

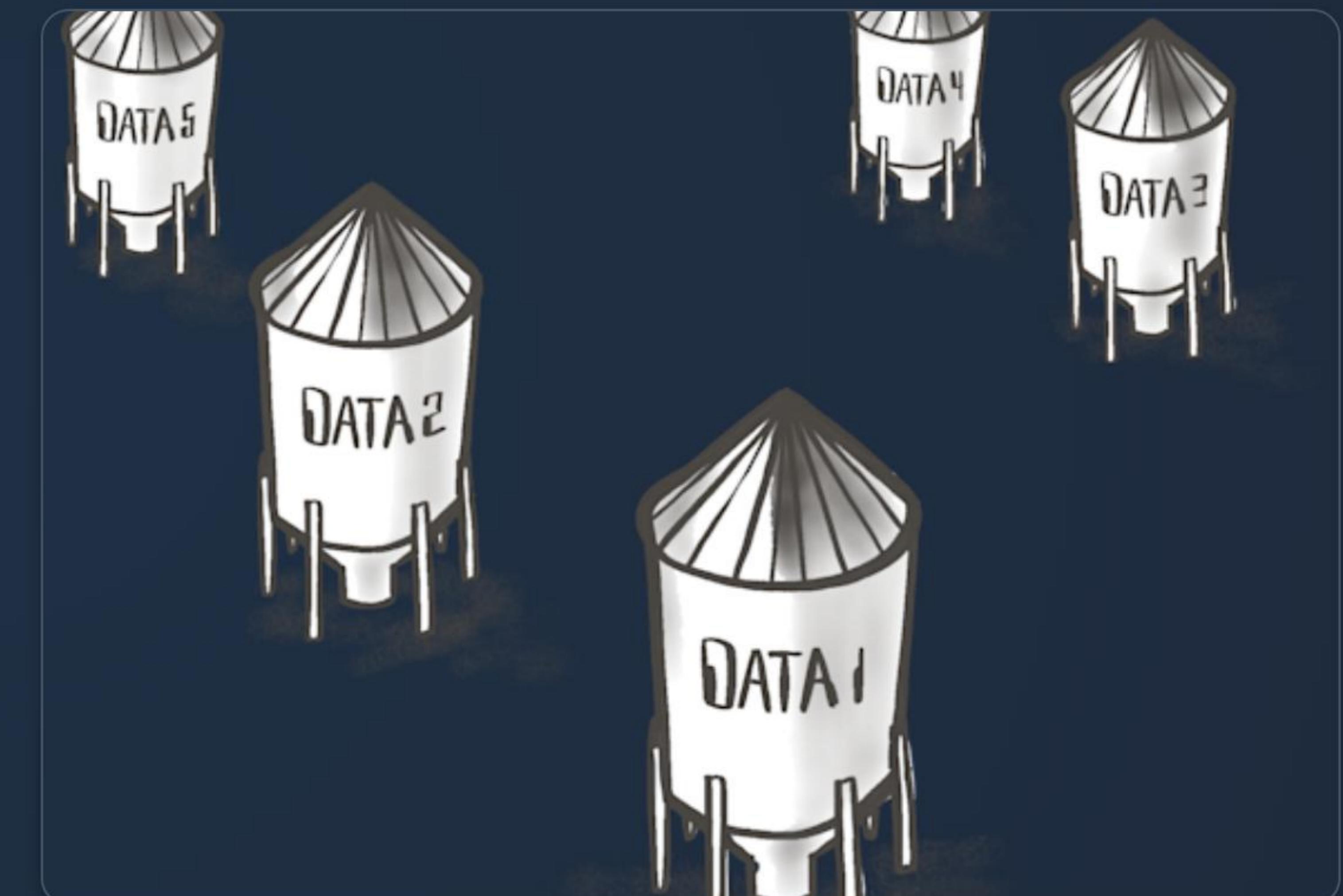
Python: Beyond Query

Kenapa Python adalah senjata wajib bagi Data Analyst &
Database Admin

Masalah Utama di Dunia Query

SQL itu powerful, tapi...

- › Query hanya bekerja optimal di dalam 1 database saja.
- › Sangat sulit melakukan pemadanan lintas sumber (Excel, CSV, API, DB lain).
- › Tidak mudah melakukan data cleaning kompleks (regex, NLP, fuzzy match).
- › Tidak didesain untuk automasi end-to-end dan pipeline data.
- › Tidak cocok untuk visualisasi interaktif yang modern.



Python: Bahasa Serbaguna



Integrasi Total

Membaca data dari berbagai sumber (CSV, Excel, DB, API) dan menggabungkannya dengan mudah.



Pengolahan Cerdas

Mengolah data besar dengan pandas & numpy, serta membuat model pencocokan data yang kompleks.



Automasi & Visual

Menjalankan automasi harian dan membuat dashboard interaktif menggunakan Streamlit.

