

/Projek DOS

Workshop

Building ETL using Apache Hop

Visual Data Engineering Made Simple



Table of contents

01 Introduction

02 ETL

01

Introduction

Introduction

Projek Freedom Open Source

Founded in 2025, This Training Center is a **project-based, real world scenario** training environments initiative focused on education in open-source technology. The goal is to empower individuals, university, and corporate with knowledge about tools and technologies such as :

- Business Intelligence
- Data Warehouse
- Data Lake
- Big Data
- Blockchain
- Artificial Intelligence
- More



Open Partnership

University



Integrating practical, hands-on training into workshops, and certification programs for students.

Corporate



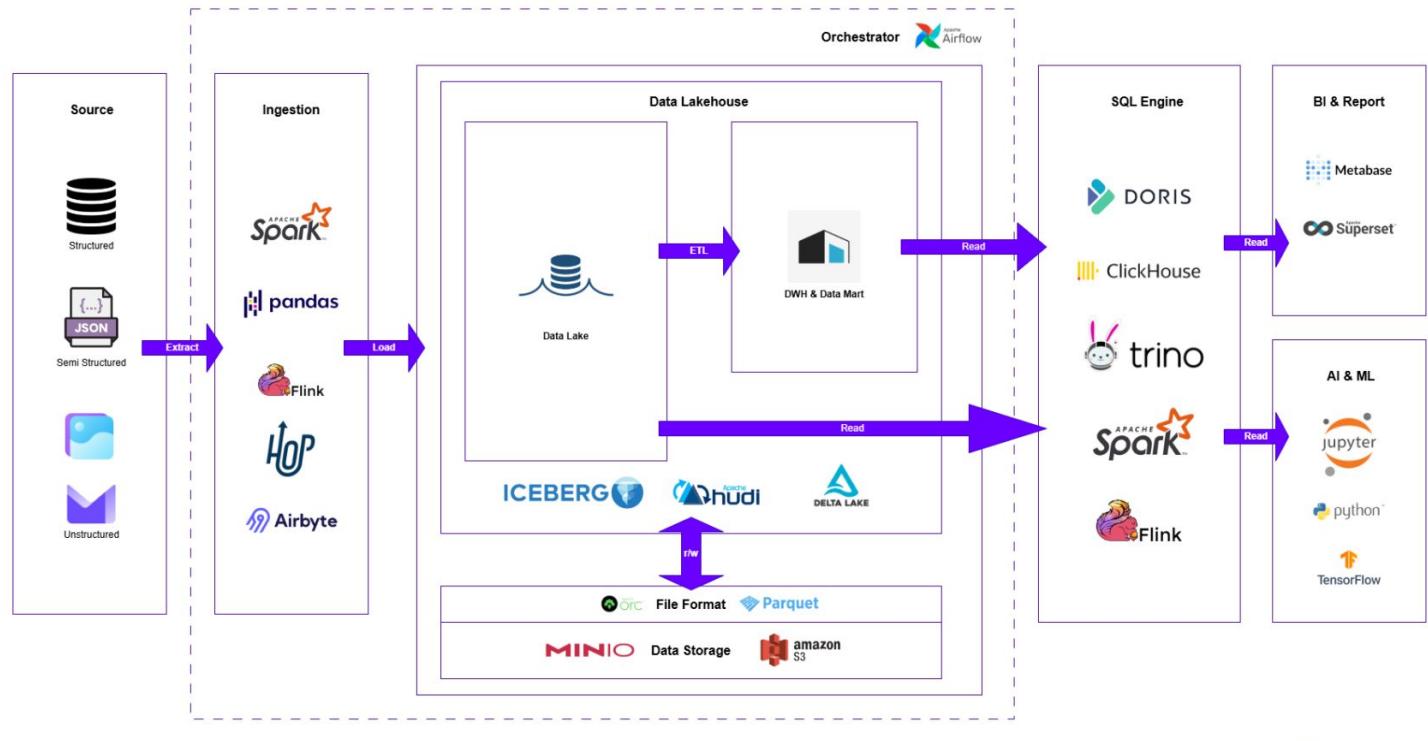
providing tailored training programs to upskill employees, and align talent with industry needs—talent acquisition partners

Government



Supporting national and regional initiatives through specialized training programs, seminars, and collaborative education projects

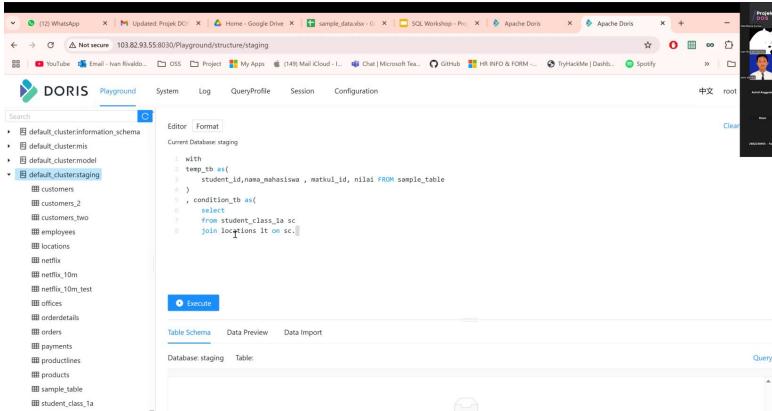
End to End - Project Based Training



Roadmap (2025)

