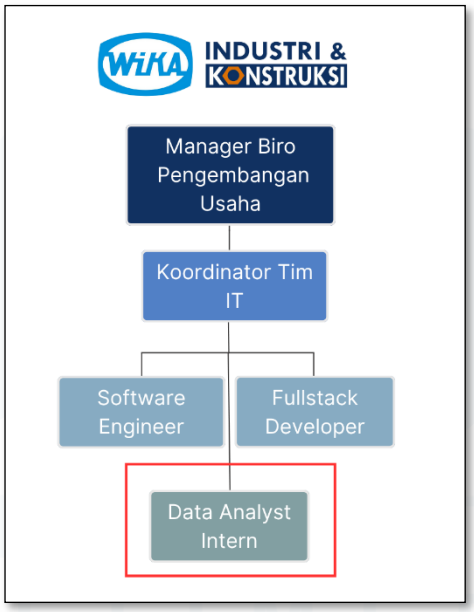
BAB III

PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

Kedudukan dalam program kerja nyata ini berada pada posisi sebagai *Data Analyst Intern* yang ditempatkan di Biro Pengembangan Usaha. Dalam menjalankan tugas, memiliki peran penting dalam mendukung pengumpulan, analisis, dan pengolahan data yang diperlukan untuk pengembangan usaha. Sebagaimana terlihat pada Gambar 3.1, memperoleh bimbingan dan pengawasan langsung dari Koordinator IT, yang juga bertindak sebagai Supervisor. Koordinator IT bertanggung jawab untuk memastikan pelaksanaan tugas berjalan dengan baik, memberikan arahan teknis, serta melakukan evaluasi berkala terhadap *progres* dan hasil kerja. Melalui bimbingan ini, diharapkan dapat mengembangkan keterampilan analisis data secara lebih mendalam.

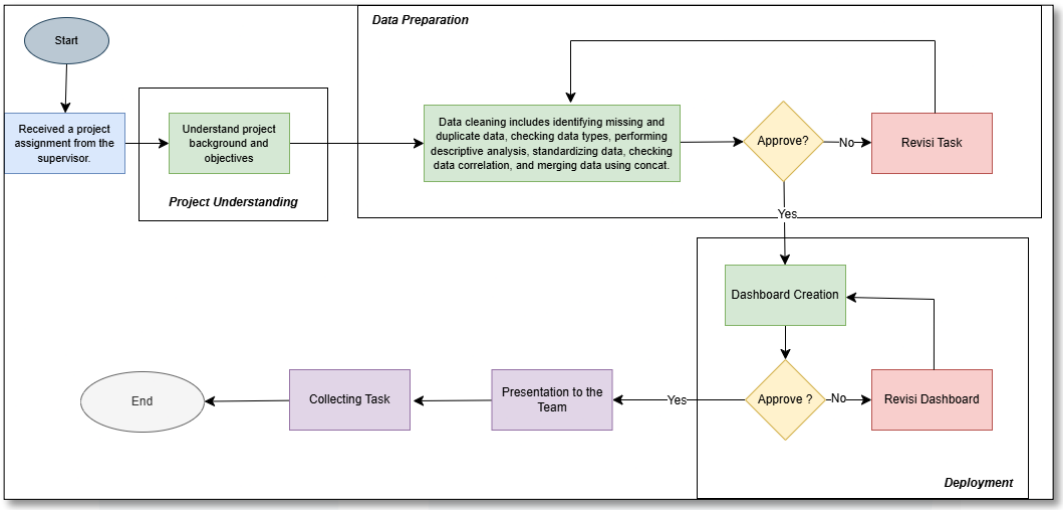


Gambar 3.1 Kedudukan Data Analyst Intern

Selama masa magang, diberikan tanggung jawab untuk terlibat dalam berbagai proyek utama yang meliputi pengumpulan, pembersihan, dan analisis data. Dalam hal ini, ditugaskan untuk memastikan bahwa data yang digunakan dalam analisis adalah akurat dan telah melalui proses pembersihan yang tepat. Selain itu, juga bertugas dalam merancang dan membuat dashboard yang dapat menyajikan hasil analisis secara visual, serta menyampaikan *insight* yang diperoleh dari olahan data tersebut. Bekerja sama dengan tim *Business Development* untuk berkoordinasi dalam mengidentifikasi dan mengkomunikasikan kebutuhan analisis yang diperlukan, sehingga hasil analisis dapat disesuaikan dengan tujuan dan strategi bisnis yang ada. Kerja sama ini memberikan kesempatan untuk memahami lebih dalam bagaimana data dapat digunakan untuk mendukung pengambilan keputusan dalam dunia bisnis.

Dalam berkoordinasi, dimanfaatkan platform *WhatsApp* dan *Zoom Meeting* serta rapat rutin setiap pagi, yang biasa disebut *Team Building Meeting (TBM),* dilaksanakan secara onsite dan terkadang secara online. TBM berfungsi untuk memastikan kelancaran komunikasi antar anggota tim, mendiskusikan perkembangan proyek yang sedang berjalan, serta menyelesaikan masalah yang muncul selama proses magang. Selain itu, TBM juga menjadi sarana untuk memperkuat kerja sama tim, berbagi pembaruan terkait tugas dan tanggung jawab masing-masing, serta memastikan bahwa seluruh anggota tim berada pada jalur yang sama dalam mencapai tujuan yang telah ditetapkan.

Setelah menjalani koordinasi rutin melalui platform *WhatsApp* dan mengikuti rapat *Team Building Meeting (TBM),* keterlibatan langsung dalam alur kerja yang lebih terstruktur mulai dijalankan. Berikut merupakan alur kerja yang diikuti selama pelaksanaan magang:



Gambar 3.2 Alur Kerja Pelaksanaan Magang

Proses alur kerja magang dimulai dengan penerimaan tugas proyek dari supervisor. Setelah itu, dilakukan pemahaman terhadap latar belakang dan tujuan proyek untuk memastikan pemahaman yang jelas mengenai ekspektasi yang diberikan. Selanjutnya, dimulai tahap *Data Preparation*, yang mencakup serangkaian langkah pembersihan data, seperti mengidentifikasi data yang hilang dan duplikat, memeriksa tipe data, melakukan analisis deskriptif, menstandarisasi data, memeriksa korelasi antar variabel, dan menggabungkan data menggunakan metode *concat*. Setelah data dipersiapkan, diminta persetujuan untuk melanjutkan ke langkah berikutnya. Jika data disetujui, proses dilanjutkan dengan *Dashboard Creation*, yaitu pembuatan *dashboard* untuk menyajikan hasil analisis secara visual. Setelah *dashboard* dibuat, kembali diminta persetujuan atas *dashboard* tersebut. Jika disetujui, hasil dan *dashboard* kemudian dipresentasikan kepada tim untuk mendapatkan umpan balik lebih lanjut. Setelah presentasi, jika disetujui, proyek siap untuk *Deployment*, yang menandakan bahwa tugas magang telah selesai dan diterapkan. Jika ada tahap yang tidak disetujui, diminta untuk melakukan revisi, baik pada tugas, *dashboard*, atau presentasi. Proses ini diakhiri ketika proyek berhasil diselesaikan dan diterapkan, menandakan bahwa project telah mencapai titik akhir.

3.2 Tugas dan Uraian Kerja Magang

Magang dilaksanakan dari tanggal 2 Februari hingga 30 April 2025 selama 704 jam, dengan tanggung jawab yang diberikan untuk mengelola tiga proyek analisis data sekaligus pembuatan *dashboard*. Proyek-proyek tersebut meliputi analisis data dan pembuatan *dashboard* untuk data cuti dan perizinan, kondisi serta panjang jembatan nasional, dan data pesanan pembelian. Dalam menjalankan proyek-proyek tersebut terdapat beberapa tools yang digunakan yaitu Jupyter Notebook, Power BI Desktop, Power BI Website, dan Canva. Selama program magang, terdapat berbagai target yang perlu dicapai, yang mencakup pemahaman yang mendalam tentang proses analisis data yang sedang dikerjakan, pelaksanaan tahapan *cleaning* data yang akan digunakan dalam analisis, serta kemampuan untuk memilih dan menerapkan visualisasi data yang sesuai agar hasil olahan data dapat dipahami oleh *user* non teknik, yang tentunya tetap mengikuti arahan yang diberikan oleh supervisor.

Sehubungan dengan pemenuhan kebijakan privasi yang diterapkan oleh perusahaan, beberapa bagian dari tangkapan layar yang berkaitan dengan pekerjaan akan disamarkan atau diburamkan sesuai dengan standar etika yang berlaku. Langkah ini diambil untuk menjaga keamanan dan integritas data yang terdapat dalam tangkapan layar tersebut. Pada Tabel 3.1 disajikan *timeline* yang menggambarkan berbagai kegiatan yang dilaksanakan selama masa magang :

Tabel 3.1 Uraian Pelaksanaan Kerja Magang

No	Daily Task	Tanggal Mulai	Tanggal Selesai	Total Hari Kerja
1	Pengenalan Lingkungan Kerja, struktur divisi, tanggung jawab pekerjaan, serta prosedur dan proyek yang akan dijalankan selama masa magang.	03/02/2025	3/2/2025	**1**
2	Mempelajari materi mengenai urutan dalam Proses Pembersihan Data mencakup identifikasi data yang tidak valid, penanganan data yang hilang, serta standarisasi format data	4/2/2025	5/2/2025	**2**
3	Melakukan *Data Cleaning* terhadap dataset perizinan dan cuti dari tahun 2023- 2024, meliputi : Memahami	6/2/2025	10/2/2025	**4**

No	Daily Task	Tanggal Mulai	Tanggal Selesai	Total Hari Kerja
	struktur data, identifikasi data hilang dan duplikat, pengecekan tipe data, memilih fitur yang digunakan, transformasi data, konversi tipe data, terakhir memastikan data sudah bersih dan siap unntuk digunakan			
4	Membuat *dashboard* Power BI untuk visualisasi data perizinan dan cuti, yang menampilkan grafik tren perizinan bulanan, pola pengajuan berdasarkan jenis dan durasi, serta pengguna dengan frekuensi cuti dan izin terbanyak.	11/2/2025	14/2/2025	**4**
5	*Brainstorming* Teknik *Storytelling* Dalam Visualisasi Data *(1)*	15/2/2025	18/2/2025	**3**
6	*Brainstorming* analisis data selanjutnya	19/2/2025	25/2/2025	**6**
7	Melakukan *Data Cleaning* terhadap dataset Kondisi Jembatan Nasional dari tahun 2023- 2024, meliputi Memahami struktur data, identifikasi data hilang dan duplikat, pengecekan tipe data, analisis deskriptif, Standarisasi, pengecekkan korelasi dan melakukan penggabungan data menggunakan concat	26/2/2025	8/3/2025	**10**
8	Mendesain dan membuat *dashboard* Power BI untuk visualisasi data Kondisi Jembatan Nasional yang menampilkan grafik tren perizinan bulanan, pola pengajuan berdasarkan jenis dan durasi, serta pengguna dengan frekuensi cuti dan izin terbanyak.	10/3/2025	13/3/2025	**4**
9	*Brainstorming* Teknik *Storytelling* Dalam Visualisasi Data (2)	14/3/2025	18/3/2025	**4**
10	Melakukan *Data Cleaning* terhadap data Panjang kondisi jembatan diindonesia, meliputi identifikasi data hilang dan duplikat, pengecekan tipe data, analisis deskriptif, Standarisasi, pengecekkan korelasi dan melakukan penggabungan data menggunakan concat	19/3/2025	12/4/2025	**13**
11	Membuat *dashboard* Power BI untuk visualisasi data jumlah kondisi jembatan Indonesia yang menampilkan grafik tren panjang kondisi jembatan, membuat top 5 dengan kondisi jembatan yang	14/4/2025	17/4/2025	**4**

No	Daily Task	Tanggal Mulai	Tanggal Selesai	Total Hari Kerja
	terpanjang, list panjang jembatan dengan berbagai kondisi perprovinsi.			
12	*Brainstorming Teknik Storytelling Dalam Visualisasi Data (3)*	18/4/2025	21/4/2025	**3**
13	Melakukan *Data Cleaning* terhadap Dataset Status Project, meliputi Meamahami struktur data, Identifikasi data *fields* dan Pengecekkan Kembali untuk memastikan data sudah bersih.	22/4/2025	28/4/2025	**6**
14	Membuat *dashboard* menggunakan Power BI untuk menganalisis cuti Status Proyek.	26/4/2025	30/4/2025	**4**

3.2.1 Pengenalan Lingkungan Kerja, Struktur Divisi, Tanggung Jawab Pekerjaan, Serta Prosedur Dan Proyek Yang Akan Dijalankan Selama Masa Magang.

Pada awal pelaksanaan magang, diberikan kesempatan untuk memahami dan mengenal lebih dalam lingkungan kerja di Divisi Biro Pengembangan Usaha PT Wijaya Karya Industri & Konstruksi (WIKAIKON). Tahap orientasi ini dirancang untuk memberikan gambaran menyeluruh mengenai dinamika operasional, proses bisnis, dan strategi pengembangan usaha yang diterapkan dalam perusahaan konstruksi dan industri besar ini. PT Wijaya Karya Industri & Konstruksi (WIKAIKON) berlokasi di Tamasari HIVE Office, Jl. DI. Panjaitan No.Kav 2 Lantai 8, Cipinang Cempedak, Kecamatan Jatinegara, Jakarta Timur, Daerah Khusus Ibukota Jakarta.

Gambar 3.3 Gedung TamanSari *HIVE Offfice*

Dalam tahap orientasi, dilakukan perkenalan dengan tim inti yang berada di Divisi Biro Pengembangan Usaha, yang menjadi mentor serta rekan kerja selama periode magang. Tim tersebut terdiri dari individu-individu dengan latar belakang profesional yang beragam, masing-masing memiliki keahlian dalam bidang yang berbeda, mulai dari pengembangan sistem, manajemen proyek konstruksi, riset pasar, hingga pengelolaan portofolio produk dan jasa. Setiap anggota tim memiliki peran yang sangat penting dalam mendukung keberhasilan operasional divisi dan pencapaian tujuan pengembangan usaha perusahaan.

Selanjutnya, diperkenalkan pula budaya kerja yang diterapkan di PT Wijaya Karya Industri & Konstruksi. Budaya kerja ini mengedepankan nilai-nilai inti perusahaan, seperti Amanah, Kompeten, Harmonis, Loyal, Adaptif, dan Kolaboratif. Selain itu, prosedur standar operasional (SOP) yang berlaku di Divisi Biro Pengembangan Usaha dijelaskan secara rinci. SOP ini mencakup berbagai tahapan, mulai dari perencanaan dan pengembangan proyek-proyek konstruksi besar, hingga penggunaan

berbagai alat dan teknologi pendukung seperti perangkat lunak manajemen proyek, sistem pengelolaan risiko, dan platform perencanaan sumber daya. Pemahaman yang mendalam mengenai SOP ini menjadi landasan yang sangat penting untuk menyelaraskan tujuan pribadi dengan visi dan misi perusahaan, serta memahami ekspektasi perusahaan terhadap kinerja yang diberikan.

Deskripsi pekerjaan sebagai bagian dari Divisi Biro Pengembangan Usaha WIKAIKON dijelaskan secara rinci oleh supervisor. Tanggung jawab utama meliputi analisis kelayakan proyek, riset pasar terkait dengan kebutuhan industri konstruksi, serta pengelolaan data proyek. Selain itu, diberikan arahan terkait ekspektasi perusahaan mengenai kontribusi yang diharapkan selama masa magang, seperti membantu tim dalam pengumpulan, pembersihan, dan pengelolaan data proyek, mematuhi peraturan dan prosedur yang berlaku, serta berkomunikasi secara efektif dengan anggota tim lainnya. Proses orientasi ini bertujuan untuk memastikan bahwa pemahaman mengenai tugas dan tanggung jawab sudah jelas, sehingga kontribusi yang optimal dapat diberikan selama masa magang.

Orientasi ini juga memberikan kesempatan untuk beradaptasi dengan lingkungan kerja profesional, yang mencakup etika kerja yang tinggi, manajemen waktu yang efektif, dan kemampuan untuk bekerja secara kolaboratif dalam tim. Dengan pemahaman yang lebih mendalam tentang proses analisis data yang dilakukan di WIKAIKON, rasa percaya diri dapat ditingkatkan untuk memulai tugas-tugas yang diberikan serta mempersiapkan diri menghadapi berbagai tantangan yang mungkin muncul selama masa magang.

3.2.2 Mempelajari Materi Mengenai Urutan Dalam Proses Pembersihan Data Mencakup Identifikasi Data Yang Tidak Valid, Penanganan Data Yang Hilang, Serta Standarisasi Format Pada Dataset Perizinan Dan Cuti.

Pada Pada hari kedua pelaksanaan magang, dilakukan pelatihan intensif mengenai proses pembersihan data yang difasilitasi langsung oleh supervisor. Pelatihan ini menjadi momen penting untuk memahami bagaimana standar operasional dan prosedur data cleaning diterapkan secara profesional di lingkungan industri, yang tentu berbeda dari pendekatan yang umumnya diajarkan di dunia akademik. Proses ini menjadi fondasi utama dalam analisis data, karena bertujuan memastikan bahwa data yang digunakan telah bebas dari duplikasi, nilai kosong, serta inkonsistensi, sehingga dapat mendukung proses analisis secara optimal [6].

