

BUSINESS KIMIA FARMA PERFORMANCE

© 2025 Pascal Brilliandy. All Rights Reserved



Hello!

I'm Pascal Brilliandy K

Experience Data Analyst skilled in Excel, SQL, Python, and Machine Learning techniques for data exploration and predictive modeling. Skilled in creating interactive dashboards using Looker Studio, Tableau, and Power BI to support data-driven business decision-making

ABOUT COMPANY

© 2025 Pascal Brilliandy. All Rights Reserved



Kimia Farma adalah perusahaan industri farmasi pertama di Indonesia yang didirikan oleh Pemerintah Hindia Belanda tahun 1817. Nama perusahaan ini pada awalnya adalah NV Chemicalien Handle Rathkamp & Co. Berdasarkan kebijaksanaan nasionalisasi atas eks perusahaan Belanda di masa awal kemerdekaan, pada tahun 1958, Pemerintah Republik Indonesia melakukan peleburan sejumlah perusahaan farmasi menjadi PNF (Perusahaan Negara Farmasi) Bhinneka Kimia Farma. Kemudian pada tanggal 16 Agustus 1971, bentuk badan hukum PNF diubah menjadi Perseroan Terbatas, sehingga nama perusahaan berubah menjadi PT Kimia Farma (Persero).