Initiative	Objective	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
SQL	Mastering data querying, manipulation, and optimization for efficient database management									
ETL	Extracting, transforming, and loading data for seamless integration and processing									
Orchestrator	Managing and coordinating end-to-end data workflows automatically									
Business Intelligence	Leveraging analytics and visualization tools to drive data-driven decisions									
Machine Learning	Building predictive models and AI solutions for advanced data analytics.									
Other	Exploring more essential technologies									

Previous Workshop - Learn SQL (26, April 2025)



The screenshot shows the Apache Doris Playground interface. On the left, there's a sidebar with a tree view of databases and tables. The main area contains a code editor with the following SQL query:

```
with
    temp_tb as(
        select student_id, name_mahasiswa , matkul_id, nilai FROM sample_table
    )
, condition_tb as(
        select
            from student_class_1a
        join locations lt on sc.||
```

Below the code editor are tabs for "Table Schema", "Data Preview", and "Data Import". At the bottom, there's a "Query" section with a "Database: staging" dropdown and a "Table: sample_table" dropdown.

In this workshop, we explored key concepts of Query and hands-on SQL examples, including:

- String Manipulation
- Subquery
- CTE (Common Table Expression)
- Windowing Function



Projek DOS



Workshop Certificate

Projek Freedoom OpenSource certify that

Joe Doe

has successfully completed

Learn SQL Query using Apache Doris

Awarded for completing the "Learn SQL using Apache Doris" workshop. This program covered SQL fundamentals and analytics using Apache Doris, equipping participants with practical skills for querying and analyzing data.

Completion Date: April 26, 2025

Expiration Date: April 26, 2026

DNAStudio DORIS

Wandhana Kurnia
Founder

Nomor ID: 12345678901234567890
Certificate ID: 12345678901234567890

02

ETL

About Me - Trainer

Firman Fakhri Mukti

ETL & BI Developer with 8+ years of experience in Banking and Insurance. Expert in building ETL pipelines, data warehouses, and BI dashboards to support data-driven decision-making



Firman Fakhri Mukti



ETL Careers Path

- **ETL Developer** (Other Skills: SQL, Store Procedure)
- **Data Engineering** (Other Skills: Python, Orchestrator)
- **Data Architect** (Other Skills: Data Governance)

Demand IT Workers will continue to increase in the coming years. ([kemnaker](#))

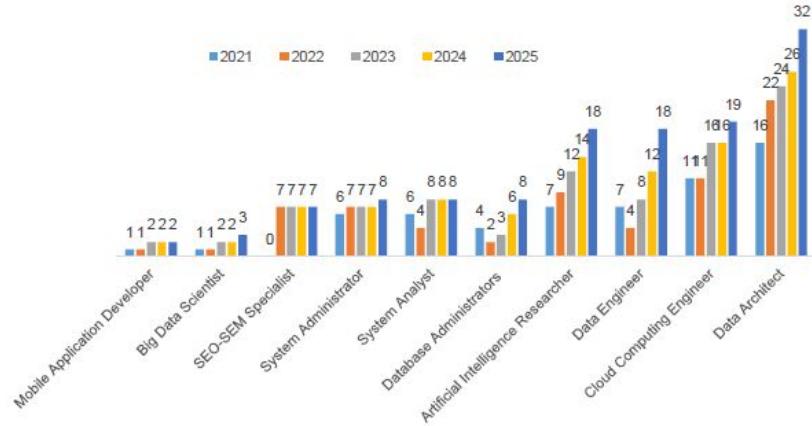


Image Credit: [kemnaker](#)

Salary Range



With **5 years of hands-on experience** in the data field, IT Consultants in Indonesia are seeing competitive compensation. The average monthly salary typically falls between **IDR 17,000,000 to IDR 20,000,000**, depending on the industry, technical skill set, and company size.

(Source: persolkelly - Salary Guide 2024)

INFORMATION TECHNOLOGY					
Data Center Service Operation (DCSO) Specialist	D3	1	18,000,000	21,000,000	
Data Center Service Provisioning (DCSP) Manager	S1	8	45,000,000	50,000,000	
Data Center Technical Support (DCTS) Lead	D3/S1	5	35,000,000	40,000,000	
Data Center Technical Support (DCTS) Manager	S1	8	45,000,000	50,000,000	
Developer Community Manager	S1	3-5	33,000,000	36,000,000	
Education Adoption Lead	S2	5-10	63,000,000	68,000,000	
Enterprise Account Manager	S2	5-10	50,000,000	55,000,000	
ESG ESG Senior Manager	S1	10	60,000,000	70,000,000	
Field Data Collection Staff	High School		5,000,000	6,000,000	
Finance Director	S1	15	85,000,000	95,000,000	
Firmware Engineer	S1	1	10,000,000	15,000,000	
Firmware Engineer Leader	S1	2	20,000,000	25,000,000	
Fullstack Software Developer	S1	5	12,000,000	16,000,000	
Education Adoption Specialist	S1	3-5	28,000,000	31,000,000	
Head of End User Services	S1	>10	50,000,000	55,000,000	
Head of HR	S1	10	30,000,000	35,000,000	
HR Business Partner	S1	1-3	8,000,000	12,000,000	
iOS Developer	S1	2-3	9,000,000	12,000,000	
IT Consultant	S1	5	17,000,000	20,000,000	
IT Developer Lead	S1	2-4	13,000,000	15,000,000	
IT Manager	S1	10	45,000,000	50,000,000	

Image Credit: persolkelly

Training Material

- What is ETL
- Most Popular ETL
- Why ETL is Important
- Who used ETL
- Pipeline Anatomy
- Multisource Integration
- Data Manipulation
- Data Load



What is ETL?