Selama sesi pelatihan, diperkenalkan berbagai teknik yang lazim digunakan dalam kegiatan pembersihan data, mulai dari penghapusan data ganda, penanganan nilai yang hilang (missing values), normalisasi data, hingga identifikasi serta koreksi data yang tidak sesuai. Materi pelatihan disampaikan melalui pendekatan studi kasus yang relevan dengan kebutuhan dan permasalahan aktual yang dihadapi perusahaan, sehingga memudahkan dalam mengaitkan teori dengan praktik yang sesungguhnya. Pemahaman ini menekankan bahwa kualitas data yang baik merupakan syarat utama untuk menghasilkan output analisis yang akurat dan dapat diandalkan sebagai dasar pengambilan keputusan bisnis.

Pada hari yang sama, juga diberikan penugasan untuk melakukan proses pembersihan data sekaligus pembuatan *dashboard* dari dataset perizinan dan percutian tahun 2023–2024. Tujuan dari penugasan ini adalah untuk menyajikan data yang telah bersih dan terstruktur dalam bentuk visual yang informatif. *Dashboard* yang dikembangkan dirancang untuk digunakan oleh Human Capital (HC) sebagai alat bantu dalam memantau kualitas kerja dan kedisiplinan karyawan secara lebih sistematis, sehingga

proses evaluasi kinerja dapat dilakukan dengan lebih efisien dan berbasis data.

3.2.3 Melakukan *Data Cleaning* Terhadap Dataset Perizinan dan Percutian dari Tahun 2023- 2024.

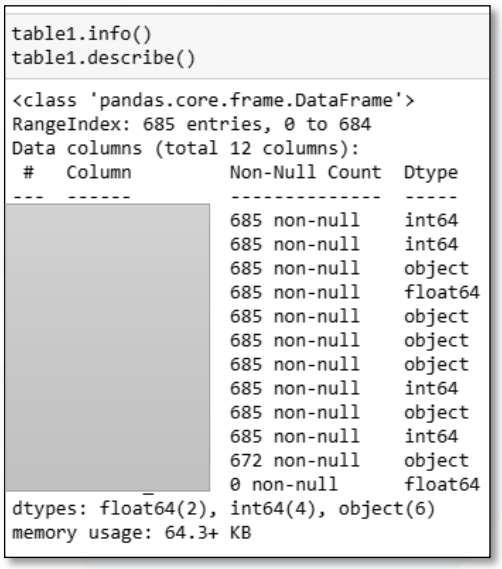
Data cleaning adalah proses yang sangat penting dalam siklus pengolahan data. Tujuan utama dari *data cleaning* adalah untuk memastikan bahwa dataset yang digunakan dalam analisis bersih, akurat, dan konsisten[7]. Pada bagian ini, dilakukan proses *data cleaning* terhadap dataset perizinan dan cuti pegawai untuk periode tahun 2023 hingga 2024. Proses ini sangat penting untuk memastikan kualitas dan integritas data yang digunakan dalam analisis lebih lanjut. Dalam tahap ini, data yang terkontaminasi, tidak konsisten, atau tidak lengkap akan diperbaiki atau dihapus untuk memastikan dataset yang lebih akurat, konsisten, dan siap untuk mendukung pengambilan keputusan yang lebih efektif. Berikut tahapan yang dilakukan untuk melakukan pembersihan data:

1. *Understanding the structure of Licensing Dataset*

   Langkah pertama yang dilakukan pada data ini adalah memahami struktur dataset dengan melihat jumlah baris dan kolom serta nilai-nilai statistik dasar dari data tersebut. Dataset ini terdiri dari 1618 baris dan 12 kolom, yang mencakup berbagai informasi terkait cuti pegawai. Beberapa kolom, seperti at_*delete*, memiliki 303 nilai non-null, yang menandakan banyaknya data kosong *(missing values)* pada kolom ini. Hal ini dapat menunjukkan bahwa sebagian besar data tidak dihapus atau tidak memiliki informasi tentang tanggal penghapusan.

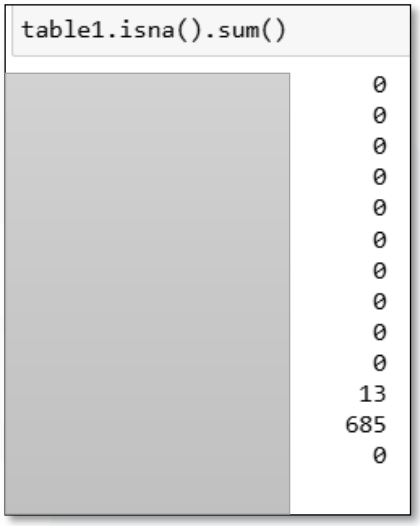
   Selain itu, kolom-kolom seperti hari_cuti, mulai_cuti, akhir_cuti, dan keperluan juga memiliki nilai yang hilang. Misalnya, kolom hari_cuti hanya memiliki 1610 nilai non-null, sedangkan kolom keperluan hanya memiliki 1579 nilai non-null, yang berarti ada beberapa catatan cuti yang tidak memiliki informasi tentang alasan

atau keperluan cuti. Untuk mempersiapkan data lebih lanjut, nilai-nilai yang hilang ini perlu ditangani agar tidak mengganggu proses analisis berikutnya.



Gambar 3.4 *Understanding the structure of Licensing Datasets*

Setelah menganalisis dataset, langkah selanjutnya adalah memeriksa adanya nilai yang hilang *(missing values)* pada setiap kolom menggunakan fungsi *.isna().sum()*. Hasil pemeriksaan menunjukkan bahwa sebagian besar kolom tidak memiliki nilai yang hilang, yang semua memiliki nilai lengkap tanpa missing values. Namun, ada dua kolom yang perlu mendapat perhatian lebih lanjut. Kolom lampiran_izin memiliki 13 nilai yang hilang, yang menunjukkan bahwa beberapa izin tidak memiliki lampiran yang terdaftar. Selain itu, kolom *deleted_at* memiliki 685 nilai yang hilang, yang bisa mengindikasikan bahwa sebagian besar data tidak memiliki tanggal penghapusan atau belum dihapus. Kolom ini perlu ditangani lebih lanjut, apakah data yang hilang akan dihapus, digantikan dengan nilai tertentu, atau ditangani dengan cara lainnya, tergantung pada relevansi kolom tersebut dalam analisis.



Gambar 3.5 *Identify Missing Data Fields of Licensing Datasets*

2. Feature Selection of Licensing Dataset

Setelah memeriksa nilai yang hilang, langkah selanjutnya adalah melakukan penghapusan kolom yang tidak relevan atau memiliki banyak data yang hilang dan tidak diperlukan dalam analisis lebih lanjut. Dalam hal ini, *kolom deleted_at, i_status,* dan lampiran_izin dihapus dari dataset menggunakan perintah *drop(columns=['deleted_at','i_status','lampiran_izin'],inplace=True).*Kolom *deleted_at* memiliki banyak nilai kosong yang tidak memberikan informasi penting untuk analisis, sementara kolom *i_status* dan lampiran_izin juga tidak diperlukan untuk tujuan analisis yang akan dilakukan. Dengan penghapusan kolom-kolom tersebut, dataset menjadi lebih bersih dan terfokus pada informasi yang relevan dan akan lebih mempermudah analisis lebih lanjut.

```
table1.drop(columns=['deleted_at','i_status','lampiran_izin'], inplace=True)
table1
```

0	1	65	2023-07-11	0.5	08:21:00	izin_dtg_terlambat	macet (test)	2023-07-11 11:22:37	65	setengah hari
1	2	65	2023-08-02	1.0	00:00:00	izin_penuh	izin ayang kawin :)	2023-08-02 09:40:42	65	satu hari
2	3	63	2023-08-07	1.0	00:00:00	izin_dlm_kerja	testing aplikasi cuti	2023-08-07 09:11:40	63	satu hari
3	4	9	2023-08-21	0.5	08:40:00	izin_dtg_terlambat	Trouble di Jalan	2023-08-21 09:09:42	9	setengah hari
4	5	9	2023-08-21	0.5	08:40:00	izin_dtg_terlambat	Trouble di Jalan	2023-08-21 09:10:03	9	setengah hari
...	...	...	...	...	...	...	...	...	...	...
680	685	197	2024-12-19	1.0	00:00:00	izin_sakit	Izin Sakit (Dengan Surat Keterangan Sakit)	2024-12-20 07:46:21	197	satu hari
681	686	197	2024-12-20	1.0	00:00:00	izin_sakit	Izin Sakit (Dengan Surat Keterangan Sakit)	2024-12-20 07:46:48	197	satu hari
682	687	137	2024-12-27	0.5	16:00:00	izin_plg_awal	Periksa kesehatan	2024-12-27 16:59:19	137	setengah hari
683	688	93	2024-12-30	1.0	00:00:00	izin_penuh	Cuti tahunan	2024-12-30 14:48:24	93	satu hari
684	689	88	2025-01-09	1.0	00:00:00	izin_dlm_kerja	kondisi alam (banjir)	2025-01-09 10:43:12	88	satu hari

685 rows × 10 columns

Gambar 3.6 *Feature Selection of Licensing Dataset*

3. *Data Transformation of Licensing Dataset*

Selanjutnya, untuk mempermudah pemahaman terkait durasi izin yang diajukan, kolom deskripsi_hari_izin ditambahkan dengan menggunakan fungsi *apply()*. Kolom ini diisi berdasarkan nilai pada kolom jml_hari_izin, yang menunjukkan jumlah hari izin yang diajukan. Dengan menggunakan logika lambda, jika jml_hari_izin bernilai 0.5, maka deskripsi akan diubah menjadi 'setengah hari'; jika jml_hari_izin bernilai 1, deskripsi akan menjadi 'satu hari'; dan jika lebih dari 1, deskripsi akan menjadi 'lebih dari satu hari'. Dengan langkah ini, informasi tentang durasi izin menjadi lebih mudah dipahami dan lebih siap untuk dianalisis lebih lanjut.

```
table1['deskripsi_hari_izin'] = table1['jml_hari_izin'].apply(
    lambda x: 'setengah hari' if x == 0.5 else ('satu hari' if x == 1 else 'lebih dari satu hari')
)
table1.head()
```

1	65	2023-07-11	0.5	08:21:00	izin_dtg_terlambat	macet (test)	0	2023-07-11 11:22:37	65	1_izin.jpeg	NaN	setengah hari
2	65	2023-08-02	1.0	00:00:00	izin_penuh	izin ayang kawin :)	0	2023-08-02 09:40:42	65	2_izin.jpg	NaN	satu hari
3	63	2023-08-07	1.0	00:00:00	izin_dlm_kerja	testing aplikasi cuti	0	2023-08-07 09:11:40	63	3_izin.jpeg	NaN	satu hari
4	9	2023-08-21	0.5	08:40:00	izin_dtg_terlambat	Trouble di Jalan	0	2023-08-21 09:09:42	9	4_izin.png	NaN	setengah hari
5	9	2023-08-21	0.5	08:40:00	izin_dtg_terlambat	Trouble di Jalan	0	2023-08-21 09:10:03	9	5_izin.png	NaN	setengah hari

Gambar 3.7 *Data Transformation of Licensing Dataset*

4. *Data Type Conversion of Licensing Dataset*

Langkah selanjutnya adalah mengonversi kolom tgl_izin menjadi tipe data *datetime* menggunakan fungsi pd.to_datetime(). Proses ini memungkinkan kita untuk menangani tanggal dengan lebih efektif dalam analisis. Jika terdapat nilai yang tidak valid (seperti format tanggal yang salah), maka nilai tersebut akan diubah menjadi *Not a Time* (NaT). Setelah itu, untuk memastikan hanya baris dengan data tanggal yang valid yang tersisa, baris-baris yang memiliki nilai NaT pada kolom tgl_izin dihapus menggunakan *dropna*(*subset*=["tgl_izin"]). Dengan langkah ini, dataset akan lebih bersih dan siap digunakan untuk analisis yang melibatkan waktu atau tanggal.

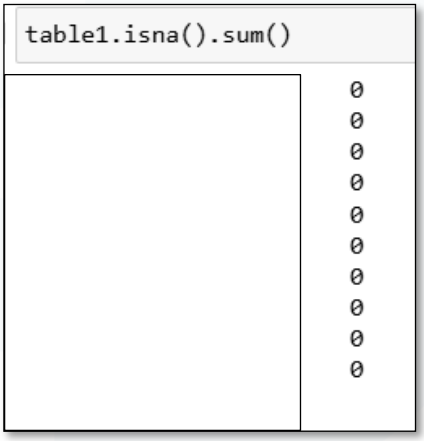
```
table1["tgl_izin"] = pd.to_datetime(table1["tgl_izin"], errors='coerce')  # Ubah ke datetime, invalid jadi NaT
table1 = table1.dropna(subset=["tgl_izin"])  # Hapus baris dengan NaT
```

Gambar 3.8 *Data Type Conversion of Licensing Dataset*

6. *Rechecking Missing Data*

Setelah melakukan proses konversi dan penghapusan data yang tidak valid, dilakukan pemeriksaan kembali terhadap dataset untuk memastikan tidak ada nilai yang hilang (missing values) pada

kolom-kolom penting. Hasilnya, dengan menggunakan table1.*isna().sum(),* terlihat bahwa semua kolom dalam dataset sekarang memiliki nilai lengkap tanpa adanya *missing values*. Hal ini menunjukkan bahwa dataset telah dibersihkan dengan baik dan siap untuk analisis lebih lanjut tanpa adanya data yang hilang yang dapat memengaruhi hasil analisis.



Gambar 3.9 *Rechecking Missing Data*

Selanjutnya, proses data cleaning juga dilakukan terhadap dataset cuti karyawan. Tahapan ini mencakup identifikasi dan penghapusan data duplikat, perbaikan data yang tidak konsisten, serta penanganan nilai yang hilang atau tidak valid. Tujuan dari pembersihan ini adalah untuk memastikan bahwa informasi terkait cuti karyawan yang digunakan dalam analisis benar-benar akurat dan dapat diandalkan. Dengan data yang bersih, perusahaan dapat memantau dan mengelola jadwal cuti tenaga kerja secara lebih efektif, sehingga alokasi sumber daya manusia dalam proyek konstruksi dapat dilakukan secara optimal dan tidak mengganggu kelangsungan operasional proyek. Berikut adalah tahapan yang dilakukan selama proses pembersihan data:

7. Understanding the Structure of Employee Leave Dataset

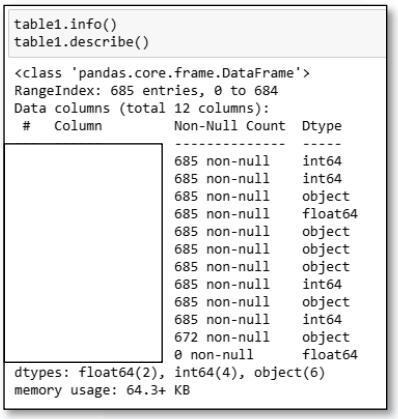
Pada tahap awal ini, dilakukan pemuatan dataset yang berisi informasi terkait perizinan pegawai menggunakan fungsi *read_csv()* dari pustaka pandas. Dataset ini berisi 685 entri data yang terdiri dari 12 kolom. Melalui perintah table1.info() dan table1.*describe(),* dilakukan pemeriksaan informasi dan statistik deskriptif terhadap dataset yang dimuat. Hasil dari table1.info() menunjukkan bahwa seluruh kolom memiliki jumlah data yang konsisten (non-null) kecuali kolom deleted_at, yang berisi data kosong (NaN). Hal ini menunjukkan bahwa data di kolom tersebut tidak relevan untuk analisis atau masih belum terisi.

Sementara itu, hasil dari table1.*describe()* memberikan gambaran statistik dasar seperti jumlah, rata-rata (*mean*), standar deviasi (std), nilai minimum (*min*), dan maksimum (*max*) untuk kolom numerik dalam dataset. Sebagai contoh, kolom id_izin memiliki nilai rata-rata 346.89 dan rentang nilai yang cukup besar (7 hingga 689), sementara jml_hari_izin memiliki rata-rata 0.81 hari dengan variasi yang lebih kecil antara 0.5 hingga 1 hari.

