

# Alif Abdul Hakim



**DATA ANALYST  
PORTFOLIO**

# INTRODUCING<sub>02</sub>

## ME



With over 1 year of experience in data analytics through academic projects, bootcamps, and teaching assistant roles, I am proficient in Python, SQL, Excel, Looker and Power BI for data processing, cleaning, and visualization. Experienced in conducting **data validation, exploratory analysis, and building interactive dashboards to deliver actionable insights that support business decision making.** Strong analytical and problem solving abilities with a proven record of ensuring data accuracy and consistency across large datasets.

# BACKGROUND

03



## Institut Teknologi Nasional Bandung

Informatics | 3.72/4.00

- **MANAGED COURSE PROJECTS:** SUCCESSFULLY HANDLED AND COMPLETED VARIOUS ACADEMIC PROJECTS, DEMONSTRATING STRONG ORGANIZATIONAL AND TIME MANAGEMENT SKILLS.
- **TEACHING ASSISTANT FOR TWO PRACTICUM COURSES:** CONDUCTED PRACTICAL SESSIONS FOR TWO COURSES, PROVIDING GUIDANCE AND SUPPORT TO STUDENTS, ENHANCING THEIR UNDERSTANDING OF THE SUBJECT MATTER.

## RevoU Tech Academy

Learn Data Analytics & Software Development with AI |  
98/100



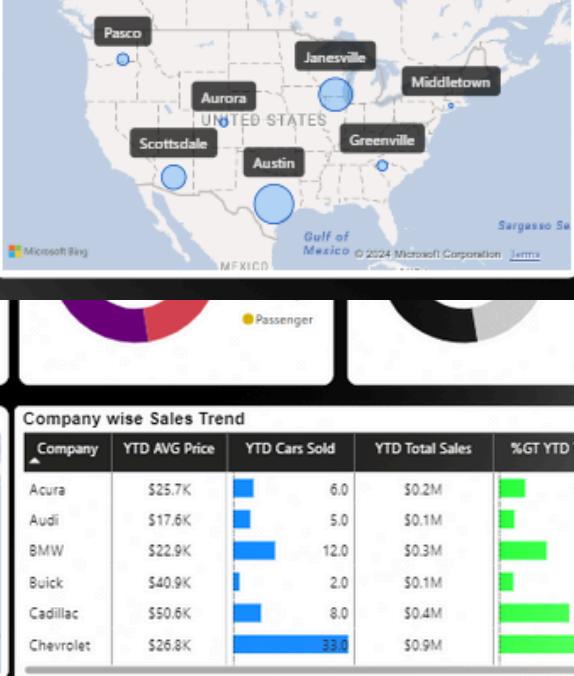
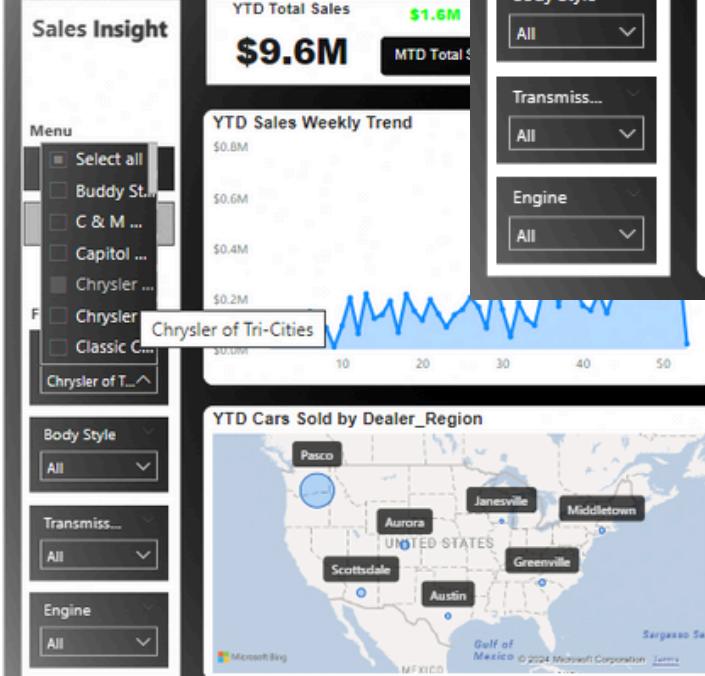
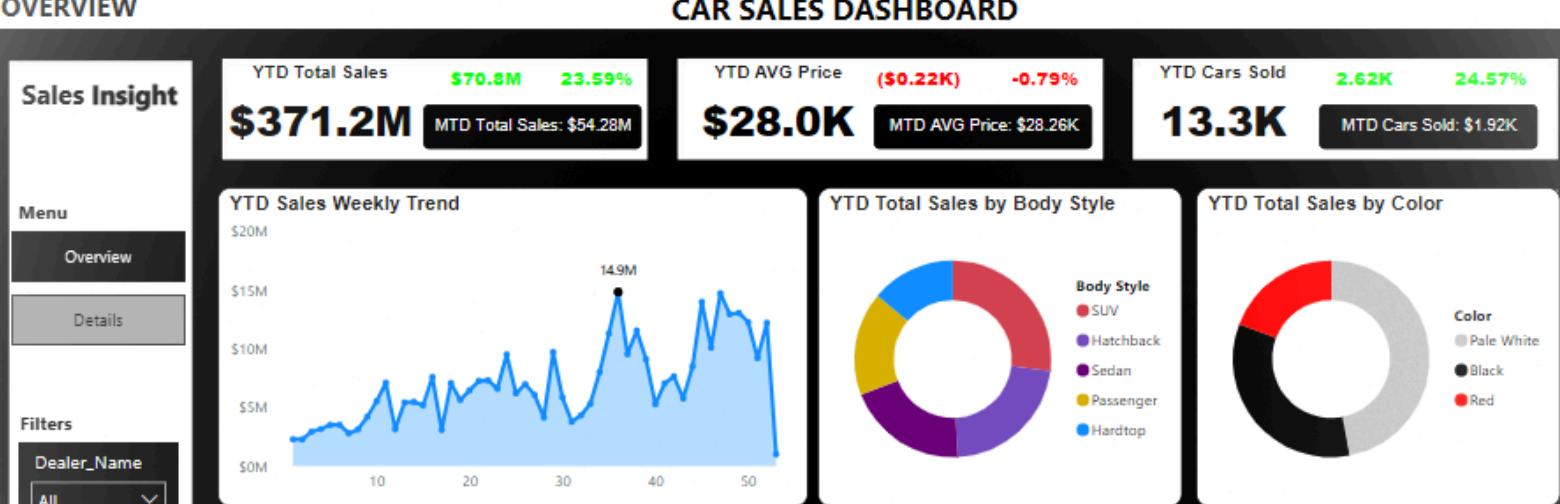
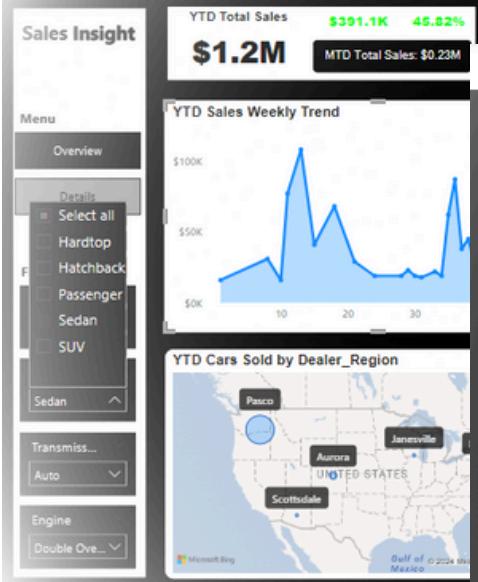
- LEARN ABOUT DATA PROCESSING IN DATA ANALYTICS, DATA VISUALIZATION AND COMMUNICATION WITH AI TOOLS, INTRO TO WEB DEVELOPMENT, FRONT-END DEVELOPMENT WITH AI TOOLS, AND CAREER DEVELOPMENT WITH AI TOOLS.
- PARTICIPATING IN A TEAM TO DEVELOP A CAPSTONE PROJECT.

