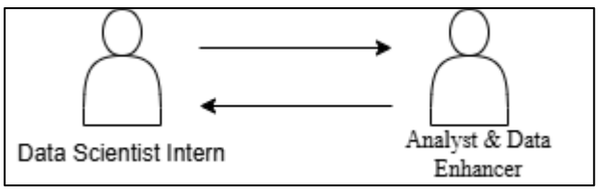
BAB III PELAKSANAAN KERJA MAGANG

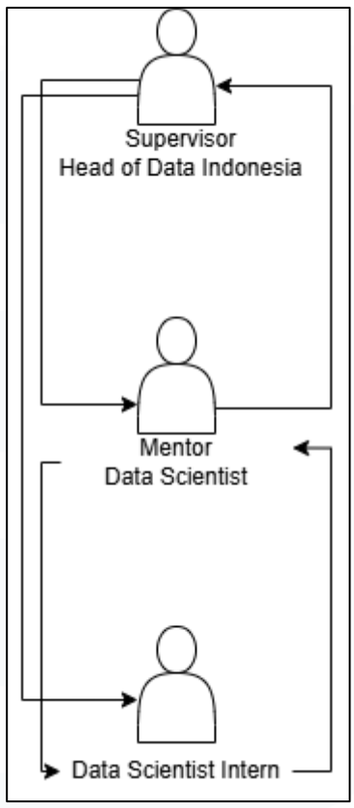
3.1 Kedudukan dan Koordinasi

Selama periode praktik kerja magang di Data Indonesia selama 5 bulan dari tanggal 16 Desember 2024 sampai 15 April 2025, mahasiswa yang mendapatkan posisi sebagai *Data Scientist* mengerjakan pekerjaan yang diberikan dengan mendapat bimbingan dari mentor yang telah ditetapkan oleh supervisor. Pendamping mentor mahasiswa sebagai *Data Scientist* magang adalah Ridhwan Mustajab, salah satu Data Scientist di Data Indonesia. Mentor bertugas sebagai pemberi materi pembelajaran, dokumentasi serta memberikan tugas terkait dengan kebutuhan Data Indonesia. Tugas yang diberikan kepada mahasiswa yang magang sebagai *Data Scientist* juga diberikan oleh supervisor sekaligus kepala atau pimpinan Data Indonesia, Bapak Setyardi Widodo.

Tugas yang diberikan oleh supervisor bisa melalui dua tahapan yaitu diberikan secara langsung oleh supervisor bersama dengan *Data Scientist* lainnya atau supervisor memberikan tugas untuk mahasiswa magang kepada mentor lalu mentor memberikan tugas kepada mahasiswa yang magang sebagai *Data Scientist.* Selain itu, *Data Scientist* magang mendapatkan tugas dengan bekerja sama dengan *Analyst & Data Enhancer* untuk menyediakan data yang diperlukan oleh *Analyst & Data Enhancer. Data Scientist Intern* akan mendapatkan feedback dari mentor jika tugas yang diberikan berasal dari mentor. Mentor juga akan memberikan feedback terhadap tugas yang diberikan oleh supervisor secara langsung sedangkan tugas dari Analyst & Data Enhancer akan mendapatkan feedback dari *Analyst & Data Enhancer.*



Gambar 3.1. Kedudukan dan Koordinasi Data Scientist dengan Analyst & Data Enhancer.



Gambar 3.2. Kedudukan dan Koordinasi Data Scientist dengan supervisor & mentor.

Selama kerja magang, *Data Scientist* mengemban kewajiban untuk menyediakan data serta menyajikan data dalam bentuk visualisasi dan dashboard. Beberapa Alat yang digunakan untuk mengerjakan tugas perlu disesuaikan agar semua *Data Scientist* dapat mengerjakan bersamaan atau saling berkolaborasi sehingga *tools* seperti google sheet, google colab, dan Looker Studio sering digunakan digunakan sebagai bentuk file terakhir sebelum diberikan kepada mentor atau supervisor untuk mendapatkan *feedback*. Alat untuk menunjang pemrograman dan visualisasi yang lebih berat seperti Visual Studio Code atau Power BI untuk bisa dikerjakan bersama masih tetap digunakan sebagai media penunjang.

Selama menjalankan program magang terdapat tantangan yang muncul ketika pengolahan data dan perlu dihadapi untuk bisa menghasilkan hasil akhir yang diinginkan. Pada program magang ini, mahasiswa diberikan tugas untuk membuat dashboard sebagai hasil akhir dari data yang telah diolah. Dashboard

yang dibuat *Data Scientist* disusun sesuai dengan pengguna yang ditujukan. Data yang diperoleh pun disesuaikan berdasarkan kebutuhan pengguna dashboard. Data yang digunakan dapat berupa data publik yang bisa diakses atau data lembaga pemerintah yang memerlukan izin untuk mengaksesnya. Mentor akan memberikan feedback pada data yang akan digunakan sebelum divisualisasikan dengan dashboard. Jika data yang sudah dirasa sudah cukup maka mentor akan membimbing untuk membuat dashboard sehingga informasi yang ditampilkan sesuai dengan kebutuhan pengguna dashboard.

3.2 Tugas dan Uraian Kerja Magang

Data Scientist di Data Indonesia memiliki tugas seperti *Data Scientist* lainnya yaitu mengumpulkan, mengolah, dan menganalisis data serta merancang visualisasi hingga menyajikan dalam bentuk dashboard. Program magang ini bertujuan untuk memberikan kesempatan kepada mahasiswa yang akan menjadi *Data Scientist* untuk bisa memperoleh pengalaman yang komprehensif terkait posisi tersebut di Industri. *Data Scientist Intern* mendapatkan tugas untuk membuat dashboard berdasarkan periode-periode yang telah ditentukan. Namun, jika terdapat permintaan untuk mengumpulkan data melalui scraping website maka periode pembuatan dashboard bisa diperpanjang untuk bisa mendapatkan format data yang sesuai. Data yang telah dipersiapkan sesuai dengan format yang diberikan oleh mentor atau sesuai dengan tugas yang diberikan juga merupakan salah satu hasil akhir selain dalam bentuk visualisasi. Data Scientist juga mendapatkan tugas pengolahan data yang perlu diselesaikan setiap hari untuk kebutuhan publikasi edisi terbaru sesuai dengan data yang diperlukan oleh *Analyst & Data Enhancer*

Data yang tidak terlalu besar dapat diolah menggunakan Microsoft Excel Power Query dan Google Excel, alat visualisasi yang digunakan adalah Looker Studio serta membuat program scraping data menggunakan bahasa pemrograman Python di IDE Visual Studio Code. Hasil dari tugas yang dikerjakan bisa berbentuk dashboard visualisasi data atau tabel data dengan format *crosstab* atau *long-tab* Berikut adalah tabel aktivitas selama program magang di Data Indonesia.

Tabel 3.1 Jadwal aktivitas program magang.

No	Nama Aktivitas	Waktu Pengerjaan	Tanggal Mulai	Tanggal Selesai
1	On-Boarding, Pengenalan, dan Pendalaman tools yang digunakan di Perusahaan	Minggu ke-1 dan Minggu ke-2	16 Desember 2024	31 Desember 2024
2	Membuat Dashboard Perkebunan dengan data lembaga pemerintah Badan Pusat Statistik	Minggu ke-3 dan Minggu ke-4	2 Januari 2025	15 Januari 2025
3	Membuat Dashboard Upah dengan data lembaga pemerintah Badan Pusat Statistik dan Kementerian Ketenagakerjaan	Minggu ke-5 dan Minggu ke-6	16 Januari 2025	31 Januari 2025
4	Membuat Dashboard Monitoring E-Commerce dengan Scraping	Minggu ke-7, Minggu ke-8, Minggu ke-9, Minggu ke-10, Minggu ke-11, Minggu ke-12,	3 Februari 2025	14 Maret 2025
5	Web Scraping Data Custom	Minggu ke-13, Minggu ke-14, Minggu ke-15 Minggu ke-16	17 Maret 2025	22 April 2025

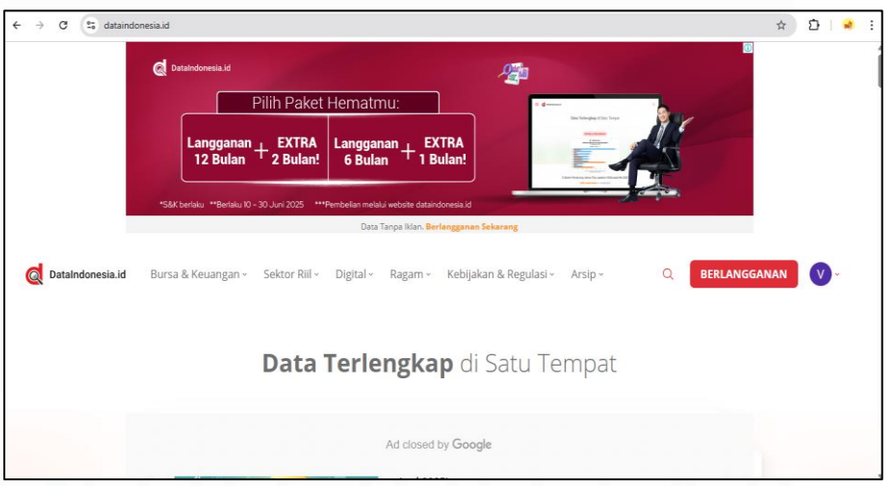
3.2.1 On-Boarding, Pengenalan, dan Pendalaman tools yang digunakan di Perusahaan

Hari pertama masuk program kerja magang adalah 16 Desember 2024. Di hari pertama magang, perwakilan *Human Resource* Bisnis Indonesia Grup mengantar mahasiswa magang untuk melakukan diskusi mengenai kontrak magang dan benefit dari progam kerja magang ini. Diskusi dilakukan di lantai 5 Wisma Bisnis Indonesia. Setelah menyelesaikan diskusi dengan *Human Resource,* mahasiswa magang di antar ke lantai ke masing-masing lini usaha Bisnis Indonesia Grup sesuai dengan tempat magang yang bersangkutan. Data Indonesia berada di lantai 6 Wisma Bisnis Indonesia dan sebelum memulai program magang mahasiswa magang perlu berdiskusi lebih lanjut dengan supervisor atau *Head of Department* Data Indonesia.

Supervisor memberikan pertanyaan seputar kemampuan sebagai *Data Scientist* seperti bahasa pemrograman yang mampu menangani proses pengolahan data yang pernah dilakukan, fokus keahlian di

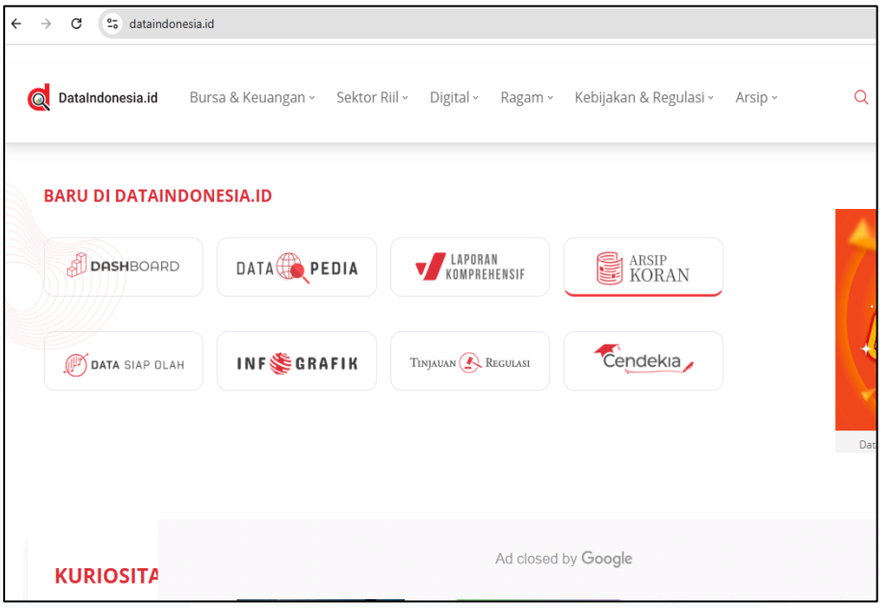
bidang data mahasiswa, hingga pengalaman dalam menggunakan alat visualisasi data seperti Power Bi, Looker Studio, atau Tableau. Penjelasan mengenai tingkat kemampuan mahasiswa akan menjadi tolak ukur terhadap tugas yang akan dibebankan kepada mahasiswa. Supervisor memberikan gambaran tentang Data Indonesia, tujuan, dan visi misi Data Indonesia. Mentor kemudian mengambil alih untuk menjelaskan tugas yang akan dilakukan *Data Scientist* selama magang di Data Indonesia meliputi pembuatan dashboard, pengolahan data, dan membuat program untuk melakukan *web scraping.*

Data Scientist diberikan akses ke website Data Indonesia untuk bisa melihat program yang berjalan di Data Indonesia. Gambar berikut adalah halaman awal Data Indonesia yang dapat diakses oleh umum. Pengguna umum bisa melihat artikel yang ditayangkan secara gratis dan dapat membayar lebih biaya berlangganan untuk bisa melihat artikel yang lebih lengkap dan fitur yang lebih luas.



Gambar 3.3. Landing Page website DataIndonesia.ID.

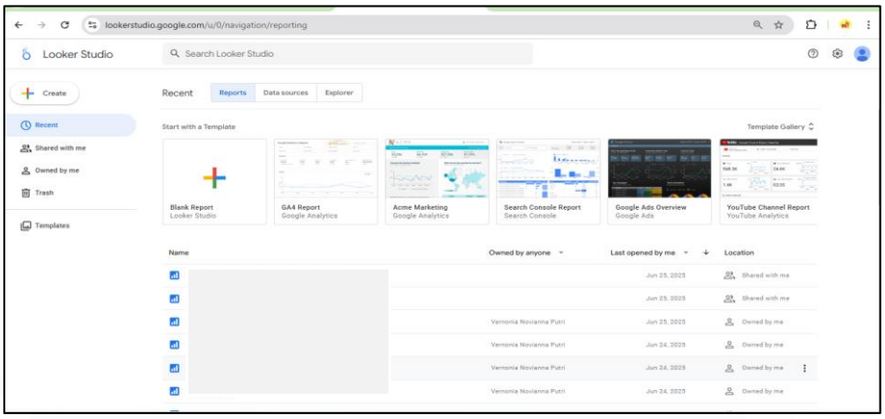
Artikel yang tersedia di website DataIndonesia.ID dapat dibagi ke dalam 6 sektor utama yaitu Bursa & Keuangan, Sektor Riil, Digital, Ragam, Kebijakan & Regulasi, dan Arsip. Tiap sektor memiliki sub sektor yang bisa dipilih. Artikel yang terbit akan dikelompokkan berdasarkan sektor-sektor tersebut.



Gambar 3.4. Fitur yang dikembangkan oleh DataIndonesia.ID

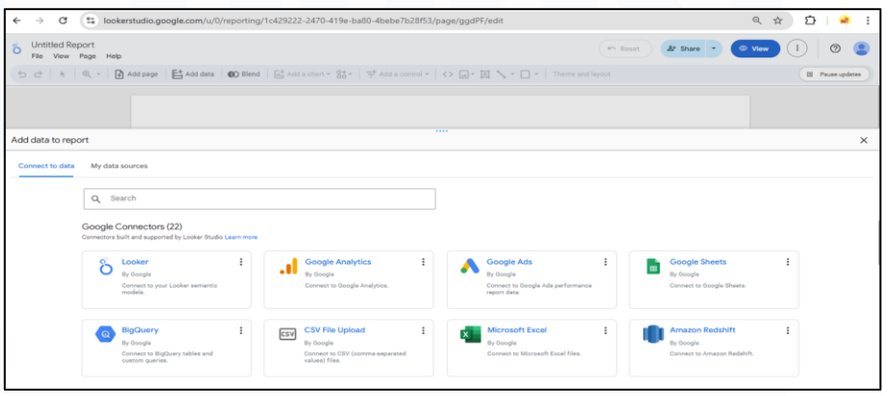
Data Scientist magang akan ditugaskan untuk mengembangan fitur Dashboard dan Data Siap Olah. Tujuan dari dashboard yang dibuat adalah bisa menjadi manfaat bagi masyarakat umum yang menggunakan dashboard Data Indonesia sehingga dashboard yang dibuat perlu disusun dengan tujuan pengguna bisa memahami antarmuka dengan mudah melalui desain navigasi komponen yang intuitif.

