tidak sama dengan perhitungan mod. panjang ataupun lebar subplot tidak boleh lebih sedikit dari urutannya")

```

## 6.4 Luthfi Muhammad Nabil/1174035

### 6.4.1 Soal 1

Buatlah librari fungsi dengan nama file chap6\_1174035\_bar.py denga jumlah subplot adalah NPM mod 3+2 :

```

1 def BarPanggil(npm):
2     import matplotlib.pyplot as plt
3     import random as rand
4     i = (int(npm)%3)+2
5     rowcol = ""
6     if i%2==0:
7         row=i/2
8         rowcol=str(int(row))+"2"
9     if i%3==0:
10        row=i/3
11        rowcol=str(int(row))+"3"
12    a=1
13    while a<=i:
14        y = [rand.randint(1,10),rand.randint(1,10),rand.randint(1,10),
15              rand.randint(1,10)]
16        plt.subplot(int(rowcol+str(a)))
17        plt.bar([1,2,3,4], y)
18        a+=1
19    plt.show()
20    return i
BarPanggil("1174035")

```

Hasil dari kompilasi terdapat pada gambar 6.5

### 6.4.2 Soal 2

Buatlah librari fungsi dengan nama file chap6\_1174035\_scatter.py denga jumlah subplot adalah NPM mod 3+2 :

```

1 def ScatterPanggil(npm):
2     import matplotlib.pyplot as plt
3     import random as rand
4     i = (int(npm)%3)+2
5     rowcol = ""
6     if i%2==0:
7         row=i/2
8         rowcol=str(int(row))+"2"
9     if i%3==0:
10        row=i/3
11        rowcol=str(int(row))+"3"
12    a=1
13    while a<=i:
14        plt.subplot(int(rowcol+str(a)))
15        y = [rand.randint(1,10),rand.randint(1,10),rand.randint(1,10),
16              rand.randint(1,10),rand.randint(1,10),rand.randint(1,10),rand.
17              randint(1,10)]
18        plt.plot([1,2,3,4,5,6,7], y, 'ro')
19        a+=1

```

```

18 plt.show()
19 return i
20 ScatterPanggil("1174035")

```

Hasil dari kompilasi terdapat pada gambar 6.6

### 6.4.3 Soal 3

Buatlah librari fungsi dengan nama file chap6\_1174035\_pie.py denga jumlah subplot adalah NPM mod 3+2 :

```

1 def PiePanggil(npm):
2     import matplotlib.pyplot as plt
3     import random as rand
4     i = (int(npm)%3)+2
5     row = ""
6     col = ""
7     if i%2==0:
8         row=i/2
9         col="2"
10    if i%3==0:
11        row=i/3
12        col="3"
13    a=1
14    labels = 'Anjing', 'Kucing', 'Kakatua', 'Jerapah'
15    # Make figure and axes
16    fig, axs = plt.subplots(int(row), int(col))
17    while a<=i:
18        c=0
19        fracs = []
20        tot=0
21        while c!=3:
22            num = rand.randint(1,33)
23            tot = tot+num
24            fracs.append(num)
25            c+=1
26        fracs.append(100-tot)
27        plt.subplot(str(int(row)),col,a)
28        plt.pie(fracs,labels=labels,startangle=90,shadow= True,explode
29        =[0,0,0,0.1],autopct='%.1f%%')
30        a+=1
31    plt.show()
32 return i
33 a = PiePanggil("1174035")

```

Hasil dari kompilasi terdapat pada gambar 6.7

### 6.4.4 Soal 4

Buatlah librari fungsi dengan nama file chap6\_1174035\_plot.py denga jumlah subplot adalah NPM mod 3+2 :

```

1 def PlotPanggil(npm):
2     import matplotlib.pyplot as plt
3     import random as rand

```

```

4 i = (int(npm)%3)+2
5 rowcol = ""
6 if i%2==0:
7     row=i/2
8     rowcol=str(int(row))+"2"
9 if i%3==0:
10    row=i/3
11    rowcol=str(int(row))+"3"
12 a=1
13 while a<=i:
14     plt.subplot(int(rowcol+str(a)))
15     y = [rand.randint(1,10),rand.randint(1,10),rand.randint(1,10),
16           rand.randint(1,10),rand.randint(1,10)]
17     plt.plot([1,2,3,4,5], y)
18     a+=1
19 plt.show()
20 return i
21 PlotPanggil("1174035")

```

Hasil dari kompilasi terdapat pada gambar 6.8

#### 6.4.5 Penanganan Error

Error yang ditemukan yaitu ValueError. Cara penanggulangannya yaitu dengan mengecek kembali panjang dan lebar subplotnya dan memastikan jika urutan pada subplot (Angka ke 3) tidak kurang atau lebih dari panjang\*lebar yang ditentukan. Berikut Penerapan dari try except terdapat pada gambar 6.9

