

Ringkasan Catatan: Joining Data with Pandas

Berdasarkan materi yang disajikan oleh Aaren Stubberfield dari Datacamp, berikut adalah poin-poin penting mengenai penggabungan data menggunakan library Pandas di Python.

1. Konsep Dasar dan Terminologi

- Di Pandas, tabel data disebut

DataFrame.

- Proses penggabungan data yang setara dengan

`JOIN` di SQL disebut **Merging.**

2. Inner Join

- *Inner join* adalah jenis penggabungan yang paling umum.
- Operasi ini menggabungkan dua DataFrame dan hanya mempertahankan baris-baris yang memiliki nilai kunci yang cocok di kedua tabel tersebut, mirip seperti irisan pada diagram Venn.
- **Sintaks Dasar:** Menggunakan metode `.merge()` pada DataFrame. Kolom yang menjadi kunci penggabungan ditentukan dengan parameter

`on.`

Python

```
wards.merge(census, on='ward')
```

- **Penanganan Kolom Duplikat (Suffixes):**
 - Jika kedua DataFrame memiliki nama kolom yang sama (selain kolom kunci), Pandas akan secara otomatis menambahkan akhiran

`_x` untuk DataFrame kiri dan `_y` untuk DataFrame kanan.

- Anda dapat menentukan akhiran kustom menggunakan parameter

`suffixes`, misalnya `suffixes=('_ward', '_cen')`.

3. Tipe Relasi Data

Relasi One-to-One

- **Definisi:** Setiap baris di tabel kiri hanya berhubungan dengan satu baris di tabel kanan.
- **Contoh:** Menggabungkan data `wards` (50 baris) dengan `census` (50 baris). Hasil penggabungannya juga akan memiliki 50 baris.

Relasi One-to-Many

- **Definisi:** Setiap baris di tabel kiri dapat berhubungan dengan satu atau lebih baris di tabel kanan.
- **Contoh:** Menggabungkan data `wards` (50 baris) dengan data `licenses` (10.000 baris). Hasil penggabungannya akan memiliki jumlah baris yang sama dengan tabel "many", yaitu 10.000 baris.

4. Menggabungkan Beberapa DataFrame

- Anda dapat menggabungkan lebih dari dua DataFrame secara bersamaan.
- **Menggabungkan dengan Beberapa Kunci:** Jika penggabungan memerlukan lebih dari satu kolom kunci, Anda dapat memberikan daftar nama kolom pada parameter `on`.

Python

```
grants.merge(licenses, on=['address', 'zip'])
```

- **Metode Berantai (Chaining):** Penggabungan beberapa DataFrame dilakukan dengan merantai metode `.merge()` secara berurutan.
 - **Tiga Tabel:** `df1.merge(df2, on='col').merge(df3, on='col')`.
 - **Empat Tabel:** `df1.merge(df2, on='col').merge(df3, on='col').merge(df4, on='col')`.

5. Aplikasi Hasil Gabungan

- Tujuan akhir dari penggabungan data adalah untuk memungkinkan analisis yang lebih kaya.
- Setelah data digabungkan, Anda dapat melakukan operasi seperti agregasi (`.groupby()`) dan visualisasi data untuk mendapatkan wawasan.