Data Scientist diberikan akses kepada Looker Studio yang digunakan untuk membuat dashboard yang akan ditayangkan di website DataIndonesia.ID untuk agar bisa mengenal alat kerja selama magang. Looker Studio atau sebelumnya dikenal sebagai Google Data Studio adalah alat visualisasi yang dikembangkan oleh Google untuk membuat dashboard yang interaktif yang bisa digunakan untuk membuat pelaporan atau *Reporting* berbasis data dan visualisasi. Looker Studio dapat diakses melalui web dan juga dapat dihubungkan dengan berbagai sumber data untuk membuat dashboard.



Gambar 3.5. Looker Studio untuk pembuatan dashboard.

Pengguna Looker Studio bisa memilih fitur *Blank Report* untuk membuat dashboard. Jika pengguna hanya ingin membuat *data source* maka aksi yang dilakukan adalah menekan tombol *Create* lalu pilih *Data Source*. Halaman yang ada pada gambar di bawah paragraf ini akan muncul ketika fitur *Blank Report* dipilih. *Data Source* yang akan digunakan bisa diambil dari koneksi yang dibuat dengan sumber data masing-masing. Di Data Indonesia, sumber data final atau yang akan terkoneksi dengan dashboard looker adalah Google Sheet walaupun untuk pengolahan data sebelum menjadi Google Sheet, *Data Scientist* dapat menggunakan DBMS atau *Database Management System* atau alat pengolahan data yang bisa digunakan oleh *Data Scientist.*



Gambar 3.6. Koneksi data source looker.

Setelah memahami antarmuka dan fitur-fitur yang tersedia di Looker Studio. *Data Scientist* perlu mempelajari dashboard yang telah

dibuat dan dipublikasikan oleh *Data Scientist* di Data Indonesia untuk memahami dashboard yang akan dibuat selama program kerja magang. Salah satu dashboard yang dipelajari pada masa *Onboarding* adalah dashboard yang membahas tentang kependudukan di Indonesia.

Pada dashboard tentang kependudukan tersebut, *Data Scientist* magang harus bisa membedah tujuan dashboard tersebut, komponen visualisasi yang digunakan, data yang diperlukan agar bisa membuat dashboard sesuai dengan tujuan, dan alasan pemilihan elemen visualisasi tersebut.

Data Scientist menganalisis tujuan dari pembuatan dashboard tersebut dan menghasilkan beberapa poin analisis yaitu.

1. Memberikan akses informasi yang mudah kepada pengguna atau masyarakat umum tentang data demografi di Indonesia sehingga masyarakat umum tidak perlu menelaah data mentah atau dokumen untuk mendapatkan informasi tersebut.
2. Dashboard dibuat dengan filter yang interaktif agar pengguna bisa memilih untuk fokus terhadap wilayah atau bidang yang ingin dicari berdasarkan kebutuhan analisis.
3. Data demografi yang berisi variabel yang bermacam-macam akan lebih mudah dicerna oleh masyarakat umum jika ditampilkan dalam bentuk visual seperti peta, tabel, dan grafik.
4. Lalu, informasi yang ditayangkan dalam bentuk dashboard tersebut bisa menjadi pendukung dalam pengambilan keputusan oleh instansi atau lembaga pemerintah.

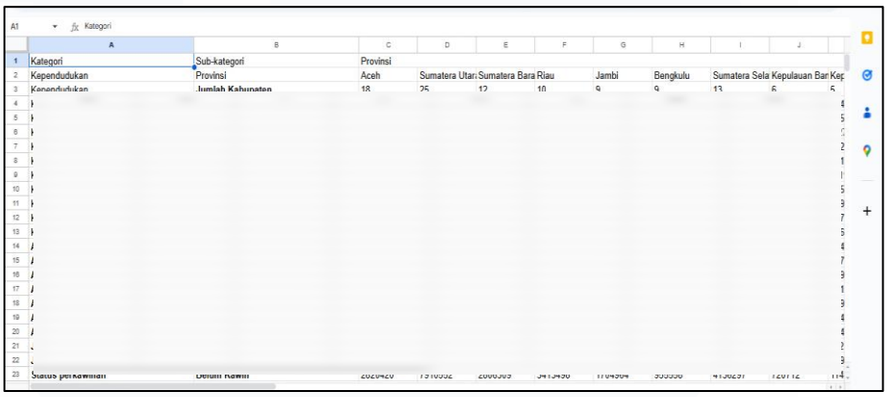
Tampilan dashboard tersebut menampilkan komponen yang digunakan dan data yang perlu dipersiapkan untuk membuat dashboard seperti yang perlu dipelajari. Berdasarkan analisis yang telah dilakukan dapat diperoleh data yang perlu dikumpulkan adalah data penduduk di tiap provinsi yang mencangkup jumlah dan kepadatan penduduk, status perkawinan penduduk di tiap provinsi, status atau tingkat pendidikan masyarakat di 38 Provinsi di Indonesia, status kesehatan penduduk dan jumlah populasi berdasarkan kepercayaan di masyarakat Indonesia. Selain itu, data wilayah seperti titik koordinat provinsi dan luas wilayah juga diperlukan untuk membuat dashboard yang dicontohkan.

Berdasarkan analisis yang telah dilakukan, beberapa komponen visualisasi yang dipilih untuk menampilkan data tersebut adalah,

1. Komponen grafik yang umum digunakan untuk visualisasi data seperti *scorecard* untuk menampilkan angka yang menjadi fokus utama dari dashboard sehingga pengguna tidak perlu menelusuri grafik. *Scorecard* dapat digunakan untuk menunjukan ringkasan statistik nasional yang menjadi informasi awal pengguna sebelum melanjutkan ke komponen yang menjadi penjelas dari ringkasan statistik tersebut.
2. Komponen peta juga digunakan untuk menampilkan sebaran penduduk di Indonesia berdasarkan warna sehingga pengguna atau *user* dapat menemukan informasi berdasarkan pola. Peta mampu menunjukan gambaran atau visual geografis dari data yang digunakan.
3. Grafik Batang digunakan untuk menampilkan kategori yang muncul pada data yang digunakan. Kategori muncul akibat adanya pengelompokan data sehingga pengguna dapat membandingkan dan mengetahui informasi yang muncul dari suatu kelompok atau populasi. Grafik batang fokus untuk menunjukan perbandingan visual.
4. *Pie Chart* digunakan untuk menampilkan proporsi atau kategori dalam satu kelompok data. Grafik *pie* dapat menunjukan kategori terbesar atau kategori dengan kuantitas data yang lebih kecil dengan cepat. Tampilan grafik ini yang sederhana dan mudah untuk dipahami menjadi pilihan untuk menyajikan data ke pengguna awam.
5. Data juga ditampilkan dalam bentuk tabel yang memiliki komponen *heatmap.* Penggunaan tabel pada dashboard didasarkan adanya data yang perlu ditampilkan secara langsung tanpa visual. Tabel juga memiliki fungsi untuk menampilkan data peringkat misalnya peringkat kepadatan penduduk antar provinsi.
6. Di Looker Studio terdapat komponen kontrol yang dapat digunakan *Data Scientist* untuk memberikan fitur interaktif di dashboard. Komponen kontrol dapat digunakan sebagai filter untuk mengurangi

kepadatan visual sehingga pengguna dapat melihat data yang dibutuhkan atau relevan terhadap kebutuhan analisis.

Setelah melakukan analisis pada dashboard yang digunakan sebagai contoh, *Data Scientist* juga perlu mempelajari format pada sumber data yang terkoneksi dengan dashboard. Sumber data yang terkoneksi dengan dashboard kependudukan tersebut adalah Google Sheet yang terlihat pada gambar dibawah ini.



Gambar 3.7. Data yang digunakan pada dashboard contoh pembelajaran.

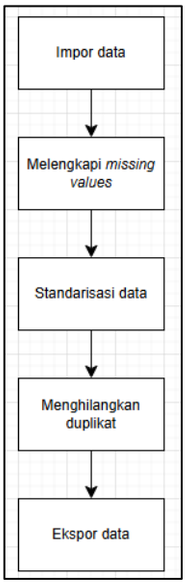
Dari penjelasan yang telah diberikan oleh mentor, data yang digunakan pada dashboard ini berasal dari lembaga statistik di Indonesia atau dikenal sebagai Badan Pusat Statistik (BPS). Badan Pusat Statistik merupakan lembaga non-kementrian yang bertugas untuk menyediakan data statistik untuk pemerintah dan masyarakat [8]. Data yang telah diperoleh dari BPS akan dikumpulkan dalam satu file yang akan diolah langsung di Google Sheet. Setiap data yang digunakan akan tersimpan di *Sheet* masing-masing sesuai dengan nama ketika diunduh. Setelah melakukan analisis pada dashboard yang menjadi contoh, Mentor memberikan evaluasi terhadap analisis tersebut serta memberikan saran untuk membuat dashboard dengan efisien.

Mentor menjelaskan bahwa sebelum menggunakan data yang diperoleh untuk dihubungkan sebagai *Data Source* dengan dashboard, *Data Scientist* perlu mempersiapkan data tersebut dengan melakukan metode *Data Cleaning* dan transformasi pada data. Data yang dimiliki perlu dibersihkan terlebih dahulu atau melakukan metode yang disebut *Data Cleaning. Data Cleaning* merupakan metode dalam pengolahan

data untuk mengidentifikasi, menemukan, menghapus, atau mengubah data yang tidak sesuai seperti data yang tidak lengkap, data yang terduplikasi, atau data yang tidak relevan sehingga data yang telah

melalui proses *Data Cleaning* aman digunakan untuk proses analisis data selanjutnya [9]. Jika terdapat kesalahan atau *miss-step* saat melakukan pembersihan data maka akurasi dan kerelevanan data yang digunakan bisa mempengaruhi proses pengolahan data selanjutnya.

Beberapa tahapan yang harus dilakukan saat *Data Cleaning* ada pada gambar berikut. Pembersihan data mencangkup dari mulainya data diimpor ke Google Sheet kemudian mencari nilai kosong atau *missing values* untuk kelengkapan data. Jika terdapat nilai kosong, *Data Scientist* perlu menentukan untuk mencari data yang hilang atau tidak mengindahkan nilai yang hilang tersebut karena tidak termasuk dalam prioritas untuk digunakan. Setelah data yang digunakan tidak ada lagi *missing values* saat dicek, makan langkah selanjutnya yang perlu dilakukan adalah menstandarkan format data yang digunakan. Misalnya, penggunaan satuan berat massa menggunakan kilogram atau ton, angka numerik ditampilkan dalam format pemisah ribuan dengan tanda titik atau koma, nama provinsi akan akan berisi huruf kapital semua atau hanya huruf pertama pada kata yang menjadi kapital, dan jumlah kategori yang akan digunakan berdasarkan standar BPS atau lembaga lain. Data yang telah terstandar berdasarkan format yang dibutuhkan selanjutnya dicari data yang terduplikasi. Data yan terduplikasi perlu dipastikan kalau data tersebut memang terdiri dari beberapa data dan normal jika terduplikat sebelum dihapus. Proses *Data Cleaning* dapat diulang kembali jika terdapat perubahan kebutuhan saat menyusun dashboard



Gambar 3.8. Tahapan Data Cleaning

Tahap Berikutnya adalah transformasi data atau disebut sebagai *Data Wrangling.* Transformasi data berupaya untuk mengatur data ke dalam format yang lebih mudah digunakan sehingga analisis yang dihasilkan lebih akurat dan proses pengolahan data menjadi lebih efisien. Pada proses pengolahan data untuk dashboard yang dicontohkan, tahapan transformasi data yang dilakukan meliputi,

1. Menggabungkan data dalam bentuk *JOIN* atau *MERGE* untuk mendapatkan sumber data (*Data Source).* Langkah ini perlu dilakukan karena sumber data awal terpisah atau tidak dalam satu file yang sama.
2. Menambahkan informasi yang berkaitan tentang kependudukan dengan memasukan data yang ada ke dalam rumus untuk menghasilkan kolom baru misalnya kolom jumlah kepadatan penduduk.
3. Mengubah bentuk atau struktur data menjadi tabular yang sesuai dengan kebutuhan visualisasi. Salah satu cara untuk mengubah struktur data adalah menggunakan fitur transpose atau pivot yang ada di Google Sheet.

Transpose adalah cara untuk mengubah orientasi data sehingga data yang awalnya dari format baris berubah menjadi kolom atau sebaliknya. Transpose dibutuhkan saat struktur data yang dimiliki memungkinkan

untuk data yang berbentuk baris menggantikan kolom data awal. Contoh transpose yang dilakukan dapat dilihat dari gambar 3.9 dibawah. Sebelum dilakukan transpose data memiliki 3 kolom dan 8 baris. Kolom atribut yang berisikan jenis-jenis agama di Indonesia terletak pada satu kolom yang sama. Setelah diterapkan transpose struktur daya berubah menjadi 8 kolom dan 2 baris. Atribut agama berpindah dari baris menjadi nama kolom yang berisi nilai jumlah pemeluk.

Provinsi	Atribut	Nilai
Aceh	Islam	5492487
Aceh	Kristen	64984
Aceh	Katholik	5982
Aceh	Hindu	90
Aceh	Buddha	6658
Aceh	Konghucu	0
Aceh	Kepercayaan ter	252

Gambar 3.9. Struktur data awal

	A	B	C	D	E	F	G	H	I
1	Provinsi	Islam	Kristen	Katholik	Hindu	Buddha	Konghucu	Kepercayaan terhadap Tuhan YME	
2	Aceh	5492487	64984	5982	90	6658	0	252	

Gambar 3.10. Struktur data setelah transpose

Fitur Pivot dapat digunakan untuk meringkas data dan merepresentasikan data dalam berbagai format bentuk sesuai dengan kebutuhan. Tabel Pivot lebih komplek dari fitur transpose data karena tabel pivot dapat menggabungkan data, mengelompokan data, dan menghitung data berdasarkan kriteria atau filter yang dipilih.

Contoh data awal yang ada pada gambar 3.11 dibawah hanya memiliki jumlah nilai yang muncul tiap kuartal atau triwulan dan kolom pilihan terdapat 2 data yaitu sektor dan negara. Setelah menggunakan tabel pivot, struktur tabel berubah dari data dengan 5 kolom menjadi tabel data dengan kolom yang hanya memuat 2 baris dengan banyak kolom bertambah sesuai tahun. Pivot table juga dapat menghasilkan total nilai per tahun dengan menghitung nilai yang muncul tiap kuartal.

A	B	C	D	E	F
Tahun	Pilihan	Uraian	Triwulan	Ket	Nilai
2017	Sektor	Sektor Primer	Q1	Jumlah	565
2017	Sektor	Sektor Primer	Q1	Nilai	1648.9
2017	Sektor	Sektor Primer	Q2	Jumlah	791
2017	Sektor	Sektor Primer	Q2	Nilai	1502.3
2017	Sektor	Sektor Primer	Q3	Jumlah	597
2017	Sektor	Sektor Primer	Q3	Nilai	1381
2017	Sektor	Sektor Primer	Q4	Jumlah	915

Gambar 3.11. Struktur data awal.