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```
file_path = 'C:/Users/User/Documents/magang/pbi/wika_cuti_magang.csv/wika_cuti_magang_table_tb_cuti.csv'
table1 = pd.read_csv(file_path, delimiter=';', quotechar='"')
table1.head()
```

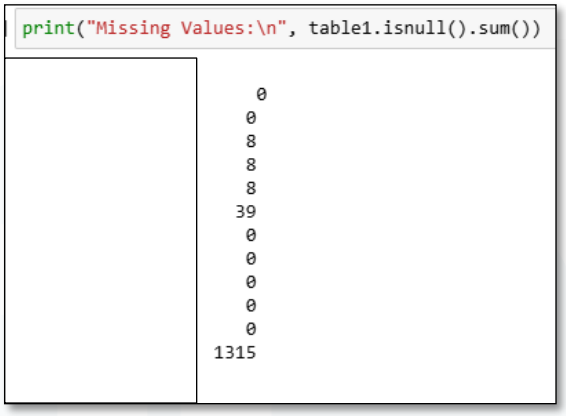
	id_cuti	id_user	jml_hari_cuti	tgl_mulai_cuti	tgl_akhir_cuti	keperluan	c_alamat_lgkp	c_status	input_date	input_user	updated_at	deleted_at
0	2	199	1.0	2022-08-26	2022-08-26	Acara keluarga	jakarta	1	2022-08-22	199	2025-02-04 11:03:59	NaN
1	4	13	1.0	2022-08-26	2022-08-26	Keperluan Keluarga	jakarta	1	2022-08-22	13	2025-02-04 11:03:59	NaN
2	6	195	1.0	2022-08-29	2022-08-29	Acara Keluarga	jakarta	1	2022-08-22	195	2025-02-04 11:03:59	NaN
3	7	7	4.0	2022-09-07	2022-09-12	Menengok Keluarga sekaligus menghadiri Pernika...	jakarta	1	2022-08-22	7	2025-02-04 11:03:59	NaN
4	9	94	4.0	2022-09-21	2022-09-26	Keperluan Keluarga	jakarta	0	2022-08-22	94	2025-02-04 11:03:59	2023-05-02 18:56:33



Gambar 3.10 *Understanding the Structure of Employee Leave Dataset*

8. Identify Missing Data Fields of Employee Leave Dataset

Setelah memahami struktur dataset dan melakukan analisis statistik dasar, langkah berikutnya adalah mengecek keberadaan nilai yang hilang (*missing values*) pada dataset. Hasil pemeriksaan menunjukkan bahwa terdapat beberapa kolom dengan missing values yang perlu ditangani sebelum melanjutkan ke tahap analisis lebih lanjut. Berdasarkan hasil pengecekan, *kolom id_cuti, id_user, c_alamat_lgkp, c_status, input_date, input_user, dan updated_at* tidak memiliki nilai yang hilang, yang menunjukkan bahwa data pada kolom-kolom ini lengkap. Sementara itu, kolom jml_hari_cuti, tgl_mulai_cuti, dan tgl_akhir_cuti masing-masing memiliki 8 missing values, yang berarti ada beberapa entri cuti yang tidak memiliki informasi terkait jumlah hari atau tanggal mulai dan akhir cuti. Kolom keperluan memiliki 39 missing values, yang mengindikasikan bahwa tidak semua entri cuti dilengkapi dengan informasi mengenai alasan atau keperluan cuti. Terakhir, kolom *deleted_at* memiliki 1315 missing values, yang menandakan bahwa sebagian besar data dalam kolom ini tidak memiliki informasi tentang tanggal penghapusan.



Gambar 3.11 *Identify Missing Data Fields of Employee Leave Dataset*

karena beberapa kolom memiliki missing values, langkah selanjutnya adalah menangani *missing values* ini, agar data dapat digunakan untuk analisis lebih mendalam. Khususnya kolom yang banyak mengandung data kosong, perlu dikaji lebih lanjut apakah kolom ini relevan dan apakah data tersebut perlu dipertahankan atau dihapus.

9. *Feature Selection of Employee Leave Dataset*

Setelah memeriksa *missing values* dan melakukan pengecekan pada kolom-kolom yang memiliki nilai hilang, langkah selanjutnya adalah menangani nilai yang hilang tersebut. Untuk memastikan dataset lebih konsisten dan siap untuk analisis lebih lanjut, dilakukan penghapusan entri yang memiliki nilai hilang menggunakan metode *dropna()*. Setelah proses penghapusan, dataset yang tersisa memiliki 1618 entri, di mana kolom-kolom yang sebelumnya mengandung *missing values* kini telah bersih dari data yang tidak lengkap.

```
table1 = table1.dropna(subset=['jml_hari_cuti', 'tgl_mulai_cuti', 'tgl_akhir_cuti'])
table1
```

0	2	199	1.0	2022-08-26	2022-08-26	Acara keluarga	jakarta	1	2022-08-22	199	2025-02-04 11:03:59	NaN
1	4	13	1.0	2022-08-26	2022-08-26	Keperluan Keluarga	jakarta	1	2022-08-22	13	2025-02-04 11:03:59	NaN
2	6	195	1.0	2022-08-29	2022-08-29	Acara Keluarga	jakarta	1	2022-08-22	195	2025-02-04 11:03:59	NaN
3	7	7	4.0	2022-09-07	2022-09-12	Menengok Keluarga sekaligus menghadiri Pernika...	jakarta	1	2022-08-22	7	2025-02-04 11:03:59	NaN
4	9	94	4.0	2022-09-21	2022-09-26	Keperluan Keluarga	jakarta	0	2022-08-22	94	2025-02-04 11:03:59	2023-05-02 18:56:33
...	...	...	...	...	...	...	...	...	...	...	...	...
1613	1667	281	1.0	2025-01-24	2025-01-24	Keperluan keluarga	jakarta	0	2025-01-07	281	2025-02-04 11:03:59	NaN
1614	1668	281	1.0	2025-01-28	2025-01-28	Keperluan keluarga	jakarta	0	2025-01-07	281	2025-02-04 11:03:59	2025-01-07 17:00:38
1615	1669	7	1.0	2025-01-10	2025-01-10	Keluarga	jakarta	1	2025-01-09	7	2025-02-04 11:03:59	NaN
1616	1670	282	3.0	2025-01-13	2025-01-15	Dampingi operasi anak	jakarta	0	2025-01-09	282	2025-02-04 11:03:59	NaN
1617	1671	60	0.0	0000-00-00	0000-00-00	NaN	jakarta	0	2025-01-10	60	2025-02-04 11:03:59	NaN

Gambar 3.12 *Feature Selection of Employee Leave Dataset*

Setelah menangani *missing values* dan menghapus entri yang tidak lengkap, langkah berikutnya adalah menghapus kolom yang tidak relevan atau tidak diperlukan untuk analisis lebih lanjut. Dalam hal ini, kolom *deleted_at* dihapus karena tidak memberikan kontribusi signifikan terhadap analisis data perizinan dan cuti pegawai. Kolom ini mengandung banyak nilai kosong dan tidak memiliki informasi yang relevan untuk tujuan analisis, sehingga dihapus untuk menyederhanakan dataset dan menjaga fokus pada data yang lebih penting. Setelah penghapusan kolom ini, dataset menjadi lebih bersih dan siap untuk tahap analisis selanjutnya.

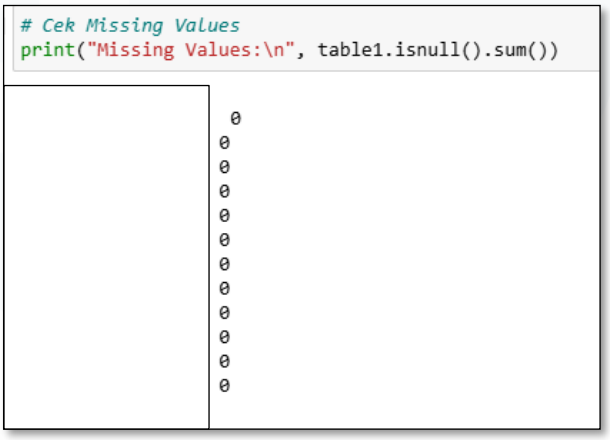
```
table1.drop(columns=['deleted_at'], inplace=True)
table1
```

0	2	199	1.0	2022-08-26	2022-08-26	Acara keluarga	jakarta	1	2022-08-22	199	2025-02-04 11:03:59
1	4	13	1.0	2022-08-26	2022-08-26	Keperluan Keluarga	jakarta	1	2022-08-22	13	2025-02-04 11:03:59
2	6	195	1.0	2022-08-29	2022-08-29	Acara Keluarga	jakarta	1	2022-08-22	195	2025-02-04 11:03:59
3	7	7	4.0	2022-09-07	2022-09-12	Menengok Keluarga sekaligus menghadiri Pernika...	jakarta	1	2022-08-22	7	2025-02-04 11:03:59
4	9	94	4.0	2022-09-21	2022-09-26	Keperluan Keluarga	jakarta	0	2022-08-22	94	2025-02-04 11:03:59
...	...	...	...	...	...	...	...	...	...	...	...
1613	1667	281	1.0	2025-01-24	2025-01-24	Keperluan keluarga	jakarta	0	2025-01-07	281	2025-02-04 11:03:59
1614	1668	281	1.0	2025-01-28	2025-01-28	Keperluan keluarga	jakarta	0	2025-01-07	281	2025-02-04 11:03:59
1615	1669	7	1.0	2025-01-10	2025-01-10	Keluarga	jakarta	1	2025-01-09	7	2025-02-04 11:03:59

Gambar 3.13 *Feature Selection of Employee Leave Dataset 2*

10. Rechecking Missing Value Data of Employee Leave Dataset

Setelah melakukan penghapusan kolom yang tidak relevan dan menangani *missing values*, langkah selanjutnya adalah memverifikasi kembali dataset untuk memastikan bahwa tidak ada nilai yang hilang pada kolom-kolom yang tersisa. Hasil pemeriksaan menunjukkan bahwa semua kolom dalam dataset sekarang telah bersih dari nilai yang hilang (*missing values*), dengan seluruh kolom memiliki 0 *missing values*.



Gambar 3.14 *Rechecking Missing Value Data of Employee Leave Dataset*

11. Data Transformation of Employee Leave Datasets

Setelah memastikan bahwa tidak ada nilai yang hilang dalam dataset, langkah selanjutnya adalah menambahkan kolom baru untuk memberikan informasi tambahan terkait kolom 'keperluan'. Kolom baru yang ditambahkan bernama 'Keterangan', yang berfungsi untuk mengklasifikasikan apakah terdapat keterangan atau tidak pada kolom 'keperluan'. Dalam kolom 'Keterangan', jika nilai pada 'keperluan' adalah 'Tidak Ada Keterangan', maka kolom tersebut akan diisi dengan label 'Tidak Ada Keterangan'. Sebaliknya, jika terdapat informasi lain, kolom 'Keterangan' akan diisi dengan label 'Ada Keterangan'. Dengan penambahan kolom ini, dataset menjadi lebih informatif dan

memudahkan dalam pengelompokan data berdasarkan keberadaan keterangan.

```
# Menambahkan kolom baru 'Keterangan' berdasarkan kolom 'keperluan'
table1['Keterangan'] = table1['keperluan'].apply(lambda x: 'Tidak Ada Keterangan' if x == 'Tidak Ada Keterangan' else 'Ada Ke
table1.head()
```

0	2	199	1.0	2022-08-26	2022-08-26	Acara keluarga	jakarta	1	2022-08-22	199	2025-02-04 11:03:59	Ada Keterangan
1	4	13	1.0	2022-08-26	2022-08-26	Keperluan Keluarga	jakarta	1	2022-08-22	13	2025-02-04 11:03:59	Ada Keterangan
2	6	195	1.0	2022-08-29	2022-08-29	Acara Keluarga	jakarta	1	2022-08-22	195	2025-02-04 11:03:59	Ada Keterangan
3	7	7	4.0	2022-09-07	2022-09-12	Menengok Keluarga sekaligus menghadiri Pernika...	jakarta	1	2022-08-22	7	2025-02-04 11:03:59	Ada Keterangan
4	9	94	4.0	2022-09-21	2022-09-26	Keperluan Keluarga	jakarta	0	2022-08-22	94	2025-02-04 11:03:59	Ada Keterangan

Gambar 3.15 *Data Transformation of Employee Leave Datasets*

3.2.4. membuat *Dashboard* Menggunakan Power Bi Untuk Menganalisis Data Perizinan Dan Cuti Yang Menampilkan Tren, Perbandingan, Dan Analisis Terkait Perizinan Karyawan.

Setelah proses pembersihan data selesai, dataset yang telah diperbaiki dan disesuaikan dengan kebutuhan analisis akan digunakan untuk menghasilkan *dashboard* yang menggambarkan informasi terkait perizinan dan cuti pegawai. *Dashboard* ini akan memvisualisasikan data secara lebih jelas dan memberikan wawasan yang lebih mendalam mengenai tren dan pola cuti pegawai selama periode yang dimaksud. *Dashboard* ini dibuat menggunakan tools Power BI.

Dashboard Perizinan ini memberikan analisis komprehensif terkait permintaan izin di perusahaan untuk periode 2023-2025. Beberapa komponen utama yang terdapat dalam *dashboard* ini antara lain Perbandingan Total Jenis Perizinan, yang menunjukkan jumlah izin berdasarkan jenisnya, seperti izin sakit, izin keterlambatan, dan izin lainnya, sehingga memudahkan untuk melihat jenis perizinan yang paling sering diajukan oleh karyawan. Kemudian, terdapat Perbandingan Total Perizinan yang memperlihatkan total permintaan izin berdasarkan

durasi, apakah satu hari penuh atau setengah hari, yang membantu menganalisis durasi izin yang diambil oleh karyawan. Tren Permintaan Izin menyajikan fluktuasi permintaan izin dari waktu ke waktu, memungkinkan analisis terhadap periode dengan permintaan izin yang lebih tinggi, sehingga perusahaan bisa melakukan perencanaan sumber daya yang lebih baik. *Top* 5 Karyawan dengan Jam Perizinan Terbanyak mengidentifikasi lima karyawan dengan izin terbanyak, yang dapat digunakan untuk evaluasi distribusi izin dan kebijakan yang diterapkan perusahaan. Terakhir, Pola Perizinan memperlihatkan tren izin berdasarkan waktu, baik jumlah orang yang mengajukan izin maupun durasi izin yang diambil setiap bulannya, memberikan wawasan terkait kapan periode-periode tertentu karyawan lebih cenderung mengambil izin.

Gambar 3.16 *Dashboard of Licensing Datasets*

Dashboard kedua ini memberikan wawasan mengenai percutian di perusahaan untuk periode 2022-2025, dengan menampilkan berbagai informasi penting terkait pengelolaan cuti karyawan. Pada bagian pertama, terdapat grafik *Top* 5 *User* dengan Cuti Terbanyak Tahun 2022-2025, yang memperlihatkan lima karyawan yang paling sering mengambil cuti, beserta persentase jumlah cuti mereka dalam total perizinan yang ada. Grafik ini berguna untuk mengidentifikasi karyawan yang sering memanfaatkan jatah cuti mereka, sehingga membantu perusahaan dalam merencanakan distribusi cuti yang lebih adil dan efisien. Selanjutnya, grafik *Trend* Pengajuan Cuti menunjukkan fluktuasi pengajuan cuti dari tahun ke tahun, memperlihatkan periode dengan lonjakan atau penurunan permintaan cuti. Hal ini sangat berguna untuk menganalisis pola musiman atau faktor lain yang memengaruhi pengajuan cuti, serta membantu perusahaan dalam merencanakan kebutuhan operasional yang lebih baik di masa depan. Terakhir, grafik Pola Perizinan memberikan gambaran mengenai jumlah hari cuti yang diajukan dibandingkan dengan jumlah orang yang mengajukan cuti pada periode tertentu. Grafik ini menunjukkan pola perilaku karyawan dalam mengambil cuti, apakah mereka lebih cenderung mengambil cuti dalam jumlah sedikit atau dalam jumlah hari yang lebih banyak.

Gambar 3.17 *Dashboard of Employee Leave Datasets*

Pemakaian *dashboard* ini dapat dilakukan oleh tim HR atau manajer untuk memantau dan menganalisis pola perizinan dan percutian secara *real-time*. Data yang diperoleh dapat digunakan sebagai acuan dalam perencanaan operasional, pengelolaan sumber daya manusia, serta kebijakan internal mengenai izin karyawan. Dengan adanya *dashboard* ini, perusahaan dapat melakukan evaluasi lebih lanjut terhadap distribusi izin karyawan, serta mengambil keputusan yang lebih tepat terkait pengelolaan waktu dan sumber daya di masa depan.

3.2.5 *Brainstorming* Teknik *Storytelling* dalam Visualisasi Data (1)

Brainstorming teknik *storytelling* dalam visualisasi data adalah langkah penting yang dilakukan oleh tim dengan tujuan mengeksplorasi berbagai cara dalam menyampaikan data secara menarik dan mudah dipahami oleh pengguna. Proses ini dipimpin oleh supervisor yang memberikan arahan kepada tim untuk berbagi ide, gagasan, dan teknik

yang dapat digunakan dalam merancang visualisasi data yang informatif sekaligus menarik secara emosional dan naratif. Dalam *brainstorming* ini, tim berfokus pada pemilihan data yang relevan dengan cerita yang ingin disampaikan, struktur naratif yang akan dibangun, serta jenis visualisasi yang paling efektif, seperti grafik batang, peta panas, atau diagram alur, yang dapat memperjelas informasi. Selain itu, juga dipertimbangkan apakah visualisasi akan bersifat statis atau interaktif, memberikan kesempatan bagi pengguna untuk menjelajahi data lebih dalam. Diskusi lebih lanjut dilakukan terkait pengguna yang akan melihat visualisasi data ini, agar teknik *storytelling* yang diterapkan sesuai dengan tingkat pemahaman mereka. Supervisor berperan dalam memberikan masukan teknis dan memastikan bahwa ide-ide yang dikembangkan dapat diterapkan dengan efektif. Dengan brainstorming ini, tim diharapkan dapat menghasilkan konsep visualisasi yang tidak hanya efektif dalam menyampaikan pesan data, tetapi juga menarik secara estetika dan mudah dipahami oleh audiens.

3.2.6 *Brainstorming* Analisis Data Selanjutnya

Tujuan dari *brainstorming* analisis data ini adalah untuk merencanakan langkah-langkah yang diperlukan dalam pengolahan dan analisis data secara mendalam. Proses ini bertujuan untuk memastikan bahwa setiap aspek dari data yang tersedia dapat dianalisis dengan cara yang tepat dan memberikan wawasan yang berguna. Wawasan ini nantinya akan mendukung pengambilan keputusan yang lebih informasional dan berbasis data, yang sangat penting dalam pembuatan kebijakan atau strategi yang lebih efektif dan efisien.

Lebih jauh lagi, brainstorming ini juga akan mencakup pertimbangan terhadap pengukuran yang akan digunakan dalam analisis, seperti metrik evaluasi atau indikator kinerja yang relevan dengan tujuan proyek. Adanya pemahaman yang jelas tentang apa yang ingin dicapai dalam analisis data akan memastikan bahwa hasil yang diperoleh tidak hanya

valid, tetapi juga dapat digunakan untuk membuat keputusan yang strategis. Sebagai contoh, analisis yang dilakukan dapat membantu manajemen dalam merencanakan kebijakan perizinan atau cuti yang lebih efisien, menilai produktivitas karyawan, atau bahkan memberikan panduan untuk perbaikan dalam proses internal perusahaan.

Dengan pemikiran yang matang dan rencana analisis yang terstruktur, diharapkan proses analisis data tidak hanya berjalan dengan efektif, tetapi juga dapat memberikan hasil yang akurat dan dapat dipertanggungjawabkan. Keakuratan hasil analisis akan memastikan bahwa keputusan yang diambil berbasis pada data yang relevan dan terpercaya, sehingga dapat meningkatkan kinerja perusahaan, memberikan keuntungan kompetitif, dan mendukung pengambilan kebijakan yang lebih baik di masa depan.

3.2.7 Melakukan *Data Cleannig* terhadap Dataset Kondisi Jembatan Nasional.

Berikut adalah tahapan yang dilakukan selama proses pembersihan data :

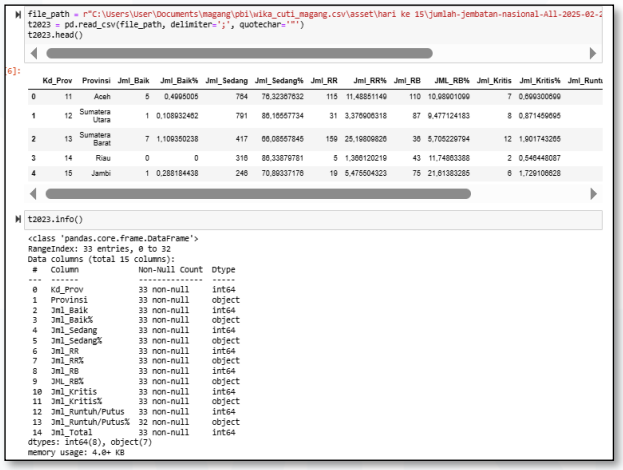
1. *Understanding the Structure of the Bridge Condition Dataset*

   Dataset yang digunakan berisi data kondisi jembatan nasional untuk setiap tahun, dengan data tahun 2023 sebagai contoh. Data ini disimpan dalam format CSV. Dataset ini memiliki 33 baris yang masing-masing mewakili satu provinsi di Indonesia dan terdiri dari 15 kolom variabel. Proses yang sama juga dilakukan untuk data tahun-tahun berikutnya.

   Setiap kolom memberikan informasi mengenai jumlah dan persentase kondisi jembatan berdasarkan klasifikasi tertentu, seperti jembatan dalam kondisi baik (Jml_Baik dan Jml_Baik%), sedang (Jml_Sedang dan Jml_Sedang%), rusak ringan (Jml_RR dan Jml_RR%), rusak berat (Jml_RB dan JML_RB%), kritis (Jml_Kritis

dan Jml_Kritis%), serta runtuh atau putus (Jml_Runtuh/Putus dan Jml_Runtuh%). Total keseluruhan jembatan di masing-masing provinsi tercantum pada kolom Jml_Total.

Hasil pemeriksaan struktur data menunjukkan bahwa delapan kolom bertipe numerik (*int64*) dan tujuh kolom lainnya bertipe objek (*object*). Kolom bertipe objek, khususnya yang memuat persentase, menggunakan tanda koma sebagai pemisah desimal. Oleh karena itu, diperlukan konversi ke tipe numerik (*float*) agar data dapat digunakan dalam proses analisis statistik dan visualisasi secara akurat. Struktur data ini memberikan dasar untuk melakukan eksplorasi dan analisis lebih lanjut terhadap kondisi infrastruktur jembatan nasional berdasarkan wilayah administratif provinsi.



