

Laporan Tugas

Instalasi dan Penggunaan

Anaconda



Dinda Majesty
NPM : 1.18.4.011

Informatics Research Center
Applied Bachelor Program of Informatics Engineering
Bandung 2019

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

Acknowledgements

Pertama-tama kami panjatkan puji dan syukur kepada Allah SWT yang telah memberikan rahmat dan hidayah-Nya sehingga Laporan Tugas Instalasi dan Penggunaan Anaconda ini dapat diselesaikan.

Abstract

Laporan Tugas Instalasi dan Penggunaan Anaconda ini dibuat dengan tujuan memenuhi tugas matakuliah Pemrograman II. Pada intinya Laporan ini menjawab secara lengkap pertanyaan yang terkait dengan silabus matakuliah Pemrograman II. Di dalamnya memuat cara instalasi anaconda hingga cara menggunakan anaconda spyder dan dijelaskan sedikit mengenai error (error indentasi).

Contents

| | | |
|----------|--|----------|
| 1 | Fungsi dan Kelas | 1 |
| 1.1 | Teori | 1 |
| 1.2 | Fungsi | 1 |
| 1.3 | Package | 1 |
| 1.4 | Kelas, Objek, Atribut, dan Method | 2 |
| 1.4.1 | Pemanggilan Library Kelas | 3 |
| 1.4.2 | From kalkulator import Penambahan | 3 |
| 1.4.3 | Pemakaian Package Fungsi Apabila File Didalam Folder | 3 |
| 1.4.4 | Pemakaian Package Kelas Apabila File didalam Folder | 4 |
| 1.5 | Keterampilan Pemrograman | 4 |
| 1.5.1 | Modulus | 4 |
| 1.5.2 | Hello NPM | 4 |
| 1.5.3 | Hello NPM (3 Digit Belakang) | 5 |
| 1.5.4 | Hello NPM (Digit ke-3) | 5 |
| 1.5.5 | Variabel Alfabet | 5 |
| 1.5.6 | Penjumlahan NPM | 5 |
| 1.5.7 | Perkalian NPM | 6 |
| 1.5.8 | Print Vertical | 6 |
| 1.5.9 | Digit Genap NPM | 6 |
| 1.5.10 | Digit Ganjil NPM | 7 |
| 1.5.11 | Bilangan Prima NPM | 7 |
| 1.5.12 | Pemanggilan Fungsi pada Main.py | 7 |
| 1.6 | Keterampilan Penanganan Error | 8 |
| 1.6.1 | Peringatan Error dan Cara Mengatasinya | 8 |

List of Figures

Listings

| | | |
|------|------------------------------|---|
| 1.1 | Modulus | 4 |
| 1.2 | Hello NPM | 5 |
| 1.3 | 3 Digit Belakang | 5 |
| 1.4 | Digit ke-3 | 5 |
| 1.5 | Variabel Alfabet | 5 |
| 1.6 | Penjumlahan NPM | 6 |
| 1.7 | Perkalian NPM | 6 |
| 1.8 | Print Vertical | 6 |
| 1.9 | Digit Genap NPM | 6 |
| 1.10 | Digit Ganjil NPM | 7 |
| 1.11 | Bilangan Prima NPM | 7 |
| 1.12 | Bilangan Prima NPM | 7 |

Chapter 1

Pengelolaan File CSV

1.1 Teori

1.2 Fungsi File CSV, Sejarah, dan Contoh

File csv (comma separated value) sering digunakan dalam dunia pemrograman untuk menampilkan data. file csv memiliki format yang sangat sederhana, setiap baris dipisahkan oleh enter dan setiap kolom dipisahkan oleh tanda koma.

Kegunaan file csv dibandingkan file dengan format lainnya adalah dari segi kompatibilitas karena file csv dapat diolah, dimodifikasi, digunakan, import/export menggunakan berbagai software dan bahasa pemrograman, salah satunya python.