A	B	C	D	E	F	G
SUM of Nilai	Tahun	Kuartal				
	2017 Total	2018				2018 Total
Pilihan		Q1 2018	Q2 2018	Q3 2018	Q4 2018	
Negara	32276642.2	8135843.7	7152849.7	6657526.7	7391927.8	29338147.9
Sektor	69164.8	13140.9	19498.5	15364.8	18311.7	66315.9
Grand Total	**32345807**	**8148984.6**	**7172348.2**	**6672891.5**	**7410239.5**	**29404463.8**

Gambar 3.12. Struktur data setelah pivot tabel.

Jika proses transformasi berhasil dilakukan maka tahap selanjutnya adalah menghubungkan atau membuat koneksi antara sumber data (*Data Source)* di Google Sheet dengan Dashboard Looker Studio. Saat memilih data yang akan digunakan untuk dashboard, pengguna Looker Studio harus memilih *Sheet* tempat data yang telah terintegrasi dan telah melewati proses pengolahan data. Program aktivitas *Onboarding* dilaksanakan selama 2 minggu hingga akhir tahun 2024. Selama 2 minggu tersebut, *Data Scientist* melakukan pendalaman *tools* yang akan digunakan dan mencoba untuk membuat dashboard untuk berlatih secara mandiri.

3.2.2 Membuat Dashboard Perkebunan dari Data Lembaga Pemerintah Badan Pusat Statistik

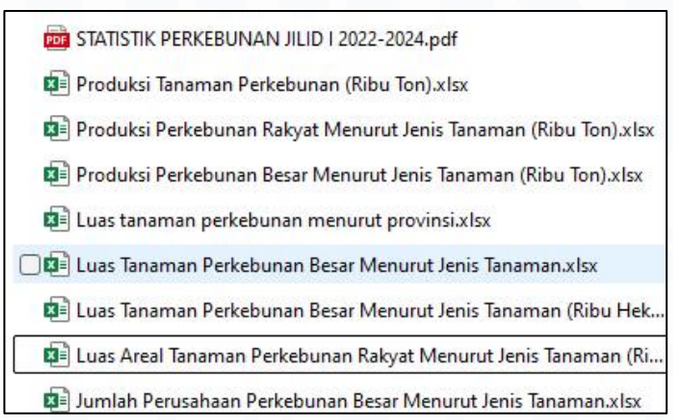
Pada bulan Januari bulan ke-3 program magang, mentor memberikan tugas kepada *Data Scientist* untuk membuat dashboard dengan menggunakan data publik dari lembaga atau institusi yang terpercaya. Mentor menyarankan untuk menggunakan data dari Badan Pusat Statistik (BPS) dan memberikan kesempatan kepada *Data*

Scientist magang untuk menggunakan pengetahuan yang telah diperoleh selama program pengenalan selama 2 minggu.

Di hari pertama, proyek pembuatan dashboard ini dimulai dari tema dashboard yang akan dibuat yaitu dashboard dengan menggunakan data tentang perkebunan. Setelah mendapatkan arahan dari mentor, *Data Scientist* membuat rancangan dashboard yang berisi tujuan, data, hingga komponen yang akan digunakan. Rancangan tersebut akan menjadi dasar dari pembuatan dashboard dan akan menjadi pembanding saat muncul perubahan selama pengerjaan proyek dashboard perkebunan. Berdasarkan rancangan dashboard perkebunan, tujuan dari pembuatan dashboard ini adalah untuk menyediakan akses informasi dan data mengenai perkebunan di Indonesia kepada masyarakat umum, dan membantu pelaku industri untuk memahami distribusi dan tren produksi komoditas perkebunan. Data yang perlu dikumpulkan perlu memiliki variabel seperti,

1. Komoditas atau jenis tanaman perkebunan.
2. Jumlah produksi komoditas tiap provinsi.
3. Luas area lahan komoditas tiap provinsi.
4. Kategori jenis usaha perkebunan.
5. Rentang waktu data 10 tahun dari tahun dashboard dibuat.

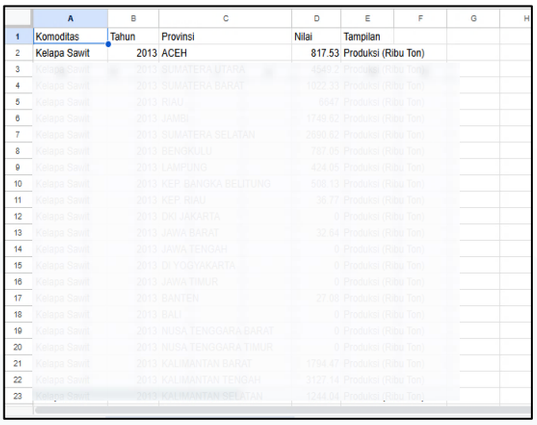
Data yang dikumpulkan harus berisi variabel dari rancangan kebutuhan data yang akan digunakan. Terdapat 6 set data yang berhasil dikumpulkan dari BPS dan memiliki variabel yang dibutuhkan.



Gambar 3.13. Data yang diperlukan untuk diolah.

Tahap yang dilakukan selanjutnya adalah eksplorasi data atau *Exploratory Data Analysis (*EDA*)* data *Data Cleaning.* Tahap ini dilakukan untuk memahami struktur dan kualitas data sebelum dilanjutkan dengan tahap visualisasi data. EDA dilakukan pada alat pengolahan data Microsoft Excel dengan alasan data yang digunakan tidak membutuhkan pengolahan data yang kompleks. Gambaran besar tahapan EDA yang dilakukan pada data untuk pembuatan dashboard perkebunan adalah,

1. Melihat data sampel dengan menampilkan sebagian data baris awal dan baris akhir di setiap set data untuk melihat struktur kolom dan baris umum datanya.
2. Memahami struktur set data yang digunakan. Salah satu cara yang dilakukan adalah mengetahui tipe data kolom di set data yang akan digunakan. Contohnya kolom provinsi yang memiliki tipe data *text* dan kolom luas yang memiliki tipe data float. Pada tahap ini, satuan yang digunakan oleh kolom tipe data numerik distandarkan agar semua data set memiliki satuan massa atau luas yang sama.
3. Menangani data yang hilang atau *Missing Values*. Kolom set data yang digunakan dicek kelengkapannya dengan menggunakan filter kolom pada excel atau menggunakan conditional formating untuk mencari *Blank Cell.* Kolom provinsi, komoditas, dan total produksi atau luas menjadi prioritas untuk diperiksa.
4. Mendeteksi data duplikat dengan menggunakan fitur yang telah tersedia di Excel atau Google Sheet yaitu fitur *Remove Duplicates.* Kolom yang rawan untuk data duplikat adalah kolom yang berisi tahun atau waktu serta komoditas pada aktivitas pembuatan dashboard perkebunan ini.
5. Melakukan analisis sederhana untuk mengetahui distribusi data dengan membuat grafik sederhana yaitu grafik batang dan grafik pai untuk melihat jangkauan nilai produksi tiap komoditas dan provinsi.
6. Menggunakan tabel pivot untuk mendapatkan ringkasan data yang bisa disesuaikan dengan kebutuhan. Misalnya, menghitung jumlah produksi tiap kategori pada setiap tahun.

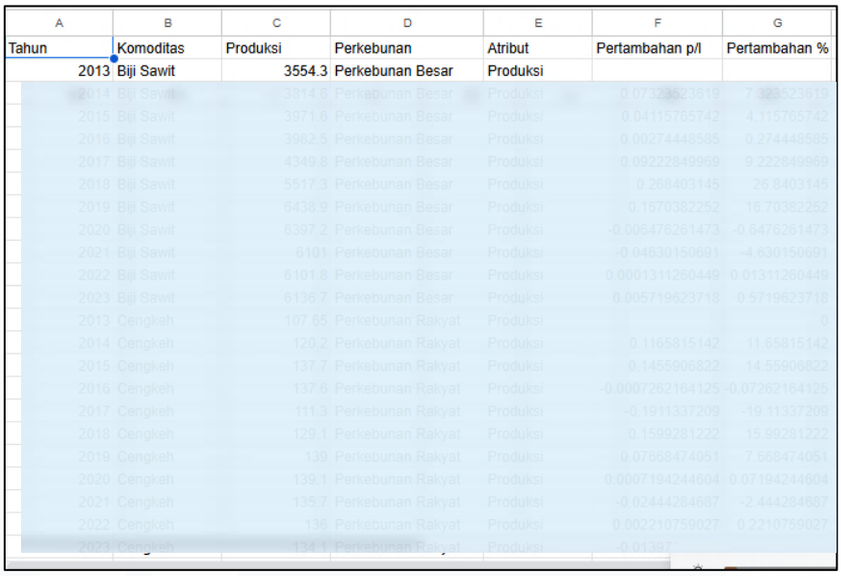


	A	B	C	D	E	F	G
1	Komoditas	Tahun	Provinsi	Nilai	Tampilan		
2	Kelapa Sawit	2013	ACEH	817.53	Produksi (Ribu Ton)		

Gambar 3.14. Data yang telah melalui tahap ekplorasi data

Tahapan persiapan data selanjutnya adalah transformasi data. Transformasi data yang dilakukan pada data tentang perkebunan ini terdiri dari menggabungkan data menjadi satu sumber data dalam satu *Sheet.* Penggabungan data yang yang dilakukan adalah *Merge* atau menumpuk data sehingga data gabungan berisi data dari jangka waktu 10 tahun. Pada tahap ini, *Data Scientist* dapat menambahkan informasi baru jika dirasa data yang diperoleh masih kurang untuk bisa merepresentasikan perkebunan dan komoditas di Indonesia dengan menambahkan kolom pertumbuhan dengan rumus menggunakan data yang tersedia.

Struktur data pada *Sheet* yang akan digunakan perlu diubah ke dalam bentuk atau format yang memudahkan untuk visualisasi menggunakan Looker Studio sehingga format diubah menjadi memanjang ke bawah dan mengurangi sebanyak-banyaknya kolom. Kolom produksi dan kolom luas lahan komoditas digabung menjadi satu kolom yaitu kolom nilai atau *value.*



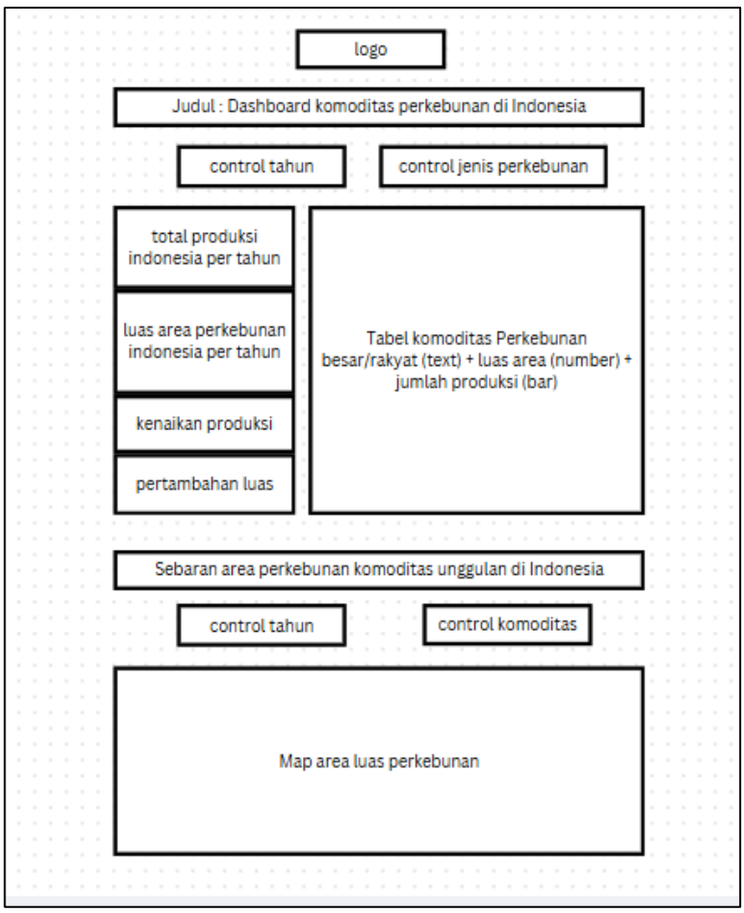
A	B	C	D	E	F	G
Tahun	Komoditas	Produksi	Perkebunan	Atribut	Pertambahan p/l	Pertambahan %
2013	Biji Sawit	3554.3	Perkebunan Besar	Produksi		
2014	Biji Sawit	3814.6	Perkebunan Besar	Produksi	0.073236523619	7.323523619
2015	Biji Sawit	3971.6	Perkebunan Besar	Produksi	0.04115765742	4.115765742
2016	Biji Sawit	3982.5	Perkebunan Besar	Produksi	0.00274448585	0.274448585
2017	Biji Sawit	4349.8	Perkebunan Besar	Produksi	0.09222849969	9.222849969
2018	Biji Sawit	5517.3	Perkebunan Besar	Produksi	0.268403145	26.8403145
2019	Biji Sawit	6438.9	Perkebunan Besar	Produksi	0.1670382252	16.70382252
2020	Biji Sawit	6397.2	Perkebunan Besar	Produksi	-0.006478261473	-0.6478261473
2021	Biji Sawit	6101	Perkebunan Besar	Produksi	-0.04630150691	-4.630150691
2022	Biji Sawit	6101.8	Perkebunan Besar	Produksi	0.0001311260449	0.01311260449
2023	Biji Sawit	6136.7	Perkebunan Besar	Produksi	0.005719623718	0.5719623718
2013	Cengkeh	107.65	Perkebunan Rakyat	Produksi		0
2014	Cengkeh	120.2	Perkebunan Rakyat	Produksi	0.1165815142	11.65815142
2015	Cengkeh	137.7	Perkebunan Rakyat	Produksi	0.1455906822	14.55906822
2016	Cengkeh	137.6	Perkebunan Rakyat	Produksi	-0.0007262164125	-0.07262164125
2017	Cengkeh	111.3	Perkebunan Rakyat	Produksi	-0.1911337209	-19.11337209
2018	Cengkeh	129.1	Perkebunan Rakyat	Produksi	0.1599281222	15.99281222
2019	Cengkeh	139	Perkebunan Rakyat	Produksi	0.07668474051	7.668474051
2020	Cengkeh	139.1	Perkebunan Rakyat	Produksi	0.0007194244604	0.07194244604
2021	Cengkeh	135.7	Perkebunan Rakyat	Produksi	-0.02444284687	-2.444284687
2022	Cengkeh	136	Perkebunan Rakyat	Produksi	0.002210759027	0.2210759027
2023	Cengkeh	134.1	Perkebunan Rakyat	Produksi	-0.01397	

Gambar 3.15. Data yang telah melalui tahap transformasi data.

Tahapan selanjutnya adalah merancang layout dan komponen visual yang akan digunakan sebelum tahap menyusun visualisasi pada dashboard. Rancangan komponen visual yang akan digunakan diantaranya adalah,

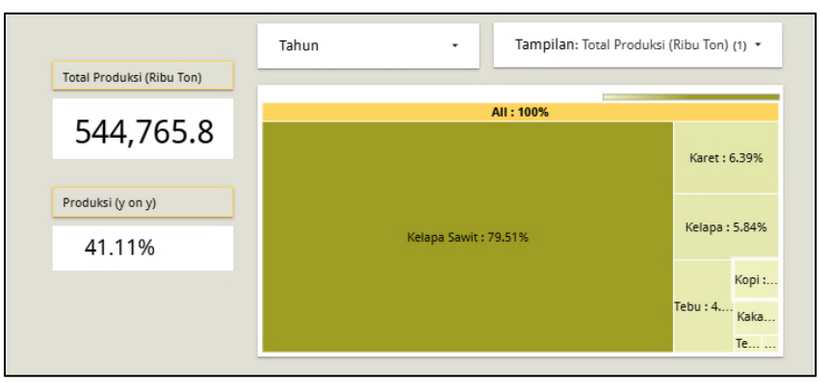
1. Scorecard untuk menampilkan total produksi nasional, komoditas, dan jumlah produksi tertinggi pada provinsi.
2. Peta yang digunakan untuk menampilkan sebaran komoditas di Indonesia dengan menggunakan gradasi warna untuk menunjukan provinsi yang memiliki tingkat produksi yang tinggi hingga provinsi dengan tingkat produksi rendah.
3. Grafik batang digunakan untuk menampilkan total produksi dan peningkatan luas area lahan per tahun.
4. Menggunakan tabel yang memiliki grafik batang yang dapat menonjolkan komoditas dengan pertumbuhan yang tinggi.
5. Grafik garis untuk menunjukan tran produksi tahunan tiap komoditas.
6. Kontrol untuk membuat dashboard interaktif sehingga calon pengguna dashboard dapat memfokuskan pada bagan yang ingin dilihat. Kontrol filter yang dipilih adalah kolom tahun, provinsi, komoditas, dan atribut. Kontrol yang digunakan berupa kontrol *Drop-down.*

Komponen-komponen yang akan digunakan dalam dashboard disusun dalam bentuk *layout* atau *wireframe* seperti gambar 3.16 dibawah ini.



