



## Snowflake Hackathon 2025

Logistics Maritime Use Case - OceanLink Logistics

PT Password Solusi Sistem  
October 2025

## 1. Downloading Dataset Files

### Download dan Unzip Inspection Document

1. Untuk langkah berikutnya, kita perlu mengunduh **3 folder** terpisah untuk **Dataset, Photos, dan Worksheet SQL, PPT & Guideline PDF**

Silakan gunakan tombol di bawah ini untuk mengakses **Repository Github - Password Snowflake Hackathon**

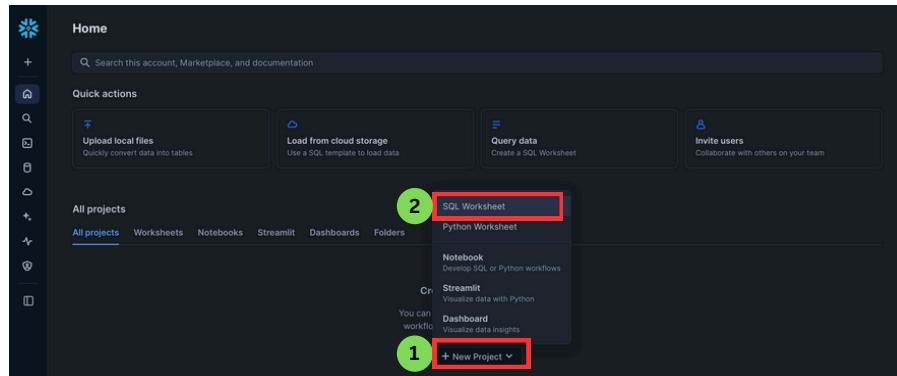


## 2. Setup the Snowflake Environment!

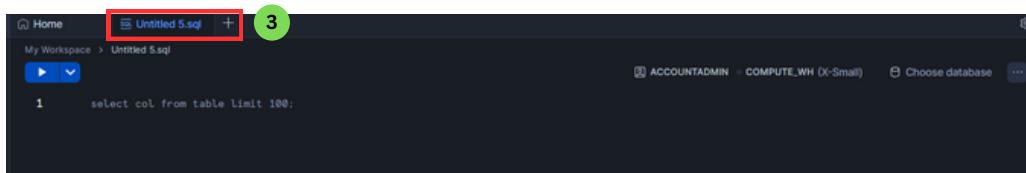
Ayo buat worksheet baru!

Pada tampilan **Snowsight**

1. Click pada tombol **+ New Project** untuk membuka opsi pembuatan **SQL Worksheet**
2. Click pada tombol **SQL Worksheet** untuk membuat **SQL Worksheet** baru



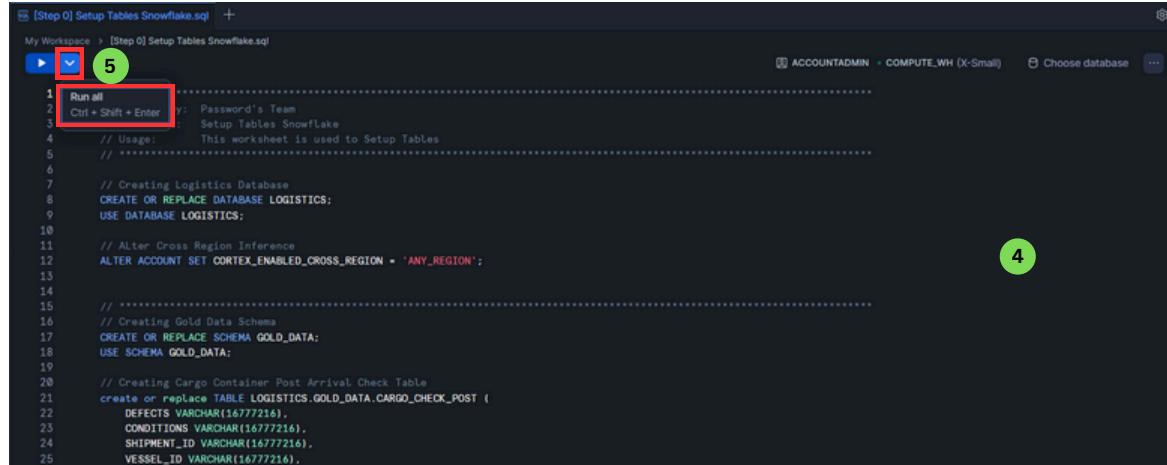
3. Klik dua kali pada judul dari **Worksheet** yang baru buat untuk mengganti namanya menjadi “[Step 0] Setup Tables Snowflake”



4. Copy dan Paste Query SQL dari **[Step 0] Setup Tables Snowflake.sql** di github repo ke **Worksheet** “[**Step 0] Setup Tables Snowflake**”