```

1 def BarPanggil(npm):
2     try :
3         import matplotlib.pyplot as plt
4         import random as rand
5         i = (int(npm)%3)+2
6         rowcol = ""
7         if i%2==0:
8             row=i/2
9             rowcol=str(int(row))+"2"
10        if i%3==0:
11            row=i/3
12            rowcol=str(int(row))+"3"
13        a=1
14        while a<=i:
15            y = [rand.randint(1,10),rand.randint(1,10),rand.randint(1,10),
16                  rand.randint(1,10)]
17            plt.subplot(int(rowcol+str(4)))
18            #plt.subplot(int(rowcol+str(a)))
19            plt.bar([1,2,3,4], y)
20            a+=1
21        plt.show()
22    except ValueError:
23        print("Salah pada index subplot")
24 BarPanggil("1174035")

```

Hasil dari kompilasi (Jika Error) terdapat pada gambar 6.9

## 6.4.6 Dika Sukma Pradana 1174050

### 6.4.7 Soal 1

```

1 from matplotlib import pyplot as plt
2
3
4
5 def barr():
6
7     print(1174050%3+2)
8
9     x = [1,3,5,7,9,11,13]
10    y = [12,22,32,42,52,67,89]
11
12    x1 = [10,7,13,4,5]
13    y1 = [2,3,2,7,9]
14
15    x2 = [20,50,60,10,40,20,90]
16    y2 = [30,50,70,90,20,80,10]
17
18    plt.subplot(221)
19    plt.bar(x,y)
20    plt.subplot(222)
21    plt.bar(x1,y1)
22    plt.subplot(223)
23    plt.bar(x2,y2)
24    plt.show()

```

untuk memunculkan hasilnya kita dapat memanggil menggunakan :

```

1 import p1174050_bar as barr
2
3 barr.barr()

```

dan akan menghasilkan grafik seperti gambar berikut:

### 6.4.8 Soal 2

```

1 from matplotlib import pyplot as plt
2
3 def scatter():
4
5     print(1174050%3+2)
6
7     x = [1,3,5,7,9,11,13]
8     y = [12,22,32,42,52,67,89]
9
10    x1 = [10,7,13,4,5]
11    y1 = [2,3,2,7,9]
12
13    x2 = [20,50,60,10,40,20,90]
14    y2 = [30,50,70,90,20,80,10]
15
16    plt.subplot(221)
17    plt.scatter(x,y)

```

```
18 plt.subplot(222)
19 plt.scatter(x1,y1)
20 plt.subplot(223)
21 plt.scatter(x2,y2)
22 plt.show()
```

untuk memunculkan hasilnya kita dapat memanggil menggunakan :

```
1 import p1174050_scatter as sct
2
3
4
5 sct.scatter()
```

dan akan menghasilkan grafik seperti gambar berikut:

#### 6.4.9 Soal 3

```
1 from matplotlib import pyplot as plt
2
3
4
5 def pie_chart():
6
7     print(1174050%3+2)
8
9     aktivity = [1,6,2,4]
10    game = [14,1,9]
11    txt = [9,2,9,17]
12    siapa = ['dika','najib','alit','iksan']
13    dimana = ['rumah','kampus','kosan']
14    apa = ['futsal','ngoding','tidur','mudik']
15    cols = ['b','m','c','y']
16
17    plt.subplot(221)
18    plt.pie(aktivity,
19            labels=siapa,
20            colors=cols,
21            startangle=0,
22            shadow=True,
23            explode=(0.2,0,0,0),
24            autopct='%1.1f%%')
25    plt.title('Siapa')
26
27    plt.subplot(222)
28    plt.pie(game,
29            labels=dimana,
30            colors=cols,
31            startangle=90,
32            shadow=True,
33            explode=(.3,0.1,0),
34            autopct='%1.1f%%')
35    plt.title('Dimana')
36
37    plt.subplot(223)
38    plt.pie(txt,
39            labels=apa,
40            colors=cols,
41            startangle=90,
```