PROJECT PORTOFOLIO

© 2025 Pascal Brilliandy. All Rights Reserved

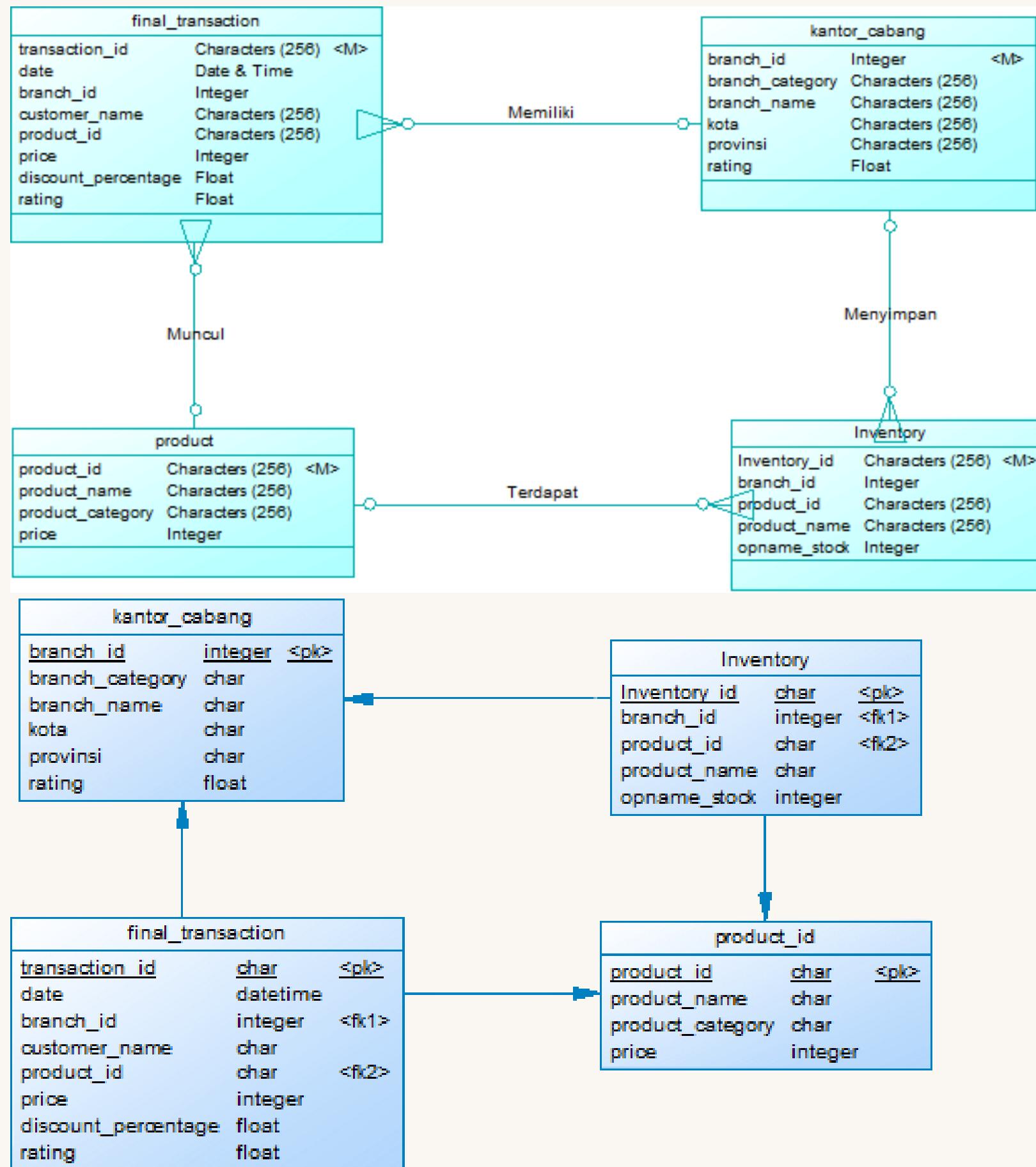
PROJECT PORTOFOLIO

Final task ini bertujuan untuk menganalisis performa bisnis PT Kimia Farma dari tahun 2020 - 2023. Proyek dimulai dengan memahami struktur data awal menggunakan pendekatan berbasis data yang diawali dengan perancangan CDM (Conceptual Data Model) dan PDM (Physical Data Model) untuk memetakan entitas penting seperti transaksi, produk, cabang, inventory, dan pelanggan.

Setelah memahami struktur dan relasi antar tabel, dilakukan proses data cleansing dan transformasi menggunakan SQL untuk menghasilkan tabel analisa yang bersih dan siap digunakan. Selanjutnya, memvisualisasikan data tersebut dalam bentuk dashboard interaktif menggunakan Google Looker Studio



PROJECT PORTOFOLIO



Project Goal :

- Kinerja penjualan tiap tahun dan wilayah,
- Efisiensi operasional cabang,
- Profitabilitas produk,
- Preferensi pelanggan,
- Dan rekomendasi perbaikan berdasarkan rating dan performa transaksi.

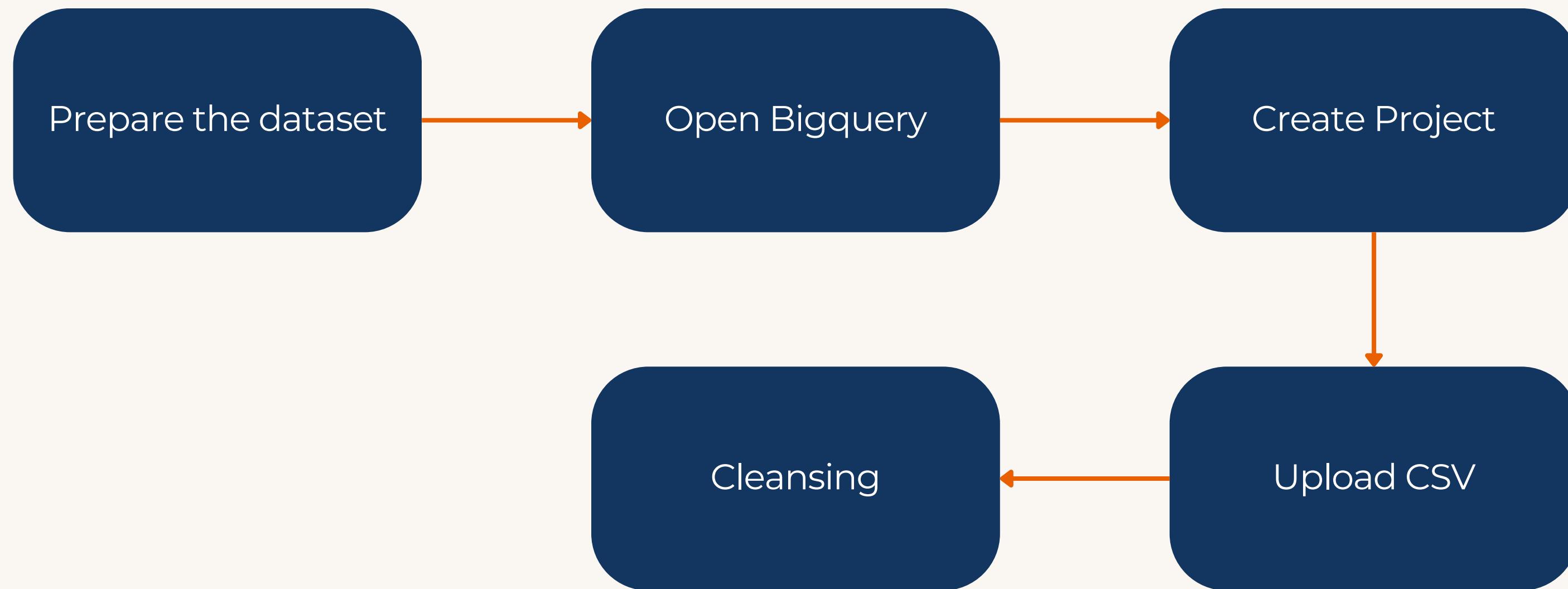
Problem Statement:

- Menentukan cabang atau wilayah yang paling menguntungkan atau bermasalah,
- Mengetahui efektivitas diskon terhadap profitabilitas,
- Mengukur kepuasan pelanggan berdasarkan rating transaksi dan cabang,
- Mengambil keputusan berbasis data dalam hal pengembangan produk atau strategi pemasaran.

IMPORTING DATASET TO BIGQUERY

© 2025 Pascal Brilliandy. All Rights Reserved

STEP BY STEP



STEP BY STEP

Row	total_rows	missing_transaction	missing_date	missing_branch	missing_cust_name	missing_product_id	missing_price	missing_discount_percent	missing_rating
1	672458	0	0	0	0	0	0	0	0

Row	total_rows	missing_inventory	missing_branch	missing_product_id	missing_product_name	missing_opname_stock
1	1035000	0	0	0	0	0

Row	total_rows	missing_branch	missing_branch_category	missing_branch_name	missing_kota	missing_provin	missing_rating
1	1725	0	0	0	0	0	0

Row	total_rows	missing_pro	missing_product_category	missing_product_name	missing_price
1	150	0	0	0	0

STEP BY STEP

The screenshot shows the Google BigQuery Studio interface. At the top, there's a navigation bar with 'Google Cloud' and 'Kimia Farma PBI' buttons, a search bar, and a user profile icon. The main area is titled 'Welcome to BigQuery Studio!' and features a 'Create new' section with buttons for 'SQL query', 'Notebook', 'Data canvas', 'Data preparation', and 'Pipeline'. Below this is a 'Recently opened' section displaying five dataset cards:

- final_transaction (Open, 1 minute ago)
- Inventory (Open, 12 minutes ago)
- vw_join_data_kimia_farma (Open, 12 minutes ago)
- join_data_kimia_farma (Open, 21 hours ago)
- product (Open, 2 days)

At the bottom left, there's a 'Repository' section with a 'Preview' tab selected, showing a message: 'No repository selected. Select a repository and a workspace to view its content.' There's also a 'View repositories' button. On the right, a 'Job history' section is partially visible. A 'Try with sample data' box for 'Google Trends Demo Query' is overlaid on the bottom right.