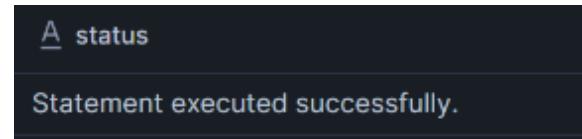
5. Run All di **Worksheet** yang sudah terisi dengan Query SQL, dengan **Klik Tombol Panah ke bawah & Klik Tombol Run All**

Query SQL ini mempersiapkan environment Snowflake untuk Tables, Stages, dan Grant Privileges



```
1 Run all
2 Ctrl + Shift + Enter y: Password's Team
3 : Setup Tables Snowflake
4 // Usage: This worksheet is used to Setup Tables
5 // ****
6
7 // Creating Logistics Database
8 CREATE OR REPLACE DATABASE LOGISTICS;
9 USE DATABASE LOGISTICS;
10
11 // ALTER Cross Region Inference
12 ALTER ACCOUNT SET CORTEX_ENABLED_CROSS_REGION = 'ANY_REGION';
13
14
15 // ****
16 // Creating Gold Data Schema
17 CREATE OR REPLACE SCHEMA GOLD_DATA;
18 USE SCHEMA GOLD_DATA;
19
20 // Creating Cargo Container Post Arrival Check Table
21 create or replace TABLE LOGISTICS.GOLD_DATA.CARGO_CHECK_POST (
22 DEFECTS VARCHAR(16777216),
23 CONDITIONS VARCHAR(16777216),
24 SHIPMENT_ID VARCHAR(16777216),
25 VESSEL_ID VARCHAR(16777216).
```

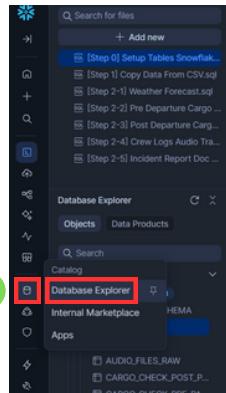
6. Akan muncul status seperti ini yang menunjukkan kalau **semua Query SQL sudah berhasil dijalankan**



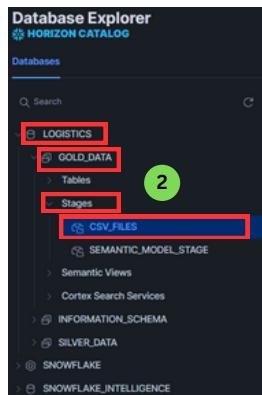
### 3. Upload Dataset to Snowflake

Saatnya melakukan loading data ke Snowflake

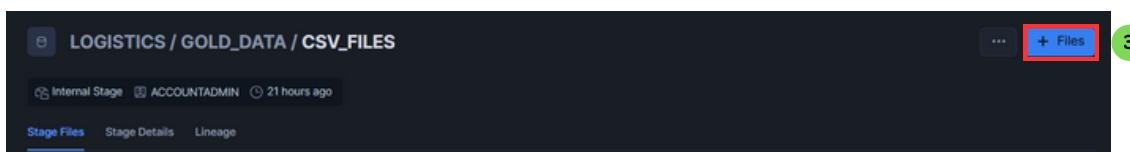
1. Klik pada tombol database & Database Explorer untuk membuka Tab Database



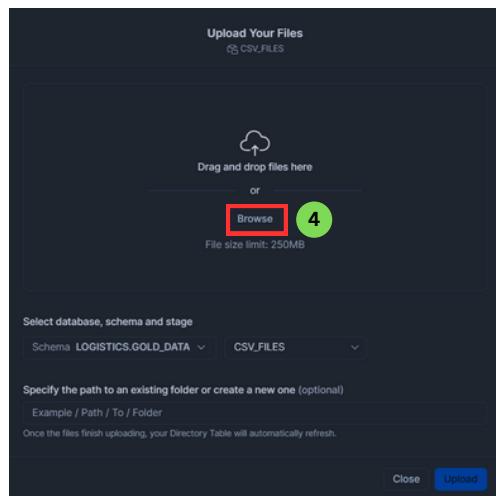
2. Klik pada Tab Database: Database LOGISTICS → Schema GOLD\_DATA → Stages → CSV\_FILES