```

42     shadow=True ,
43     explode=(.1 ,0 ,0 ,0) ,
44     autopct='%.1f%%' )
45 plt.title('Ngapain')
46 plt.show()

```

untuk memunculkan hasilnya kita dapat memanggil menggunakan :

```

1 import p1174050_pie as pie
2
3 pie.pie_chart()

```

dan akan menghasilkan grafik seperti gambar berikut:

#### 6.4.10 Soal 4

```

1 from matplotlib import pyplot as plt
2
3 def plot():
4
5     print(1174050%3+2)
6
7     x = [1 ,3 ,5 ,7 ,9 ,11 ,13]
8     y = [12 ,22 ,32 ,42 ,52 ,67 ,89]
9
10    x1 = [10 ,7 ,13 ,4 ,5]
11    y1 = [2 ,3 ,2 ,7 ,9]
12
13    x2 = [20 ,50 ,60 ,10 ,40 ,20 ,90]
14    y2 = [30 ,50 ,70 ,90 ,20 ,80 ,10]
15
16    plt.subplot(221)
17    plt.plot(x,y)
18    plt.subplot(222)
19    plt.plot(x1,y1)
20    plt.subplot(223)
21    plt.plot(x2,y2)
22    plt.show()

```

untuk memunculkan hasilnya kita dapat memanggil menggunakan :