Beyond Query: Kekuatan Python

Kemampuan	SQL	Python
Gabung data lintas sumber	✗ Terbatas	✓ Sangat Mudah
Fuzzy matching / pemanfaatan	✗ Tidak Ada	✓ (RapidFuzz, FuzzyWuzzy)
NLP untuk pembersihan teks	✗ Tidak Ada	✓ Lengkap
Automasi Schedule	— Tergantung DB	✓ Fleksibel (Airflow/Cron)
Visualisasi	✗ Minimal	✓ Interaktif (Streamlit)

Konteks Data Ketenagakerjaan

Contoh masalah nyata yang sulit dipecahkan hanya dengan SQL:

- **Variasi Nama:** "PT. Maju" vs "PT Maju Jaya" sulit dipadankan.
- **Alamat Non-Standar:** Format alamat berantakan perlu parsing cerdas.
- **Multi-Sumber:** Data tersebar di Excel, CSV, Oracle, dan PostgreSQL.
- **Logika Rumit:** Perlu hitung kedekatan kata (string distance).



Contoh: Pemadanan Fuzzy Matching

Perbedaan Signifikan

Di SQL: Hampir mustahil dilakukan tanpa ekstensi kompleks atau fungsi buatan yang lambat.

Di Python: Hanya butuh 3-5 baris kode untuk mendapatkan skor kemiripan.

```
from rapidfuzz import fuzz

score = fuzz.ratio("PT Bahagia Abadi", "PT Bahagia
Abady")

# Hasil: 92% Match
```

```
        - self.autoDetermineLanguageFromInputting)
        if lang is None:
            raise Exception("Input language could not be determined")
        return None
    parsedInput = self.parseInputToLanguageModel(inputString, inputLanguage, context)
    if not parsedInput or not self.model:
        return None
    context.append(parsedInput) # Add new conversation entry to context
    return (self.model.generateLLMOutput(parsedInput), context)

def parseInputToLanguageModel(inputString, inputLanguage, context):
    if self.model is None or self.model.language != inputLanguage:
        # LLM is not initialised or has wrong language, load LLM
        self.model = self.loadAILanguageModelFromDatabase(inputLanguage)
        if self.model is None or not self.runModelSelfDiagnostic():
            raise Exception("AI language model load failed")
        return None
    self.model.setLLMContext(context) # Put past conversation context into
    llmInputParser = self.model.getInputParser()
    return llmInputParser.parseInput(inputString)

def generateLLMOutput(parsedInput):
    llmContext = self.model.getLLMContext()
    byteCodeResponse = self.model.convertInputToIntermediateResponse(
        parsedInput)
    if byteCodeResponse is None:
        raise Exception("LLM failed to produce intermediate response")
```

Python Sebagai Middleware



Workflow Pemadanan Data

-  **Load Data:** Tarik data perusahaan/pekerja dari berbagai sumber mentah.
-  **Normalisasi:** Standarisasi nama, alamat, NPWP, dan kode KBLI.
-  **Deduplication:** Hapus data ganda sebelum pemrosesan lebih lanjut.
-  **Matching:** Jalankan fuzzy matching lintas dataset dengan Python.
-  **Finalisasi:** Simpan hasil akhir ke PostgreSQL untuk analisis lanjutan.

Python + SQL = Kombinasi Terbaik



Terbaik untuk penyimpanan data terstruktur, menjaga integritas data, melakukan join tabel besar, dan agregasi cepat.



Terbaik untuk pemrosesan logika lanjutan, fuzzy matching, automasi workflow, dan integrasi lintas platform.

Hasilnya: Proses lebih cepat, Error menurun, Konsistensi meningkat.

**II Data Analyst & DBA bukan cuma penulis query.
Mereka adalah engineer yang mengalirkan data lintas
sistem.**

Mindset Baru

Sesi Quick Demo

1

Ingest & Norm

Membaca CSV & Excel, lalu melakukan normalisasi teks secara otomatis.

2

Merge & Match

Menggabungkan dataset dan melakukan fuzzy match untuk mencari duplikasi.

3

Load & View

Load hasil ke PostgreSQL dan visualisasi dashboard mini dengan Streamlit.

The Superpower

Python bukan sekadar teknis tambahan.

Python adalah superpower untuk semua pekerjaan data.

Image Sources



<https://mailchimp.com/ctf/images/yzco4xsimv0y/1dCLdw7waiqbE8NeYaaLj1/b41a6af00094b57ec39e4a9df8c683cf/data-silos-1.png?w=580&q=70>

Source: mailchimp.com



<https://communityhealthnapavalley.org/wp-content/uploads/2022/10/pile-of-papers.jpg>

Source: communityhealthnapavalley.org



<https://www.careersingovernment.com/tools/wp-content/uploads/2025/11/2-7-scaled.jpg>

Source: www.careersingovernment.com



https://plus.unsplash.com/premium_photo-1664297950425-99a968926a74?fm=jpg&q=60&w=3000&ixlib=rb-4.1.0&ixid=M3wxMjA3fDB8MHxzZWFiY2h8NXx8ZGF0YSUyMHNjaWVuY2V8ZW58MHx8MHx8fDA%3D

Source: unsplash.com



<https://static.vecteezy.com/system/resources/thumbnails/057/628/894/small/futuristic-digital-network-connectivity-concept-with-blue-and-gold-nodes-photo.jpg>

Source: www.vecteezy.com