Definition:

ETL stands for **Extract, Transform, Load**. It is a data integration process that combines data from multiple sources into a central repository, such as a data warehouse. The process involves:

1. **Extracting** data from various sources like databases, APIs, or files.
2. **Transforming** the data by cleaning, formatting, and applying business rules to make it suitable for analysis.
3. **Loading** the transformed data into a target system for further use.

This process is fundamental in preparing data for analytics, reporting, and decision-making.

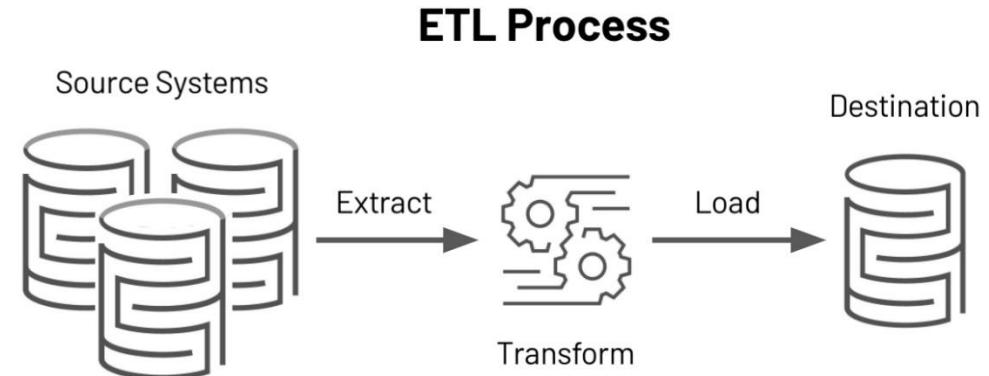


Image Credit: [Extract Transform Load \(ETL\) – Databricks](#)

Most Popular ETL

ETL tools are well-established in the data industry. Some of the most popular and mature solutions include:

- **SQL Server Integration Services (SSIS)** – A Microsoft tool for building enterprise-level data integration and workflow solutions.
- **IBM InfoSphere DataStage** – An enterprise-grade ETL platform from IBM, designed for large-scale and complex data environments.
- **Apache Hop** – An open-source ETL tool developed by the Apache Software Foundation.
- **Talend Data Integration** – A widely used ETL tool for enterprise versions, known for its flexibility and wide range of connectors.
- **Pentaho Data Integration (PDI)** – An Enterprise ETL solution with a graphical interface, also known as Kettle, used for batch and real-time data processing.



Why ETL is important

1. Combines Data from Many Sources

ETL helps you bring together data from different places — like Excel files, databases, or APIs — into one single storage. Think of it like collecting puzzle pieces from different boxes and putting them together into one clear picture.

2. Cleans and Organizes Messy Data

Raw data is often messy or incomplete. The “Transform” step in ETL fixes the data — making it clean, consistent, and ready for use.

3. Makes Reports More Accurate

Clean and complete data means better dashboards and reports — so teams can make smarter, data-based decisions.

4. Saves Time and Effort

ETL processes can run automatically on a schedule. That means no more manual data cleaning every day or week.



Who uses ETL?

Data Engineers

They design and build ETL pipelines to move and prepare data for use.

Data Analysts

They use the cleaned and organized data from ETL to create reports, dashboards, and insights.

Business Intelligence (BI) Teams

BI professionals rely on ETL to feed accurate data into analytics tools like Power BI or Tableau.

Data Scientists

They need structured and clean data (often from ETL) to train machine learning models.

Business Users & Decision Makers

They don't build ETL – but benefit from it! Thanks to ETL, they can view dashboards and reports to support decisions.

Companies in All Industries

Retail, banking, healthcare, startups, government – all use ETL to understand their data better.



Data Engineer



Data Analyst



BI Teams

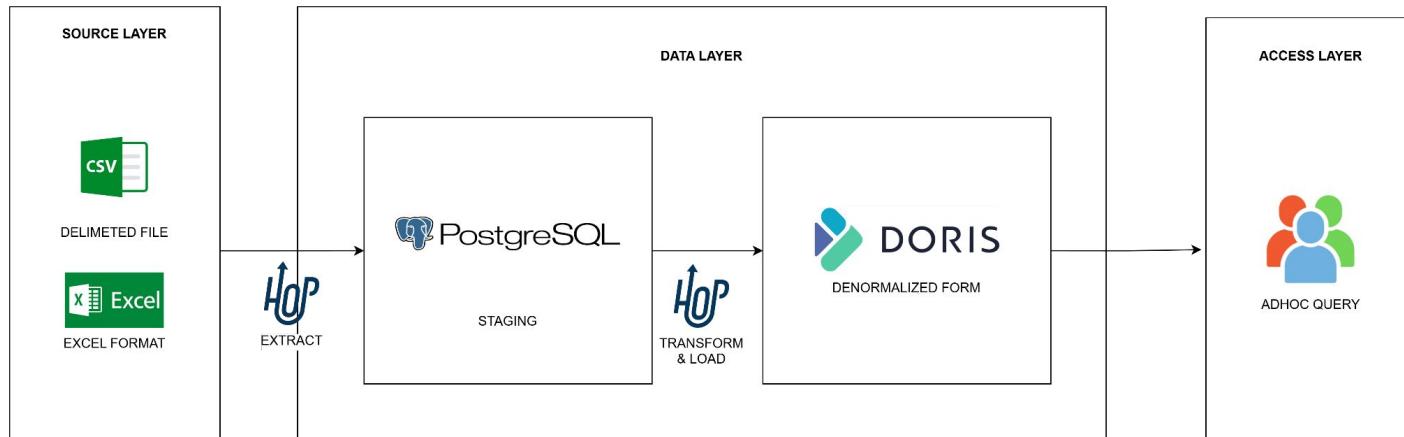


Data Scientists



Business Users & Decision Makers

Pipeline Anatomy



Tech Stack (ETL)