TABEL ANALISA

© 2025 Pascal Brilliandy. All Rights Reserved

TABEL ANALISA

<input type="checkbox"/>	Field name	Type	Mode
<input type="checkbox"/>	transaction_id	STRING	NULLABLE
<input type="checkbox"/>	date	DATE	NULLABLE
<input type="checkbox"/>	branch_id	INTEGER	NULLABLE
<input type="checkbox"/>	branch_category	STRING	NULLABLE
<input type="checkbox"/>	branch_name	STRING	NULLABLE
<input type="checkbox"/>	kota	STRING	NULLABLE
<input type="checkbox"/>	provinsi	STRING	NULLABLE
<input type="checkbox"/>	customer_name	STRING	NULLABLE
<input type="checkbox"/>	product_id	STRING	NULLABLE
<input type="checkbox"/>	product_name	STRING	NULLABLE
<input type="checkbox"/>	product_category	STRING	NULLABLE
<input type="checkbox"/>	total_price	INTEGER	NULLABLE
<input type="checkbox"/>	discount_percentage	FLOAT	NULLABLE
<input type="checkbox"/>	item_price	INTEGER	NULLABLE
<input type="checkbox"/>	nett_sales	FLOAT	NULLABLE
<input type="checkbox"/>	persentase_gross_laba	FLOAT	NULLABLE
<input type="checkbox"/>	nett_profit	FLOAT	NULLABLE
<input type="checkbox"/>	rating_transaction	FLOAT	NULLABLE

Dilakukan pembuatan tabel analisa dengan membuat bigquery syntax berdasarkan hasil agregasi dari 4 dataset sebelumnya. Pada gambar, tabel analisa dibuat di dalam dataset baru yaitu “join_data_kimia_farma”

TABEL ANALISA

Row	transaction_id	date	branch_id	branch_category	branch_name	kota
1	TRX1000078	2023-07-19	83862	Klinik-Apotek-Laboratorium	Kimia Farma - Klinik-Apotek-La...	Ternate
2	TRX1001390	2020-04-22	51871	Klinik & Apotek	Kimia Farma - Klinik & Apotek	Balikpapan
3	TRX1003024	2021-04-30	66696	Klinik-Apotek-Laboratorium	Kimia Farma - Klinik-Apotek-La...	Denpasar
4	TRX1005577	2022-12-05	85805	Klinik-Apotek-Laboratorium	Kimia Farma - Klinik-Apotek-La...	Yogyakarta
5	TRX1007025	2022-06-29	97507	Apotek	Kimia Farma - Apotek	Pekanbaru

provinsi	rating_cabang	customer_name	product_id	product_name	product_category	total_price	discount_percentage
Maluku Utara	4.7	Lisa Lam	KF151	Other analgesics and antipyreti...	N02BE/B	723800	0.05
Kalimantan Timur	3.9	Nicholas Cooper	KF953	Drugs for obstructive airway di...	R06	997500	0.02
Bali	4.0	Pamela Anderson	KF166	Psycholeptics drugs, Anxiolytic...	N02BA	823800	0.06
DI Yogyakarta	4.5	Joan Castro	KF678	Psycholeptics drugs, Hypnotic...	N05B	426700	0.09
Riau	4.1	Brandon Mathis	KF961	Other analgesics and antipyretics, Salicylic acid and	N02BE/B	438100	0.12

*Kolom Tabel Join

nett_sales	percentase_gross_sales	nett_profit	rating_transaction
723438.1	0.3	217031.43	4.0
997300.5	0.3	299190.15	3.4
823305.72	0.3	246991.72	3.3
426315.97	0.25	106578.99	3.0
437574.28	0.25	109393.57	4.8

Creative Presentation

BIGQUERY SYNTAX

© 2025 Pascal Brilliandy. All Rights Reserved

BIGQUERY SYNTAX

```
1 CREATE OR REPLACE TABLE `kimia-farma-pbi-458804.kimia_farma_pbi.join_data_kimia_farma` AS
2 with
3
4 -- CTE Tabel Inventory
5 table_inv AS (
6   SELECT * FROM `kimia-farma-pbi-458804.kimia_farma_pbi.Inventory`
7 ),
8 -- CTE Tabel Transaction
9 table_ft AS (
10  SELECT * FROM `kimia-farma-pbi-458804.kimia_farma_pbi.final_transaction`
11 ),
12 -- CTE Tabel Kantor Cabang
13 table_kc AS (
14   SELECT * FROM `kimia-farma-pbi-458804.kimia_farma_pbi.kantor_cabang`
15 ),
16 -- CTE Tabel Product
17 table_pro AS (
18   SELECT * FROM `kimia-farma-pbi-458804.kimia_farma_pbi.product`
19 )
```