Gambar 3.16. Layout komponen visual dashboard.

Gambar *layout* komponen visual dashboard yang ditampilkan merupakan rancangan awal sebelum mendapat saran dan masukan dari mentor.



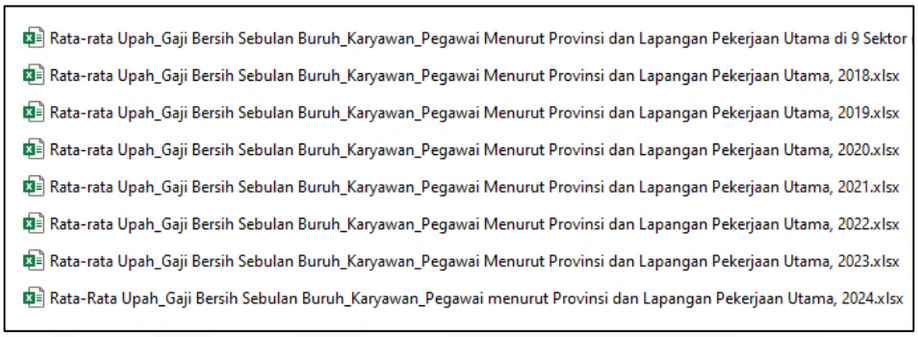
Gambar 3.17. Contoh salah satu komponen dashboard visualisasi data perkebunan.

Hasil akhir dashboard tidak berbeda terlalu jauh dengan rancangan. Salah satu perubahan yang ditambahkan ke dalam visualisasi dashboard adalah penambahan grafik *treemaps* untuk menunjukan hierarki kategori berdasarkan ukuran.

3.2.3 Membuat Dashboard Upah dari Data Lembaga Pemerintah Badan Pusat Statistik dan Kementrian Ketenagakerjaan

Pada minggu ke-5 sejak menjalankan program magang di bulan Januari, mentor memberikan tugas untuk membuat dashboard dengan data dan topik yang bisa dipilih sendiri. Topik yang dipilih tetap ditujukan untuk memberikan manfaat kepada masyarakat umum. Setelah melakukan pendalaman pada variasi data yang tersedia dan berdiskusi dengan mentor, topik dashboard yang dipilih berkaitan dengan ketenagakerjaan. Dashboard akan berisi data tentang upah pekerja formal. Pekerja formal merupakan pegawai, karyawan, atau buruh serta pekerja berstatus berusaha yang dibantu buruh tetap. Tujuan dari pembuatan dashboard ini adalah untuk menyediakan informasi tentang gaji bersih pekerja formal (pegawai/karyawan/buruh) di Indonesia serta membantu pengambilan kebijakan terkait kesejahteraan tenaga kerja formal. Analisis yang dihasilkan akan menunjukkan wilayah dan sektor pekerjaan yang memerlukan perhatian kebijakan atau investasi.

Langkah pengolahan data setelah memperoleh topik yang dipilih adalah mengumpulkan data yang akan digunakan. Data yang dicari memiliki kata kunci “pekerja formal”, “buruh”, “karyawan”, “pegawai”, dan “upah”. Selain itu, Data UMP (Upah Minimum Provinsi) juga diperlukan untuk membandingkan upah atau gaji bersih yang diterima oleh pekerja formal. Data yang dibutuhkan bisa dicari melalui website lembaga statistik di Indonesia atau berasal dari instansi yang berkutat di bidang ketenagakerjaan. Setelah dipilah, data yang dikumpulkan disimpan dalam satu file berdasarkan variabel pembeda untuk menghindari kesalahan dalam memasukkan atau impor data.



Gambar 3.18 Contoh pengumpulan data mentah variabel 1.

Dari data yang dikumpulkan, data mentah yang tidak diperlukan karena tidak mengandung kata kunci atau tidak termasuk dalam kategori pekerja formal misalnya muncul kata kunci pekerja tetapi pekerja yang dimaksud pada data adalah pekerja bebas akan diarsipkan untuk digunakan pada pembuatan dashboard selanjutnya. Data mentah yang digunakan merupakan data hasil survei yang dilakukan 2 kali dalam setahun sehingga pada 1 tahun terdapat 2 nilai rata-rata upah atau gaji bersih berdasarkan provinsi atau sektor.

Rata-rata Upah/Gaji Bersih Sebulan (rupiah) Buruh/Karyawan/Pegawai Menurut Provinsi dan Jenis Pekerjaan Utama, 2018

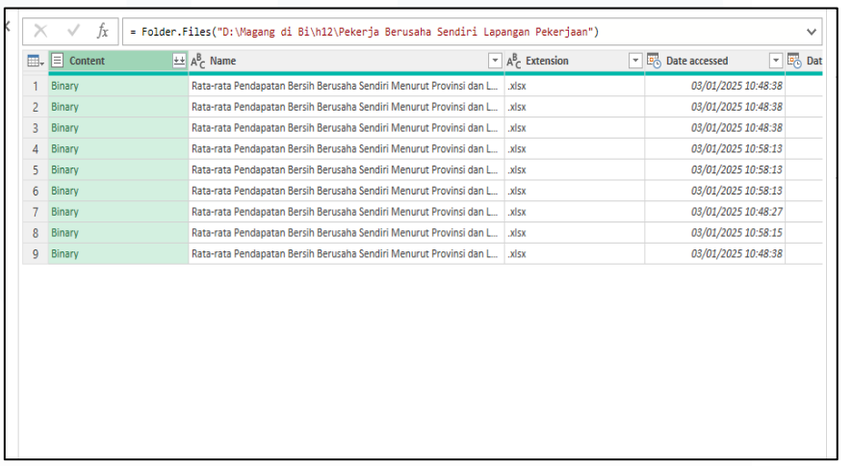
Perkotaan+Perdesaan																				Laki-laki+Pere
	Februari 2018										Agustus 2018									
	Jenis Pekerjaan Utama 1)										Jenis Pekerjaan Utama 1)									
Provinsi	0/1	2	3	4	5	6	7/8/9	X/00	Jumlah		0/1	2	3	4	5	6	7/8/9	X/		
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)		(11)	(12)	(13)	(14)	(15)	(16)	(17)	(18		
Aceh	2751582	5691997	2982391	1617806	958730,4	1823148	1775609	3261047	2312847		2523800	3709854	2591376	1818198	1421811	1820065	1888835	3		
Sumatera Utara	2698222	5397535	2529331	1918761	1187034	1941266	1907314	3324699	2202517		2822337	4430650	2813935	2058843	1326077	1896319	2094584	3		
Sumatera Barat	3043302	6111124	2769726	1692126	1818481	1699894	1947243	3232997	2465428		3285705	5711596	2618926	2033959	1590331	1941202	2132077	3		
Riau	2651982	3426841	2725537	2041099	1321391	1742671	2282361	3647033	2358662		2953038	5123949	2953023	2052462	1678782	1988205	2478210	3		
Jambi	2797157	3704969	2571371	1997132	1151582	1281025	1987324	3185631	2081612		2756634	3722828	2684668	1999184	1617610	1580975	2147156	3		
Sumatera Selatan	2880496	6490955	2353077	1866139	1329258	1169891	2115958	2871196	2123387		2651767	4940900	2940236	2086412	1302955	1433567	2290046	3		
Bengkulu	2766223	4641859	3126881	2344628	1149503	1779876	1810849	3352715	2449841		2994749	5489033	3145262	2171864	1594244	1683814	1993187	2		
Lampung	3108819	4326612	2756305	1931074	1256466	1555024	1896776	3412113	2287798		2918390	4379870	2624297	1926593	1281704	1708000	2004687	3		
Kepulauan Bangka Be	3225606	5334312	2785042	1926840	1570712	2097571	2463082	2915483	2521591		3590486	7054240	3021482	2056546	1669138	2249380	2455033	3		
Kepulauan Riau	4412438	8008264	3478535	3408205	2553186	2408438	3208091	4365358	3604388		5162231	8163746	4460148	3744772	2812679	2232187	3942528	4		
DKI Jakarta	5837562	11195140	5337982	4780463	2301205	3200000	3086325	3451069	4156334		6393021	9408896	5166832	4609117	2679442	2707727	3304611	3		
Jawa Barat	3887092	8674637	3932297	2531456	1922742	1459177	2489295	2616409	2978524		4207409	8165955	4173033	2775482	2102778	1654216	2759677	3		
Jawa Tengah	2759056	5872562	2592385	1770804	1300530	1678237	1631814	3279674	1965156		3188073	5705761	2746208	1888530	1468996	1752884	1770416	2		

Gambar 3.19 Contoh tampilan data mentah.

Tahap yang bisa dilakukan selanjutnya adalah *Exploratory Data Analysis (EDA)* atau eksplorasi data. Eksplorasi data dilakukan untuk memahami karakteristik data dan menjadi proses pemeriksaan awal. Tahapan Eksplorasi data yang dilakukan yaitu,

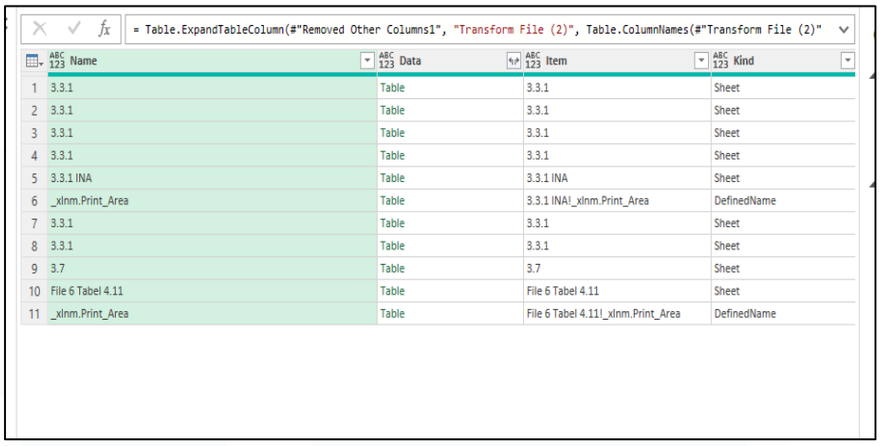
1. Menggabungkan set data yang akan digunakan untuk melihat struktur tabel data. Data yang digunakan memiliki format .xlsx atau dalam bentuk Microsoft Excel sehingga alat pengolahan data yang

digunakan adalah Power Query yang terdapat di Microsoft Excel untuk mempermudah pengolahan data. Power Query di Excel dapat digunakan untuk proses ETL atau *Extract, Transform,* dan *Load.* Cara menggabungkan banyak file data excel dengan menekan tombol perintah *GET DATA* di tab DATA lalu pilih tampilan menu *FROM FILE* dan *FROM FOLDER* yang menunjukan untuk mengambil data pada satu folder yang dituju. Kemudian pilih menu *TRANSFORM & LOAD TO* untuk melanjutkan proses penggabungan di Power Query.



Gambar 3.20 Rangkuman data yang telah load di power query.

2. Di Power Query, nama set data yang digunakan akan muncul bersama dengan karakteristik dari data tersebut. Setelah tampilan yang muncul pada Power Query sama dengan gambar di atas makan tekan tombol yang berada di pojok kanan *header column* content untuk memperluas kolom data untuk aksi eksplorasi selanjutnya.

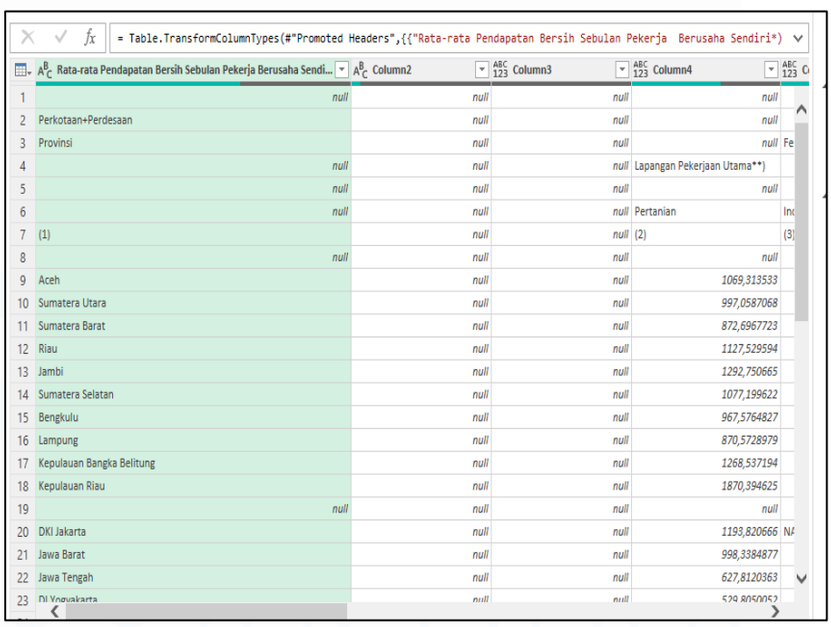


= Table.ExpandTableColumn(#"Removed Other Columns1", "Transform File (2)", Table.ColumnNames(#"Transform File (2)"

	Name	Data	Item	Kind
1	3.3.1	Table	3.3.1	Sheet
2	3.3.1	Table	3.3.1	Sheet
3	3.3.1	Table	3.3.1	Sheet
4	3.3.1	Table	3.3.1	Sheet
5	3.3.1 INA	Table	3.3.1 INA	Sheet
6	_xlnm.Print_Area	Table	3.3.1 INA!_xlnm.Print_Area	DefinedName
7	3.3.1	Table	3.3.1	Sheet
8	3.3.1	Table	3.3.1	Sheet
9	3.7	Table	3.7	Sheet
10	File 6 Tabel 4.11	Table	File 6 Tabel 4.11	Sheet
11	_xlnm.Print_Area	Table	File 6 Tabel 4.11!_xlnm.Print_Area	DefinedName

Gambar 3.21 Perluasan kolom tabel.