# PROJECT

# CAR SALES

OVERVIEW

CAR SALES DASHBOARD



# Power BI Project 05

[Get the Github link](#)

SHBOARD

<b>(\$0.22K)</b> <b>-0.79%</b>	<b>MTD AVG Price: \$28.26K</b>	<b>YTD Cars Sold</b> <b>2.62K</b> <b>24.57%</b>	<b>MTD Cars Sold: \$1.92K</b>			
<b>Date</b>	<b>Company</b>	<b>Color</b>	<b>Model</b>			
Sunday, January 02, 2022	Ford	Black	Expedition	\$26.00K		
Sunday, January 02, 2022	Dodge	Black	Durango	\$19.00K		
Sunday, January 02, 2022	Cadillac	Red	Eldorado	\$31.50K		
Sunday, January 02, 2022	Toyota	Pale White	Celica	\$14.00K		
Sunday, January 02, 2022	Acura	Red	TL	\$24.50K		
Sunday, January 02, 2022	Mitsubishi	Pale White	Diamante	\$12.00K		
Sunday, January 02, 2022	Toyota	Pale White	Corolla	\$14.00K		
Sunday, January 02, 2022	Mitsubishi	Pale White	Galant	\$42.00K		
Sunday, January 02, 2022	Cadillac	Red	Malibu	\$82.00K		
Sunday, January 02, 2022	Ford	Pale White	Escort	\$15.00K		
Sunday, January 02, 2022	Acura	Pale White	RL	\$31.00K		
Sunday, January 02, 2022	Nissan	Pale White	Pathfinder	\$46.00K		
Sunday, January 02, 2022	Mercury	Black	Grand Marquis	\$9.00K		
Sunday, January 02, 2022	Harrison	Scrivener Performance Engineering	BMW	\$15.00K		
Sunday, January 02, 2022	Zainab	Buddy Storbeck's Diesel Service Inc	BMW	323i	\$15.00K	
Sunday, January 02, 2022	Zara	C & M Motors Inc	Subaru	Pale White	Forester	\$17.00K
Sunday, January 02, 2022	Zoe	Capitol KIA	Hyundai	Black	Accent	\$18.00K
Sunday, January 02, 2022	Zoey	Chrysler of Tri-Cities	Cadillac	Pale White	Eldorado	\$31.00K
Sunday, January 02, 2022	Aaliyah	Chrysler Plymouth	Cadillac	Pale White	Land Cruiser	\$33.00K
Sunday, January 02, 2022	Ariannah	Classic Chevy	Honda	Pale White	Accord	\$21.00K
		Total	\$671,525.47K			

# MAVEN ROASTERS COFFEE SHOP SALES

# Sql ft Power BI Project 06

SqlIt Power BI

The screenshot displays three separate SQLite database windows, each showing a query editor and a results pane.

- Query 1:** A complex SQL script for calculating monthly sales and orders. It includes CTEs for total sales and total orders, and a main query that joins these with monthly sales data to calculate growth percentages.
- Query 2:** A simplified version of the first query, likely a step in the development process. It shows the core logic for calculating total sales and total orders.
- Query 3:** The final, optimized version of the query. It uses a WITH clause to define the monthly sales CTE and includes a detailed explanatory comment at the top.

Each query editor has standard buttons for RUN, SAVE QUERY, SHARE, SCHEDULE, and MORE. The results panes show the query results and execution details.

# Get the Github link

This project aims to explore insights into sales data from **149,117** Maven Roasters **transaction records**, focusing on **total sales, total number of orders, and total quantity sold performance**. The data source comes from **Maven Analytics**. This dataset includes historical sales data for a coffee shop operating out of three NYC locations. Each store consists of several departments, and the task is to derive insights from total sales, total number of orders, and total quantity sold performance.