```

1 import p1174050_plot as plt
2
3 plt.plot()

```

dan akan menghasilkan grafik seperti gambar berikut:

#### 6.4.11 Penanganan Error

Macam - macam error :

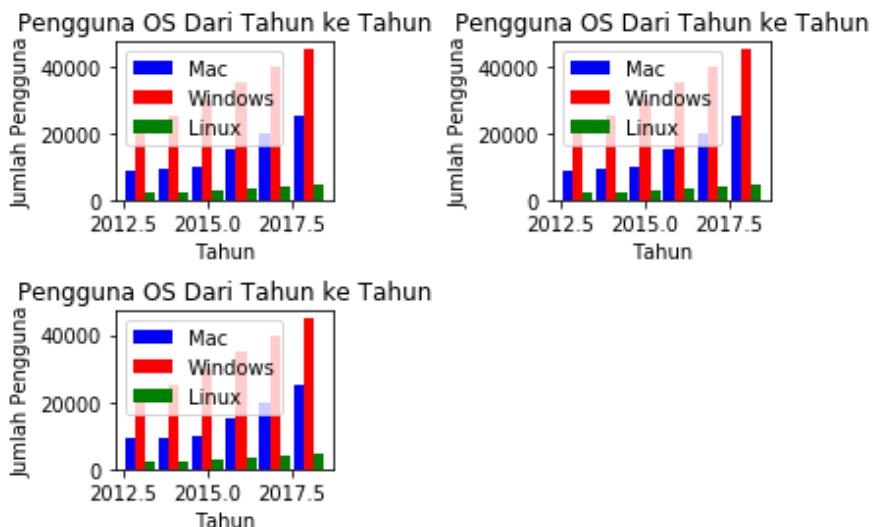
- Syntax Errors Syntax Errors adalah suatu keadaan saat kode python mengalami kesalahan penulisan. Solusinya adalah memperbaiki penulisan kode yang salah.
- Name Error NameError adalah exception yang terjadi saat kode melakukan eksekusi terhadap local name atau global name yang tidak terdefinisi. Solusinya

adalah memastikan variabel atau function yang dipanggil ada atau tidak salah ketik.

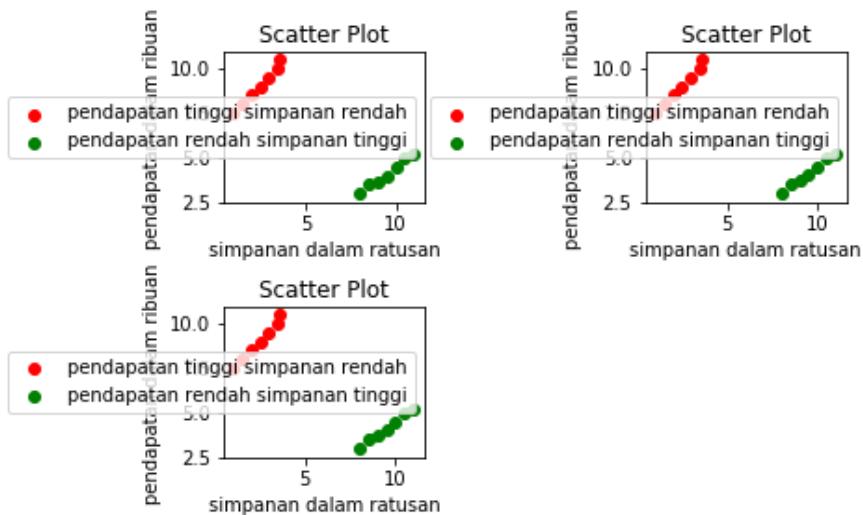
- Type Error TypeError adalah exception yang akan terjadi apabila pada saat dilakukannya eksekusi terhadap suatu operasi atau fungsi dengan type object yang tidak sesuai. Solusi dari error ini adalah mengkoversi varibelnya sesuai dengan tipe data yang akan digunakan.

Apabila terjadi suatu ke-eror-an maka dapat ditangani dengan cara sebagai berikut :

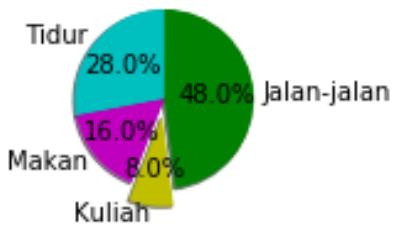
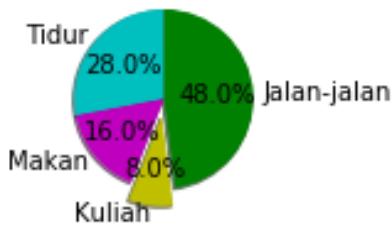
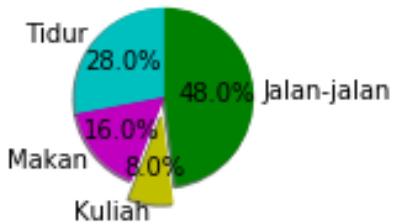
```
1 def tryExceptError():
2     try:
3         from p1174050_scatter import scatter as sct
4     except SyntaxError:
5         print("ERRORoooooooooooooOR")
6
7 tryExceptError()
```

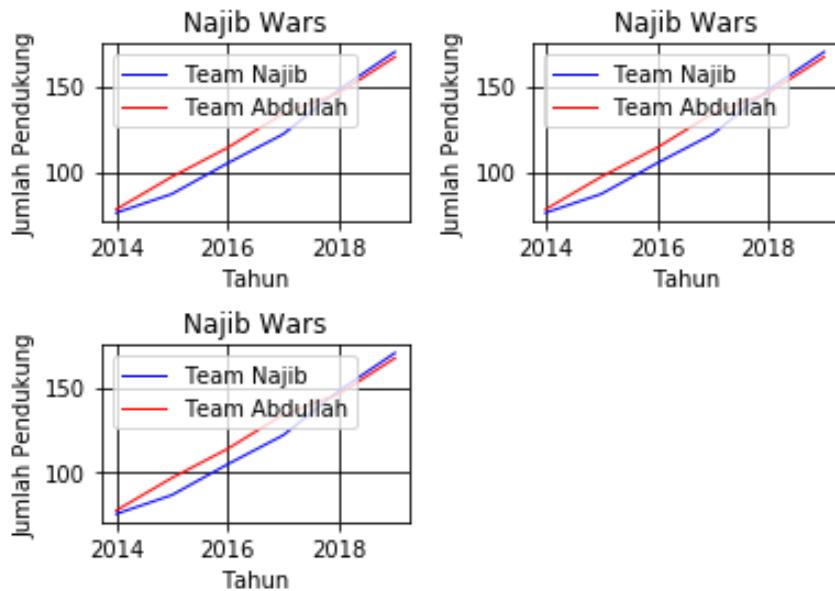


**Gambar 6.1** Hasil compile membuat fungsi Bar Plot menggunakan Matplotlib.

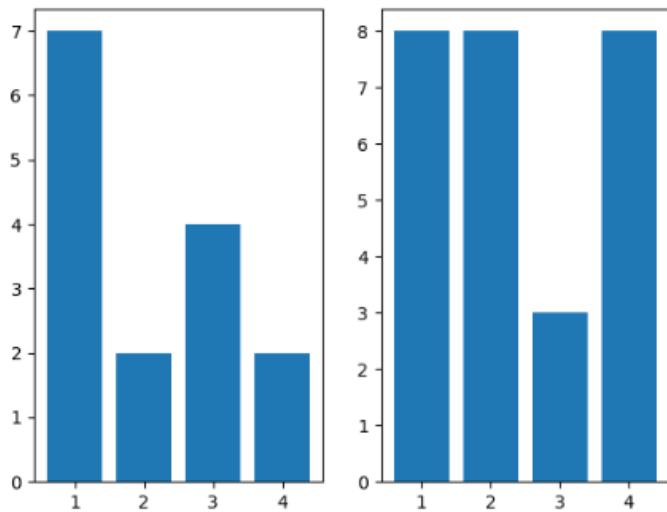