- Membuat Tabel Baru dengan nama “join data kimia farma”
- Mengambil seluruh data menggunakan CTE

BIGQUERY SYNTAX

```
21 -- Cek Missing Value Final Transaction
22 missing_ft as(
23   select
24     count(*) as total_rows, -- Jumlah Row
25     countif(transaction_id is null) as missing_transaction, -- Jumlah missing transaction
26     countif(date is null) as missing_date, -- Jumlah missing date
27     countif(branch_id is null) as missing_branch, -- Jumlah missing branch id
28     countif(customer_name is null) as missing_cust_name, -- Jumlah missing customer name
29     countif(product_id is null) as missing_product_id, -- Jumlah missing product id
30     countif(price is null) as missing_price, -- Jumlah missing price
31     countif(discount_percentage is null) as missing_discount_percentage, -- Jumlah missing discount
32     countif(rating is null) as missing_rating -- Jumlah missing rating
33   from table_ft
34 ),
35
36 -- Cek Missing Value Inventory
37 missing_inv as(
38   select
39     count(*) as total_rows, -- Jumlah Row
40     countif(Inventory_id is null) as missing_inventory, -- Jumlah missing inventory id
41     countif(branch_id is null) as missing_branch, -- Jumlah missing branch
42     countif(product_id is null) as missing_product_id, -- Jumlah missing product id
43     countif(product_name is null) as missing_product_name, -- Jumlah missing product name
44     countif(opname_stock is null) as missing_opname_stock -- Jumlah missing opname stock
45   from table_inv
46 ),
47
```

- Membuat query untuk missing value (null)

BIGQUERY SYNTAX

```
48 -- Cek Missing Value Kantor Cabang
49 missing_kc as(
50   select
51     count(*) as total_rows, -- Jumlah Row
52     countif(branch_id is null) as missing_branch, -- Jumlah missing branch
53     countif(branch_category is null) as missing_branch_category, -- Jumlah missing
54     countif(branch_name is null) as missing_branch_name, -- Jumlah missing branch name
55     countif(kota is null) as missing_kota, -- Jumlah missing kota
56     countif(provinsi is null) as missing_provinsi, -- Jumlah missing provinsi
57     countif(rating is null) as missing_rating -- Jumlah missing rating
58   from table_kc
59 ),
60
61 -- Cek Missing Value Product
62 missing_pro as[
63   select
64     count(*) as total_rows, -- Jumlah row
65     countif(product_id is null) as missing_pro, -- Jumlah missing product id
66     countif(product_category is null) as missing_product_category, -- Jumlah missing product category
67     countif(product_name is null) as missing_product_name, -- Jumlah missing product name
68     countif(price is null) as missing_price -- Jumlah missing price
69   from table_pro
70 ],
71
```

BIGQUERY SYNTAX

```
72 -- CTE Tabel Inventory: Bersihkan missing values
73 cleaned_inventory AS (
74     SELECT *
75     FROM (
76         SELECT *,
77             ROW_NUMBER() OVER(PARTITION BY inventory_id, branch_id, product_id) AS nRow
78         from table_inv
79         WHERE Inventory_id IS NOT NULL
80         AND branch_id IS NOT NULL
81         AND product_id IS NOT NULL
82         AND product_name IS NOT NULL
83         AND opname_stock IS NOT NULL
84     ) AS sub
85     WHERE nRow = 1
86 ),
87
88 -- CTE Tabel Final Transaction: Bersihkan missing values
89 cleaned_transaction AS (
90     SELECT *
91     FROM (
92         SELECT *,
93             ROW_NUMBER() OVER(PARTITION BY transaction_id, date, branch_id) AS nRow
94         from table_ft
95         WHERE transaction_id IS NOT NULL
96         AND date IS NOT NULL
97         AND branch_id IS NOT NULL
98         AND customer_name IS NOT NULL
99         AND product_id IS NOT NULL
100        AND price IS NOT NULL
101        AND discount_percentage IS NOT NULL
102        AND rating IS NOT NULL
103    ) AS sub
104    WHERE nRow = 1
105 ),
106
```