Contoh file csv:

```
1 nomor, nama klub, jumlah main, poin, tanggal
2 1, Manchester City, 8, 19, 1/2/1999
3 2, Arsenal, 8, 18, 1/3/1999
4 3, Tottenham Hotspurs, 8, 18, 1/4/1999
5 4, Liverpool, 8, 17, 1/5/1999
6 5, Chelsea, 8, 16, 1/6/1999
7 6, Everton, 8, 15, 1/7/1999
8 7, Manchester United, 8, 14, 1/8/1999
9 8, Southampton, 8, 12, 1/9/1999
10 9, AFC Bournemouth, 8, 12, 1/10/1999
11 10, Crystal Palace, 8, 11, 1/11/1999
```

Listing 1.1: Contoh CSV

Contoh kode program python untuk membaca file csv.

```
1 #CSV Reader
2 import csv
3
4 with open('FCSV.csv') as csvfile:
```



```

5 readCSV = csv.reader(csvfile, delimiter=',')
6 for row in readCSV:

```

Listing 1.2: Contoh Kode Python to Read CSV

Contoh kode program python untuk membaca file csv menggunakan library pandas.

```

1 import pandas as pd
2 df1=pd.read_csv("FCSV.csv")
3 print(df1)

```

Listing 1.3: Contoh Kode Python to Read CSV

Sejarah CSV.

CSV sudah digunakan sejak tahun 1972, CSV dapat dikompilasi pada bahasa pemrograman IBM Fortran. Data yang dipisahkan oleh koma apabila terdapat spasi didalamnya maka harus diberi tanda petik di awal dan akhir isi dari data tersebut. Nama CSV digunakan pada tahun 1983. Pada panduan dari Osborne Executive Computer terdapat kutipan yang membolehkan isi karakter memiliki koma. Pada tahun 2005 dengan RFC4180, CSV didefinisikan sebagai MIME Content Type. Lalu pada tahun 2013, defisiensi 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 tentang bagaimana baris, kolom dapat digunakan dalam dokumen CSV menggunakan indeks posisi. Pada Tahun 2015, W3C mempublikasikan draft rekomendasi untuk CSV-metadata standard yang dimulai pada bulan Desember 2015.

1.3 Aplikasi yang bisa Menciptakan File CSV

Aplikasi yang dapat kita gunakan untuk membuat file csv ada banyak, diantaranya:

1. Spreadsheet

Spreadsheet merupakan aplikasi pembuatan file CSV dengan cara memasukkan data sesuai baris dan kolom yang diinginkan. Contoh spreadsheet seperti Google Spreadsheet, Microsoft Excel, dan aplikasi lainnya.

2. Bahasa

Pemrograman

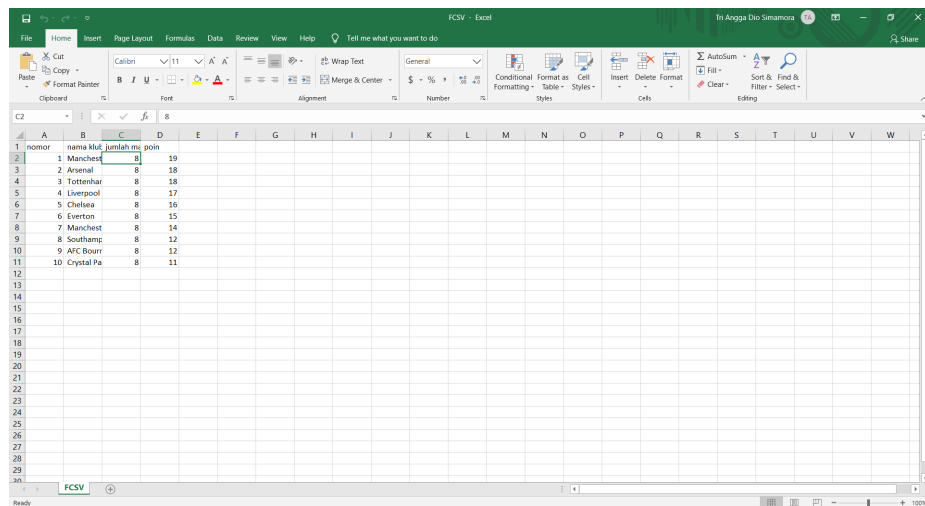
Bahasa pemrograman merupakan media untuk membuat aplikasi yang dapat digunakan untuk membuat file CSV khusus dengan bahasa pemrograman tertentu yang support dengan pembuatan file CSV. Seperti Python, C Sharp, dan lain sebagainya.