**Gambar 6.2** Hasil compile membuat fungsi Scatter Plot menggunakan Matplotlib.

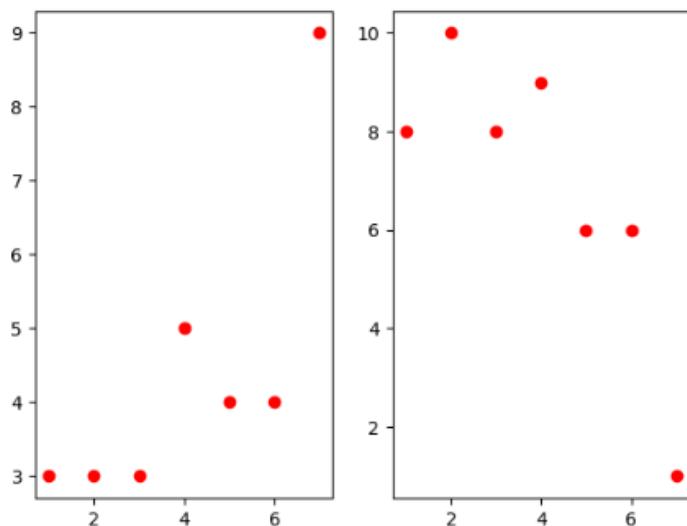
**Kegiatan Sehari-hari****Kegiatan Sehari-hari****Kegiatan Sehari-hari****Gambar 6.3** Hasil compile membuat fungsi Pie Plot menggunakan Matplotlib.



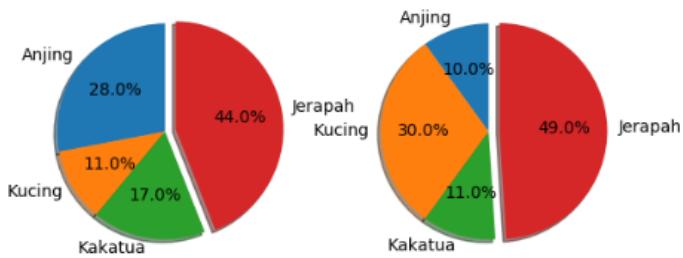
**Gambar 6.4** Hasil compile membuat fungsi Plot menggunakan Matplotlib.



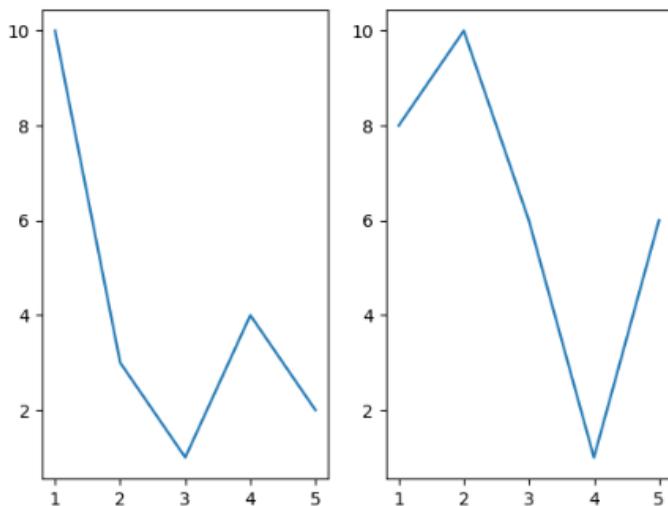
**Gambar 6.5** Grafik Bar



Gambar 6.6 Grafik Scatter



Gambar 6.7 Grafik Pie

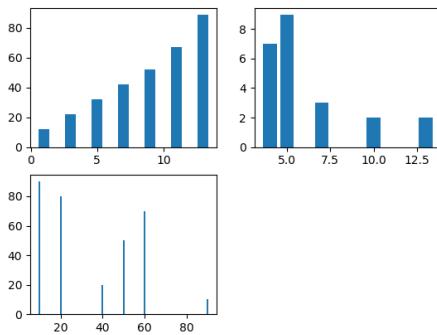


Gambar 6.8 Grafik Plot

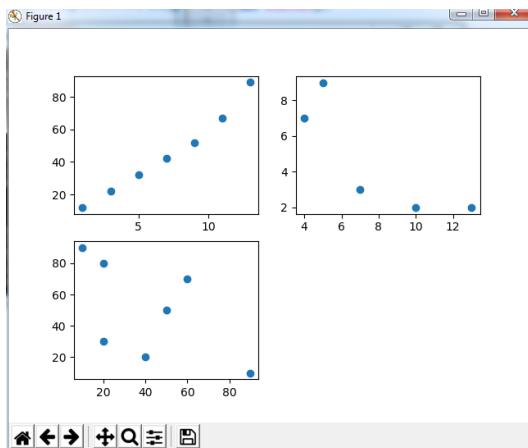
```
C:\Windows\system32\cmd.exe
D:\Coba\MatPlotPraktek>
D:\Coba\MatPlotPraktek>python chap6_1174035_error.py
D:\Coba\MatPlotPraktek>python chap6_1174035_error.py
Traceback (most recent call last):
  File "chap6_1174035_error.py", line 20, in <module>
    BarPanggil("1174035_error.py")
  File "chap6_1174035_error.py", line 15, in BarPanggil
    plt.subplot(int('0' + str(4)))
  File "C:\Users\Luthfi\AppData\Local\Programs\Python\Python37\lib\site-packages\matplotlib\pyplot.py", line 1884, in subplot
    a = fig.add_subplot(*args, **kwargs)
  File "C:\Users\Luthfi\AppData\Local\Programs\Python\Python37\lib\site-packages\matplotlib\figure.py", line 1367, in add_subplot
    a = subplot_class_factory(projection_class)(self, *args, **kwargs)
  File "C:\Users\Luthfi\AppData\Local\Programs\Python\Python37\lib\site-packages\matplotlib\axes\_subplots.py", line 68, in __init__
    _format(maxn-rows+cols, num=num)
ValueError: num must be 1 <= num <= 2, not 4