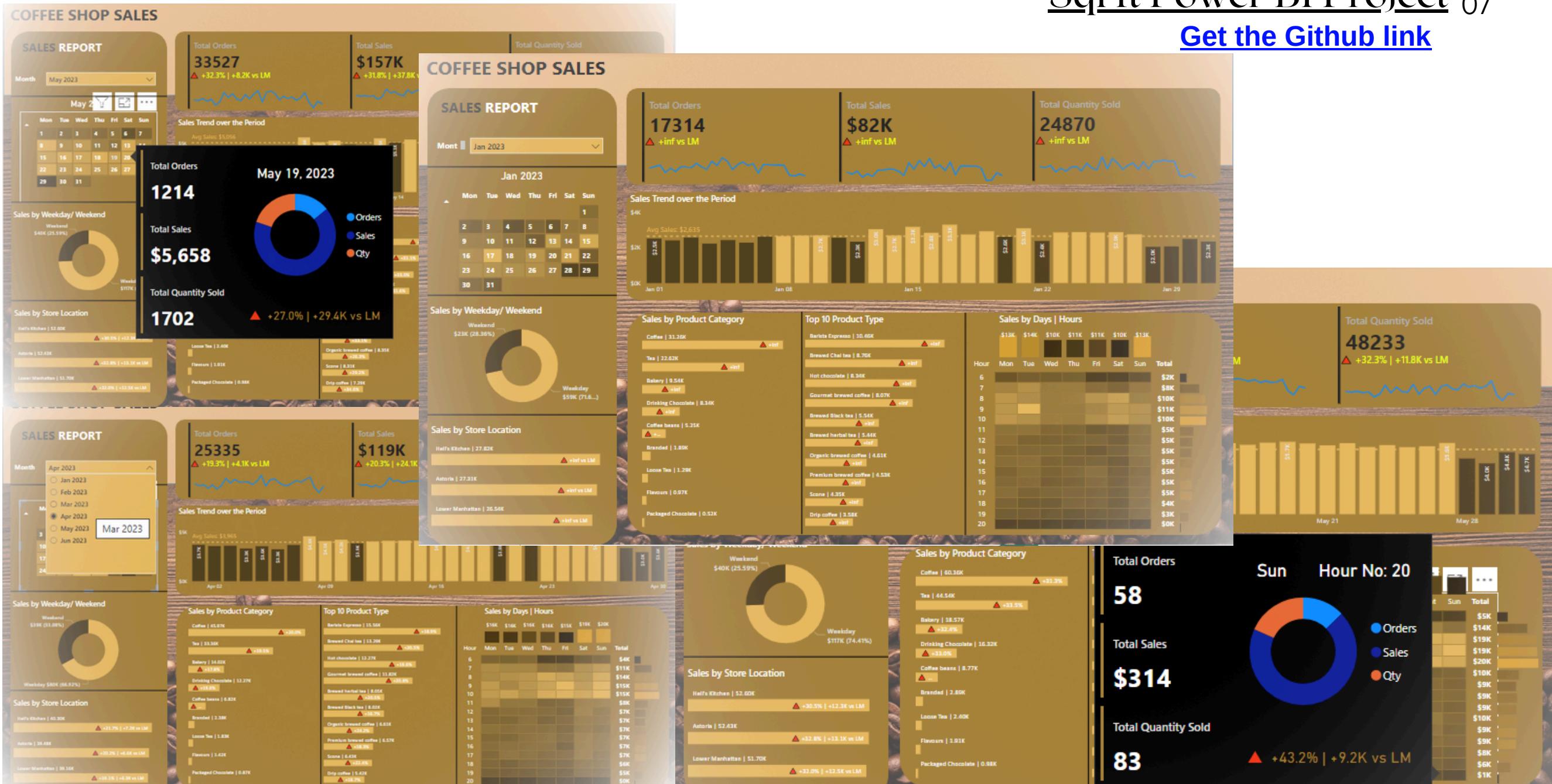
In the product category analysis, the number of unique product IDs will be identified along with the product distribution of each Maven Roasters branch. On the transaction quantity side, the number of best-selling products will be explored, and total sales, total number of orders, and total quantity sold performance will be **broken down by month**. Sales analysis will involve understanding the number of sales at **various times**, as well as on **weekends and weekdays**. A **Month-over-Month (MOM)** analysis will also be performed to identify sales trends and growth patterns.