Apache Hop, short for **Hop Orchestration Platform**, is a data orchestration and data engineering platform that aims to facilitate all aspects of data and metadata orchestration

Hop is an entirely new **open source** data integration platform that is easy to use, fast and flexible



Source :
<https://hop.apache.org/manual/latest/getting-started/hop-what-is-hop.html>

Tech Stack (Database)

PostgreSql, PostgreSQL is a powerful, open source object-relational database system that uses and extends the SQL language combined with many features that safely store and scale the most complicated data workloads.



Tech Stack (OLAP)

Apache Doris , Apache Doris is an MPP-based data warehouse known for its high query speed. For queries on large datasets, it returns results in sub-seconds

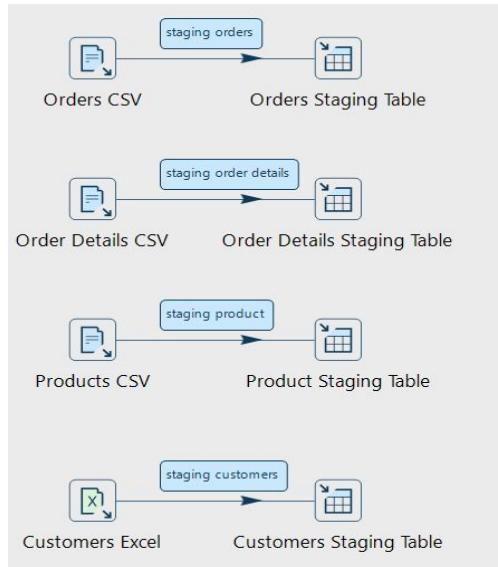


DORIS

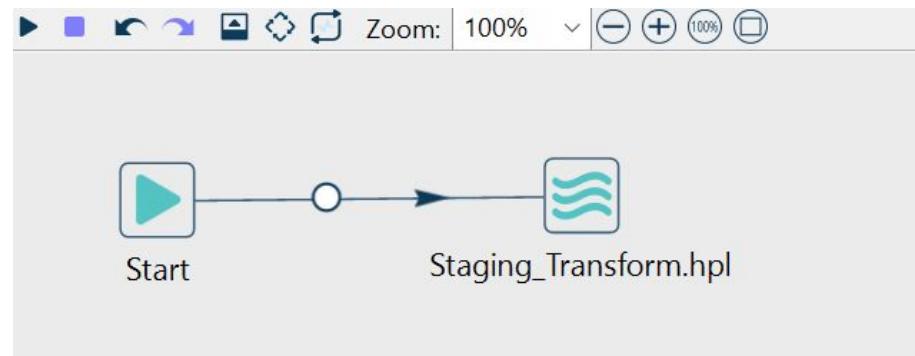
Source :
<https://doris.apache.org/docs/gettingStarted/what-is-apache-doris>

Pipelines Vs Workflows

Pipelines are Hop's work horse: read from sources, write to targets and perform just about any manipulation on your data through hundreds of **transforms**.



Workflows are Hop's tool to orchestrate workflows and pipelines, perform environment validations, error handling and much more with the available **actions**

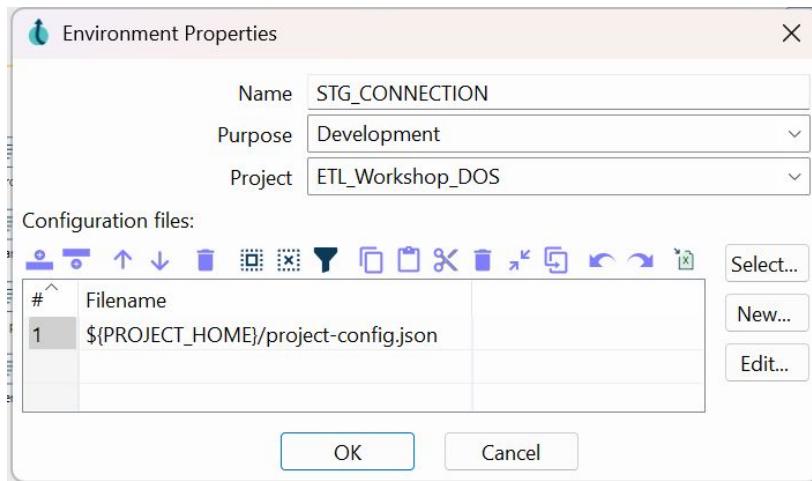


Project Setup

Project Properties

Name	ETL_Workshop_DOS	X	
Home folder	C:\FirmanFakhri\ETL_Workshop_DOS	<input type="button" value="Browse..."/>	
Configuration file (relative path)	project-config.json	<input type="button" value="Browse..."/>	
Parent project to inherit from		<input type="button"/>	
Description			
Company			
Department			
Metadata base folder (HOP_METADATA_FOLDER)	\${PROJECT_HOME}/metadata		
Unit tests base path (HOP_UNIT_TESTS_FOLDER)	\${PROJECT_HOME}		
Data Sets CSV Folder (HOP_DATASETS_FOLDER)	\${PROJECT_HOME}/datasets		
Enforce executions in project home?	<input checked="" type="checkbox"/>		
Project variables to set :			
#	Name	Value	Description (optional information)
1			