```
file_path = r"C:\Users\User\Documents\magang\pbi\wika_cuti_magang.csv\asset\hari ke 15\jumlah-jembatan-nasional-All-2025-02-2
t2023 = pd.read_csv(file_path, delimiter=';', quotechar='"')
t2023.head()
```

	Kd_Prov	Provinsi	Jml_Baik	Jml_Baik%	Jml_Sedang	Jml_Sedang%	Jml_RR	Jml_RR%	Jml_RB	JML_RB%	Jml_Kritis	Jml_Kritis%	Jml_Runtu
0	11	Aceh	5	0,4995005	764	76,32367632	115	11,48851149	110	10,98901099	7	0,699300699	
1	12	Sumatera Utara	1	0,108932462	791	86,16557734	31	3,376906318	87	9,477124183	8	0,871459695	
2	13	Sumatera Barat	7	1,109350238	417	66,08557845	159	25,19809826	36	5,706229794	12	1,901743265	
3	14	Riau	0	0	316	86,33879781	5	1,366120219	43	11,74863388	2	0,546448087	
4	15	Jambi	1	0,288184438	246	70,89337176	19	5,475504323	75	21,61383285	6	1,729106628	

```
t2023.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 33 entries, 0 to 32
Data columns (total 15 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   Kd_Prov             33 non-null     int64
 1   Provinsi            33 non-null     object
 2   Jml_Baik            33 non-null     int64
 3   Jml_Baik%           33 non-null     object
 4   Jml_Sedang          33 non-null     int64
 5   Jml_Sedang%         33 non-null     object
 6   Jml_RR              33 non-null     int64
 7   Jml_RR%             33 non-null     object
 8   Jml_RB              33 non-null     int64
 9   JML_RB%             33 non-null     object
 10  Jml_Kritis          33 non-null     int64
 11  Jml_Kritis%         33 non-null     object
 12  Jml_Runtuh/Putus    33 non-null     int64
 13  Jml_Runtuh/Putus%   32 non-null     object
 14  Jml_Total           33 non-null     int64
dtypes: int64(8), object(7)
memory usage: 4.0+ KB
```

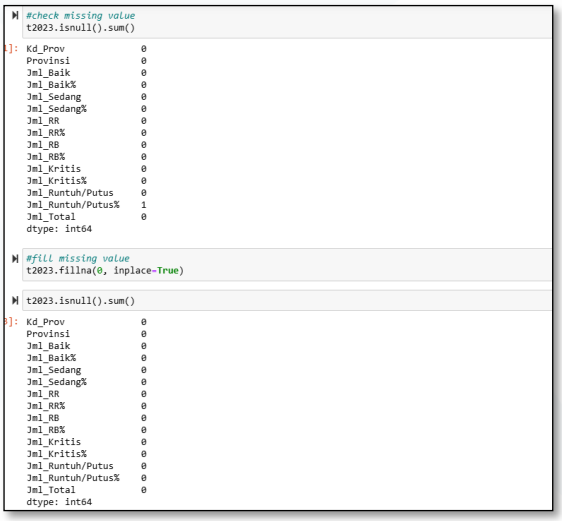
Gambar 3.18 Understanding the Structure of the Bridge Condition Dataset

2. *Identify Missing Data Fields of the Bridge Condition dataset*

Setelah memahami struktur dataset dan melakukan analisis statistik dasar, langkah berikutnya adalah memastikan dataset

t2023 .tidak mengandung nilai yang hilang, pertama-tama dilakukan pengecekan menggunakan t2023.*isnull().sum(),* yang menunjukkan

bahwa hanya kolom Jml_Runtuh/Putus% yang memiliki 1 nilai yang hilang, sementara kolom lainnya tidak mengandung nilai yang hilang. Selanjutnya, untuk mengatasi masalah nilai hilang tersebut, digunakan metode fillna(0, *inplace=True*) yang menggantikan nilai yang hilang dengan angka 0. Setelah itu, dilakukan pengecekan ulang dengan t2023.*isnull().sum()* yang menunjukkan bahwa sekarang seluruh kolom tidak mengandung nilai hilang lagi, karena semua kolom menunjukkan jumlah nilai hilang sebesar 0. Proses yang sama juga dilakukan untuk data tahun-tahun berikutnya. Dengan demikian, dataset t2023 dan dataset tahun-tahun lainnya telah bersih dari nilai yang hilang dan siap digunakan untuk analisis lebih lanjut.



Gambar 3.19 Identify Missing Data Fields of the Bridge Condition datasets

3. *Standardization column of the Bridge Condition dataset*

Pada langkah ini, dilakukan pembaruan pada kolom Kd_Prov untuk dua provinsi, yaitu Papua dan Papua Barat, dalam dataset

t2023. Untuk provinsi Papua, nilai Kd_Prov diubah menjadi 94, sedangkan untuk Papua Barat, nilai Kd_Prov diubah menjadi 91. Pembaruan ini dilakukan dengan menggunakan fungsi loc pada pustaka pandas, yang memungkinkan pemilihan baris berdasarkan kondisi tertentu (nama provinsi) dan penggantian nilai pada kolom yang sesuai. Setelah pembaruan dilakukan, dataset t2023 menampilkan nilai Kd_Prov yang baru untuk provinsi tersebut, yang dapat digunakan untuk analisis lebih lanjut. Proses yang sama juga dilakukan untuk data tahun-tahun berikutnya. Proses ini memastikan bahwa kode provinsi yang digunakan sudah sesuai dengan ketentuan yang berlaku, khususnya untuk provinsi Papua dan Papua Barat.

```
# Mengubah kode Papua dan Papua Barat
t2023.loc[t2023["Provinsi"] == "Papua", "Kd_Prov"] = 94
t2023.loc[t2023["Provinsi"] == "Papua Barat", "Kd_Prov"] = 91
# Menampilkan DataFrame yang telah diperbarui
t2023
```

8]:

	Kd_Prov	Provinsi	Jml_Baik	Jml_Baik%	Jml_Sedang	Jml_Sedang%	Jml_RR	Jml_RR%	Jml_RB	Jml_RB%	Jml_Kritis	Jml_Kritis%	Jml_Ru
0	11	Aceh	5	0,4995005	764	76,32367632	115	11,48851149	110	10,98901099	7	0,699300699	
1	12	Sumatera Utara	1	0,108932462	791	86,16557734	31	3,376906318	87	9,477124183	8	0,871459695	
2	13	Sumatera Barat	7	1,109350238	417	66,08557845	159	25,19809826	36	5,705229794	12	1,901743265	
3	14	Riau	0	0	316	86,33879781	5	1,366120219	43	11,74863388	2	0,546448087	
4	15	Jambi	1	0,288184438	246	70,89337176	19	5,475504323	75	21,61383285	6	1,729106628	
5	16	Sumatera Selatan	3	0,617283951	393	80,86419753	12	2,469135802	75	15,43209877	3	0,617283951	
6	17	Bengkulu	2	0,673400673	215	72,39057239	10	3,367003367	60	20,2020202	9	3,03030303	
7	18	Lampung	0	0	384	88,0733945	17	3,899082569	31	7,110091743	3	0,688073394	
8	19	Kepulauan Bangka Belitung	2	1,709401709	75	64,1025641	10	8,547008547	28	23,93162393	2	1,709401709	

Gambar 3.20 *Standardization column of the Bridge Condition datasets*

Selain itu, kolom nama provinsi dan kode provinsi juga telah diperiksa untuk memastikan bahwa tidak ada spasi tersembunyi yang dapat menyebabkan kesalahan dalam pemrosesan data lebih lanjut. Proses penggantian koma dengan titik pada kolom persentase juga penting untuk memastikan data dapat dihitung dengan tepat. Setelah kolom persentase dikonversi menjadi tipe data numerik (*float*), data siap untuk dianalisis dan divisualisasikan dengan metode statistik yang tepat. Setiap langkah yang dilakukan pada dataset t2023 juga diterapkan pada data tahun-tahun berikutnya untuk menjaga konsistensi dan kualitas data.

```
# Pastikan nama kolom bersih dari spasi tersembunyi
t2023.columns = t2023.columns.str.strip()

# Daftar kolom persentase yang ingin diubah
cols_to_convert = [
    "Jml_Baik%",
    "Jml_Sedang%",
    "Jml_RR%",
    "Jml_RB%",
    "Jml_Kritis%",
    "Jml_Runtuh/Putus%"
]

# Ganti koma dengan titik pada kolom bertipe object agar bisa dikonversi ke float
t2023[cols_to_convert] = t2023[cols_to_convert].replace(',', '.', regex=True)

