Laporan Praktikum Pertemuan 9

**Data Science Lanjut**

Dibuat Oleh

Nama : Muhamad Faisal Halim

NIM : 19.240.0163

Kelas : -

Mata Kuliah : Data Science Lanjut

Mahasiswa Pertukaran Mahasiswa.

**Universitas Muhammadiyah Kalimantan Timur**

~ STMIK Widya Pratama Pekalongan

**Note**

Data pada praktikum ini disamakan dengan data yang ada pada contoh yang diberikan di Openlearning UMKT.

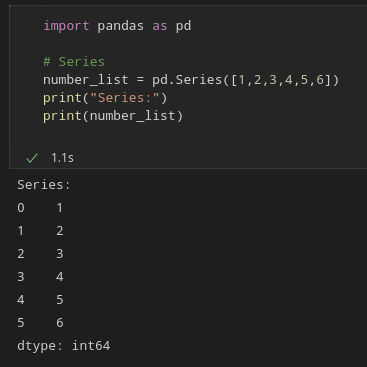
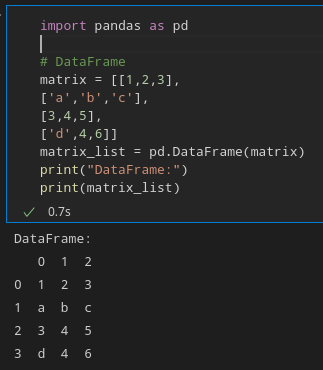
**Materi dan Praktikum**

title : **Manipulasi Data Dengan Pandas**

Pandas adalah sebuah library di Python yang berlisensi BSD dan open source yang menyediakan struktur data dan analisis data yang mudah digunakan. Pandas biasa digunakan untuk membuat tabel, mengubah dimensi data, mengecek data, dan lain sebagainya. Struktur data dasar pada Pandas dinamakan DataFrame, yang memudahkan kita untuk membaca sebuah file dengan banyak jenis format seperti file .txt, .csv, dan .tsv. Fitur ini akan menjadikannya table dan juga dapat mengolah suatu data dengan menggunakan operasi seperti join, distinct, group by, agregasi, dan teknik lainnya yang terdapat pada SQL.

Dapat disimpulkan, bahwa Pandas merupakan library analisis data yang diperlukan untuk membersihkan data mentah ke dalam sebuah bentuk yang bisa untuk diolah.

1. Series: satu kolom bagian dari tabel dataframe yang merupakan 1 dimensional numpy array sebagai basis data nya, terdiri dari 1 tipe data (integer, string, float, dll).
2. DataFrame: gabungan dari Series, berbentuk rectangular data yang merupakan tabel spreadsheet itu sendiri (karena dibentuk dari banyak Series, tiap Series biasanya punya 1 tipe data, yang artinya 1 dataframe bisa memiliki banyak tipe data).

Pandas dataframe dan series memiliki banyak sekali attribut, dan pada pembelajaran ini ada beberapa yang sudah di pakah pada sub bab sebelumnya. untuk tahu lebih mengenai attribut apa yang ada bisa dilihat pada link berikut

Dataframe : <https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.html>

Series : <https://pandas.pydata.org/docs/reference/series.html>

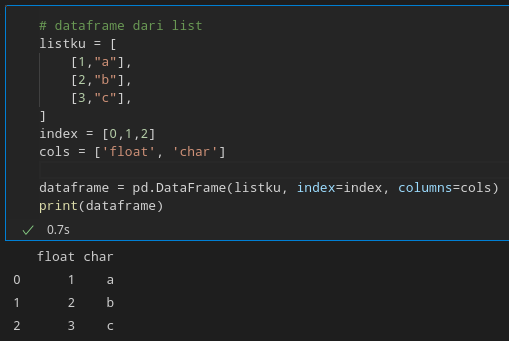
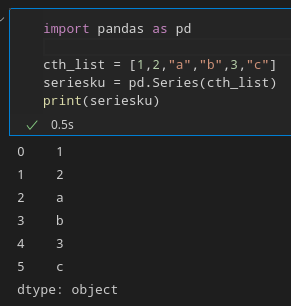
|  |  |
| --- | --- |
| DataFrame.info() | Metode ini mencetak informasi tentang DataFrame termasuk indeks dtype dan kolom, nilai non-null dan penggunaan memori |
| DataFrame.shape | Mencetak tuple yang mewakili dimensi DataFrame |
| DataFrame.dtype | Mencetak tipe data ditiap kolom yang ada |
| DataFrame.astype() | digunakan untuk convert tipe data berdasarkan tipe data seperti float, int, str, numpy.float dll. |
| DataFrame.copy() | digunakan untuk duplikasi data yang tersimpan didalam sebuah variable dan disimpan di variable baru. |
| Series.to\_list() | Merubah series menjadi list. |
| Series.unique() | Mengembalikan nilai unik dari suatu kolom dalam bentuk aray |
| Series.index() | The index (axis labels) of the Series |
| DataFrame.index() | Indeks (label baris) dari DataFrame |
| DataFrame.column | Mengetahui kolom apa saja dari dataframe. |
| DataFrame.loc[]  Series.loc[] | Akses grup baris dan kolom berdasarkan label atau array boolean |
| DataFrame.iloc[]  Series.iloc[] | Akses grup baris dan kolom berdasarkan index kolom atau index |
| DataFrame.size | Kembalikan int yang mewakili jumlah elemen dalam objek ini. |

