

Dataframe I

7 Mar 2024



Agenda

1. Sorting in Dataframe
2. Filtering Dataframe
3. Creating Additional Column

SORTING DATAFRAME

SORTING DATAFRAME

- Sorting dalam DataFrame pandas merujuk pada pengaturan ulang baris berdasarkan nilai pada satu atau lebih kolom. Anda dapat mengurutkan data secara ascending (menaik) atau descending (menurun). Pandas menyediakan dua metode utama untuk sorting:
 - `sort_values()`
 - `sort_index()`
- Sorting merupakan operasi fundamental dalam analisis data, yang memungkinkan untuk mengatur data, mempersiapkannya untuk analisis, atau meningkatkan penyampaian.

SORTING DATAFRAME : `sort_values`

- Metode ini mengurutkan DataFrame berdasarkan nilai dari satu atau lebih kolom.
- Syntax:
 - `DataFrame.sort_values(by, axis=0, ascending=True, inplace=False, kind='quicksort', na_position='last')`
- Parameter
 - **by**: Label tunggal atau daftar. Nama atau daftar nama kolom untuk diurutkan.
 - **ascending**: Boolean atau daftar boolean, default True. Mengurutkan secara ascending vs. descending.
 - **inplace**: Boolean, default False. Jika True, operasi dilakukan inplace.
 - **kind**: {'quicksort', 'mergesort', 'heapsort'}, default 'quicksort'. Pilihan algoritma sorting.
 - **na_position**: {'first', 'last'}, default 'last'. Posisi untuk nilai NaN.

SORTING DATAFRAME : `sort_index`

- Metode ini mengurutkan DataFrame berdasarkan indeksinya, baik indeks baris maupun kolom.
- Syntax:
 - `DataFrame.sort_index(axis=0, level=None, ascending=True, inplace=False, kind='quicksort', na_position='last', sort_remaining=True)`
- Parameter
 - **axis**: {0 atau 'index', 1 atau 'columns'}, default 0. Axis yang akan diurutkan.
 - **level**: int atau nama level atau daftar int atau daftar nama level. Jika tidak None, mengurutkan berdasarkan nilai pada level indeks tertentu.
 - **ascending**: Boolean atau daftar boolean, default True. Mengurutkan secara ascending vs. descending.
 - **inplace**: Boolean, default False. Jika True, operasi dilakukan inplace.
 - **kind**: {'quicksort', 'mergesort', 'heapsort'}, default 'quicksort'. Pilihan algoritma sorting.
 - **na_position**: {'first', 'last'}, default 'last'. Posisi untuk nilai NaN.
 - **sort_remaining**: Boolean, default True. Jika True dan sorting berdasarkan level dan indeks multilevel, mengurutkan berdasarkan level lainnya juga.

FILTERING DATAFRAME

FILTERING DATAFRAME

- Memfilter data dalam DataFrame merupakan cara yang efektif untuk melihat dan menganalisis subset data berdasarkan kondisi tertentu.
- Memfilter data dari sebuah DataFrame adalah salah satu operasi paling umum saat membersihkan data.

FILTERING DATAFRAME

- Memfilter data dalam DataFrame merupakan cara yang efektif untuk melihat dan menganalisis subset data berdasarkan kondisi tertentu.
- Memfilter data dari sebuah DataFrame adalah salah satu operasi paling umum saat membersihkan data.
- Secara umum dapat dibagi menjadi:
 - Filter berdasarkan posisi
 - Filter berdasarkan kondisi
 - Filter berdasarkan query

FILTERING DATAFRAME : Posisi

- menggunakan **df.loc()** → loc memerlukan nama label dari indeks baris dan kolom untuk filtering.
- menggunakan **df.iloc()** → iloc memerlukan nilai indeks numerik untuk baris dan kolom.

FILTERING DATAFRAME : Kondisi

- Memfilter DataFrame dengan kondisi merupakan salah satu operasi yang paling umum dan penting dalam analisis data menggunakan pandas. Ini memungkinkan Anda untuk memilih subset data berdasarkan satu atau lebih kondisi logis.
- Syntax = `df[kondisi]`

FILTERING DATAFRAME : Query

- Pandas menyediakan metode `.query()` yang memungkinkan Anda untuk memfilter DataFrame dengan cara yang lebih ringkas dan mudah dibaca, mirip dengan sintaks SQL.
- Syntax = `df.query(kondisi_sql)`

MENAMBAH KOLOM

MENAMBAH KOLOM

- Menambah kolom pada DataFrame di pandas merupakan operasi yang sering dilakukan selama analisis data.
- Anda bisa menambahkan kolom baru berdasarkan data yang sudah ada atau dengan nilai yang sepenuhnya baru.



Terima Kasih