D:\Coba\MatPlotPraktek>python chap6_1174035_error.py
Salah pada indeks subplot
D:\Coba\MatPlotPraktek>
```

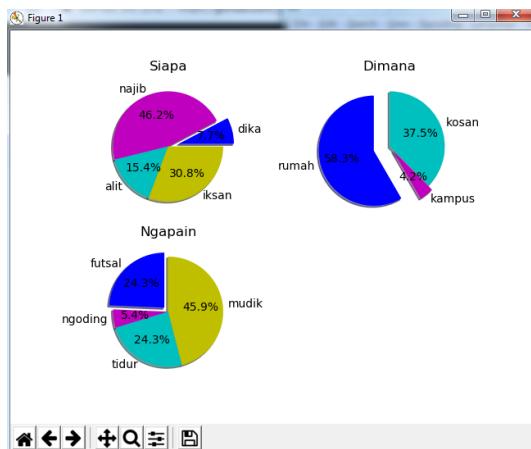
Gambar 6.9 Hasil Kompilasi Error



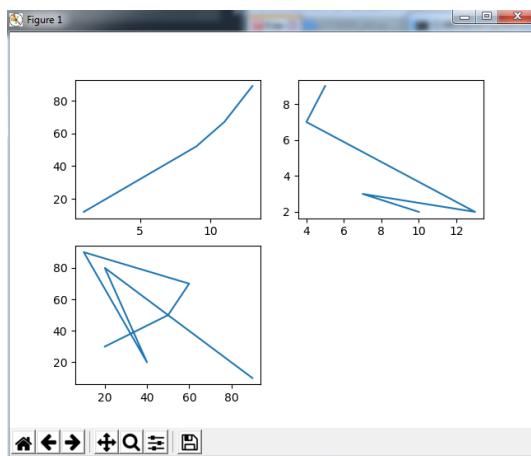
**Gambar 6.10** Grafik Batang



**Gambar 6.11** Grafik Scatter



Gambar 6.12 Grafik Pie



Gambar 6.13 Grafik Plot