3. Notepad atau Text Editor

Text editor juga dapat membuat file CSV, cukup dengan membuat file sesuai format CSV dan save file tersebut dengan ekstensi .csv.

1.4 Menulis dan Membaca File CSV pada Ms.Excel

Membuat file csv melalui ms.excel sangatlah mudah, isikan nomor pada kolom, kemudian isikan variabel sebagai judul pada baris, lalu isikan data sesuai dengan variabel yang telah ditentukan, contoh:



The screenshot shows the Microsoft Excel interface with a file named 'FCSV'. The data is organized in a table with the following content:

| | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W |
|----|-------|------------|-------------|------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | nomor | nama klub | jumlah main | poin | | | | | | | | | | | | | | | | | | | |
| 2 | 1 | Manchester | 8 | 19 | | | | | | | | | | | | | | | | | | | |
| 3 | 2 | Arsenal | 8 | 18 | | | | | | | | | | | | | | | | | | | |
| 4 | 3 | Tottenham | 8 | 18 | | | | | | | | | | | | | | | | | | | |
| 5 | 4 | Liverpool | 8 | 17 | | | | | | | | | | | | | | | | | | | |
| 6 | 5 | Chelsea | 8 | 16 | | | | | | | | | | | | | | | | | | | |
| 7 | 6 | Everton | 8 | 15 | | | | | | | | | | | | | | | | | | | |
| 8 | 7 | Manchester | 8 | 14 | | | | | | | | | | | | | | | | | | | |
| 9 | 8 | Southamp | 8 | 12 | | | | | | | | | | | | | | | | | | | |
| 10 | 9 | AFC Bourn | 8 | 12 | | | | | | | | | | | | | | | | | | | |
| 11 | 10 | Crystal Pa | 8 | 11 | | | | | | | | | | | | | | | | | | | |

Figure 1.1: Contoh Penulisan CSV pada Excel

Setelah mengetikkan data pada excel, lalu pilih menu file, pilih menu export, pilih menu change file type, pilih csv, lalu klik tombol save as. Maka file excel akan menjadi file csv.

```
1 nomor, nama klub, jumlah main, poin, tanggal
2 1, Manchester City, 8, 19, 1/2/1999
3 2, Arsenal, 8, 18, 1/3/1999
4 3, Tottenham Hotspurs, 8, 18, 1/4/1999
5 4, Liverpool, 8, 17, 1/5/1999
6 5, Chelsea, 8, 16, 1/6/1999
7 6, Everton, 8, 15, 1/7/1999
8 7, Manchester United, 8, 14, 1/8/1999
9 8, Southampton, 8, 12, 1/9/1999
10 9, AFC Bournemouth, 8, 12, 1/10/1999
11 10, Crystal Palace, 8, 11, 1/11/1999
```

Listing 1.4: Contoh CSV

1.4.1 Sejarah Library CSV

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, Kevin Altis dan lainnya telah merilis versi final untuk library Python CSV.

1.4.2 Sejarah Library Pandas

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.

1.4.3 Fungsi - fungsi yang terdapat di library CSV

Berikut fungsi-fungsi yang terdapat pada library csv.

1. `csv.reader(csvfile, dialect='excel', **fmtparams)`

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

```
1 #CSV Reader
2 import csv
3
4 with open('FCSV.csv') as csvfile:
5     readCSV = csv.reader(csvfile, delimiter=',')
6     for row in readCSV:
7         print(row)
```

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

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

```
1 #CSV Writer
2 import csv
3
4 with open('asal.csv', mode='w') as csvfile:
5     fieldnames = ['nomor', 'nama klub', 'jumlah main', 'poin']
6     csv_writer = csv.DictWriter(csvfile, fieldnames=fieldnames)
7
8     csv_writer.writeheader()
```

```

9     csv_writer.writerow({'nomor': '1', 'nama klub': 'Asal-Asalan',
10                          'jumlah main': '8', 'poin': '10'})
    csv_writer.writerow({'nomor': '2', 'nama klub': 'Asal-Asalan2',
                          'jumlah main': '8', 'poin': '12'})

```

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

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

4. `csv.unregister_dialect(name)`

Menghapus asosiasi dialek dengan nama pada registry dialek.