3. Setelah data berhasil digabung, maka eksplorasi dilanjutkan dengan melihat struktur data yang sudah terupload di Power Query. Pada gambar di tahap sebelumnya, pilih salah satu baris yang berisi teks *table* untuk membuka tabel set data yang akan digunakan. Pengamatan sekilas menunjukan jika nama kolom masih menggunakan nama format, terdapat kolom yang berisikan nama provinsi, terdapat beberapa kolom yang berisikan angka, dan terdapat baris yang berisi kategori lapangan kerja. Di data yang digunakan juga memiliki banyak data *null.*

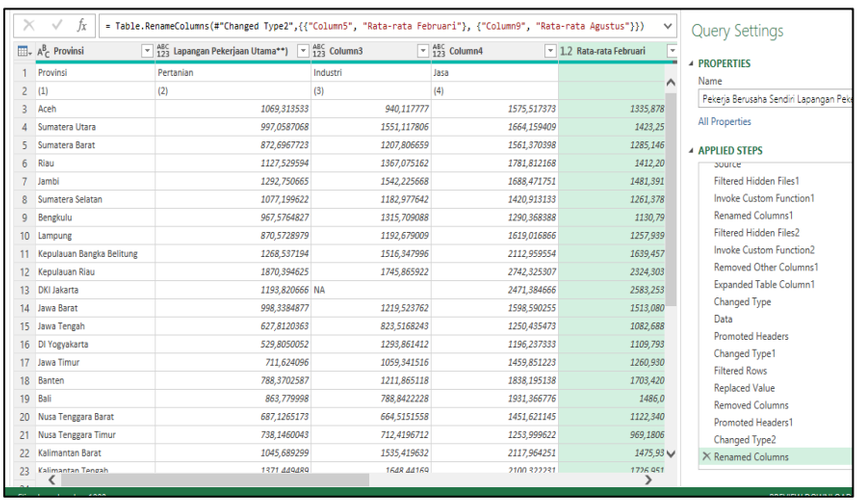


= Table.TransformColumnTypes(#"Promoted Headers",{{"Rata-rata Pendapatan Bersih Sebulan Pekerja Berusaha Sendiri*)

	Rata-rata Pendapatan Bersih Sebulan Pekerja Berusaha Sendi...	Column2	Column3	Column4	
1	null	null	null	null	
2	Perkotaan+Perdesaan	null	null	null	
3	Provinsi	null	null	null	Fe
4	null	null	null	Lapangan Pekerjaan Utama**)	
5	null	null	null	null	
6	null	null	null	Pertanian	In
7	(1)	null	null	(2)	(3
8	null	null	null	null	
9	Aceh	null	null	1069,313533	
10	Sumatera Utara	null	null	997,0587068	
11	Sumatera Barat	null	null	872,6967723	
12	Riau	null	null	1127,529594	
13	Jambi	null	null	1292,750665	
14	Sumatera Selatan	null	null	1077,199622	
15	Bengkulu	null	null	967,5764827	
16	Lampung	null	null	870,5728979	
17	Kepulauan Bangka Belitung	null	null	1268,537194	
18	Kepulauan Riau	null	null	1870,394625	
19	null	null	null	null	
20	DKI Jakarta	null	null	1193,820666	NA
21	Jawa Barat	null	null	998,3384877	
22	Jawa Tengah	null	null	627,8120363	
23	DI Yogyakarta	null	null	529,8050952	

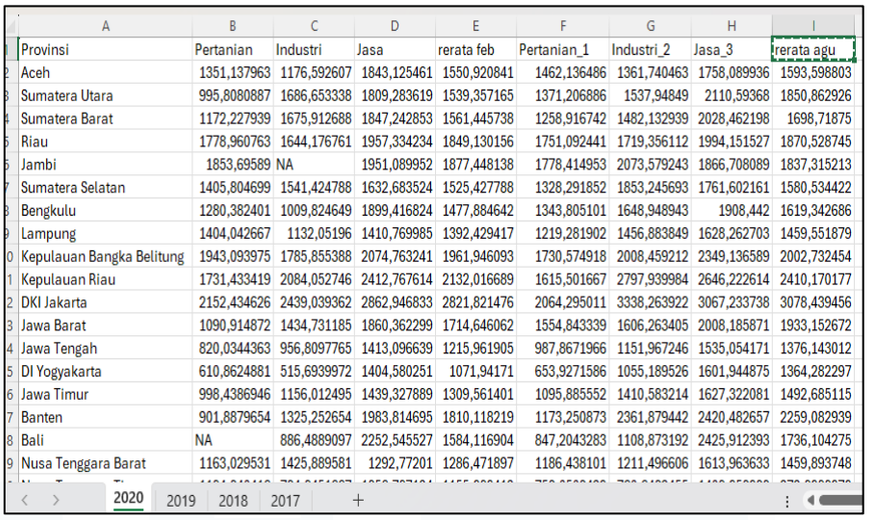
Gambar 3.22 Gambar struktur tabel salah satu dataset.

4. Tahap eksplorasi selanjutnya adalah mengidentifikasi kolom kosong atau *missing values.* Di tabel data yang digunakan muncul banyak kolom kosong atau *null*. Melihat kerelevanan data *null* bisa dilakukan dengan menggunakan filter yang muncul ketika menekan kolom dengan klik kanan lalu pilih *SELECT ALL* dan *UNCHECK* pilihan null sehingga tampilan akan berubah untuk menutup baris yang berisi data *null.* Gunakan kolom yang berisi data untuk menghindari data yang satu baris dengan null terfilter. Hapus kolom yang tidak memiliki data menggunakan menu *USE FIRST ROW AS HEADER* untuk membuat data baris pertama menjadi nama kolom. Setelah tidak ada lagi data yang hilang, *load* hasil tabel ke *Sheet* dan lakukan cara yang sama untuk tabel lainnya.



	Provinsi	Lapangan Pekerjaan Utama**)	Column3	Column4	Rata-rata Februari
1	Provinsi	Pertanian	Industri	Jasa	
2	(1)	(2)	(3)	(4)	
3	Aceh	1069,313533	940,117777	1575,517373	1335,878
4	Sumatera Utara	997,0587068	1551,117806	1664,159409	1423,25
5	Sumatera Barat	872,6967723	1207,806659	1561,370398	1285,146
6	Riau	1127,529594	1367,075162	1781,812168	1412,20
7	Jambi	1292,750665	1542,225668	1688,471751	1481,391
8	Sumatera Selatan	1077,199622	1182,977642	1420,913133	1261,378
9	Bengkulu	967,5764827	1315,709088	1290,368388	1130,79
10	Lampung	870,5728979	1192,679009	1619,016866	1257,939
11	Kepulauan Bangka Belitung	1268,537194	1516,347996	2112,959554	1639,457
12	Kepulauan Riau	1870,394625	1745,865922	2742,325307	2324,303
13	DKI Jakarta	1193,820666	NA	2471,384666	2583,253
14	Jawa Barat	998,3384877	1219,523762	1598,590255	1513,080
15	Jawa Tengah	627,8120363	823,5168243	1250,435473	1082,688
16	DI Yogyakarta	529,8050052	1293,861412	1196,237333	1109,793
17	Jawa Timur	711,624096	1059,341516	1459,851223	1260,930
18	Banten	788,3702587	1211,865118	1838,195138	1703,420
19	Bali	863,779998	788,8422228	1931,366776	1486,0
20	Nusa Tenggara Barat	687,1265173	664,5151558	1451,621145	1122,340
21	Nusa Tenggara Timur	738,1460043	712,4196712	1253,999622	969,1806
22	Kalimantan Barat	1045,689299	1535,419632	2117,964251	1475,93
23	Kalimantan Tengah	1371,449489	1648,44169	2100,322231	1726,951

Gambar 3.23 Struktur tabel data setelah identifikasi missing values.

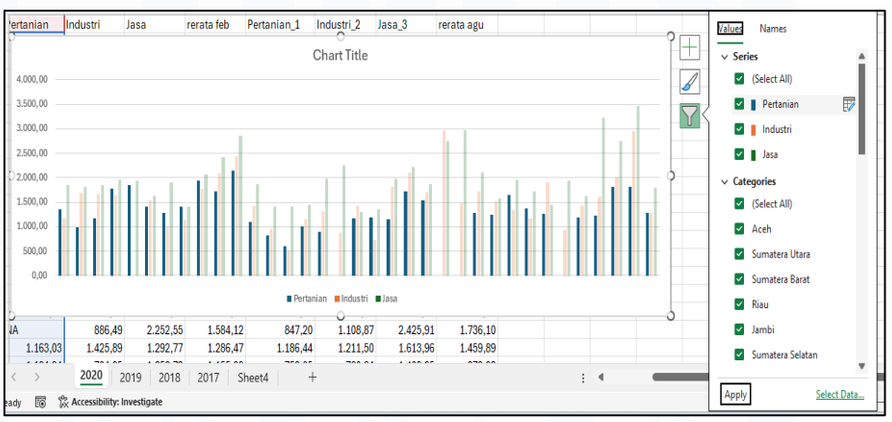


	A	B	C	D	E	F	G	H	I
1	Provinsi	Pertanian	Industri	Jasa	rerata feb	Pertanian_1	Industri_2	Jasa_3	rerata agu
2	Aceh	1351,137963	1176,592607	1843,125461	1550,920841	1462,136486	1361,740463	1758,089936	1593,598803
3	Sumatera Utara	995,8080887	1686,653338	1809,283619	1539,357165	1371,206886	1537,94849	2110,59368	1850,862926
4	Sumatera Barat	1172,227939	1675,912688	1847,242853	1561,445738	1258,916742	1482,132939	2028,462198	1698,71875
5	Riau	1778,960763	1644,176761	1957,334234	1849,130156	1751,092441	1719,356112	1994,151527	1870,528745
6	Jambi	1853,69589	NA	1951,089952	1877,448138	1778,414953	2073,579243	1866,708089	1837,315213
7	Sumatera Selatan	1405,804699	1541,424788	1632,683524	1525,427788	1328,291852	1853,245693	1761,602161	1580,534422
8	Bengkulu	1280,382401	1009,824649	1899,416824	1477,884642	1343,805101	1648,948943	1908,442	1619,342686
9	Lampung	1404,042667	1132,05196	1410,769985	1392,429417	1219,281902	1456,883849	1628,262703	1459,551879
10	Kepulauan Bangka Belitung	1943,093975	1785,855388	2074,763241	1961,946093	1730,574918	2008,459212	2349,136589	2002,732454
11	Kepulauan Riau	1731,433419	2084,052746	2412,767614	2132,016689	1615,501667	2797,939984	2646,222614	2410,170177
12	DKI Jakarta	2152,434626	2439,039362	2862,946833	2821,821476	2064,295011	3338,263922	3067,233738	3078,439456
13	Jawa Barat	1090,914872	1434,731185	1860,362299	1714,646062	1554,843339	1606,263405	2008,185871	1933,152672
14	Jawa Tengah	820,0344363	956,8097765	1413,096639	1215,961905	987,8671966	1151,967246	1535,054171	1376,143012
15	DI Yogyakarta	610,8624881	515,6939972	1404,580251	1071,94171	653,9271586	1055,189526	1601,944875	1364,282297
16	Jawa Timur	998,4386946	1156,012495	1439,327889	1309,561401	1095,885552	1410,583214	1627,322081	1492,685115
17	Banten	901,8879654	1325,252654	1983,814695	1810,118219	1173,250873	2361,879442	2420,482657	2259,082939
18	Bali	NA	886,4889097	2252,545527	1584,116904	847,2043283	1108,873192	2425,912393	1736,104275
19	Nusa Tenggara Barat	1163,029531	1425,889581	1292,77201	1286,471897	1186,438101	1211,496606	1613,963633	1459,893748

2020 2019 2018 2017

Gambar 3.24 Data yang telah dibersihkan.

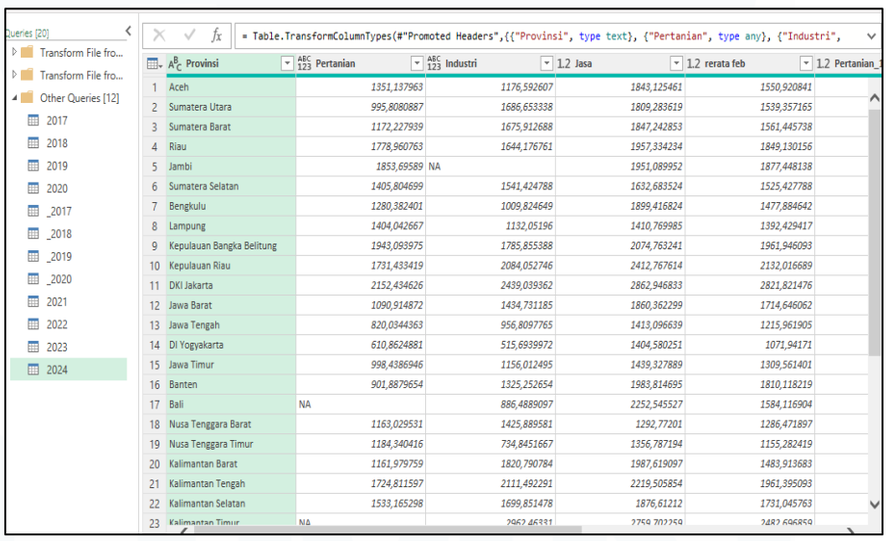
5. Tahap eksplorasi data selanjutnya adalah membuat analisis permulaan atau sementara untuk melihat distribusi dan frekuensi data yang digunakan. Dari data yang ditampilkan terdapat 3 kategori jenis pekerjaan, data yang digunakan tersedia dari 2017-2024, dan rata-rata upah tiap jenis pekerjaan di Indonesia
6. Pada tahap eksplorasi, visualisasi dengan membuat grafik awal seperti menggunakan grafik batang untuk menunjukan upah atau gaji bersih tiap provinsi dan menyoroti perbandingan visualisasi antar provinsi. Selain itu, membuat grafik awal dapat membantu untuk melihat apakah distribusi data normal atau terdapat data yang berbeda dari banyak data (*outlier).*



Gambar 3.25 Grafik awal tahap eksplorasi data

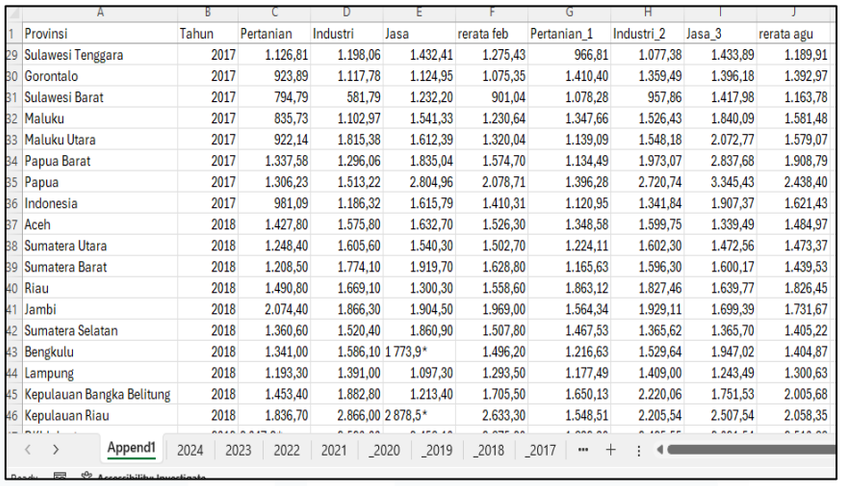
Tahapan persiapan data selanjutnya adalah transformasi data. Transformasi data dilakukan agar sumber data yang digunakan untuk dashboard Looker Studio adalah sumber data terpadu atau berisi data upah dalam jangka waktu 8 tahun. Tahapan transformasi data yang dilakukan adalah menggabungkan data yang berada di *Sheet* data Excel yang berbeda-beda menjadi satu *Sheet* data. Berikut uraian dari tahapan transformasi ,

1. Mengambil data yang sudah dibersihkan di dalam file Excel yang dihasilkan pada tahapan eksplorasi data. Pilih tab DATA dan tekan tombol perintah *GET DATA* lalu pilih FROM EXCEL WORKBOOK. Pilih nama file dan SELECT ALL semua sheet yang akan digunakan dan pilih *TRANSFORM DATA.*



Gambar 3.26 Menggabungkan data dengan power query.

