

# Tugas Pemrograman Chapter 4

Aditya Luthfi Maulana Harahap (1184090)

December 4, 2019

# Chapter 1

## Teori

### 1.1 Apa itu fungsi file CSV, jelaskan sejarah dan contoh

#### 1.1.1 Fungsi file CSV

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

#### 1.1.2 Sejarah file CSV

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

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

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

- Contoh

```
1 npm,nama,kelas
2 1184090,Aditya Luthfi Maulana Harahap,D4 TI 2C
3 1184076,Ariq Rafi Kusumah,D4 TI 2C
```

## 1.2 Aplikasi-aplikasi yang bisa membuat file CSV

1. Text Editor (Notepad, Wordpad, dll)
2. Spreadsheet (Microsoft Excel)

## 1.3 Cara menulis dan membaca file CSV pada Excel

1. Membuka aplikasi MS.Excel
2. K adalah kolom dan B adalah baris

3. Kemudian K1 dan B1 di isi dengan Npm, K2 dan B1 di isi dengan Nama, K1 dan B3 di isi dengan Kelas
4. Kemudian pada baris ke selanjutnya adalah record(isi data).
5. Selanjutnya jika telah seperti atas maka selanjutnya kita save as dan pada save as type kita ganti jadi csv (Comman delimited)
6. Maka file CSV telah di buat.

## 1.4 Sejarah library CSV

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

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

## 1.5 Sejarah library Pandas

Pandas adalah toolkit yang powerful sebagai alat analisis data dan struktur untuk bahasa pemrograman Python. Dengan menggunakan pandas kita dapat mengolah data dengan mudah, salah satu fiturnya adalah Dataframe. Dengan adanya fitur dataframe kita dapat membaca sebuah file dan menjadikannya tabble serta juga dapat mengolah suatu data dengan menggunakan operasi seperti join, distinct, group by, agregasi, dan lain-lain

yang terdapat pada SQL. Banyak format file yang dapat dibaca menggunakan Pandas, seperti file .txt, .csv, .tsv dan lainnya. Agar lebih jelas mari kita mencobanya secara langsung.

## 1.6 Fungsi-fungsi yang terdapat pada library CSV

### 1. reader

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

```
1 #Membaca File CSV dengan Fungsi reader dengan library CSV
2 import csv
3
4 with open('prak1.csv') as csv_file:
5     csv_reader = csv.reader(csv_file, delimiter=',')
6     for row in csv_reader:
7         print(row[0], row[1], row[2])
```

### 2. DictReader

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

```
1 #Membaca File CSV dengan Fungsi DictReader dengan library CSV
2 import csv
3
4 with open('prak1.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'])
```

### 3. write

Fungsi ini digunakan untuk menulis file berformat CSV dari list.

```
1 #Menulis File CSV dengan Fungsi writer dengan library CSV
2 import csv
3
4 with open('prak2.csv', mode='w') as csv_file:
5     csv_writer = csv.writer(csv_file, delimiter=',',
6                             quotechar='"', quoting=csv.QUOTE_MINIMAL)
7     csv_writer.writerow(['npm', 'nama', 'kelas'])
8     csv_writer.writerow(['1184090', 'Aditya Luthfi Maulana
9                           Harahap', 'D4TI2C'])
```

```

8     csv_writer.writerow(['1184076', 'Ariq Rafi Kusumah', '
    D4TI2C'])

```

#### 4. DictWrite

Fungsi ini digunakan untuk menulis file berformat CSV dari dictionary.

```

1 #Menulis File CSV dengan Fungsi DictWriter dengan library
  CSV
2 import csv
3
4 with open('prak3.csv', mode='w') as csv_file:
5     fieldnames = ['npm', 'nama', 'kelas']
6     writer = csv.DictWriter(csv_file, fieldnames=fieldnames)
7
8     writer.writeheader()
9     writer.writerow({'npm': '1184090', 'nama': 'Aditya
    Luthfi Maulana Harahap', 'kelas': 'D4TI2C'})
10    writer.writerow({'npm': '1184076', 'nama': 'Ariq Rafi
    Kusumah', 'kelas': 'D4TI2A'})

```

## 1.7 Fungsi-fungsi yang terdapat pada library Pandas

#### 1. read\_csv

Fungsi ini digunakan untuk membaca isi file berformat CSV.

```

1 #Membaca File CSV dengan Fungsi read_csv dengan Library
  Pandas
2 import pandas
3
4 df = pandas.read_csv('prak1.csv')
5 print(df)

```

#### 2. to\_csv

Fungsi ini digunakan untuk menulis file berformat CSV.

```

1 #Menulis File CSV dengan Fungsi to_csv dengan Library Pandas
2 import pandas
3
4 df = pandas.read_csv('prak1.csv')
5 df.to_csv('prak4.csv')

```

## Chapter 2

# Keterampilan Pemrograman

### Soal 1

```
1 def bukaModeListCsv():
2     with open('praktikum1.csv') as csv_file:
3         csv_reader = csv.reader(csv_file, delimiter=',')
4         for row in csv_reader:
5             print(row[0], row[1], row[2])
```

### Soal 2

```
1 def bukaModeDictCsv():
2     with open('praktikum1.csv', mode='r') as csv_file:
3         csv_reader = csv.DictReader(csv_file)
4         for row in csv_reader:
5             print(row['npm'], row['nama'], row['kelas'])
```

### Soal 3

```
1 def bukaModeListPandas():
2     df = pandas.read_csv('praktikum1.csv')
3     print(df)
```

### Soal 4

```
1 def bukaModeDictPandas():
2     df = pandas.read_csv('praktikum1.csv')
3     dt = pandas.DataFrame.from_dict(df)
```

```
4 print(dt)
```

## Soal 5

```
1 def ubahFormatTanggal():
2     df = pandas.read_csv('praktikum1.csv', parse_dates=['tanggal
3     print(df)
```

## Soal 6

```
1 def ubahIndexKolom():
2     df = pandas.read_csv('praktikum1.csv')
3     df.index = ['Row_1', 'Row_2']
4     print(df)
```

## Soal 7

```
1 def ubahNamaKolom():
2     df = pandas.read_csv('praktikum1.csv')
3     df.columns = ['Col_1', 'Col_2', 'Col_3', 'Col_4']
4     print(df)
```

## Soal 8

```
1 lib = __import__('1184090csv')
2
3 lib.bukaModeListCsv()
4 lib.bukaModeDictCsv()
5
6 lib.tulisCsv()
```

## Soal 9

```
1 lib = __import__('1184090pandas')
2
3 lib.bukaModeListPandas()
4 lib.bukaModeDictPandas()
5
6 lib.tulisCsvPandas()
```



# Chapter 3

## Penanganan Error

menggunakan Try Except agar agar dapat memudahkan bagian yang error

```
1 def bacaCsvPandas():
2     try:
3         df = pandas.read_csv('praktikum1.csv')
4         print(df)
5     except SyntaxError:
6         print("Kesalahan penulisan (syntax)")
7     except NameError:
8         print("Variable tidak ada")
9     except TypeError:
10        print("Tipe data salah")
11    except:
12        print("Terjadi sebuah kesalahan")
```