Project Environment Variable



Hop described variables dialog

Editing configuration file: C:\FirmanFakhri\ETL_Workshop_DOS/project-config.json

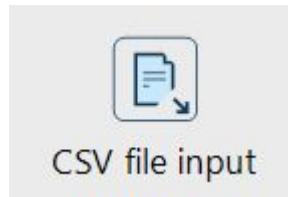
#	Variable name	Value	Description
1	STAGING_SERVER	localhost	
2	STAGING_DATABASE	postgres	
3	STAGING_PORT	5432	
4	STAGING_USERNAME	postgres	

Metadata Connection

The screenshot shows a software interface for managing metadata connections. On the left is a sidebar with a tree view of connection types:

- Asynchronous Web Service
- Azure Authentication
- Beam File Definition
- Cassandra Connection
- Data Set
- Relational Database Connection
 - local_postgresql
- Execution Data Profile
- Execution Information Location
- Neo4j Graph Model
- Hop Server
- MongoDB Connection
- Neo4j Connection
- Partition Schema
- Pipeline Log
- Pipeline Probe
- Pipeline Run Configuration
 - local
 - Pipeline Unit Test
 - Rest Connection
 - Static Schema Definition
 - Splunk Connection
 - Variable Resolver
 - Web Service
 - Workflow Log
- Workflow Run Configuration
 - local

Transform (CSV File Input)



CSV file input

Transform name: Products CSV

Filename: \${PROJECT_HOME}/files/20051231.products.csv

Delimiter:

Enclosure: "

NIO buffer size: 50000

Lazy conversion?

Header row present?

Add filename to result

The row number field name (optional)

Running in parallel?

New line possible in fields?

File encoding:

Schema definition

# ^	Name	Type	Format	Length	Precision	Currency	Decimal	Group	Trim type
1	productCode	String		168		\$.	,	none
2	productName	String		43		\$.	,	none
3	productLine	String		16		\$.	,	none
4	productScale	String		6		\$.	,	none
5	productVendor	String		25		\$.	,	none
6	productDescription	String		391		\$.	,	none
7	quantityInStock	Integer	#	15	0	\$.	,	none
8	buyPrice	Number	#.#	15	1	\$.	,	none
9	MSRP	Number	#.#	15	1	\$.	,	none

Transform (Table Output)

Table output

Table output

Transform name: Product Staging Table
Connection: local_postgresql
Target schema: staging
Target table: products
Commit size: 1000
Truncate table: Truncate on first row: Ignore insert errors: Specify database fields:

Main options Database fields

Fields to insert:

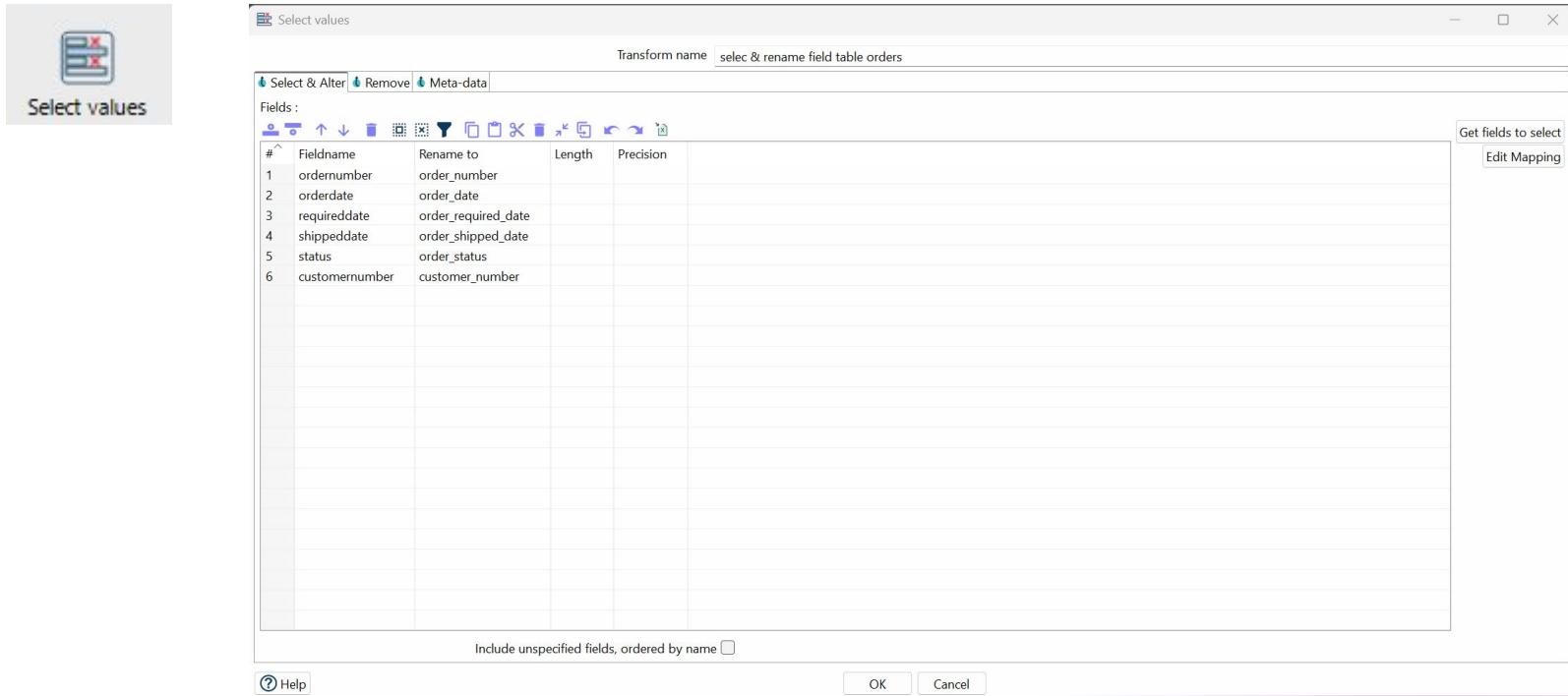
#	Table field	Stream field
1	productCode	productCode
2	productName	productName
3	productLine	productLine
4	productScale	productScale
5	productVendor	productVendor
6	productDescription	productDescription
7	quantityInStock	quantityInStock
8	buyPrice	buyPrice
9	MSRP	MSRP