# Konversi ke float
t2023[cols_to_convert] = t2023[cols_to_convert].astype(float)

# Cek hasilnya
t2023.head()
```

9]:

Jml_Baik%	Jml_Sedang	Jml_Sedang%	Jml_RR	Jml_RR%	Jml_RB	Jml_RB%	Jml_Kritis	Jml_Kritis%	Jml_Runtuh/Putus	Jml_Runtuh/Putus%	Jml_Total
0.499501	764	76.323676	115	11.488511	110	10.989011	7	0.699301	0	0.0	1001
0.108932	791	86.165577	31	3.376906	87	9.477124	8	0.871460	0	0.0	918
1.109350	417	66.085578	159	25.198098	36	5.705230	12	1.901743	0	0.0	631

Gambar 3.21 *Standardization column of the Bridge Condition dataset 2*

Selain itu, ditambahkan kolom baru bernama "Tahun" yang diisi dengan nilai 2023 secara seragam di seluruh baris. Penambahan kolom ini dilakukan dengan perintah t2023["Tahun"] = 2023, dan bertujuan untuk mempertahankan informasi asal tahun data, terutama saat seluruh dataset dari berbagai tahun akan digabung dalam satu dataframe gabungan (concatenated dataframe). Hasil akhir dari tahap ini adalah dataset t2023 yang bersih, lengkap, dan siap untuk proses integrasi data lintas tahun.

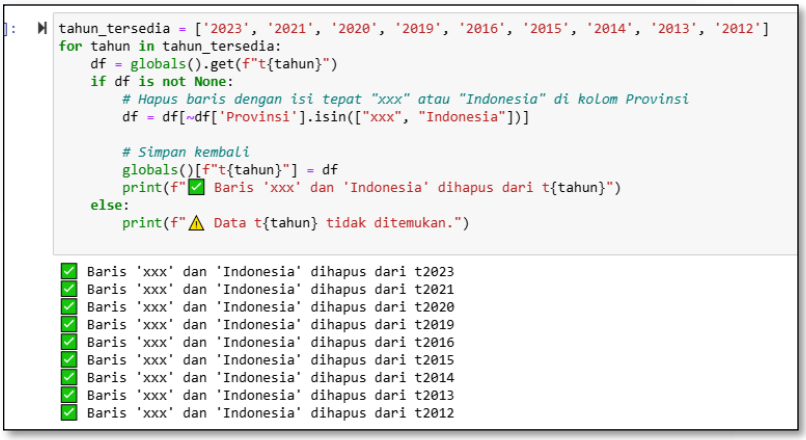
```
t2023["Tahun"] = 2023
```

Gambar 3.22 *Year Attribute Standardization*

4. *Filtering Unwanted Rows from the Bridge Condition Dataset*

Langkah selanjutnya adalah menggabungkan semua *DataFrame* yang mewakili data dari tahun Langkah selanjutnya adalah menggabungkan semua DataFrame yang mewakili data dari tahun. Pada tahap ini, dilakukan proses penyaringan data untuk menghapus baris-baris yang tidak relevan atau tidak mewakili data tingkat provinsi yang sah. Beberapa file CSV tahunan mengandung baris tambahan dengan nilai "xxx" atau "Indonesia" pada kolom Provinsi. Baris-baris tersebut tidak sesuai dengan unit analisis yang

diinginkan, yaitu data per provinsi. Oleh karena itu, dilakukan penghapusan baris dengan nilai "xxx" dan "Indonesia" dari setiap dataset tahunan menggunakan fungsi isin() dan operator negasi ~ dari pustaka pandas.Proses ini memastikan bahwa analisis tidak terdistorsi oleh data agregat nasional atau entri yang tidak sah. Penyaringan dilakukan secara otomatis melalui perulangan for pada semua tahun yang tersedia, sehingga konsistensi data antar tahun tetap terjaga. Dataset yang telah dibersihkan kemudian disimpan kembali ke dalam variabelnya masing-masing menggunakan globals().



Gambar 3.23 *Filtering Unwanted Rows from the Bridge Condition Dataset*

5. *Dataset Consistency Checking of the Bridge Condition Dataset*

Langkah berikutnya dalam proses pembersihan data adalah melakukan pengecekan konsistensi struktur dataset pada setiap tahun. Hal ini dilakukan dengan melihat jumlah baris dan kolom dari masing-masing dataset menggunakan fungsi .shape. Hasil pemeriksaan menunjukkan bahwa sebagian besar dataset, seperti tahun 2023, 2021, 2020, dan 2019 memiliki jumlah baris yang konsisten, yaitu 33 baris yang merepresentasikan seluruh provinsi di Indonesia setelah dilakukan penghapusan baris tidak relevan. Namun, terdapat ketidakkonsistenan pada beberapa tahun lainnya.

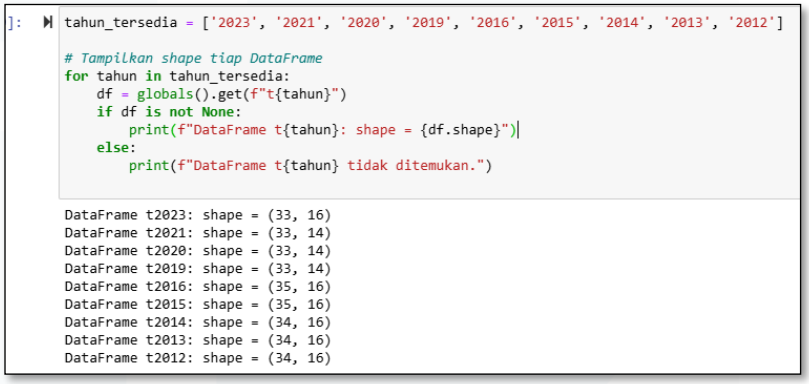
Dataset tahun 2016 dan 2015 memiliki 35 baris, sedangkan dataset tahun 2014, 2013, dan 2012 masing-masing memiliki 34 baris. Ketidaksesuaian ini mengindikasikan adanya baris tambahan yang kemungkinan merupakan entri tidak sah atau duplikat, sehingga perlu dilakukan pemeriksaan lebih lanjut untuk memastikan validitas data. Dari sisi kolom, sebagian besar dataset memiliki 16 kolom, kecuali data tahun 2021 hingga 2019 yang hanya memiliki 14 kolom, yang kemungkinan disebabkan oleh absennya data persentase pada tahun-tahun tersebut. Pemeriksaan ini penting untuk memastikan bahwa seluruh dataset memiliki struktur yang seragam sehingga dapat digunakan untuk analisis komparatif antar tahun secara valid dan konsisten.

Sebelum proses penggabungan seluruh dataset dilakukan, dilakukan pengecekan struktur data untuk memastikan konsistensi jumlah baris dan kolom pada setiap tahun. Pemeriksaan ini dilakukan dengan menampilkan shape dari masing-masing dataset, yang mencerminkan jumlah baris (representasi provinsi) dan jumlah kolom (variabel panjang jembatan).

Hasil pengecekan menunjukkan bahwa terdapat beberapa ketidakkonsistenan. Dari sisi jumlah baris, sebagian besar dataset memiliki 33 baris, namun terdapat dataset seperti tahun 2018 hingga 2016 yang memiliki 34 baris, kemungkinan karena sudah melalui proses penyamaan berdasarkan referensi provinsi. Sementara dari sisi kolom, sebagian besar dataset memiliki 10 kolom, namun dataset tahun 2020, 2019, dan 2012 hanya memiliki 9 kolom. Hal ini bisa disebabkan oleh hilangnya satu variabel pada tahun-tahun tersebut atau ketidakterbacaan kolom saat proses awal.

Pemeriksaan ini penting untuk mengidentifikasi perbedaan struktur sebelum semua dataset digabung menjadi satu kesatuan.

Ketidakkonsistenan pada jumlah baris atau kolom perlu ditangani terlebih dahulu agar hasil agregasi tidak menghasilkan data yang tidak seimbang atau hilang informasi. Selanjutnya, dataset perlu distandarisasi baik dari sisi kolom maupun jumlah baris agar proses integrasi dan analisis data lintas tahun dapat dilakukan secara akurat.



```
]:  tahun_tersedia = ['2023', '2021', '2020', '2019', '2016', '2015', '2014', '2013', '2012']

    # Tampilkan shape tiap DataFrame
    for tahun in tahun_tersedia:
        df = globals().get(f"t{tahun}")
        if df is not None:
            print(f"DataFrame t{tahun}: shape = {df.shape}")
        else:
            print(f"DataFrame t{tahun} tidak ditemukan.")

    DataFrame t2023: shape = (33, 16)
    DataFrame t2021: shape = (33, 14)
    DataFrame t2020: shape = (33, 14)
    DataFrame t2019: shape = (33, 14)
    DataFrame t2016: shape = (35, 16)
    DataFrame t2015: shape = (35, 16)
    DataFrame t2014: shape = (34, 16)
    DataFrame t2013: shape = (34, 16)
    DataFrame t2012: shape = (34, 16)
```

Gambar 3.24 *Dataset Consistency Checking of the Bridge Condition Dataset*

6. *Standardizing the Number of Rows Across All Bridge Condition Dataset*

Langkah selanjutnya adalah menggabungkan semua *DataFrame* yang mewakili Setelah dilakukan pengecekan terhadap jumlah baris yang tidak konsisten pada masing-masing dataset tahunan, langkah selanjutnya adalah menyamakan jumlah baris agar seluruh dataset memiliki struktur yang seragam. Proses ini dilakukan dengan menggunakan data tahun 2013 sebagai referensi acuan, karena data tersebut sudah bersih dan memiliki representasi lengkap untuk setiap provinsi. Kolom kd_prov dan provinsi dari dataset referensi ini digunakan sebagai basis penyamaan. Setiap dataset dari tahun 2012 hingga 2023 kemudian disamakan dengan melakukan *merge* berdasarkan kolom kd_prov dan provinsi menggunakan metode *left join*.

```
t2013.columns = t2013.columns.str.strip().str.lower()
provinsi_ref = t2013[['kd_prov', 'provinsi']].drop_duplicates()
provinsi_ref['kd_prov'] = provinsi_ref['kd_prov'].astype(str)

# Loop tiap tahun
for tahun in tahun_tersedia:
    df = globals().get(f"t{tahun}")
    if df is not None:
        df.columns = df.columns.str.strip().str.lower()
        df['kd_prov'] = df['kd_prov'].astype(str)

        df_fix = provinsi_ref.merge(df, on=['kd_prov', 'provinsi'], how='left')
        globals()[f"t{tahun}"] = df_fix
        print(f"✅ Data t{tahun} sudah disamakan (jumlah baris: {df_fix.shape[0]})")
    else:
        print(f"⚠️ Data t{tahun} tidak ditemukan.")
```

```
✅ Data t2023 sudah disamakan (jumlah baris: 34)
✅ Data t2021 sudah disamakan (jumlah baris: 34)
✅ Data t2020 sudah disamakan (jumlah baris: 34)
✅ Data t2019 sudah disamakan (jumlah baris: 34)
✅ Data t2016 sudah disamakan (jumlah baris: 34)
✅ Data t2015 sudah disamakan (jumlah baris: 34)
✅ Data t2014 sudah disamakan (jumlah baris: 34)
✅ Data t2013 sudah disamakan (jumlah baris: 34)
✅ Data t2012 sudah disamakan (jumlah baris: 34)
```

Gambar 3.25 *Standardizing the Number of Rows Across All Bridge Condition Dataset*

Dengan demikian, hanya data yang sesuai dengan referensi provinsi akan dipertahankan, dan data yang tidak sesuai akan diabaikan. Selain itu, kolom kd_prov juga dikonversi ke tipe string untuk memastikan kesesuaian format saat penggabungan data dilakukan. Hasil dari proses ini menunjukkan bahwa seluruh dataset kini memiliki jumlah baris yang sama, yaitu 34 baris, yang mencerminkan 34 provinsi sesuai dengan acuan dari dataset tahun 2013. Penyamaan ini penting untuk menjaga integritas data dan memudahkan proses analisis komparatif lintas tahun, tanpa terpengaruh oleh perbedaan jumlah entri yang sebelumnya ditemukan.

7. *Merging Data of the Bridge Condition Dataset*

Langkah selanjutnya adalah menggabungkan semua *DataFrame* yang mewakili data dari tahun 2012 hingga 2023 menjadi satu *DataFrame* utama. Hal ini dilakukan dengan menggunakan fungsi pd.*concat()* dari pustaka pandas, yang digunakan untuk

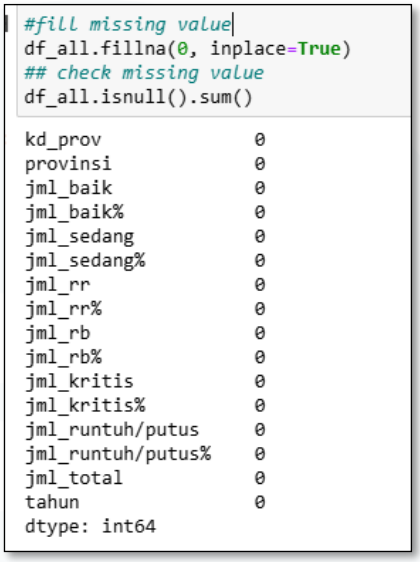
menggabungkan beberapa DataFrame menjadi satu. Fungsi ini menggabungkan data dari tahun 2012 (t2012), 2013 (t2013), 2014 (t2014), 2015 (t2015), 2016 (t2016), 2019 (t2019), 2020 (t2020), 2021 (t2021), 2022 (t2022), dan 2023 (t2023) ke dalam satu DataFrame yang disebut df_master. Parameter *ignore_index=True* digunakan untuk mengatur ulang indeks secara otomatis setelah penggabungan, memastikan bahwa indeks setiap baris di *DataFrame* yang baru unik dan terurut. Proses yang sama juga dilakukan untuk data tahun-tahun berikutnya.

```
# Menggabungkan semua DataFrame menjadi satu
df_all = pd.concat([t2012, t2013, t2014, t2015, t2016, t2019, t2020, t2021, t2023], ignore_index=True)
```

Gambar 3.26 *Merging Data of the Bridge Condition Dataset*

8. Rechecking Missing Value Data of the Bridge Condition Datasets

Setelah penggabungan semua data, dilakukan pengecekan terhadap data yang hilang (*missing data*) menggunakan fungsi *isnull().sum()*. Hasilnya menunjukkan bahwa tidak ada nilai yang hilang pada seluruh kolom dalam dataset df_master, termasuk kolom-kolom seperti Kd_Prov, Provinsi, Jml_Baik, Jml_Sedang%, Jml_RR%, Jml_RB%, Jml_Kritis%, Jml_Runtuh/Putus%, Jml_Total, Tahun, Jml_RS, dan Jml_RS%. Sebelum pengecekan ini, langkah pengisian nilai yang hilang (*missing value*) telah dilakukan dengan mengganti nilai yang hilang dengan angka nol menggunakan *fillna(0)*. Proses ini memastikan bahwa tidak ada nilai yang hilang pada dataset, sehingga siap untuk analisis lebih lanjut. Proses yang sama juga dilakukan untuk data tahun-tahun berikutnya.



```
#fill missing value
df_all.fillna(0, inplace=True)
## check missing value
df_all.isnull().sum()
```

```
kd_prov              0
provinsi             0
jml_baik             0
jml_baik%            0
jml_sedang           0
jml_sedang%          0
jml_rr               0
jml_rr%              0
jml_rb               0
jml_rb%              0
jml_kritis           0
jml_kritis%          0
jml_runtuh/putus     0
jml_runtuh/putus%    0
jml_total            0
tahun                0
dtype: int64
```

Gambar 3.27 *Rechecking Missing Value Data of the Bridge Condition Dataset*

9. *Exporting the Cleaned and Standardized Dataset to CSV Format*

Setelah seluruh proses pembersihan, penyamaan struktur kolom, serta standarisasi jumlah baris selesai dilakukan, langkah akhir dalam tahap ini adalah menyimpan dataset yang telah digabung dan dibersihkan ke dalam format CSV. File CSV digunakan karena bersifat ringan, mudah dibaca oleh berbagai perangkat lunak analisis data, serta kompatibel dengan Power BI yang digunakan dalam proyek visualisasi.

Dataset df_all, yang berisi gabungan seluruh data kondisi jembatan nasional dari berbagai tahun, diekspor ke direktori lokal dalam bentuk file bernama master_data2..csv. Proses penyimpanan dilakukan menggunakan fungsi .to_csv() dari pustaka pandas, dengan parameter index=False untuk memastikan bahwa kolom indeks tidak ikut tersimpan sebagai bagian dari file. Penyimpanan ini penting untuk menjamin ketersediaan data dalam format yang siap digunakan untuk

visualisasi lebih lanjut dalam Power BI dan untuk keperluan dokumentasi serta analisis lanjutan.

```
# Menyimpan hasil ke CSV
df_all.to_csv(r'C:\Users\User\Documents\magang\pbi\wika_cuti_magang.csv\asset\hari ke 15\jumlah-jembatan-nasional-All-2025-0
```

Gambar 3.28 *Exporting the Cleaned and Standardized Dataset to CSV Format*

3.2.8 Membuat *Dashboard* Menggunakan Power BI Untuk Menganalisis Data Kondisi Jembatan Nasional Di Indonesia.

Setelah proses pembersihan data selesai, visualisasi data dilakukan untuk membantu dalam menganalisis kondisi jembatan nasional di Indonesia. *Dashboard* yang dibuat menggunakan Power BI ini berfokus pada analisis kondisi jembatan berdasarkan provinsi serta tahun. Tampilan *dashboard* ini menggambarkan aspek kondisi jembatan jembatan per provinsi. *Dashboard* ini dibuat untuk membantu tim Business Development dalam mengidentifikasi wilayah yang memerlukan perhatian khusus, baik untuk pembangunan jembatan baru maupun pemeliharaan infrastruktur yang sudah ada. Dengan menyajikan informasi secara visual dan interaktif, *dashboard* ini digunakan oleh pengguna untuk menganalisis perbandingan antarprovinsi, mengevaluasi perubahan kondisi jembatan dari tahun ke tahun, serta mendukung penyusunan strategi pengembangan proyek yang lebih tepat sasaran.

Pada dataset ini yang menggambarkan kondisi jembatan, terdapat beberapa elemen utama dapat di lihat pada Gambar 3.29. Pengguna dapat memilih data berdasarkan provinsi dan tahun menggunakan filter yang tersedia. Grafik tren jumlah jembatan runtuh menunjukkan perubahan jumlah jembatan yang runtuh setiap tahun, dengan puncak keruntuhan pada tahun 2015. *Pie chart* menunjukkan lima provinsi dengan jumlah jembatan runtuh terbanyak, di mana Papua, Maluku, Kalimantan Tengah, Maluku Utara, dan Papua Barat menempati urutan

teratas. Selain itu, grafik batang horizontal memperlihatkan kondisi jembatan di setiap provinsi, dengan warna yang menunjukkan kondisi jembatan dari baik hingga rusak berat, sehingga memudahkan dalam mengetahui kondisi infrastruktur di tiap provinsi.

Gambar 3.29 *Dashboard of the Bridge Condition Datasets [8]*

Dashboard kondisi jembatan nasional ini diharapkan dapat membantu tim Business Development dalam mengidentifikasi wilayah prioritas pembangunan maupun pemeliharaan infrastruktur berdasarkan data kondisi jembatan nasional. Dengan visualisasi data yang sistematis dan interaktif, tim dapat memantau distribusi infrastruktur jembatan secara lebih menyeluruh, menganalisis tren historis berdasarkan tahun, serta merumuskan strategi pengembangan proyek secara lebih terarah.

3.2.9 *Brainstorming* Teknik *Storytelling* Dalam Visualisasi Data (2).

Dilakukan riset mandiri melalui berbagai sumber pembelajaran seperti tutorial di YouTube yang relevan dengan data cleaning. Riset ini dilakukan untuk memperdalam pemahaman tentang cara-cara memperbagus tampilan *dashboard* agar lebih terstruktur dan informatif, Proses ini tidak hanya membantu dalam meningkatkan keterampilan teknis yang sangat relevan dengan kebutuhan proyek yang sedang dikerjakan yaitu ditemukannya cara unpivot table menggunakan Power

BI, lalu diterapkan secara langsung pada dataset cuti karyawan yang memiliki format lebar (wide format), di mana masing-masing jenis cuti ditampilkan dalam kolom terpisah. Dengan menggunakan fitur Unpivot Columns pada Power Query Editor, data tersebut diubah ke dalam format panjang (long format) agar lebih mudah dianalisis dan divisualisasikan. Transformasi ini digunakan untuk memisahkan setiap jenis kondisi jembatan dikategorikan secara dinamis dan ditampilkan secara efisien dalam *dashboard* interaktif. Selain meningkatkan keterbacaan data, teknik ini juga memudahkan dalam pembuatan *filter* dan segmentasi kondisi per kategori serta. Proses riset mandiri ini tidak hanya memperluas pemahaman teknis, tetapi juga secara langsung mendukung tujuan proyek magang, yaitu membuat *dashboard* yang mudah dipahami oleh pengguna *non-teknik*.

3.2.10 Melakukan *Data Cleaning* Terhadap *Dataset* Panjang Jembatan Nasional .

Berikut adalah tahapan yang dilakukan selama proses pembersihan data :

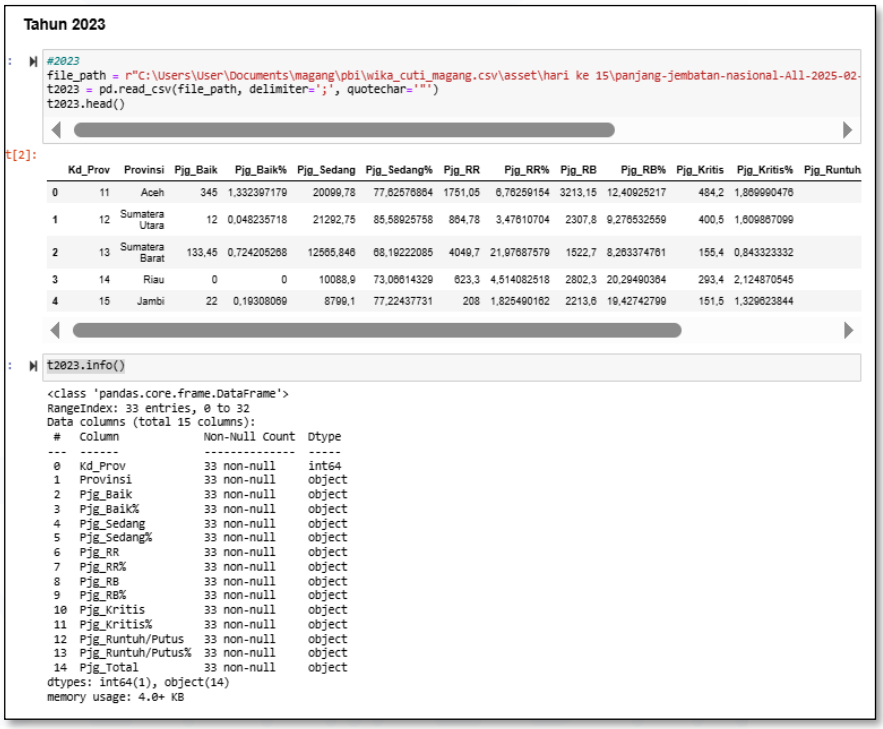
1. *Understanding the Structure of the Bridge Length Dataset*

   Tahapan awal yang dilakukan adalah membaca dan melakukan inspeksi awal terhadap dataset panjang jembatan nasional tahun 2023. File data dibaca dari direktori lokal menggunakan fungsi read_csv() dari pustaka pandas dengan parameter delimiter=';' karena format CSV yang digunakan memisahkan nilai dengan titik koma. Dataset dimuat ke dalam variabel t2023, lalu diperiksa menggunakan fungsi .head() untuk melihat lima baris pertama dan .info() untuk mengetahui struktur dan tipe data.

   Hasil pemeriksaan menunjukkan bahwa dataset terdiri dari 33 baris yang merepresentasikan masing-masing provinsi di Indonesia, serta 15 kolom yang mencakup informasi panjang jembatan

berdasarkan klasifikasi kondisi (baik, sedang, rusak ringan, rusak berat, kritis, runtuh/putus), baik dalam satuan panjang maupun persentasenya. Dari sisi tipe data, hanya kolom Kd_Prov yang memiliki tipe numerik (int64), sementara 14 kolom lainnya masih bertipe object. Hal ini terjadi karena adanya penggunaan tanda koma sebagai pemisah desimal dalam file, yang menyebabkan nilai-nilai numerik dibaca sebagai teks.

Dengan demikian, meskipun data tidak memiliki nilai yang hilang (null), diperlukan proses lanjutan berupa konversi format desimal dan tipe data agar dataset dapat digunakan untuk analisis statistik dan visualisasi secara optimal. Proses inspeksi dan pembacaan data ini tidak hanya dilakukan untuk tahun 2023, tetapi juga diterapkan pada seluruh dataset tahun-tahun sebelumnya agar standar dan kualitas data tetap konsisten dalam analisis lintas tahun.



Tahun 2023