5. `csv.get_dialect(name)`

Mengambil dialek yang telah diasosiasikan dengan nama.

6. `csv.list_dialects()`

Mengembalikan dialek yang telah diregistrasi.

7. `csv.field_size_limit([new_limit])`

Mengembalikan maksimal kolom data yang diperbolehkan oleh pembaca.

1.4.3.1 Fungsi - fungsi yang terdapat di library Pandas

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

Untuk membaca file CSV dan menyimpannya ke DataFrame. Contohnya:

```

1 import pandas as pd
2 df1=pd.read_csv("FCSV.csv")
3 print(df1)

```

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

Membaca file excel dan menyimpannya ke DataFrame. Contohnya:

```

1 import pandas as pds
2 df2=pds.read_excel("FCSV.xlsx", index_col=None, header=None)
3 print(df2)

```

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

Untuk membuat file CSV dari data yang ada

1.5 Keterampilan Pemrograman

1.5.1 CSV List

```
1 import csv
2
3 def CsvList():
4     with open('FCSV.csv') as csv_file:
5         csv_reader = csv.reader(csv_file, delimiter=',')
6         line_count = 0
7         for row in csv_reader:
8             if line_count == 0:
9                 print(f'Nama Kolom adalah {"", ".join(row)}')
10                line_count += 1
11            else:
12                print(f'\t Nomor : {row[0]}, Nama Klub : {row[1]},
13                Jumlah Main : {row[2]}, Poin : {row[3]}')
14                line_count += 1
15                print(f'Jumlah Total {line_count} lines')
```

Listing 1.5: CSV List

1.5.2 CSV Dictionary

```
1 def CsvDict():
2     with open('dictionary.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'Nama Kolom adalah {"", ".join(row)}')
8                 line_count += 1
9             else:
10                print(f'\t First Name : {row["first_name"]} Last
11                Name : {row["last_name"]}')
12                line_count += 1
13                print(f'Jumlah Total {line_count} lines')
```

Listing 1.6: CSV Dictionary

1.5.3 Hello NPM (Pandas List)

```
1 import pandas
2
3 def pandasList():
4     pd = pandas.read_csv("FCSV.csv")
5     print(pd)
```

Listing 1.7: Pandas List

1.5.4 Hello NPM (Pandas Dictionary)

```
1 def pandasDict():
2     pds = pandas.read_csv("dictionary.csv")
```

```

3     crot = pandas.DataFrame.from_dict(pds)
4     print(crot)

```

Listing 1.8: Pandas Dictionary

1.5.5 Pandas Date

```

1 def Tanggal():
2     tanggal = pandas.read_csv("FCSV.csv")
3     tanggal['tanggal']=pandas.to_datetime(tanggal['tanggal'])
4     print(tanggal)

```

Listing 1.9: Pandas Date

1.5.6 Pandas Ubah Index

```

1 def ubahIndex():
2     baris = pandas.read_csv("FCSV.csv")
3     barisindex = ['1', '2', '0', '3', '5', '4', '6', '7', '9', '8']
4     oke = baris.reindex(barisindex)
5     print(oke)

```

Listing 1.10: Pandas Ubah Index

1.5.7 Pandas Ubah Column

```

1 def ubahColumn():
2     ubah = pandas.read_csv("FCSV.csv")
3     okey = ubah.rename(columns={"nomor" : "number"})
4     print(okey)

```

Listing 1.11: Pandas Ubah Column

1.5.8 Main CSV

```

1 import D1184011_csv
2
3 csvList = D1184011_csv.CsvList()
4 csvDict = D1184011_csv.CsvDict()

```

Listing 1.12: Main CSV

1.5.9 Main Pandas

```

1 import D1184011_pandas
2
3 pandasList = D1184011_pandas.pandasList()
4 pandasDict = D1184011_pandas.pandasDict()
5 pandasDate = D1184011_pandas.Tanggal()
6 pandasUbah = D1184011_pandas.ubahIndex()
7 pandasUbahCol = D1184011_pandas.ubahColumn()

```

Listing 1.13: Digit Genap NPM