Get fields Enter field mapping

Mapping Table

No.	Column	Source		
		Table	Column	Business Rule
1	order_date	orders	orderDate	
2	order_required_date	orders	requiredDate	
3	order_shipped_date	orders	shippedDate	
4	order_number	orders	orderNumber	
5	order_status	orders	status	
6	order_quantity	orderDetails	quantityOrdered	join orders to order detail on ordernumber = ordernumber
7	order_price_each	orderDetails	priceEach	case when currency USD then priceEach * 1 else when currency IDR then priceEach / 16500 end
8	order_amount	orderDetails	order_amount	quantityOrdered * priceEach
9	order_line_number	orderDetails	order_line_number	
10	product_code	products	productCode	join orders to order detail on ordernumber = ordernumber --> join order detail to product on productcode = productcode
11	product_name	products	productName	cleansing special character (space,#,\$)
12	product_scale	products	productScale	
13	product_description	products	productDescription	
14	product_category	products	productLines	
15	customer_number	customers	customerNumber	join orders to customers on customernumber = customernumber
16	customer_name	customers	customerName	
17	customer_city	customers	city	
18	customer_state	customers	state	
19	customer_country	customers	country	

Transform (Select Values)



The screenshot shows the 'Select values' dialog box, which is part of a larger application interface. On the left, there is a small icon labeled 'Select values' with a blue square containing three white 'X' marks. The main window has a title bar 'Select values' and a sub-title 'Transform name: selec & rename field table orders'. Below the title bar is a toolbar with three buttons: 'Select & Alter' (highlighted in green), 'Remove' (blue), and 'Meta-data' (green). A 'Fields' section follows, containing a table with six rows of field mappings:

#	Fieldname	Rename to	Length	Precision
1	ordernumber	order_number		
2	orderdate	order_date		
3	requireddate	order_required_date		
4	shippeddate	order_shipped_date		
5	status	order_status		
6	customernumber	customer_number		

On the right side of the dialog, there are two buttons: 'Get fields to select' and 'Edit Mapping'. At the bottom, there is a checkbox 'Include unspecified fields, ordered by name' and standard 'OK' and 'Cancel' buttons.

Transform (String Operations)