```
#2023
file_path = r"C:\Users\User\Documents\magang\pbi\wika_cuti_magang.csv\asset\hari ke 15\panjang-jembatan-nasional-All-2025-02-
t2023 = pd.read_csv(file_path, delimiter=';', quotechar='"')
t2023.head()
```

t[2]:

	Kd_Prov	Provinsi	Pjg_Baik	Pjg_Baik%	Pjg_Sedang	Pjg_Sedang%	Pjg_RR	Pjg_RR%	Pjg_RB	Pjg_RB%	Pjg_Kritis	Pjg_Kritis%	Pjg_Runtuh
0	11	Aceh	345	1,332397179	20099,78	77,62576864	1751,05	6,76259154	3213,15	12,40925217	484,2	1,869990476	
1	12	Sumatera Utara	12	0,048235718	21292,75	85,58925758	864,78	3,47610704	2307,8	9,276532559	400,5	1,609867099	
2	13	Sumatera Barat	133,45	0,724205268	12565,846	68,19222085	4049,7	21,97687579	1522,7	8,263374761	155,4	0,843323332	
3	14	Riau	0	0	10088,9	73,06614329	623,3	4,514082518	2802,3	20,29490364	293,4	2,124870545	
4	15	Jambi	22	0,19308069	8799,1	77,22437731	208	1,825490162	2213,6	19,42742799	151,5	1,329623844	

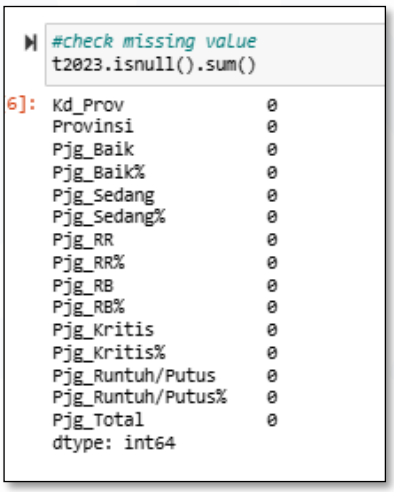
```
t2023.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 33 entries, 0 to 32
Data columns (total 15 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   Kd_Prov            33 non-null     int64
 1   Provinsi           33 non-null     object
 2   Pjg_Baik           33 non-null     object
 3   Pjg_Baik%          33 non-null     object
 4   Pjg_Sedang         33 non-null     object
 5   Pjg_Sedang%        33 non-null     object
 6   Pjg_RR             33 non-null     object
 7   Pjg_RR%            33 non-null     object
 8   Pjg_RB             33 non-null     object
 9   Pjg_RB%            33 non-null     object
 10  Pjg_Kritis         33 non-null     object
 11  Pjg_Kritis%        33 non-null     object
 12  Pjg_Runtuh/Putus   33 non-null     object
 13  Pjg_Runtuh/Putus%  33 non-null     object
 14  Pjg_Total          33 non-null     object
dtypes: int64(1), object(14)
memory usage: 4.0+ KB
```

Gambar 3.30 *Understanding the Structure of the Bridge Length Dataset*

2. *Identify Missing Data Fields of the Bridge Length dataset*

Setelah memahami struktur dataset dan melakukan analisis statistik dasar, langkah berikutnya adalah memastikan dataset t2023 tidak mengandung nilai yang hilang, pertama-tama dilakukan pengecekan menggunakan t2023.*isnull().sum(),* yang menunjukkan bahwa hanya kolom Jml_Runtuh/Putus% yang memiliki 1 nilai yang hilang, sementara kolom lainnya tidak mengandung nilai yang hilang. Selanjutnya, untuk mengatasi masalah nilai hilang tersebut, digunakan metode fillna(0, *inplace=True*) yang menggantikan nilai yang hilang dengan angka 0. Setelah itu, dilakukan pengecekan ulang dengan t2023.*isnull().sum()* yang menunjukkan bahwa sekarang seluruh kolom tidak mengandung nilai hilang lagi, karena semua kolom menunjukkan jumlah nilai hilang sebesar 0. Proses yang sama juga dilakukan untuk data tahun-tahun berikutnya. Dengan demikian, dataset t2023 dan dataset tahun-tahun lainnya telah bersih dari nilai yang hilang dan siap digunakan untuk analisis lebih lanjut.



```
#check missing value
t2023.isnull().sum()
```

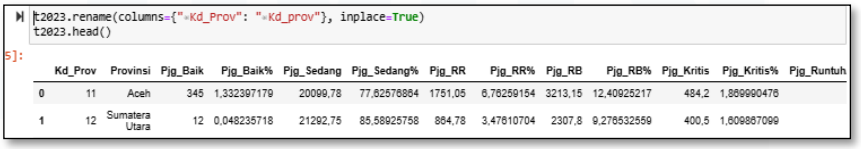
```
6]: Kd_Prov              0
    Provinsi             0
    Pjg_Baik             0
    Pjg_Baik%            0
    Pjg_Sedang           0
    Pjg_Sedang%          0
    Pjg_RR               0
    Pjg_RR%              0
    Pjg_RB               0
    Pjg_RB%              0
    Pjg_Kritis           0
    Pjg_Kritis%          0
    Pjg_Runtuh/Putus     0
    Pjg_Runtuh/Putus%    0
    Pjg_Total            0
    dtype: int64
```

Gambar 3.31 *Identify Missing Data Fields of the Bridge Length dataset*

3. *Standardization of Columns in the Bridge Length Dataset*

Langkah selanjutnya adalah melakukan standarisasi penamaan kolom pada dataset kondisi jembatan nasional tahun 2023. Dalam tahap ini ditemukan bahwa nama kolom Kd_Prov memiliki spasi tersembunyi di awal nama kolom, yang dapat menyebabkan kendala dalam pemanggilan kolom saat proses analisis data. Untuk mengatasi hal tersebut, dilakukan proses penggantian nama kolom dengan menggunakan fungsi rename() dari pustaka pandas. Kolom Kd_Prov diubah menjadi Kd_prov agar konsisten dengan format penamaan kolom lainnya dan mempermudah proses manipulasi data selanjutnya.

Proses ini penting untuk memastikan bahwa semua kolom memiliki format penamaan yang seragam tanpa karakter tersembunyi seperti spasi atau tab yang tidak terlihat. Setelah dilakukan standarisasi, ditampilkan lima baris pertama dari dataset menggunakan fungsi .head() untuk memastikan bahwa perubahan kolom telah berhasil diterapkan. Standarisasi nama kolom seperti ini juga diterapkan pada dataset tahun-tahun sebelumnya, sehingga semua data memiliki struktur kolom yang konsisten dan siap digunakan dalam proses pembersihan lanjutan maupun analisis komparatif antar tahun.



```
t2023.rename(columns={"Kd_Prov": "Kd_prov"}, inplace=True)
t2023.head()
```

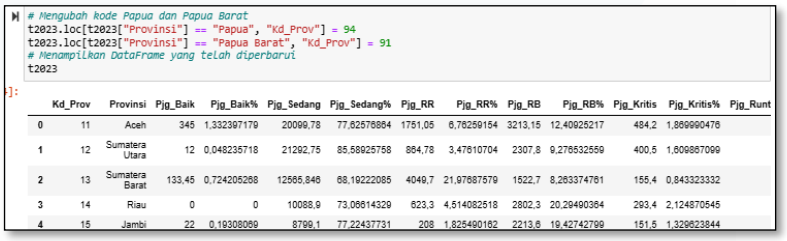
5]:

	Kd_Prov	Provinsi	Pjg_Baik	Pjg_Baik%	Pjg_Sedang	Pjg_Sedang%	Pjg_RR	Pjg_RR%	Pjg_RB	Pjg_RB%	Pjg_Kritis	Pjg_Kritis%	Pjg_Runtuh
0	11	Aceh	345	1,332397179	20099,78	77,62576884	1751,05	6,76259154	3213,15	12,40925217	484,2	1,869990476	
1	12	Sumatera Utara	12	0,048235718	21292,75	85,58925758	864,78	3,47610704	2307,8	9,276532559	400,5	1,609867099	

Gambar 3.32 *Standardization of Columns in the Bridge Length Dataset*

langkah berikutnya adalah memperbarui kode provinsi (Kd_Prov) untuk dua wilayah, yaitu Papua dan Papua Barat, agar sesuai dengan kode resmi yang digunakan secara nasional. Dalam dataset awal, terdapat ketidaksesuaian pada nilai kode provinsi untuk

kedua wilayah tersebut. Untuk memastikan keseragaman dan akurasi dalam analisis, nilai kode provinsi Papua diubah menjadi 94 dan Papua Barat menjadi 91 menggunakan fungsi loc dari pustaka pandas, yang memungkinkan pemilihan baris berdasarkan kondisi tertentu dan melakukan pembaruan nilai pada kolom tertentu. Perubahan ini bersifat penting karena kode provinsi sering digunakan sebagai kunci relasi antar tabel atau saat dilakukan penggabungan data (merge) antar dataset. Dengan memperbarui kode provinsi sesuai standar nasional, proses integrasi data lintas tahun maupun lintas sumber data dapat berjalan lebih lancar. Pembaruan ini juga dilakukan untuk seluruh dataset tahun-tahun lainnya agar seluruh data memiliki struktur yang seragam dan akurat.



```
# Mengubah kode Papua dan Papua Barat
t2023.loc[t2023["Provinsi"] == "Papua", "Kd_Prov"] = 94
t2023.loc[t2023["Provinsi"] == "Papua Barat", "Kd_Prov"] = 91
# Menampilkan DataFrame yang telah diperbarui
t2023
```

	Kd_Prov	Provinsi	Pjg_Baik	Pjg_Baik%	Pjg_Sedang	Pjg_Sedang%	Pjg_RR	Pjg_RR%	Pjg_RB	Pjg_RB%	Pjg_Kritis	Pjg_Kritis%	Pjg_Runt
0	11	Aceh	345	1,332397179	20099,78	77,62576884	1751,05	6,76259154	3213,15	12,40925217	484,2	1,869990476	
1	12	Sumatera Utara	12	0,048235718	21292,75	85,58925758	864,78	3,47610704	2307,8	9,276532559	400,5	1,609867099	
2	13	Sumatera Barat	133,45	0,724205268	12565,846	68,19222085	4049,7	21,97687579	1522,7	8,263374761	155,4	0,843323332	
3	14	Riau	0	0	10088,9	73,08614329	623,3	4,514082518	2802,3	20,29490364	293,4	2,124870545	
4	15	Jambi	22	0,19308069	8799,1	77,22437731	208	1,825490162	2213,6	19,42742799	151,5	1,329623844	

Gambar 3.33 *Updating Province Codes for Data Consistency*

Memperjelas identifikasi data berdasarkan tahun, ditambahkan kolom baru bernama Tahun pada dataset panjang jembatan nasional tahun 2023. Kolom ini diisi dengan nilai 2023 secara seragam untuk seluruh baris dalam dataset. Penambahan kolom tahun penting dilakukan agar data dari masing-masing tahun dapat digabungkan (concatenate) ke dalam satu tabel master secara akurat, serta mempermudah proses analisis tren data dari waktu ke waktu. Langkah ini juga diterapkan pada seluruh dataset tahun-tahun lainnya agar setiap entri dalam tabel gabungan memiliki informasi waktu yang jelas dan dapat dibandingkan secara konsisten dalam analisis longitudinal.

```
t2023["Tahun"] = 2023
```

Gambar 3.34 *Year Attribute Standardization 2*

4. *Filtering Unwanted Rows from the Length Condition Dataset*

Beberapa file dataset memuat baris tambahan dengan nilai xxx atau Indonesia pada kolom Provinsi. Baris-baris ini tidak mewakili data tingkat provinsi yang sah, sehingga perlu dihapus agar tidak memengaruhi hasil analisis. Penghapusan dilakukan dengan menggunakan metode filter pada pandas, yakni menghapus baris yang nilai kolom Provinsi-nya termasuk dalam daftar ["xxx", "Indonesia"]. Proses ini dilakukan pada seluruh dataset dari tahun 2012 hingga 2023 untuk menjaga konsistensi antar tahun. Hasilnya, semua dataset telah berhasil dibersihkan dari entri yang tidak relevan dan kini hanya memuat data untuk masing-masing provinsi secara sah, yang siap digunakan untuk proses agregasi dan visualisasi lebih lanjut.