2. Selanjutnya pilih menu *APPEND QUERIES AS NEW* untuk menggabungkan data dari tahun 2017 hingga 2024. Setiap tabel data sudah disesuaikan untuk memiliki jumlah kolom dan urutan yang sama. Tambahkan semua tabel yang akan digabungkan ke dalam kolom menu *tables to append* lalu pilih *Load* untuk mengunggahnya di *Sheet Excel.*



	A	B	C	D	E	F	G	H	I	J
1	Provinsi	Tahun	Pertanian	Industri	Jasa	rerata feb	Pertanian_1	Industri_2	Jasa_3	rerata agu
29	Sulawesi Tenggara	2017	1.126,81	1.198,06	1.432,41	1.275,43	966,81	1.077,38	1.433,89	1.189,91
30	Gorontalo	2017	923,89	1.117,78	1.124,95	1.075,35	1.410,40	1.359,49	1.396,18	1.392,97
31	Sulawesi Barat	2017	794,79	581,79	1.232,20	901,04	1.078,28	957,86	1.417,98	1.163,78
32	Maluku	2017	835,73	1.102,97	1.541,33	1.230,64	1.347,66	1.526,43	1.840,09	1.581,48
33	Maluku Utara	2017	922,14	1.815,38	1.612,39	1.320,04	1.139,09	1.548,18	2.072,77	1.579,07
34	Papua Barat	2017	1.337,58	1.296,06	1.835,04	1.574,70	1.134,49	1.973,07	2.837,68	1.908,79
35	Papua	2017	1.306,23	1.513,22	2.804,96	2.078,71	1.396,28	2.720,74	3.345,43	2.438,40
36	Indonesia	2017	981,09	1.186,32	1.615,79	1.410,31	1.120,95	1.341,84	1.907,37	1.621,43
37	Aceh	2018	1.427,80	1.575,80	1.632,70	1.526,30	1.348,58	1.599,75	1.339,49	1.484,97
38	Sumatera Utara	2018	1.248,40	1.605,60	1.540,30	1.502,70	1.224,11	1.602,30	1.472,56	1.473,37
39	Sumatera Barat	2018	1.208,50	1.774,10	1.919,70	1.628,80	1.165,63	1.596,30	1.600,17	1.439,53
40	Riau	2018	1.490,80	1.669,10	1.300,30	1.558,60	1.863,12	1.827,46	1.639,77	1.826,45
41	Jambi	2018	2.074,40	1.866,30	1.904,50	1.969,00	1.564,34	1.929,11	1.699,39	1.731,67
42	Sumatera Selatan	2018	1.360,60	1.520,40	1.860,90	1.507,80	1.467,53	1.365,62	1.365,70	1.405,22
43	Bengkulu	2018	1.341,00	1.586,10	1 773,9*	1.496,20	1.216,63	1.529,64	1.947,02	1.404,87
44	Lampung	2018	1.193,30	1.391,00	1.097,30	1.293,50	1.177,49	1.409,00	1.243,49	1.300,63
45	Kepulauan Bangka Belitung	2018	1.453,40	1.882,80	1.213,40	1.705,50	1.650,13	2.220,06	1.751,53	2.005,68
46	Kepulauan Riau	2018	1.836,70	2.866,00	2 878,5*	2.633,30	1.548,51	2.205,54	2.507,54	2.058,35

Append1 | 2024 | 2023 | 2022 | 2021 | _2020 | _2019 | _2018 | _2017

Gambar 3.27 Hasil tabel data yang telah digabungkan.

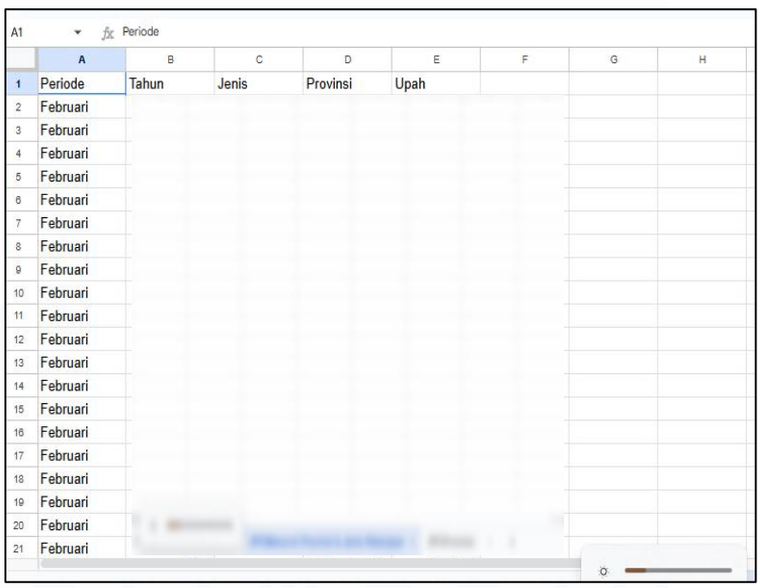
3. Selanjutnya, data yang akan digunakan untuk Looker Studio membutuhkan format tabel yaitu *long table* atau setiap baris mewakili *record* atau observasi. Data yang sedang diolah memiliki format tabel crosstab seperti nama kategori atau tahun menjadi nama/ judul kolom sehingga data yang dimiliki sekarang perlu diubah menggunakan transpose. Transpose akan mengubah

4. Orientasi data dari bentuk horizontal ke bentuk vertikal ataupun sebaliknya. Selain menggunakan transpose, mengubah susunan tabel dapat dilakukan dengan menu *UNPIVOT DATA di* Power Query.

	Provinsi	Tahun	Attribute	Value
1	Aceh	2017	Pertanian	1069,313533
2	Aceh	2017	Industri	940,117777
3	Aceh	2017	Jasa	1575,517373
4	Aceh	2017	rerata feb	1335,878202
5	Aceh	2017	Pertanian_1	1292,241251
6	Aceh	2017	Industri_2	1247,6074
7	Aceh	2017	Jasa_3	1950,604681
8	Aceh	2017	rerata agu	1630,625397
9	Sumatera Utara	2017	Pertanian	997,0587068
10	Sumatera Utara	2017	Industri	1551,117806
11	Sumatera Utara	2017	Jasa	1664,159409
12	Sumatera Utara	2017	rerata feb	1423,25875
13	Sumatera Utara	2017	Pertanian_1	1117,706408
14	Sumatera Utara	2017	Industri_2	1281,368425
15	Sumatera Utara	2017	Jasa_3	1795,158539
16	Sumatera Utara	2017	rerata agu	1499,885875
17	Sumatera Barat	2017	Pertanian	872,6967723
18	Sumatera Barat	2017	Industri	1207,806659
19	Sumatera Barat	2017	Jasa	1561,370398
20	Sumatera Barat	2017	rerata feb	1285,146655
21	Sumatera Barat	2017	Pertanian_1	991,4157474
22	Sumatera Barat	2017	Industri_2	1315,552906
23	Sumatera Barat	2017	Jasa_3	1795,041047

Gambar 3.28 Tabel data yang telah diubah ke bentuk long table.

5. Tahap terakhir dari transformasi data adalah melakukan ekspor data Excel ke Google Sheet karena format sumber data yang digunakan untuk dashboard adalah Google Sheet.



Gambar 3.29 Impor data ke Google Sheet.

Transformasi data diperlukan karena susunan tabel mempengaruhi kemampuan untuk memfilter, membandingkan, dan memvisualisasikan

data secara dinamis saat menyusun visualisasi di dashboard. Setelah memperoleh data yang telah melalui proses eksplorasi data dan proses transformasi data maka tahap selanjut adalah merancang *layout* visualisasi pada dashboard. Rancangan dashboard berisi kolom atau variabel data yang akan digunakan, komponen visual, serta tujuan dari penggunaan komponen visual tersebut. Komponen variabel atau kolom utama untuk pembuatan dashboard upah adalah,

1. Kolom Rata-rata Upah yang berisi angka rata-rata upah berdasarkan faktor.
2. Kolom UMP yang berisi angka upah minimum tiap provinsi
3. Jangka waktu atau tahun nilai upah yang tercatat.
4. Kolom nama provinsi.
5. Kolom atribut pilihan yang berisi nama kategori "Jenis Pekerjaan, dan "Sektor Lapangan Usaha".
6. Kolom atribut pilihan yang berisi uraian jenis-jenis pekerjaan dan macam-macam sektor lapangan usaha.

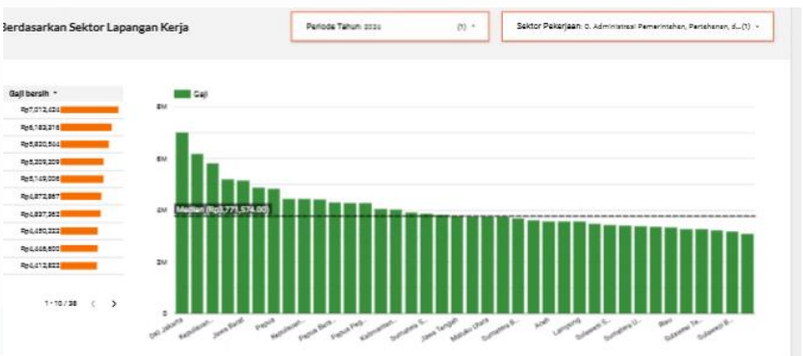
Pemilihan variabel tersebut bertujuan untuk menyoroti ketimpangan yang berasal dari perbedaan upah antara sektor pekerjaan dan lapangan kerja berdasarkan provinsi. Komponen visual yang direncanakan berdasarkan variabel atau kolom utama yang telah ditentukan seperti,

1. Grafik Batang digunakan untuk menunjukan perbandingan gaji bersih antara provinsi. Terdapat dua grafik batang yang digunakan pada dashboard ini yaitu grafik batang yang menunjukan angka upah berdasarkan jenis pekerjaan dan grafik batang yang menunjukan angka upah berdasarkan sektor lapangan kerja. Grafik batang hanya menampilkan sebagian provinsi yang memiliki upah atau gaji bersih yang tinggi.
2. Tabel Interaktif yang digunakan memiliki fitur untuk merepresentasikan angka di dalam sel dengan visual grafik batang. Tabel interaktif dipilih untuk ditampilkan pada dashboard agar data yang telah ditampilkan dengan grafik batang yang berfokus untuk
3. menunjukan data secara visual dapat diketahui angka aslinya. Pengguna dashboard bisa memilih untuk menggunakan tabel

interaktif jika tertarik dengan angka rata-rata upah tiap provinsi. Pada tabel interaktif, kolom upah minimum provinsi digunakan sebagai pembanding upah atau gaji bersih tiap provinsi. Pengguna bisa melihat upah dengan mengurutkan berdasarkan provinsi atau gaji.

4. Kontrol Waktu dan Kategori. Kontrol pada Looker Studio berfungsi untuk melakukan filter berdasarkan kolom atau variabel yang ditetapkan. Pada pembuatan dashboard ini, variabel atau kolom yang digunakan adalah kolom waktu yang berupa tahun dan kolom uraian kategori yang berisi jenis-jenis pekerjaan dan macam-macam sektor lapangan kerja. Bentuk kontrol yang digunakan adalah kontrol *Drop-down* yang akan memberikan pilihan filter kepada pengguna dengan tampilan memanjang kebawah.

Dari komponen visual yang telah dipilih untuk digunakan, layout dashboard dibagi menjadi 2 bagian kolom atas dan bawah. Kolom atas akan berisi visualisasi data upah berdasarkan provinsi dan jenis pekerjaan sedangkan kolom bawah adalah visualisasi data upah atau gaji bersih berdasarkan provinsi dan sektor lapangan kerja.

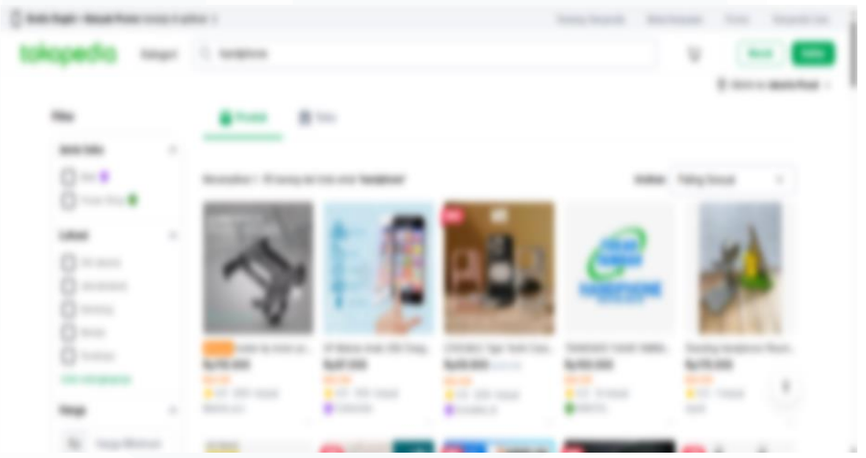


Gambar 3.30 Salah satu komponen dashboard upah.

4.2.4 Membuat Dashboard Monitoring E-Commerce dengan Web Scraping

Pada minggu ke-7 program magang di Data Indonesia, *Data Scientist* mendapatkan tugas untuk membuat percobaan dashboard monitoring e-commerce menggunakan data hasil *scraping*. *Data*

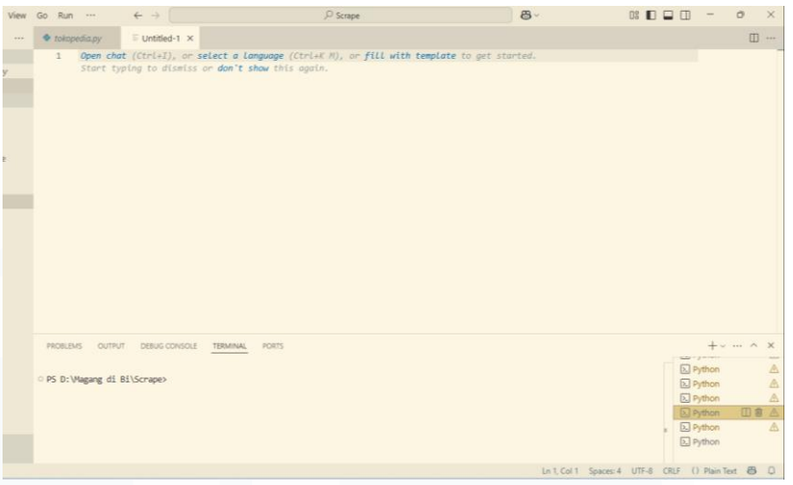
Scientist diberikan pilihan untuk memilih salah satu e-commerce atau website tempat jual barang yang akan diambil datanya. Pada proyek pembuatan dashboard monitoring ini, e-commerce yang dipilih adalah e-commerce T***p****. Proyek ini dikerjakan bersama dengan *Data Scientist* lain di Data Indonesia untuk mempercepat waktu pengerjaan proyek. Mentor menentukan salah satu produk yang dijual yaitu handphone dan tablet sebagai produk yang datanya akan disajikan dalam bentuk dashboard.



Gambar 3.31 Tampilan E-commerce yang dipilih

Salah satu cara untuk mengambil data di website adalah menggunakan metode *Scraping data. Scraping data* adalah metode pengambilan data dengan cara mengekstrak informasi dari situs web yang dilakukan secara otomatis. Proses otomasi yang dimaksudkan adalah dengan membuat program dengan bahasa pemrograman yang sesuai. Berdasarkan diskusi dengan mentor, bahasa pemrograman yang digunakan untuk *scraping data* di e-commerce ini adalah bahasa pemrograman python dan menggunakan IDE Visual Studio Code sebagai teks editor. Bahasa pemrograman python dipilih karena sudah memiliki modul yang mendukung untuk melakukan proses *scraping*

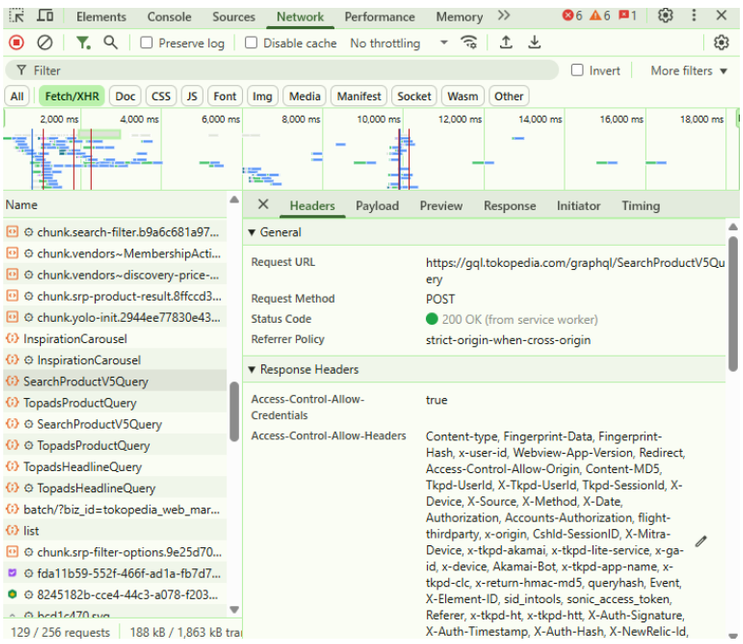
data sedangkan visual studio code dipilih karena ringan untuk digunakan perangkat atau laptop yang memiliki spesifikasi *mid-range* sehingga tidak bisa menangani proses pengolahan data dan pemrograman yang berat. IDE Visual Studio Code atau VS Code mendukung berbagai bahasa pemrograman.