AB String operations

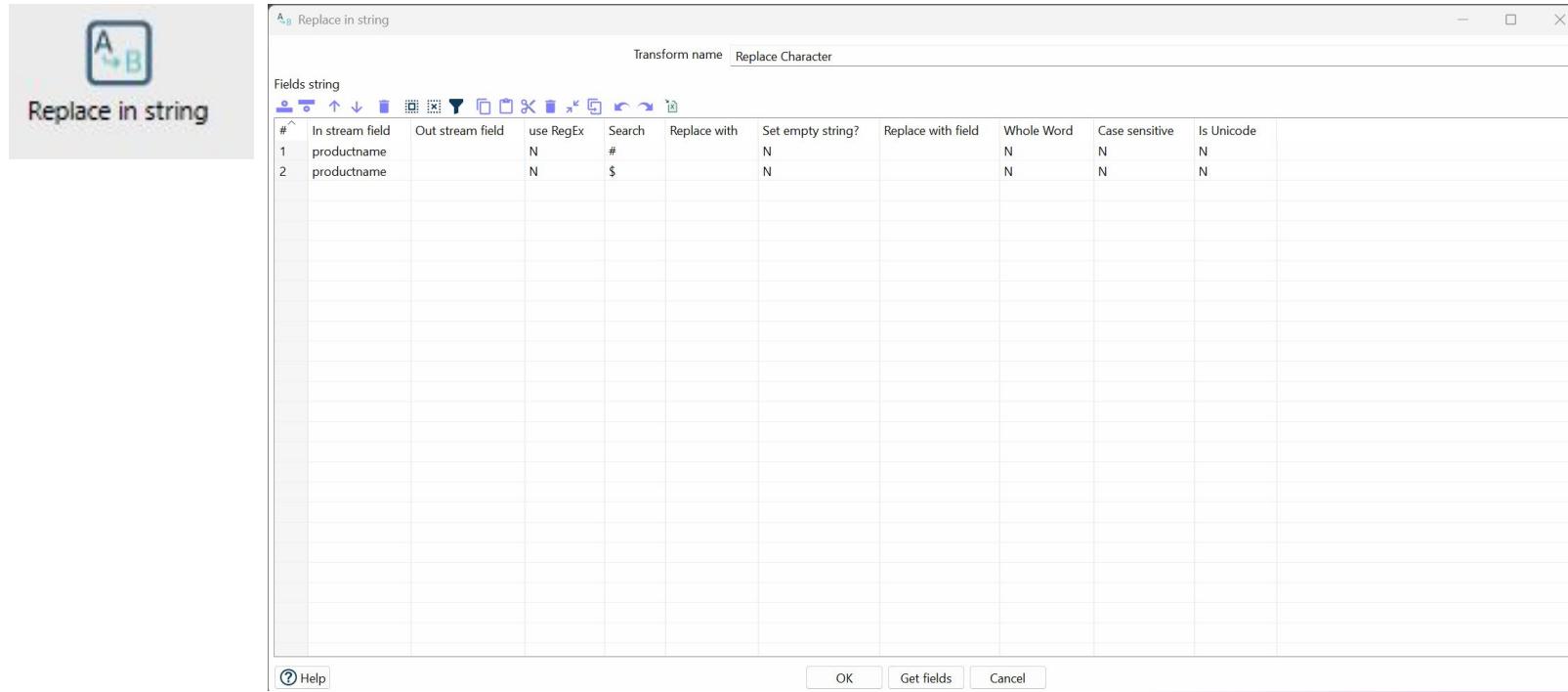
Transform name: trim product name field

The fields to process:

#	In stream field	Out stream field	Trim type	Lower/Upper	Padding	Pad char	Pad Length	InitCap	Escape	Digits	Remove Special character
1	productname		both	none	none			N	None	none	none

OK Get fields Cancel

Transform (Replace in String)



The screenshot shows the 'Replace in string' transform configuration dialog. The title bar reads 'A B Replace in string'. The transform name is 'Replace Character'. The interface includes a toolbar with various icons and a table for defining the replace operation.

Fields string

#	In stream field	Out stream field	use RegEx	Search	Replace with	Set empty string?	Replace with field	Whole Word	Case sensitive	Is Unicode
1	productname		N	#		N		N	N	N
2	productname		N	\$		N		N	N	N

Buttons at the bottom include 'Help', 'OK', 'Get fields', and 'Cancel'.

Transform (Filter rows)

The screenshot shows the configuration dialog for the 'Filter rows' transform. On the left, there is a preview icon with a blue square containing a white 'E' and a yellow funnel, labeled 'Filter rows'. The main window has a title bar 'Filter rows' and several configuration fields:

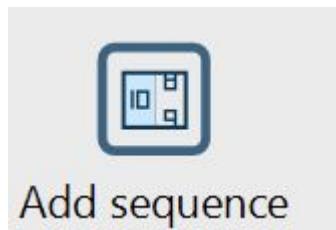
- Transform name:** Filter rows
- Send 'true' data to transform:** Add Group ID
- Send 'false' data to transform:** Select values no split rows

Below these fields is a section titled 'The condition:' with a plus sign (+) button. The condition is defined as:

productcode CONTAINS [] ; [] (String)

At the bottom of the dialog are 'Help', 'OK', and 'Cancel' buttons.

Transform (Add sequence)



Add sequence

Transform name

Name of value

Use a database to generate the sequence

Use DB to get sequence

Connection

Schema name

Sequence name

Use a pipeline counter to generate the sequence

Use counter to calculate sequence

Counter name (optional)

Start at value

Increment by

Maximum value

Help OK Cancel

Transform (Split Field to Rows)



Split field to rows

Transform name: Split field product code

Field to split: productcode

Delimiter: ;

Delimiter is a Regular:

New field name: split_product_code

Additional fields

Include rownum in output: Rownum fieldname: row_num

Reset Rownum at each input row

Transform (Sort Rows)

The screenshot shows the 'Sort rows' configuration dialog. On the left, there is a preview icon with a downward arrow and the text 'Sort rows'. The main dialog has the following fields:

- Transform name: Sort rows product code
- Sort directory: \${java.io.tmpdir}
- TMP-file prefix: out
- Sort size (rows in memory): 1000000
- Free memory threshold (in %):
- Compress TMP files
- Only pass unique rows (verifies keys only)

Below these fields is a 'Fields:' section containing a table:

# [^]	Fieldname	Ascending	Case sensitive compare	Sort based on current locale	Collator Strength	Presorted
1	ordernumber	Y	N	N	0	N

At the bottom of the dialog are buttons for Help, OK, Get Fields, and Cancel.