```
tahun_tersedia = ['2023', '2021', '2020', '2019', '2016', '2015', '2014', '2013', '2012']
for tahun in tahun_tersedia:
    df = globals().get(f"t{tahun}")
    if df is not None:
        # Hapus baris dengan isi tepat "xxx" atau "Indonesia" di kolom Provinsi
        df = df[~df['Provinsi'].isin(["xxx", "Indonesia"])]

        # Simpan kembali
        globals()[f"t{tahun}"] = df
        print(f"✅ Baris 'xxx' dan 'Indonesia' dihapus dari t{tahun}")
    else:
        print(f"⚠️ Data t{tahun} tidak ditemukan.")
```

```
✅ Baris 'xxx' dan 'Indonesia' dihapus dari t2023
✅ Baris 'xxx' dan 'Indonesia' dihapus dari t2021
✅ Baris 'xxx' dan 'Indonesia' dihapus dari t2020
✅ Baris 'xxx' dan 'Indonesia' dihapus dari t2019
✅ Baris 'xxx' dan 'Indonesia' dihapus dari t2016
✅ Baris 'xxx' dan 'Indonesia' dihapus dari t2015
✅ Baris 'xxx' dan 'Indonesia' dihapus dari t2014
✅ Baris 'xxx' dan 'Indonesia' dihapus dari t2013
✅ Baris 'xxx' dan 'Indonesia' dihapus dari t2012
```

Gambar 3.35 *Filtering Unwanted Rows from the Length Condition Dataset*

langkah lanjutan berupa penghapusan kolom yang tidak relevan dari masing-masing dataset tahunan. menyederhanakan struktur data dan fokus pada nilai absolut panjang jembatan, beberapa kolom

persentase dihapus dari dataset. Kolom yang dihapus meliputi Pjg_Baik%, Pjg_Sedang%, Pjg_RR%, Pjg_RB%, Pjg_Kritis%, dan Pjg_Runtuh/Putus%, karena kolom-kolom tersebut tidak diperlukan dalam analisis yang hanya berfokus pada satuan panjang fisik jembatan dalam meter.

Proses penghapusan dilakukan menggunakan fungsi drop() dari pustaka pandas, dengan parameter columns yang berisi daftar nama kolom yang akan dihapus, serta inplace=True untuk menerapkan perubahan langsung pada dataset t2023. Setelah kolom-kolom tersebut dihapus, struktur dataset menjadi lebih ringkas, hanya mencakup data provinsi, panjang jembatan berdasarkan kondisi fisik, total panjang, dan tahun. Langkah ini juga diterapkan pada dataset tahun-tahun lainnya untuk memastikan struktur data tetap konsisten dan memudahkan proses penggabungan serta analisis lebih lanjut di tahap berikutnya.

```
t2023.drop(columns=['Pjg_Baik%', 'Pjg_Sedang%', 'Pjg_RR%', 'Pjg_RB%', 'Pjg_Kritis%', 'Pjg_Runtuh/Putus%'], inplace=True)
t2023.head()
```

	Kd_Prov	Provinsi	Pjg_Baik	Pjg_Sedang	Pjg_RR	Pjg_RB	Pjg_Kritis	Pjg_Runtuh/Putus	Pjg_Total	Tahun
0	11	Aceh	345	20099,78	1751,05	3213,15	484,2	0	25893,18	2023
1	12	Sumatera Utara	12	21292,75	864,78	2307,8	400,5	0	24877,83	2023
2	13	Sumatera Barat	133,45	12565,846	4049,7	1522,7	155,4	0	18427,096	2023
3	14	Riau	0	10088,9	623,3	2802,3	293,4	0	13807,9	2023
4	15	Jambi	22	8799,1	208	2213,6	151,5	0	11394,2	2023

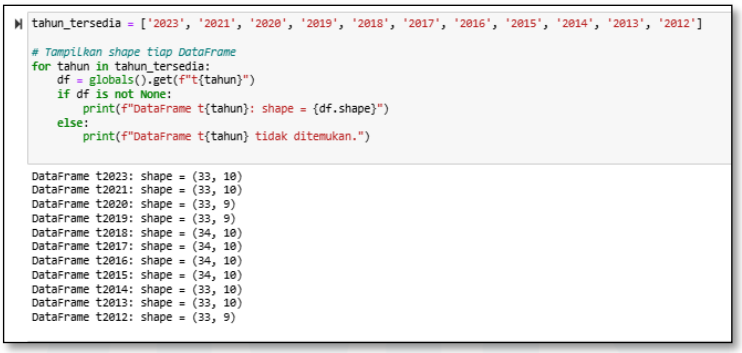
Gambar 3.36 *Removing Percentage Columns from the Dataset*

5. *Dataset Consistency Checking of the Bridge Length Dataset*

Sebelum proses penggabungan seluruh dataset dilakukan, dilakukan pengecekan struktur data untuk memastikan konsistensi jumlah baris dan kolom pada setiap tahun. Pemeriksaan ini dilakukan dengan menampilkan shape dari masing-masing dataset, yang mencerminkan jumlah baris (representasi provinsi) dan jumlah kolom (variabel panjang jembatan).

Hasil pengecekan menunjukkan bahwa terdapat beberapa ketidakkonsistenan. Dari sisi jumlah baris, sebagian besar dataset memiliki 33 baris, namun terdapat dataset seperti tahun 2018 hingga 2016 yang memiliki 34 baris, kemungkinan karena sudah melalui proses penyamaan berdasarkan referensi provinsi. Sementara dari sisi kolom, sebagian besar dataset memiliki 10 kolom, namun dataset tahun 2020, 2019, dan 2012 hanya memiliki 9 kolom. Hal ini bisa disebabkan oleh hilangnya satu variabel pada tahun-tahun tersebut atau ketidakterbacaan kolom saat proses awal.

Pemeriksaan ini penting untuk mengidentifikasi perbedaan struktur sebelum semua dataset digabung menjadi satu kesatuan. Ketidakkonsistenan pada jumlah baris atau kolom perlu ditangani terlebih dahulu agar hasil agregasi tidak menghasilkan data yang tidak seimbang atau hilang informasi. Selanjutnya, dataset perlu distandarisasi baik dari sisi kolom maupun jumlah baris agar proses integrasi dan analisis data lintas tahun dapat dilakukan secara akurat.



```
tahun_tersedia = ['2023', '2021', '2020', '2019', '2018', '2017', '2016', '2015', '2014', '2013', '2012']

# Tampilkan shape tiap DataFrame
for tahun in tahun_tersedia:
    df = globals().get(f"t{tahun}")
    if df is not None:
        print(f"DataFrame t{tahun}: shape = {df.shape}")
    else:
        print(f"DataFrame t{tahun} tidak ditemukan.")
```

```
DataFrame t2023: shape = (33, 10)
DataFrame t2021: shape = (33, 10)
DataFrame t2020: shape = (33, 9)
DataFrame t2019: shape = (33, 9)
DataFrame t2018: shape = (34, 10)
DataFrame t2017: shape = (34, 10)
DataFrame t2016: shape = (34, 10)
DataFrame t2015: shape = (34, 10)
DataFrame t2014: shape = (33, 10)
DataFrame t2013: shape = (33, 10)
DataFrame t2012: shape = (33, 9)
```

Gambar 3.37 *Dataset Consistency Checking of the Bridge Length Dataset*

6. *Standardizing the Number of Rows Across All Bridge Length Dataset*

Selanjutnya dilakukan proses penyamaan jumlah baris untuk menjaga konsistensi struktur data. Dataset tahun 2018 digunakan sebagai referensi karena memiliki format yang bersih, lengkap, dan

representatif terhadap seluruh provinsi di Indonesia. Kolom kd_prov dan provinsi dari dataset referensi diambil untuk membentuk struktur acuan bernama provinsi_ref.

Setiap dataset tahunan kemudian diolah dengan langkah-langkah sebagai berikut: pertama, seluruh nama kolom diseragamkan menggunakan metode str.strip().str.lower() untuk menghindari perbedaan penamaan akibat spasi atau huruf kapital. Selanjutnya, tipe data pada kolom kd_prov dikonversi menjadi string agar proses pencocokan antar tabel berjalan lancar. Lalu, masing-masing dataset digabungkan (merge) dengan provinsi_ref menggunakan metode left join berdasarkan kolom kd_prov dan provinsi.

Hasil dari proses ini menunjukkan bahwa seluruh dataset tahun 2012 hingga 2023 telah disamakan menjadi 34 baris, yang mencerminkan jumlah provinsi sesuai referensi. Proses penyamaan ini sangat penting untuk memastikan integritas data, memudahkan agregasi, serta menjamin keakuratan dalam analisis lintas tahun yang akan dilakukan pada tahap selanjutnya.

```
# Pastikan t2018 sudah ada dan digunakan sebagai referensi
t2018.columns = t2018.columns.str.strip().str.lower()
provinsi_ref = t2018[['kd_prov', 'provinsi']].drop_duplicates()
provinsi_ref['kd_prov'] = provinsi_ref['kd_prov'].astype(str)

# Loop tiap tahun
for tahun in tahun_tersedia:
    df = globals().get(f"t{tahun}")
    if df is not None:
        df.columns = df.columns.str.strip().str.lower()
        df['kd_prov'] = df['kd_prov'].astype(str)

        df_fix = provinsi_ref.merge(df, on=['kd_prov', 'provinsi'], how='left')
        globals()[f"t{tahun}"] = df_fix
        print(f"✅ Data t{tahun} sudah disamakan (jumlah baris: {df_fix.shape[0]})")
    else:
        print(f"⚠️ Data t{tahun} tidak ditemukan.")
```

```
✅ Data t2023 sudah disamakan (jumlah baris: 34)
✅ Data t2021 sudah disamakan (jumlah baris: 34)
✅ Data t2020 sudah disamakan (jumlah baris: 34)
✅ Data t2019 sudah disamakan (jumlah baris: 34)
✅ Data t2018 sudah disamakan (jumlah baris: 34)
✅ Data t2017 sudah disamakan (jumlah baris: 34)
✅ Data t2016 sudah disamakan (jumlah baris: 34)
✅ Data t2015 sudah disamakan (jumlah baris: 34)
✅ Data t2014 sudah disamakan (jumlah baris: 34)
✅ Data t2013 sudah disamakan (jumlah baris: 34)
✅ Data t2012 sudah disamakan (jumlah baris: 34)
```

Gambar 3.38 *Standardizing the Number of Rows Across All Bridge Length Dataset*

7. *Merging Data of the Bridge Condition Dataset*

Setelah seluruh dataset dari tahun 2012 hingga 2023 diperiksa dan disesuaikan, langkah berikutnya adalah menggabungkan seluruh data tersebut menjadi satu tabel utama. Proses penggabungan dilakukan menggunakan fungsi pd.concat() dari pustaka pandas, dengan menyusun setiap DataFrame tahunan ke dalam satu kesatuan dan menyetel parameter ignore_index=True agar indeks baris diperbarui secara otomatis.

Penggabungan ini menghasilkan sebuah DataFrame baru bernama df_all yang memuat informasi panjang jembatan nasional berdasarkan kondisi fisik di seluruh provinsi dan tahun. Dengan struktur yang sudah distandarisasi, data ini siap untuk digunakan dalam analisis komparatif lintas waktu, visualisasi tren infrastruktur, serta pembuatan dashboard di Power BI. Langkah ini merupakan tahap akhir dari proses pembersihan dan persiapan data, yang memastikan bahwa seluruh dataset memiliki format yang seragam dan dapat dianalisis sebagai satu kesatuan yang utuh.