dari daftar diatas masih banyak lagi yang belum bisa saya sebutkan.

Membuat Dataframe Dan Series

Untuk membuat DataFrame atau Series bisa dari berbagai macam tipe data di python, seperti list, dictionary, maupun numpy array.

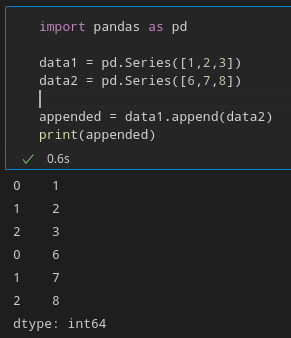
pada praktik dibawah ini saya akan mencoba membuat DataFrame dan Series dari list di python.

Menggabungkan Dataset atau Series

pada pandas ada beberapa metode menggabunkan series atau dataframe, misal merge, concat, append, join.

berikut adalah contoh mengabungkan series dengan append



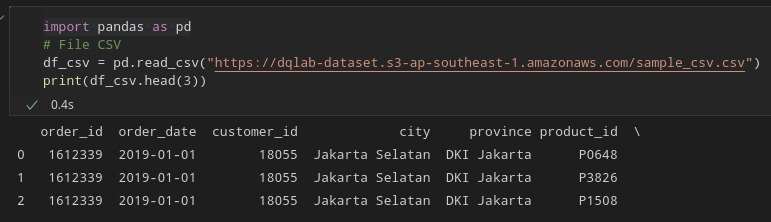
Dataset I/O

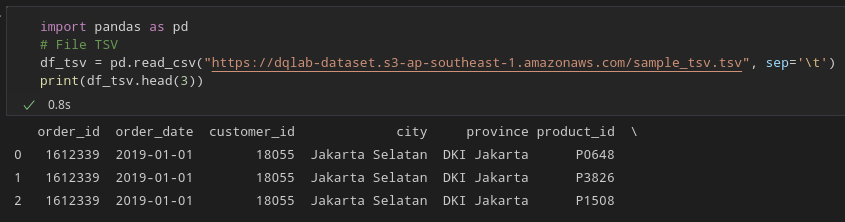
Terdapat sangat banyak file yang bisa dibaca dengan pandas, namun pada kebanyakan kasus ada beberapa file yang sering digunakan seperti. CSV, TSV, Excel, Google BigQuery, SQL Query, JSON. dan hasil output yang diberikan dapat berupa series ataupun dataframe.

Pada dasarnya tipe data CSV dan TSV itu sama, hanya berbeda metode pemisahan data pada kedua file itu, jika CSV menggunakan comma dan TSV menggunakan tab.

dengan menggunakan fungsi Pandas.read\_csv() kita dapat dengan mudah membaca file csv, dengan sparator pemisah comma “,” namun kita juda dapat meribah value dari sparaator tersebut, misal diubah menadi “;”, “|”, atau tab “\t” tergantung pemisah pada file CSV atau kita, karena pada beberapa kasus terdapat file CSV yang pemisahnya menggunakan “;”.

|  |  |
| --- | --- |
| pandas.read\_csv() | Membaca File CSV atau TSV |
| pandas.read\_excel() | Membaca FIle Excel |
| pandas.read\_json() | Membaca File Json |





Dalam praktik dibidang data scientist ada kalanya kita harus membuat dataset. dan pandas udah menyediakan fungsi yang dapat kita gunakan.

|  |  |
| --- | --- |
| .to\_csv() | Tulis objek ke file nilai yang dipisahkan koma (csv) |
| .to\_json() | Ubah objek menjadi string JSON.  Catatan NaN's dan None akan dikonversi menjadi null dan objek datetime akan dikonversi ke stempel waktu UNIX. |
| .to\_excel() | Tulis objek ke lembar Excel.  Untuk menulis satu objek ke file .xlsx Excel, Anda hanya perlu menentukan nama file target. |
| .to\_sql() | Menulis catatan yang disimpan dalam DataFrame ke database SQL. |