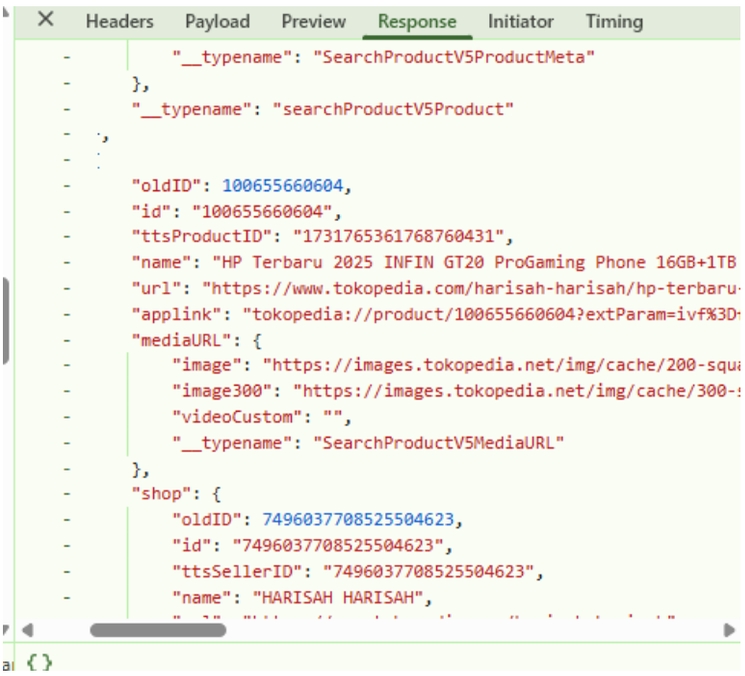
Gambar 3.32 Tampilan IDE VS Code.

Setelah menetapkan produk yang akan dicari maka selanjutnya menentukan karakteristik data produk yang juga akan diekstrak. Karakteristik nama produk, harga, toko, kota penjual, rating, serta kategori produk adalah data yang akan diambil dari produk handphone dan tablet serta menjadi kata kunci pencarian saat pembuatan *script scraping data.*

Sebelum membuat skrip pemrograman, *Data Scientist* perlu tahu cara situs yang akan di-*scrapping* mengirimkan data menggunakan fitur Inspect Element dan Network Tab halaman website pada browser. Pilih filter Fetch/XHR untuk mencari file JSON menggandung alamat endpoint atau API yang berfungsi saat pencarian produk. Melalui proses identifikasi, ditemukan situs e-commerce T***p**** menggunakan endpoint GraphQL. (https://gql.********.com/graphql/SearchProductV5Query) untuk mengirimkan dan menerima data produk. Selain melihat alamat endpoint, tab response juga berisi data produk. Temuan-temuan tersebut menjadi alasan untuk melakukan *scraping data* dengan API karena tujuan data sudah ditemukan.



Gambar 3.33 Identifikasi alamat endpoint.



Gambar 3.34 Cek tab response.

Tahap selanjutnya adalah membuat *script scraping data* menggunakan bahasa pemrograman python. *Scraping data* dengan python membutuhkan *library* yang diperlukan untuk mengimpor fungsi *library*. *Library* yang digunakan meliputi *Library* requests

untuk melakukan HTTP request, *Library* json untuk memproses respon dalam format JSON, *Library* pandas untuk mengolah data menjadi DataFrame, dan *Library* openpyxl untuk menyimpan hasil ke file Excel.

Saat ini, website sudah menerapkan keamanan untuk mendeteksi pengambilan data oleh bots berdasarkan pola sedangkan pengambilan data dengan metode *scraping* akan dikenali sebagai salah satu bots oleh website karena menghasilkan pola yang berbeda dari hasil pola manusia saat mengakses website. Jika proses *scraping data* terdeteksi sebagai bot maka perangkat yang digunakan akan diblokir aksesnya ke website e-commerce. Salah satu cara untuk menghindari blokir dari server karena permintaan dianggap sebagai bot adalah dengan menyusun header permintaan (HTTP headers) yang menyerupai permintaan dari browser sungguhan berisi User-Agent, Accept, Origin, dan Referer yang bisa didapatkan saat mencari alamat endpoint di tahap sebelumnya.

```
headers = {
    "User-Agent": (
        "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 "
        "(KHTML, like Gecko) Chrome/91.0.4472.124 Safari/537.36"
    ),
    "Accept": "application/json, text/plain, */*",
    "Content-Type": "application/json",
    "Origin": "https://www.tokopedia.com",
    "Referer": "https://www.tokopedia.com/search",
}
```

Gambar 3.35 Headers pada script.

Tahap selanjutnya adalah membuat fungsi get_params() yang menghasilkan parameter pencarian untuk mengambil data dari banyak halaman sekaligus misalnya 100 halaman pertama. Setiap parameter berisi kata kunci yaitu “handphone” dan”tablet”, nomor halaman, dan titik awal data (start) untuk digunakan dalam proses *pagination*.

```
def get_params():
    params = []
    for i in range (1,101):
        for product in cari:
            param = "device=desktop&l_name=sre&navsource=&ob=23&page={}&q={}&related=true&rows=60&safe_search=fals
            params.append(param)
    return params
```

Gambar 3.36 Fungsi parameter pencarian.

Setelah menentukan banyak halaman yang akan di ekstrak, maka tahap selanjutnya adalah membuat fungsi untuk ekstraksi data (scrape_data()). Fungsi utama scrape_data() bertugas untuk,

1. Mengirim permintaan POST ke endpoint GraphQL.
2. Mengambil data produk dari struktur JSON.
3. Mengekstrak informasi yang relevan seperti ID produk, nama produk, harga, jumlah terjual, rating, nama toko, kota asal, kategori, dan gambar.
4. Menyimpan hasil ekstraksi dalam struktur data *list of tuples.*

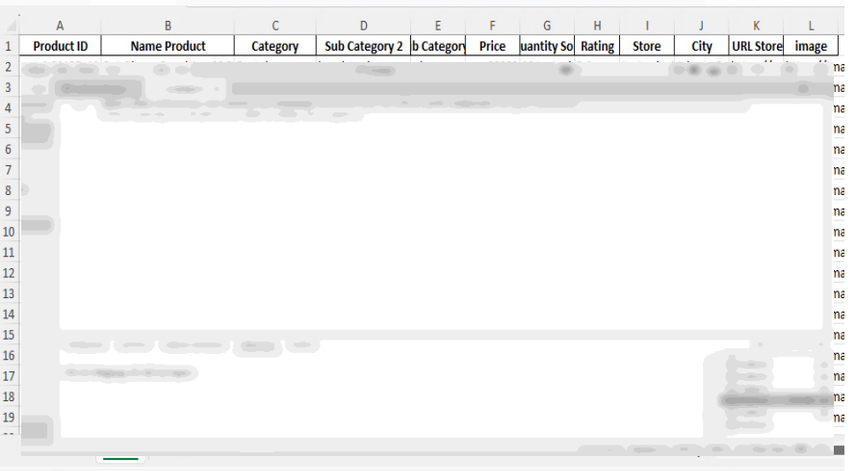
Gambar 3.37 Fungsi scrape data untuk ekstraksi.

Fungsi scrape_data() dijalankan dalam loop untuk semua parameter pencarian lalu hasilnya dikumpulkan ke dalam satu list besar yaitu list all_data dan diubah ke dalam pandas.DataFrame dan disimpan dalam file Excel menggunakan df.to_excel('data.xlsx').

```python
if __name__ == '__main__':
    params = get_params()
    all_data=[]
    for i in range(0, len(params)):
        param = params[i]
        data = scrape_data(param)
        all_data.extend(data)
```

Gambar 3.38 Fungsi pengulangan untuk pencarian setiap parameter.

Hasil data dari program *scaping data* ada pada gambar di bawah menunjukan nama produk yang tertera pada website e-commerce, kategori produk, harga, dan karakteristik lain dari kata kunci produk yang dicari sebagai nama kolo.



Gambar 3.39 Tabel data hasil running skrip.

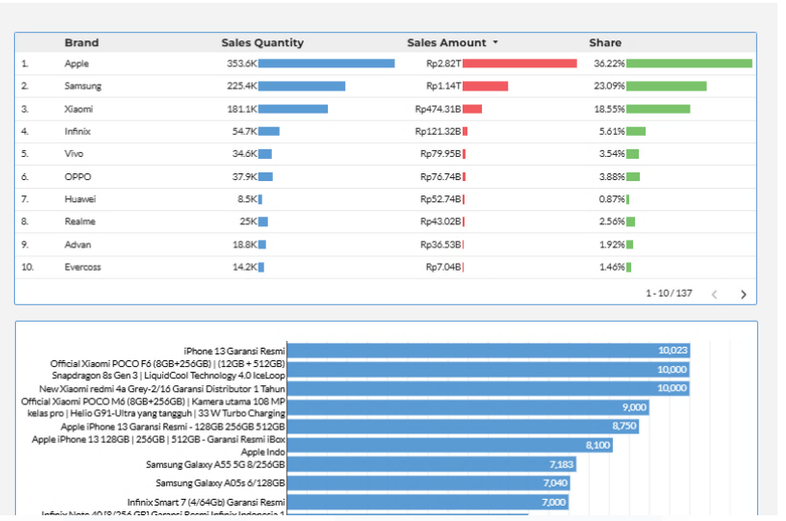
Data yang dihasil sudah memiliki format *long-table* yang sesuai dengan format Looker Studio sehingga tidak perlu mengubah susunan tabel. Tahapan eksplorasi data tetap dilakukan untuk mengidentifikasi pola dan menemukan ketidaksesuaian pada data. Data dibersihkan dengan Python dan Excel untuk menyamakan format kolom dan menangani nilai null serta format tipe data disesuaikan agar dapat terbaca di Looker Studio. Proses ini mengubah format harga dan jumlah penjualan ke tipe numerik dan penyesuaian struktur tabel agar konsisten. Data yang sudah

dipersiapkan diekspor ke Google Sheet untuk dihubungkan ke Looker Studio.

Setelah menerima data yang telah melewati proses eksplorasi data, tahap berikutnya adalah merancang tata letak visualisasi yang akan diterapkan di dashboard. Desain dashboard mencakup variabel kolom atau data untuk digunakan sebagai komponen visual serta tujuan menggunakan komponen visual ini. Visualisasi yang akan ditampilkan di dashboard monitor e-commerce adalah,

1. Kontrol atau Filter Interaktif yang berfungsi untuk mempersempit pencarian data dan menyesuaikan tampilan sesuai kebutuhan pengguna. Variabel yang
2. digunakan sebagai filter adalah kolom *Category, Sub Category, Brand, Product Name, Price*, dan *City*.
3. Bar chart horizontal yang digunakan untuk menunjukkan skala penjualan dan pangsa pasar. Variabel yang digunakan adalah *Brand, Sales Quantity, Sales Amount,* dan *Share (%)*.
4. Grafik batang yang menunjukan produk dengan penjualan tertinggi. Kolom: yang digunakan adalah *Product Name, Price, Sales Quantity,* dan *Sales Amount*.
5. Tabel yang menggunakan heatmap untuk menunjukan toko dengan jumlah penjualan handphone dan tablet terbesar. Tabel akan menampilkan daftar toko teratas berdasarkan jumlah unit handphone dan tablet yang terjual serta total nilai penjualan.
6. Peta dipilih untuk menunjukkan persebaran toko di seluruh wilayah Indonesia. Tabel digunakan bersama dengan peta untuk menunjukan total toko per kota jika salah satu kota dipilih.

Dari susunan komponen visual yang digunakan maka bentuk realisasi visualisasi data di dashboard ada pada gambar dibawah ini.



Gambar 3.40 Salah satu komponen dashboard dengan data hasil scraping.

Dashboard hasil scraping data T***p**** menyajikan sejumlah insight penting terkait performa penjualan berbagai brand dan produk elektronik di platform tersebut. Secara umum, visualisasi dibagi ke dalam beberapa segmen utama dan masing-masing memberikan gambaran mendalam terhadap aspek tertentu dari aktivitas penjualan.

1. Distribusi Penjualan berdasarkan Brand

   Pada bagian atas dashboard, ditampilkan tabel berisi sepuluh brand teratas dengan total kuantitas dan nilai penjualan tertinggi. Berdasarkan data yang ditampilkan, Apple menempati posisi pertama dengan kuantitas penjualan sebesar lebih dari 350 ribu unit dan total nilai penjualan mencapai lebih dari Rp2 triliun. Hasil ini menunjukkan tingginya minat konsumen terhadap produk Apple meskipun harganya relatif tinggi dibandingkan produk pesaing. Brand lain seperti Samsung dan Xiaomi juga memiliki kontribusi signifikan terhadap volume penjualan, mencerminkan dominasi merek-merek besar dalam pasar elektronik.

2. Performa Produk Berdasarkan Penjualan

   Bagian berikutnya menampilkan produk-produk dengan angka penjualan tertinggi. Produk iPhone 13 Garansi Resmi

menjadi yang paling laris dengan jumlah penjualan mencapai 10.000 unit dan total transaksi lebih dari Rp900 miliar. Visualisasi ini memungkinkan identifikasi produk unggulan yang mendominasi pasar serta dapat dijadikan dasar analisis tren konsumen terhadap jenis produk dan rentang harga tertentu.

3. Peringkat Toko Berdasarkan Penjualan

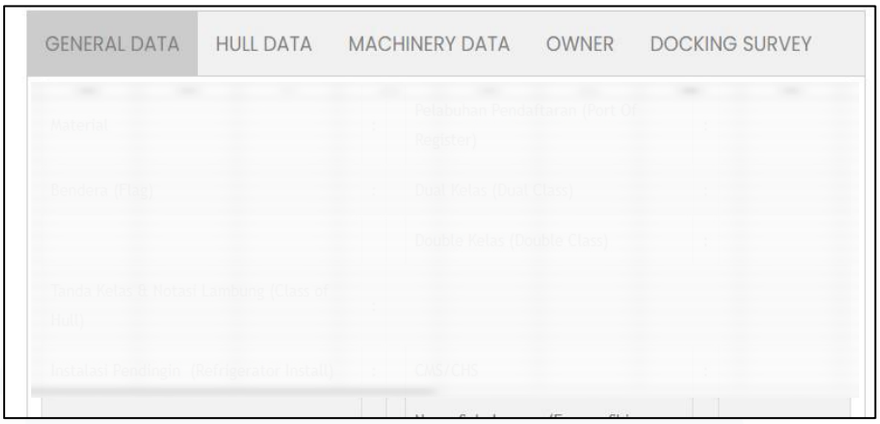
   Kolom dashboard berikutnya menampilkan toko-toko yang mencatatkan performa terbaik dalam penjualan handphone dan tablet. Toko seperti Xiaomi Official Store dan Ini Toko Budi muncul sebagai pemain kuat dengan angka penjualan tinggi. Visualisasi ini penting untuk mengidentifikasi toko-toko dominan dalam rantai distribusi serta memberikan gambaran tentang peran toko resmi dan non-resmi dalam mendistribusikan produk.