- Query untuk menghapus duplikasi

BIGQUERY SYNTAX

```
107 -- CTE Tabel Kantor Cabang: Bersihkan missing values
108 cleaned_branch AS (
109     SELECT *
110     FROM (
111         SELECT *,
112             ROW_NUMBER() OVER(PARTITION BY branch_id, branch_name, kota) AS nRow
113         from table_kc
114         WHERE branch_id IS NOT NULL
115             AND branch_category IS NOT NULL
116             AND branch_name IS NOT NULL
117             AND kota IS NOT NULL
118             AND provinsi IS NOT NULL
119             AND rating IS NOT NULL
120     ) AS sub
121     WHERE nRow = 1
122 ),
123
124 -- CTE Tabel Product: Bersihkan missing values
125 cleaned_product AS (
126     SELECT *
127     FROM (
128         SELECT *,
129             ROW_NUMBER() OVER(PARTITION BY product_id, product_name, product_category) AS nRow
130         from table_pro
131         WHERE product_id IS NOT NULL
132             AND product_category IS NOT NULL
133             AND product_name IS NOT NULL
134             AND price IS NOT NULL
135     ) AS sub
136     WHERE nRow = 1
137 ),
138
```

BIGQUERY SYNTAX

```
139 -- Gabungkan semua tabel dengan INNER JOIN  
140 joined_data AS ()  
141     SELECT  
142         ft.transaction_id,  
143         ft.date,  
144         ft.branch_id,  
145         kc.branch_category,  
146         kc.branch_name,  
147         kc.kota,  
148         kc.provinsi,  
149         ft.customer_name,  
150         ft.product_id,  
151         pro.product_name,  
152         pro.product_category,  
153         ft.price AS total_price,  
154         ft.discount_percentage,  
155         pro.price AS item_price,
```

1

```
156     -- Kalkulasi nett_sales  
157     ft.price * (1 - ft.discount_percentage / 100.0) AS nett_sales,  
158     -- Kalkulasi persentase_gross_laba  
159     CASE  
160         WHEN ft.price <= 50000 THEN 0.10  
161         WHEN ft.price <= 100000 THEN 0.15  
162         WHEN ft.price <= 300000 THEN 0.20  
163         WHEN ft.price <= 500000 THEN 0.25  
164         ELSE 0.30  
165     END AS persentase_gross_laba,  
166     -- Kalkulasi nett_profit  
167     (ft.price * (1 - ft.discount_percentage / 100.0)) *  
168     CASE  
169         WHEN ft.price <= 50000 THEN 0.10  
170         WHEN ft.price <= 100000 THEN 0.15  
171         WHEN ft.price <= 300000 THEN 0.20  
172         WHEN ft.price <= 500000 THEN 0.25  
173         ELSE 0.30  
174     END AS nett_profit,
```

2

```
175  
176  
177     FROM cleaned_transaction AS ft  
178     JOIN cleaned_branch AS kc  
179         ON ft.branch_id = kc.branch_id  
180     JOIN cleaned_product AS pro  
181         ON ft.product_id = pro.product_id  
182  
183
```