By breaking it down into these aspects, this project aims to provide a comprehensive picture of Maven Roasters' sales performance and provide strategic insights for improving and optimizing sales strategies.

# MAVEN ROASTERS COFFEE SHOP SALES

Sql ft Power BI Project 07

[Get the Github link](#)



# STARTUP LAYOFFS

# Sql Project 08

```
phk EDA
Limit to 1000 rows
13 -- Membuat table baru untuk proses selanjutnya
14 • CREATE TABLE phk_copyy
15   LIKE phk;
16
17 • SELECT *
18   FROM phk_copyys;
19
20 -- Memasukkan data
21 • INSERT phk_copyy
22   SELECT *
23   FROM phk;
24
25
26 -----DATA
27 -- STEP:
28 -- 1. Remove Duplicates
29 -- 2. Standardize Data
30 -- 3. Null/Blank Values
31 -- 4. Remove any Columns
32
33 -- Step 1. Remove Duplicates
34
35 -- Cek Duplikat
36 • SELECT * FROM phk_copyys;
37

94      }
95      DELETE
96        FROM DELETE_CTE
97      ;
98
99 • WITH DELETE_CTE AS (
100   SELECT company, location, industry, total_laid_off, percentage_laid_off,
101     ROW_NUMBER() OVER (PARTITION BY company, location, industry, total_laid_off)
102       FROM phk_copyy
103     )
104     DELETE FROM phk_copyy
105 WHERE (company, location, industry, total_laid_off, percentage_laid_off,
106       SELECT company, location, industry, total_laid_off, percentage_laid_off
107       FROM DELETE_CTE
108     ) AND row_num > 1;
109
110 -- Membuat & menambahkan kolom baru row_num, selanjutnya menghapus duplikat
111
112 • ALTER TABLE phk_copyy ADD row_num INT;
113
114 • SELECT *
115   FROM phk_copyys;
116
117
118 • CREATE TABLE `phk_copyv2` (
```

The image shows two MySQL Workbench windows side-by-side. Both windows have tabs 'phk' and 'EDA\*' active. The left window contains a series of SQL queries for Exploratory Data Analysis (EDA) on a table named 'phk\_copyyy2'. The right window shows a more complex query involving a Common Table Expression (CTE) named 'Company\_Year' and a derived table 'Company\_Year\_Rank'.

```
1 -- Exploratory Data Analysis
2
3 • SELECT *
4   FROM phk_copyyy2;
5
6 • SELECT *
7   FROM phk_copyyy2
8   ORDER BY total_laid_off DESC;
9
10 • SELECT MAX(total_laid_off)
11   FROM phk_copyyy2;
12
13
14 -- Mengubah tipe data kolom ke int
15
16 -- Periksa apakah ada nilai non-numerik
17 • SELECT *
18   FROM phk_copyyy2
19   WHERE CAST(total_laid_off AS UNSIGNED) IS NOT NULL;
20
21 -- Ubah tipe data kolom setelah memastikan
22 • ALTER TABLE phk_copyyy2
23   MODIFY total_laid_off INT;
24
25
91
92
93 • WITH Company_Year AS
94 (
95   SELECT company, YEAR(date) AS years, SUM(total_laid_off) AS total_laid_off
96     FROM phk_copyyy2
97   GROUP BY company, YEAR(date)
98 )
99 , Company_Year_Rank AS (
100   SELECT company, years, total_laid_off, DENSE_RANK() OVER (PARTITION BY years ORDER BY total_laid_off DESC) AS ranking
101     FROM Company_Year
102 )
103   SELECT company, years, total_laid_off, ranking
104     FROM Company_Year_Rank
105   WHERE ranking <= 3
106   AND years IS NOT NULL
107   ORDER BY years ASC, total_laid_off DESC;
108
109
110
111   -- Rolling Total of Layoffs Per Month
112 • SELECT SUBSTRING(date,1,7) as dates, SUM(total_laid_off) AS total_laid_off
113   FROM layoffs_staging2
114   GROUP BY dates
115   ORDER BY dates ASC;
```

# Get the Github link

This project aims to explore insights into **layoffs data from 3,626 entries**, focusing on total layoffs, percentage laid off, and funds raised. The data source comes from **Kaggle**, with information available from the declaration of COVID-19 as a pandemic on **March 11, 2020, to May 30, 2024**. The dataset includes **startup layoffs reported on Layoffs.fyi since COVID-19**.