Transform (Merge join)



Merge join

Transform name: Merge join

First transform: Sort rows product code

Second transform: Sort rows quantityordered

Join Type: INNER

Keys for 1st transform:

#	Key field
1	group_id
2	row_num

Keys for 2nd transform:

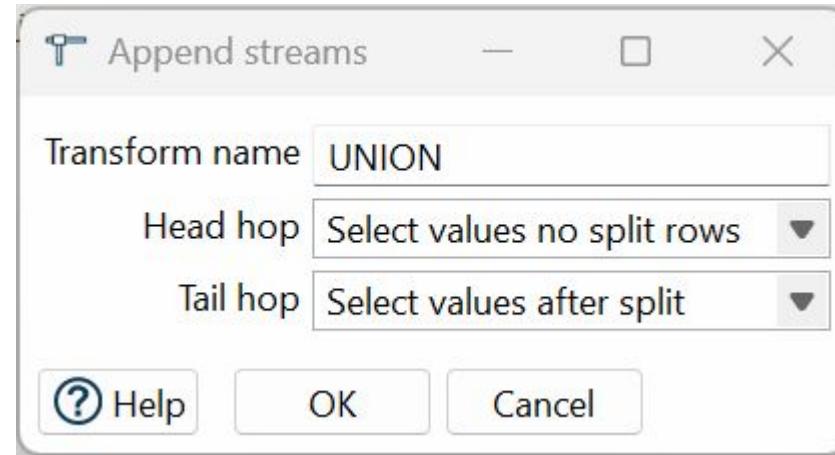
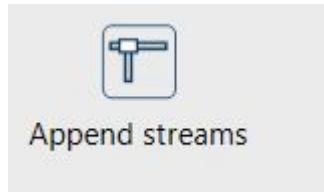
#	Key field
1	group_id
2	row_num

Get key fields

Get key fields

Help OK Cancel

Transform (Append Stream)

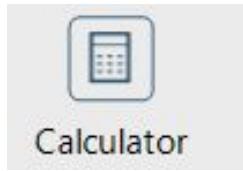


Transform (Formula)

The screenshot shows a 'Formula' dialog box with the following details:

- Formula name:** Currency Conditional
- Fields:** A toolbar with various icons for managing fields.
- Table:** A grid for defining new fields. The first row contains:
 - #[^]
 - New field
 - order_price_each_update
 - Formula
 - IF([currency] = "USD", [order_price_each]*1, IF([currency] = "IDR", [order_price_each]/16500, NA()))
 - Value type: Number
 - Length: (empty)
 - Precision: (empty)
 - Replace value: (empty)
 - Set Null to #N/A: N
- Buttons at the bottom:** Help, OK, Cancel.

Transform (Calculator)



Calculator

Transform name: calculate order amount

Throw an error on non existing files

Fields:

#	New field	Calculation	Field A	Field B	Field C	Value type	Length	Precision	Remove	Conversion mask	Decimal symbol	Grouping symbol	Currency syr
1	order_amount	A * B	order_price_each_update	order_quantity		Number			N				

OK Cancel

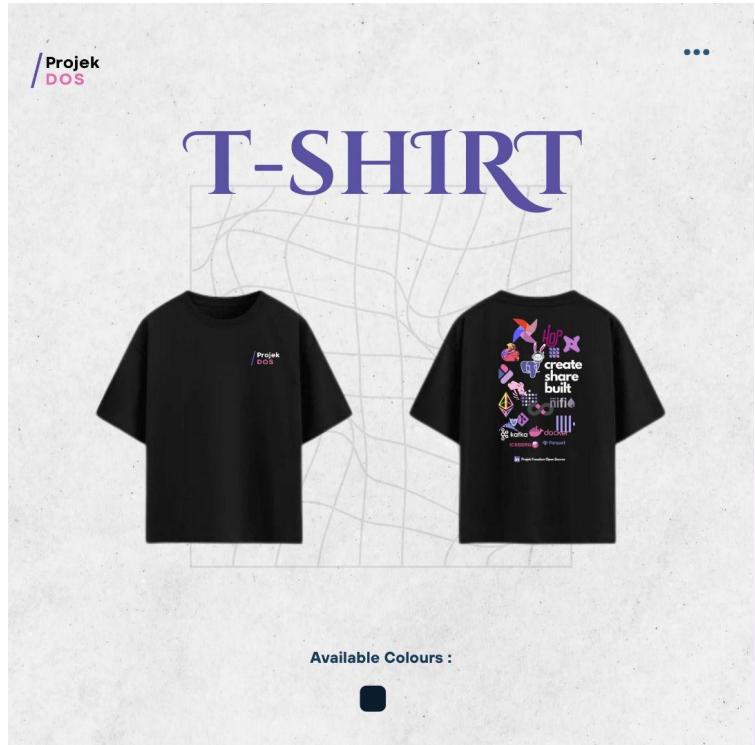
Demo

Quiz



- URL: <https://github.com/projekdos>
- Repository: etl_workshop_apache_hop_batch1
- Read the Instruction
- Solve the Problem

Please submit the result to info@projekdos.com. The winner will get Free Merchandise from us!



What's next

Building Data Pipeline using Apache Airflow & Apache Doris – FREE (June 2025)

Understand how data orchestrators like Apache Airflow manage workflows from data sources to your target systems. Perfect for beginners and aspiring data engineers.

What You'll Learn:

- ◆ What is data orchestration and why it matters
- ◆ Introduction to Apache Airflow & DAG concepts
- ◆ Hands-on: build and schedule your first data pipeline
- ◆ Simple integration using Python, SQL & Apache Doris

Projek DOS

BUILDING DATA PIPELINE
USING APACHE AIRFLOW & APACHE DORIS

FREE

Apache Airflow DORIS

JUNE 2025

LIMITED SEAT

FOR MORE INFORMATION VISIT OUR LINKEDIN

in PROJEK FREEDOM OPEN SOURCE

Thanks!

Business Contact:

 info@projekdos.com

 +6281385368844 (Whatsapp)



Whatsapp Community



LinkedIn



Youtube Channel