4. Peta Persebaran Toko

   Komponen peta interaktif memberikan gambaran mengenai distribusi toko-toko aktif di T***p****. Berdasarkan grafik, Kota Jakarta Pusat, Jakarta Barat, dan Jakarta Utara merupakan wilayah dengan jumlah toko tertinggi. Hasil tersebut menunjukkan bahwa aktivitas e-commerce masih sangat terpusat di wilayah urban, khususnya di wilayah Jabodetabek.

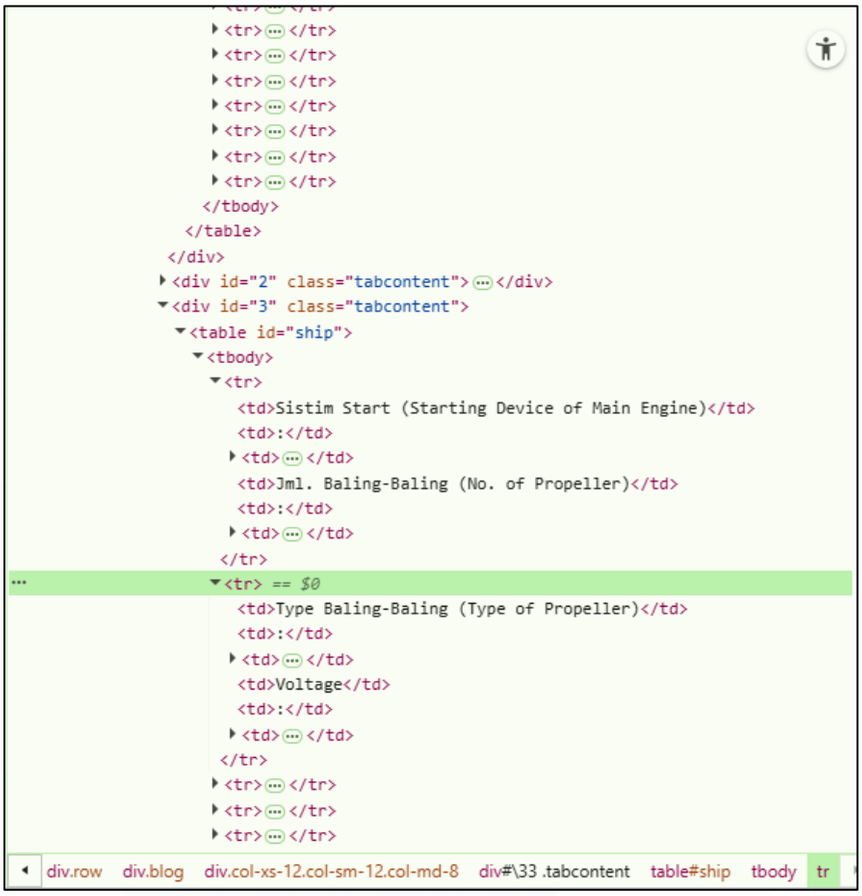
4.2.5 Web Scraping Data Kustom

Pada Minggu ke-13 program magang di Data Indonesia, *Data Scientist* ditugaskan untuk membantu mengumpulkan data transportasi K**** yang terdaftar di lembaga pemerintahan di Indonesia. Metode yang digunakan adalah *scraping data* pada situs atau website. Jika ditelusuri tiap nomor registrasi transportasi sama dengan nama alamat halaman sehingga bisa disimpulkan satu halaman hanya memuat data registrasi satu jenis transportasi. Ketika *Data Scientist* mencari dara registrasi transportasi lain maka halam website akan berubah denngan bentuk tabel yang sama di halaman yang berbeda.



Gambar 3.41 Gambar tabel pada halaman website yang akan digunakan.

Pendekatan scraping data yang akan dilakukan ditentukan berdasarkan struktur halaman web. *Data Scientist* bisa menggunakan menu Inspect dan bergeser ke tab Network untuk melihat apakah ada data yang dikirim. Jika tidak ada data yang dikirim dan tidak menemukan alamat endpoint maka cek tab Element untuk melihat apakah data berada di susunan HTML website. Struktur halaman website tidak menyediakan API publik atau format data terstruktur seperti JSON atau XML yang bisa langsung diambil dan diolah. Pendekatan scraping berbasis HTML (HyperText Markup Language) menjadi satu-satunya pilihan yang memungkinkan untuk mengekstrak informasi yang ditampilkan secara visual di halaman web.



Gambar 3.42 Struktur HTML halaman website.

Berdasarkan hasil temuan pada struktur halaman website maka proyek *scraping data* dilakukan dengan menggunakan bahasa pemrograman Python di Visual Studio Code. Pendekatan yang digunakan bersifat asynchronous dengan memanfaatkan *Library* aiohttp untuk melakukan permintaan HTTP secara paralel dan *Library*

BeautifulSoup untuk mengekstrak elemen HTML dari setiap halaman. Tahapan yang dilakukan untuk mengerjakan pengambilan data dengan volume yang besar terdiri dari:

1. Penentuan Sumber Data dan Identifikasi Pola URL. Setiap halaman yang berisi data kapal dapat diakses melalui URL dengan format https://www.***.co.id/***register-{nomor}.html. Angka {nomor} adalah angka unik yang menunjukkan halaman informasi masing-masing kapal. Rentang scraping ditetapkan mulai dari halaman nomor 1 hingga 29.000.

2. Ekstraksi data utama kapal dengan fungsi extract_data_from_url. Program mengambil informasi utama seperti nama transportasi, nomor register tramsportasi (diambil dari nomor halaman), nomor IMO, dan status registrasi transportasi. Data tersebut berada dalam tabel di elemen div dengan kelas blog-content.

Gambar 3.43 Contoh skrip fungsi extract_data_from_url.

3. Membuat fungsi untuk ekstraksi spesifikasi kapal. Fungsi extract_data_from_specifications digunakan untuk mengambil spesifikasi teknis yang mencakup panjang kapal, tipe kapal, serta data struktural lainnya. Selain itu, fungsi extract_ship_specs

4. juga mengambil informasi dari tab lainnya seperti galangan tempat pembuatan transportasi, tahun pembuatan, dan lokasi pembuatan.

```
# Function to extract data from a given URL asynchronously
async def extract_data_from_specifications(session, url, page_num):
    try:
        headers = {
            "User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWeb
        }
        async with session.get(url, headers=headers) as response:
            response.raise_for_status()
            html = await response.text()

        # Parse the HTML content using BeautifulSoup
        soup = BeautifulSoup(html, 'html.parser')

        # Find the table with ID "ship" within the div
        table = content_div.find('table', {'id': 'ship'})
        if not table:
            print(f"No table with ID 'ship' found for URL: {url}")
            return None
```

Gambar 3.44 Contoh skrip fungsi extract_data_from_specifications.

5. Mengekstrak data mesin utama (*Main Machine*) dan mesin bantu (*Auxiliary Engine*) menggunakan fungsi extract_machine_data. Hasilnya dari fungsi tersebut ditransformasikan menjadi susunan horizontal agar setiap transportasi direpresentasikan oleh satu baris data meskipun memiliki beberapa unit mesin.

```
# Function to extract main machine and auxiliary engine data asynchronously
async def extract_machine_data(session, page_number):
    url = f"https://www.bki.co.id/shipregister-{page_number}.html#"
    headers = {
        "User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/12
    }
    try:
        async with session.get(url, headers=headers) as response:
            response.raise_for_status()
            html = await response.text()

        soup = BeautifulSoup(html, 'html.parser')

        # Find the tables by their headers
        main_machine_table = soup.find('h4', string='MAIN MACHINE')

        # Convert data to JSON format
        main_machine_data = [dict(zip(main_machine_headers, row)) for row in main_machine_rows]
        auxiliary_engine_data = [dict(zip(auxiliary_engine_headers, row)) for row in auxiliary_engine_rows]

        # Create DataFrames
```

Gambar 3.45 Contoh skrip fungsi extract_machine_data.

6. Semua data yang telah diperoleh dari berbagai bagian halaman digabungkan menggunakan pandas.merge dengan kolom kunci

(*foreign key*) berupa *Nomor Register*. Penggabungan dilakukan untuk memastikan bahwa data tabel transportasi utama, spesifikasi, dan mesin berada dalam satu tabel utuh.

7. Data akhir disimpan dalam file Excel dengan nama final_table1-10000.xlsx atau diganti dengan rentang nama halaman lainnya, yang berisi struktur data lengkap dan siap digunakan untuk proses analisis atau visualisasi lebih lanjut.
8. Data yang sudah diambil/ *scrapping* digabung menjadi satu sumber data dan diekspor menjadi file Spreadsheet.

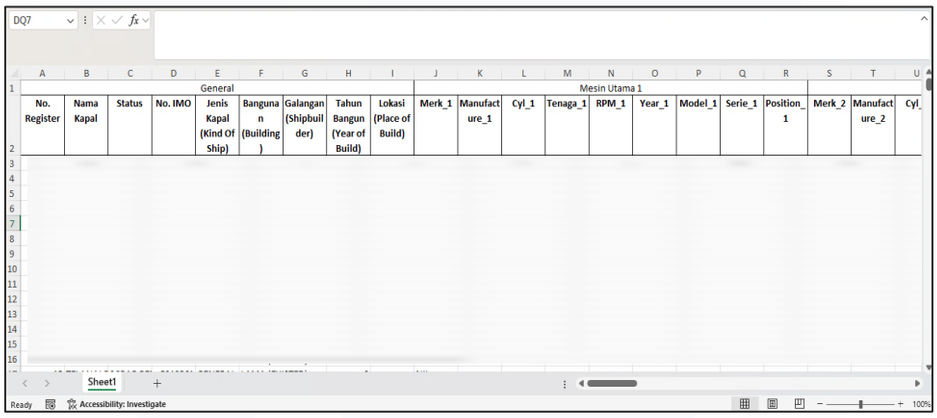
```python
# Function to extract data from a given URL asynchronously
async def extract_data_from_url(session, url, page_num):
    try:
        headers = {
            "User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
        }
        async with session.get(url, headers=headers) as response:
            response.raise_for_status()  # Raise an error for bad status codes
            html = await response.text()

        # Parse the HTML content using BeautifulSoup
        soup = BeautifulSoup(html, 'html.parser')

        # Find the div with class "blog-content"
        content_div = soup.find('div', {'class': 'blog-content'})
        if not content_div:
            print(f"No content found for URL: {url}")
            return None

        # Find the table with ID "sh" within the div
        table = content_div.find('table', {'id': 'ship'})
        if not table:
            print(f"No table with ID 'sh' found for URL: {url}")
            return None

        # Initialize a list to store data
        data = []

        # Iterate through each row to extract data (skip header row)
        rows = table.find_all('tr')[1:]  # Start from the second row to skip the header
```

Gambar 3.46 Contoh skrip scraping data untukk mengambil data transportasi.

Terdapat tantangan yang muncul selama proses scraping seperti halaman kosong, ketidaksesuaian struktur HTML, hingga pembatasan dari sisi server. Cara yang dilakukan untuk mengatasi hal tersebut adalah menggunakan mekanisme retry dengan *exponential backoff* yang memungkinkan sistem untuk menunggu secara acak sebelum mencoba kembali halaman yang gagal diakses. Selain itu, validasi elemen HTML

diperiksa/ cek sebelum mengambil data untuk menghindari error saat *parsing*.

Hasil dari scraping ini adalah file .xlsx yang berisi data dari ribuan transportasi dengan data identitas, status, spesifikasi teknis, hingga rincian mesin. Hasil data ini dapat dikembangkan lebih lanjut ke dalam bentuk dashboard atau sebagai dataset untuk analisis terkait kondisi industri maritim Indonesia.



Gambar 3.47 Contoh tabel data hasil gabungan.

3.3 Kendala yang Ditemukan

Berikut merupakan kesulitan dan kendala yang ditemukan selama pelaksanaan program kerja magang di Data Indonesia yang berlangsung kurang lebih empat bulan,

1. Keterbatasan data yang diperoleh dari berbagai sumber (seperti BPS atau hasil web scraping). Data memiliki format yang tidak konsisten seperti perbedaan penamaan kolom, struktur tabel yang tidak seragam, atau nilai numerik yang tidak dapat langsung diproses karena adanya perbedaan antar kolom mengenai tanda titik atau koma sebagai pemisah ribuan.
2. Kendala lain yang cukup signifikan adalah jumlah file Excel yang harus digabungkan kadang terlalu banyak untuk menggunakan alat pengolahan data yang umum dipakai awam. Dalam salah satu proyek, terdapat proses untuk mengolah ribuan file Excel yang masing-masing

memiliki variasi kolom yang berbeda yang tidak memungkinkan untuk dikerjakan secara manual.

3. Saat membangun dashboard dengan Looker Studio muncul keterbatasan jumlah filter yang kurang kompleks, performa dashboard lambat saat data terlalu besar, dan kesulitan menampilkan metrik kompleks dengan
4. kalkulasi atau rumus serta keterbatasan fitur untuk kalkulasi yang tidak fleksibel seperti di spreadsheet.
5. Permasalahan yang muncul saat memulai program magang akibat adanya kendala dalam memahami alur pengolahan data dari data mentah hingga menjadi dashboard yang utuh.
6. Saat menjalankan proyek *scraping data,* perangkat sering terblokir atau server menolak permintaan akibat proses scraping dilakukan dalam jumlah besar atau data yang diambil ada ribuan halaman. Permintaan data yang terlalu besar, terlalu cepat, dan terlalu sering dapat mengakibatkan kegagalan akses halaman sebab server website lambat.

3.4 Solusi atas Kendala yang Ditemukan

Dari permasalahan yang telah ditemukan, berikut adalah solusi atau opsi alternatif untuk mengurangi atau menyelesaikan kendala tersebut,

1. Permasalahan terkait format data yang tidak konsisten dapat diatasi dengan melakukan proses eksplorasi data secara menyeluruh menggunakan Microsoft Excel dan Python. Beberapa cara yang dilakukan adalah standarisasi penamaan kolom, perbaikan format angka (menghapus pemisah ribuan yang berbeda, menyamakan format desimal), serta pemetaan ulang struktur tabel agar data dari berbagai sumber memiliki format yang seragam dan siap digunakan untuk analisis lebih lanjut.
2. Kendala penggabungan banyak file Excel dengan struktur kolom yang berbeda dapat diatasi dengan membangun skrip otomatisasi menggunakan Python, khususnya dengan bantuan modul pandas, dan openpyxl. Cara tersebut memungkinkan seluruh file dalam direktori diproses secara batch lalu digabungkan menjadi satu data master yang telah distandarkan.
3. Solusi untuk menghadapi keterbatasan teknis seperti lambatnya performa saat memuat data dalam jumlah besar adalah dengan

membersihkan dataset dari kolom yang tidak relevan dan membatasi ukuran dataset yang digunakan. Selain itu, solusi lain yang dilakukan seperti merancang ulang struktur agar data dapat mendukung agregasi langsung dari dalam Looker Studio tanpa perlu rumus tambahan yang berat.

4. Kendala dalam memahami alur proses pengolahan data dari data mentah hingga menjadi dashboard di awal magang dapat diatasi dengan memperbanyak studi mandiri serta berdiskusi aktif dengan pembimbing dan rekan magang. *Data Scientist Intern* di Data Indonesia dapat mempelajari ulang prinsip dasar data pipeline dari eksplorasi data, transformasi data hingga visualisasi, serta mempelajari contoh dashboard yang sudah ada untuk memahami bagaimana setiap komponen data bekerja secara keseluruhan.
5. Solusi jika penolakan permintaan muncul dari sisi server adalah menerapkan strategi retry mechanism dengan pendekatan exponential backoff. Setiap kali scraping terhadap satu halaman gagal, sistem akan mencoba kembali hingga lima kali dengan penambahan waktu tunggu yang meningkat secara eksponensial. Penambahan delay acak juga dapat digunakan untuk menyerupai pola akses pengguna manusia sehingga terhindar dari deteksi bots.