```
# Menggabungkan semua DataFrame menjadi satu
df_all = pd.concat([t2012, t2013, t2014, t2015, t2016, t2017,t2018,t2019, t2020, t2021, t2023], ignore_index=True)
```

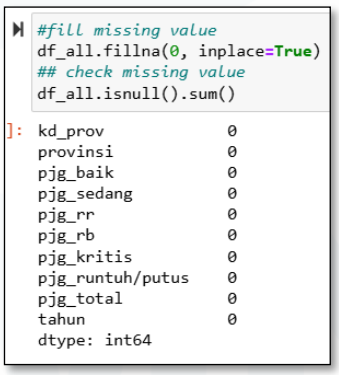
Gambar 3.39 *Merging Data of the Bridge Condition Dataset*

8. *Rechecking Missing Value Data of the Bridge Condition Datasets*

Setelah seluruh dataset tahunan berhasil digabung menjadi satu dalam variabel df_all, dilakukan langkah akhir dalam proses pembersihan data, yaitu memastikan tidak ada nilai yang hilang (missing value) pada kolom manapun. Proses ini dilakukan dengan menggunakan fungsi fillna(0) dari pustaka pandas, yang berfungsi untuk menggantikan seluruh nilai kosong dengan angka nol. Angka

nol ini merepresentasikan bahwa tidak terdapat panjang jembatan dalam kategori tertentu di suatu provinsi pada tahun tertentu, dan sekaligus menjaga agar proses analisis tidak terganggu oleh nilai NaN.

Selanjutnya, dilakukan verifikasi dengan fungsi isnull().sum() untuk memastikan bahwa semua kolom telah bersih dari nilai hilang. Dari hasil tersebut, dapat disimpulkan bahwa tidak ada lagi nilai kosong di seluruh kolom, baik pada informasi identitas wilayah (kd_prov, provinsi), data panjang jembatan berdasarkan kondisi (pjg_baik hingga pjg_runtuh/putus), total panjang jembatan (pjg_total), maupun tahun pengamatan (tahun). Dengan demikian, dataset df_all telah sepenuhnya bersih, lengkap, dan siap digunakan sebagai master dataset dalam proses analisis lanjutan serta pengembangan dashboard visualisasi kondisi infrastruktur jembatan nasional di Power BI.



Gambar 3.40 *Rechecking Missing Value Data of the Bridge Length Dataset*

9. *Exporting the Cleaned and Standardized Dataset to CSV Format*

Setelah seluruh proses pembersihan, penyamaan struktur kolom, serta standarisasi jumlah baris selesai dilakukan, langkah akhir dalam tahap ini adalah menyimpan dataset yang telah digabung dan dibersihkan ke dalam format CSV. File CSV digunakan karena

bersifat ringan, mudah dibaca oleh berbagai perangkat lunak analisis data, serta kompatibel dengan Power BI yang digunakan dalam proyek visualisasi.

Dataset df_all, yang berisi gabungan seluruh data kondisi jembatan nasional dari berbagai tahun, diekspor ke direktori lokal dalam bentuk file bernama master_data2..csv. Proses penyimpanan dilakukan menggunakan fungsi .to_csv() dari pustaka pandas, dengan parameter index=False untuk memastikan bahwa kolom indeks tidak ikut tersimpan sebagai bagian dari file. Penyimpanan ini penting untuk menjamin ketersediaan data dalam format yang siap digunakan untuk visualisasi lebih lanjut dalam Power BI dan untuk keperluan dokumentasi serta analisis lanjutan.

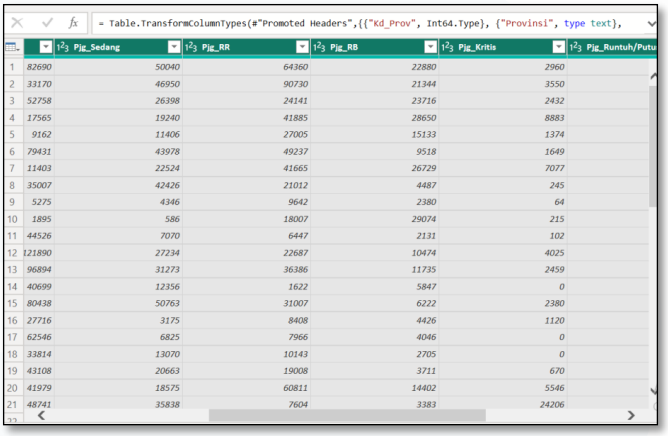
Gambar 3.41 *Exporting Dataset to CSV Format*

3.2.11 Membuat *Dashboard* Menggunakan Power BI Untuk Menganalisis Data Panjang Jembatan Nasional Di Indonesia.

Setelah proses pembersihan data selesai, visualisasi data dilakukan untuk membantu dalam menganalisis panjang jembatan nasional di Indonesia. *Dashboard* yang dibuat menggunakan Power BI ini berfokus pada analisis panjang jembatan berdasarkan provinsi serta tahun. Tampilan utama *dashboard* ini menggambarkan aspek panjang jembatan per provinsi [1].

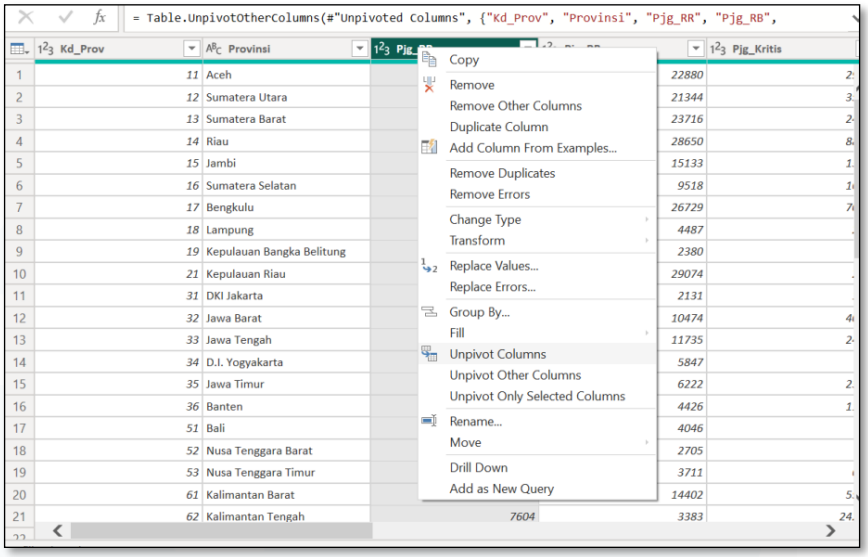
Sebagai Tahap tambahan yaitu Unpivoting digunakan untuk mengubah format data yang awalnya dalam bentuk lebar (wide format) menjadi lebih panjang (long format), yang digunakan untuk mempermudah analisis lebih lanjut [9]. Pada gambar 3.25, terlihat data awal yang memiliki beberapa kolom, seperti "Provinsi", "Pjg_Sedang", "Pjg_RR", "Pjg_RB", dan "Pjg_Kritis". Kolom-kolom ini berisi nilai-

nilai berdasarkan provinsi yang ingin dianalisis. Unpivot ini akan digunakan untuk membuat slicer agar dapat di pilih berdasarkan slicer.



Gambar 3.42 *unpivot Column Initial Stage*

Langkah selanjutnya adalah menggunakan fungsi Unpivot di Power Query (ditampilkan pada gambar kedua). Fungsi ini akan mengubah data dari format lebar menjadi panjang, di mana nilai-nilai pada kolom "Pjg_Sedang", "Pjg_RR", "Pjg_RB", dan "Pjg_Kritis" akan dipindahkan ke dua kolom baru, satu untuk kategori nilai (misalnya, Pjg_Sedang, Pjg_RR, dll.) dan satu untuk nilai itu sendiri. Proses ini membuat kolom "Attribute" untuk mewakili tipe jembatan (baik, rusak ringan, rusak berat, dll.), sedangkan kolom “Value” akan berisi data angka yang terkait dengan masing-masing provinsi. kolom “Attribute” kemudian di rename menjadi kolom “Kondisi”, Selanjutnya untuk kolom value in di rename menjadi “Panjang_Jembatan”.



Gambar 3.43 *Unvipot Column*

Selanjutnya juga dibuatkan measure untuk menghitung total panjang jembatan. Dapat dilihat pada gambar 3.27 rumus ini berfungsi untuk menjumlahkan semua nilai dalam kolom Panjang_Jembatan pada tabel masterdatapanjangjembatan, dan kemudian mengonversinya ke dalam format angka dengan pemisah ribuan. Setelah perhitungan total panjang jembatan selesai, hasilnya ditampilkan dengan menambahkan satuan "m" (meter) di akhir angka. Sebagai contoh, jika hasil perhitungan total panjang jembatan adalah 1,000, maka output yang ditampilkan pada *dashboard* adalah "1,000 m". Dengan demikian, informasi tentang total panjang jembatan dapat disajikan dalam format yang lebih mudah dipahami dan lebih terstruktur.

Gambar 3.44 *Formula for adding m2*

Dengan data yang sudah di-unpivot, langkah selanjutnya adalah membuat visualisasi yang sesuai di Power BI, seperti grafik tren, pie chart, atau tabel distribusi berdasarkan kondisi dan provinsi. Proses unpivot ini sangat penting agar data dapat diproses dengan mudah dan

efisien dalam Power BI, yang akan mendukung pembuatan *dashboard* yang informatif dan mudah dipahami.

Pada *dashboard* ini fokus utama adalah panjang jembatan berdasarkan Panjang provinsi dapat dilihat pada gambar 3.28 *Filter* untuk provinsi, kondisi, dan tahun dapat digunakan pengguna untuk memilih data yang lebih spesifik lagi. Total panjang jembatan ditampilkan sebagai angka utama, yaitu 1.126 m² . selanjutnya Grafik tren panjang jembatan menunjukkan perubahan panjang jembatan berdasarkan kondisi setiap tahun. *Pie chart* menampilkan lima provinsi dengan total panjang jembatan dalam kondisi tertentu, dengan Maluku, Papua, dan Jawa Barat sebagai provinsi dengan panjang jembatan terbanyak. Grafik batang horizontal menunjukkan panjang jembatan berdasarkan kondisi di setiap provinsi, memberikan gambaran tentang distribusi panjang jembatan yang masih baik dan yang rusak di seluruh Indonesia.

Gambar 3.45 *Dashboard of Bridge Length [10]*

3.2.12 *Brainstorming* Teknik *Storytelling* Dalam Visualisasi Data (3)

Brainstorming ini dilakukan sebagai upaya untuk menentukan arah pengembangan project *dashboard* berikutnya selama masa magang. Tujuan dari kegiatan ini adalah untuk menggali berbagai ide visualisasi

data yang tidak hanya relevan dengan kebutuhan perusahaan, tetapi juga mampu menyampaikan insight secara efektif melalui pendekatan storytelling. Dalam konteks ini, storytelling berperan penting untuk mengubah data mentah menjadi informasi yang bermakna dan mudah dipahami oleh pemangku kepentingan, terutama bagi manajemen non-teknis.

Proses *brainstorming* mempertimbangkan sejumlah faktor seperti ketersediaan data internal, urgensi informasi yang dibutuhkan oleh pengguna akhir, serta potensi *dashboard* untuk membantu monitoring dan pengambilan keputusan secara cepat. Beberapa ide yang sempat muncul dalam diskusi antara lain: pemantauan distribusi alat berat, evaluasi progres pekerjaan berdasarkan divisi, dan visualisasi performa mingguan tenaga kerja proyek.

Setelah mempertimbangkan seluruh opsi, diputuskan untuk mengembangkan *dashboard* status proyek sebagai fokus utama dari project selanjutnya. *Dashboard* ini dirancang untuk menampilkan informasi terkait progres proyek, status penyelesaian pekerjaan, dan indikator waktu pelaksanaan secara visual dan terstruktur. Tujuan utamanya adalah untuk mendukung manajer dalam memantau status proyek secara menyeluruh, mengidentifikasi keterlambatan, serta mengambil keputusan berbasis data secara lebih cepat dan tepat. *Dashboard* ini diharapkan dapat menjadi alat bantu yang efektif dalam pengawasan proyek-proyek yang tersebar di berbagai lokasi dan memiliki tingkat kompleksitas tinggi.

3.2.13 Melakukan *Data Cleaning* Terhadap *Dataset* Status Proyek

Berikut adalah tahapan yang dilakukan selama proses pembersihan data :

1. *Understanding the Structure of Project Status Dataset*

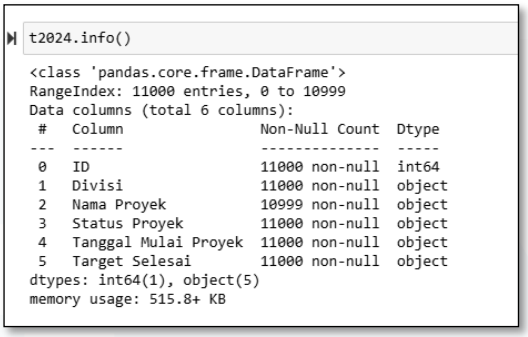
Dataset yang digunakan dalam analisis ini berasal dari file *Status_Proyek_2024.xlsx* yang memuat informasi terkait proyek-proyek yang dijalankan oleh berbagai divisi dalam perusahaan selama tahun 2024. Data ini terdiri dari beberapa kolom utama, yaitu ID, Divisi, Nama Proyek, Status Proyek, Tanggal Mulai Proyek, dan Target Selesai. Kolom ID berfungsi sebagai identifikasi unik untuk setiap proyek. Kolom Divisi menunjukkan unit kerja yang bertanggung jawab atas pelaksanaan proyek, seperti Divisi *Casting dan Steel Fabrication*. Nama Proyek berisi penamaan unik untuk masing-masing proyek, contohnya "Proyek-0184" dan "Proyek-0397". Kolom Status Proyek menggambarkan perkembangan terkini proyek yang dikategorikan menjadi tiga status, yaitu Tertunda, On Progress, dan Selesai. Sementara itu, Tanggal Mulai Proyek dan Target Selesai mencerminkan rentang waktu pelaksanaan proyek. Dataset ini dibaca menggunakan pustaka pandas dalam Python dan ditampilkan lima data awal untuk memperoleh gambaran umum mengenai struktur data. Informasi ini menjadi dasar dalam proses analisis selanjutnya, seperti perhitungan durasi proyek, distribusi status proyek, dan evaluasi kinerja proyek berdasarkan divisi terkait.

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import os

import warnings
warnings.filterwarnings('ignore')

file_path = "C:\\Users\\User\\Documents\\Status_Proyek_2024.xlsx"
t2024 = pd.read_excel(file_path)
t2024.head()
```

	ID	Divisi	Nama Proyek	Status Proyek	Tanggal Mulai Proyek	Target Selesai
0	1	Casting	Proyek-0184	Tertunda	2024-02-21	2024-07-08
1	2	Casting	Proyek-0411	Tertunda	2024-12-21	2025-06-04
2	3	Casting	Proyek-0397	Tertunda	2024-12-20	2025-05-31
3	4	Steel Fabrication	Proyek-0152	On Progress	2024-05-25	2024-08-21
4	5	Steel Fabrication	Proyek-0391	Selesai	2024-01-23	2024-03-15

Gambar 3.46 *Understanding the Structure of Project Status Dataset*

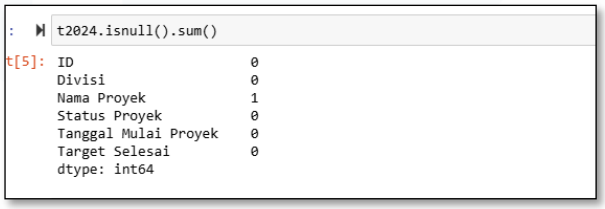
Dataset ini terdiri dari 11.000 baris data dan 6 kolom. Enam kolom tersebut mencakup ID, Divisi, Nama Proyek, Status Proyek, Tanggal Mulai Proyek, dan Target Selesai. Kolom ID bertipe data numerik (int64), sedangkan lima kolom lainnya bertipe objek (object) yang umumnya merepresentasikan data dalam format *string* atau tanggal yang belum dikonversi. Dari segi kelengkapan data, seluruh kolom memiliki jumlah nilai non-null yang sama, yaitu 11.000 entri, kecuali kolom Nama Proyek yang memiliki satu nilai kosong (null), sehingga hanya terdapat 10.999 entri yang lengkap. Informasi ini penting untuk mengetahui apakah ada data yang perlu dibersihkan atau ditangani sebelum dilakukan analisis lebih lanjut, seperti konversi tipe data tanggal serta imputasi terhadap nilai yang hilang.



Gambar 3.47 *Identify Data Info of Project Status Dataset*

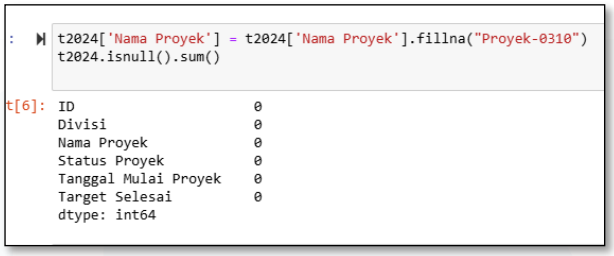
2. *Identify Missing Data Fields of Project Status Dataset*

Setelah dilakukan pemeriksaan terhadap data yang hilang menggunakan fungsi *isnull().sum()*, ditemukan bahwa terdapat satu nilai kosong (*missing value*) pada kolom Nama Proyek, sedangkan lima kolom lainnya tidak memiliki nilai kosong.



Gambar 3.48 *Identify Missing Data Fields of Project Status Dataset*

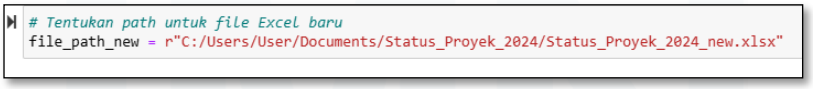
Untuk menangani hal tersebut, dilakukan diskusi bersama tim serta pencarian data tambahan guna memastikan bahwa nilai yang akan diisi sesuai dengan informasi yang semestinya. Berdasarkan hasil pencarian dan kesepakatan yang dicapai, nilai kosong tersebut diisi dengan nama proyek "Proyek-0310" menggunakan fungsi fillna(). Setelah proses pengisian dilakukan, pengecekan ulang memastikan bahwa seluruh kolom kini telah terisi penuh tanpa ada nilai kosong, sehingga data telah siap untuk dianalisis lebih lanjut dengan kondisi yang bersih dan lengkap.



Gambar 3.49 *Rechecking Missing Data of Project Status Dataset*

3. *The cleaned data is stored in .xlsx format*

Setelah proses pembersihan dan pengisian data selesai dilakukan, langkah selanjutnya adalah menyimpan dataset yang telah diperbarui ke dalam file Excel baru. Untuk keperluan tersebut, ditentukan path penyimpanan dengan nama file Status_Proyek_2024_new.xlsx yang diletakkan di direktori yang sama, yaitu di folder *Status_Proyek_2024*. Path tersebut didefinisikan ke dalam variabel file_path_new menggunakan format *raw string* untuk menghindari kesalahan pembacaan karakter escape. Penentuan path ini bertujuan untuk memastikan bahwa file hasil transformasi data dapat diakses dan digunakan kembali untuk proses analisis lanjutan maupun dokumentasi.



Gambar 3.50 *The cleaned data is stored in .xlsx format*

3.2.14 Membuat *Dashboard* Menggunakan Power Bi Untuk Menganalisis Dataset Status Proyek

Monitoring status proyek tahun 2024 ditampilkan dalam bentuk visual yang interaktif untuk mempermudah pemantauan perkembangan setiap proyek. Informasi dapat *difilter* berdasarkan divisi, nama proyek, status proyek, serta bulan pelaksanaan, sehingga analisis dapat difokuskan pada data tertentu sesuai kebutuhan. Pada tampilan yang ditampilkan, proyek yang dipilih berasal dari divisi *Steel Fabrication*

dengan status masih dalam proses, dimulai pada tanggal 1 Januari 2024 dan ditargetkan selesai pada 3 Februari 2024.

Data ditampilkan dalam berbagai bentuk grafik seperti grafik garis untuk melihat tren jumlah proyek yang dimulai tiap bulan, grafik batang untuk menunjukkan jumlah proyek berdasarkan status, serta diagram lingkaran yang memperlihatkan persentase proyek berdasarkan statusnya. Selain itu, ditampilkan juga daftar proyek lengkap dengan ID, nama proyek, divisi, dan statusnya. Penyajian ini membantu dalam memahami kondisi proyek secara keseluruhan dan memudahkan dalam mengambil keputusan berdasarkan data yang tersedia.

Gambar 3.51 *Dashboard of Project Status Dataset*

3.3 kendala dan kesulitan yang Ditemukan

Selama proses magang sebagai *Data Analyst* di PT Wijaya Karya Industri & Konstruksi, terdapat beberapa kendala dan kesulitan yang dihadapi yaitu :

1. Dalam proses pembersihan data, ditemukan banyak nilai kosong pada beberapa kolom penting, yang jika tidak ditangani secara tepat dapat menyebabkan hasil analisis menjadi bias atau tidak akurat, sehingga memerlukan perhatian khusus dalam proses penanganannya.
2. Terdapat perbedaan atau ketidakkonsistenan dalam penamaan kolom antar berbagai dataset yang digunakan, baik dari segi struktur penulisan maupun format, sehingga menyulitkan proses penyatuan data dan menimbulkan potensi kesalahan dalam pengolahan data lebih lanjut.
3. Data yang diperlukan untuk analisis tidak tersedia secara terpusat, melainkan tersebar di beberapa file dengan format yang berbeda-beda, sehingga memerlukan upaya tambahan untuk mengintegrasikan seluruh data menjadi satu kesatuan yang utuh dan siap dianalisis.
4. Dalam proses pembuatan dashboard, ditemukan kesulitan dalam memilih jenis grafik atau chart yang tepat karena banyaknya pilihan teknik visualisasi yang tersedia, serta perlunya pemahaman yang mendalam terkait konteks data agar informasi yang ditampilkan dapat disampaikan secara efektif kepada pihak manajemen.

3.4 Solusi atas Kendala yang Ditemukan

Adapun solusi yang diterapkan untuk kendala-kendala yang ditemukan selama magang sebagai *Data Analyst* di PT Wijaya Karya Industri & Konstruksi, yakni :

1. Dilakukan teknik imputasi data, yaitu mengganti nilai kosong tersebut menggunakan nilai rata-rata, median, atau modus tergantung pada jenis data yang bersangkutan, atau melakukan penghapusan kolom jika tidak digunakan.
2. Dalam menghadapi ketidakkonsistenan nama kolom, dilakukan proses standarisasi secara menyeluruh dengan mengubah nama kolom menjadi

format yang seragam, baik melalui proses pencocokan string secara otomatis maupun dengan menggantinya secara manual untuk memastikan kesesuaian antar dataset.

3. Untuk menyatukan data yang tersebar di berbagai sumber, dilakukan proses integrasi data menggunakan tools Python dengan bantuan pustaka pandas, khususnya melalui fungsi concat, sehingga seluruh informasi dari berbagai file dapat digabungkan ke dalam satu dataset terpadu yang siap dianalisis lebih lanjut.
4. Dalam menghadapi tantangan pemilihan teknik visualisasi yang tepat, solusi yang dilakukan adalah dengan mempelajari berbagai referensi dan tutorial terkait visualisasi data, baik melalui sumber online maupun dokumentasi resmi, serta berkonsultasi secara aktif dengan supervisor guna menentukan jenis visualisasi yang paling relevan dengan kebutuhan pengguna dan tujuan analisis data.