Conducting comprehensive data cleaning using MySQL involved identifying and removing duplicates, standardizing data, correcting errors, addressing null values, and eliminating unnecessary columns and rows. New tables were created to streamline the dataset and ensure consistency and accuracy for analysis.

**Exploratory Data Analysis (EDA)** was performed to explore the dataset, identify trends, uncover patterns, and analyze key metrics such as total layoffs, percentage laid off, and funds raised. **Advanced SQL queries** were executed to determine companies with the **highest layoffs by year**, **calculate rolling totals** of layoffs per month, and group and summarize data by various dimensions, including company, location, industry, stage, and country.

Outliers and significant events were identified, providing insights into the impact of layoffs across different sectors and time periods.

By breaking it down into these aspects, **this project aims to provide a comprehensive picture of the layoffs data and offer strategic insights for understanding and addressing the impact of layoffs across various industries and time periods.**

# COFFEE SHOP SALES

## Capstone Project



09

```

10
11 # % Revenue of Generated
12 -- Step 1
13 SELECT SUM(transaction_qty * unit_price) as Total_Revenue,
14 | store_location
15 FROM `first-a0.kopi.kopii`
16 GROUP BY 2
17
18 -- Step 2 REVENUE GENERATED BY STORE LOCATION
19 SELECT ((SUM(transaction_qty * unit_price) / 1006.83) * 100) as Revenue_Generated,
20 | store_location
21 FROM `first-a0.kopi.kopii`
22 GROUP BY 2
23
24 -- Step 3
25 SELECT SUM((total_revenue / 1006.83) * 100) as Revenue_Generated
26 FROM (
27 | | SELECT SUM(transaction_qty * unit_price) as total_revenue
28 | | FROM `first-a0.kopi.kopii`
29 ) as subquery
30
31 #Transaction by Each Day
32 SELECT SUM(transaction_qty * unit_price) as Total_Revenue,
33 | EXTRACT(DAYOFWEEK FROM transaction_date) Day,
34 | store_location
35 FROM `first-a0.kopi.kopii`
36 GROUP BY 2

```



[ ]	136443	5	Lower Manhattan	64	0.80	Flavours
→	115939	3	Astoria	37	3.00	Coffee
→	90364	8	Hell's Kitchen	25	2.20	Coffee
→	110972	3	Astoria	56	2.55	Tea
	44171	3	Astoria	63	0.80	Flavours
				product_type	product_detail	
	116605	Barista Espresso		Latte		
	119259	Scone		Jumbo Savory Scone		
	105265	Gourmet brewed coffee		Ethiopia Rg		
	29485	Barista Espresso		Latte		
	126885	Premium brewed coffee		Jamaican Coffee River Lg		
	...	...		...		
	136443	Regular syrup		Hazelnut syrup		
	115939	Barista Espresso		Espresso shot		
	90364	Organic brewed coffee		Brazilian Sm		
	110972	Brewed Chai tea		Spicy Eye Opener Chai Rg		
	44171	Regular syrup		Carmel syrup		
						[200 rows x 11 columns]

```

[ ] sampled_data.to_csv('sampled_data.csv', index=False)
print("Data yang telah di-sampling disimpan ke 'sampled_data.csv'")
→ Data yang telah di-sampling disimpan ke 'sampled_data.csv'

```



# REACH ME OUT



[Alif Abdul Hakim](#)



[alifabdulhakim.ah@gmail.com](mailto:alifabdulhakim.ah@gmail.com)



[alversonn](#)