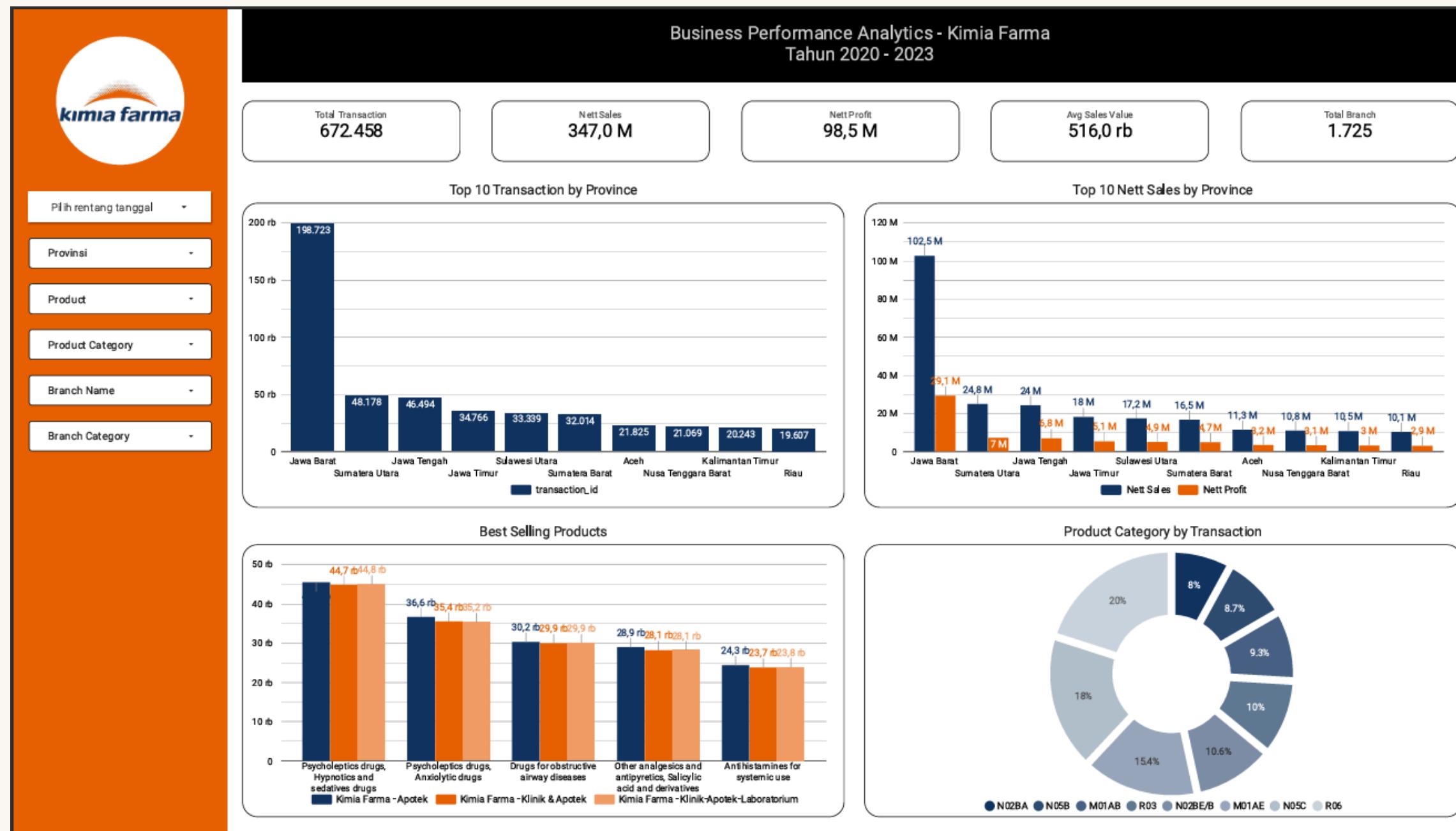
3

Creative Presentation

DASHBOARD PERFORMANCE ANALYTICS

© 2025 Pascal Brilliandy. All Rights Reserved

DASHBOARD PERFORMANCE ANALYTICS



LINK FINAL TASK

Link Dashboard Looker Studio

Link Video Presentasi

Link Repository Github

**Thank You
So Much!**