3. Klik tombol + Files untuk menambahkan files ke dalam Stages CSV\_FILES

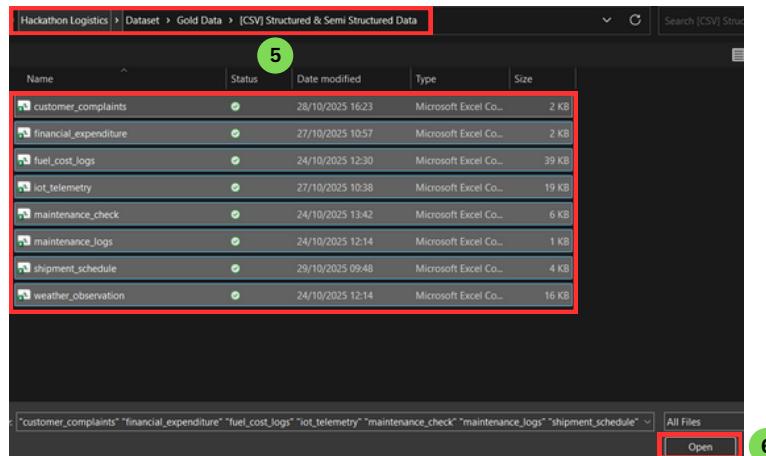


4. Klik Tombol Browse untuk memilih Files yang akan diupload



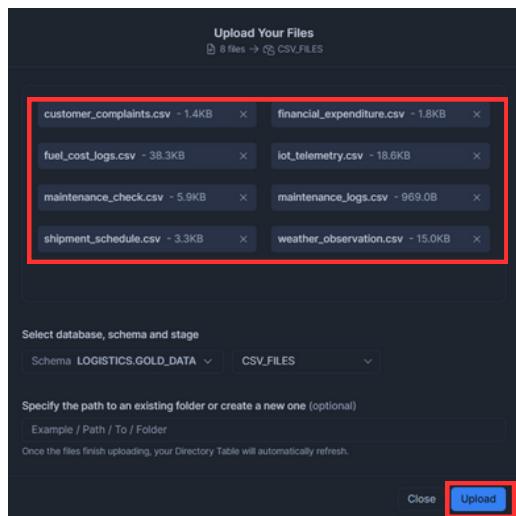
**5. Blok semua File CSV yang ada di dalam Folder**  
Hackathon Logistics → Dataset → Gold Data → [CSV] Structured & Semi Structured Data

6. Klik tombol Open untuk finalize file yang ingin diupload



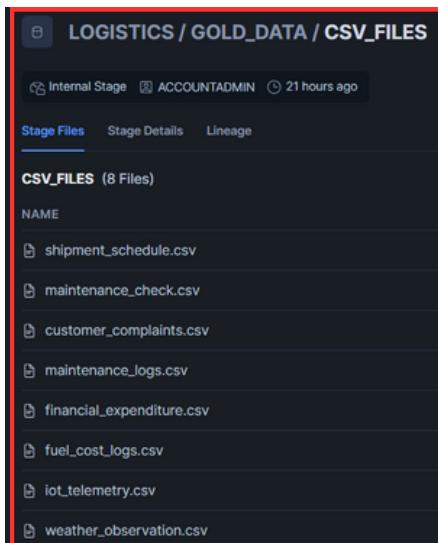
The screenshot shows a file browser interface with a list of CSV files. The files listed are: customer\_complaints, financial\_expenditure, fuel\_cost\_logs, iot\_telemetry, maintenance\_check, maintenance\_logs, shipment\_schedule, and weather\_observation. Each file entry includes a small icon, the file name, a status indicator, the date modified, the type (Microsoft Excel Co...), and the size. A red box highlights the entire list of files. A green circle with the number '5' is located above the list. At the bottom right of the browser, there is a search bar and a dropdown menu, followed by an 'All Files' button and an 'Open' button. A green circle with the number '6' is positioned to the right of the 'Open' button.

7. Pastikan ada **8 Files** yang akan diupload pada step ini, dan **Klik Upload** untuk **finalize proses Upload File ke Stages**



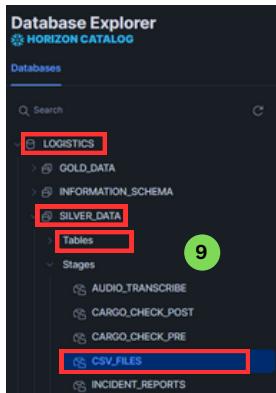
The screenshot shows a modal dialog titled 'Upload Your Files'. It displays a list of eight CSV files that have been selected for upload: customer\_complaints.csv, financial\_expenditure.csv, fuel\_cost\_logs.csv, iot\_telemetry.csv, maintenance\_check.csv, maintenance\_logs.csv, shipment\_schedule.csv, and weather\_observation.csv. Each file entry shows its name, size (e.g., 1.4KB, 1.8KB, etc.), and a close button. Below the file list, there are fields for selecting a database, schema, and stage, and for specifying a folder path. At the bottom right of the dialog, there are 'Close' and 'Upload' buttons, with the 'Upload' button highlighted by a red box. A green circle with the number '7' is located above the file list.

8. Pastikan tampilan **CSV\_FILES** seperti pada gambar di bawah ini

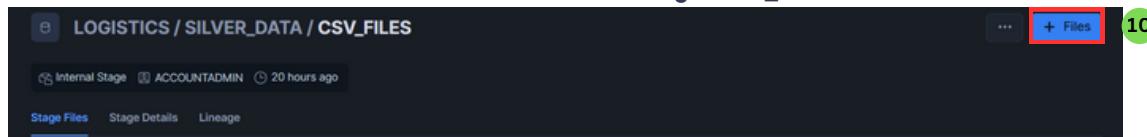


The screenshot shows a stage details page for 'LOGISTICS / GOLD\_DATA / CSV\_FILES'. The page indicates it is an 'Internal Stage' created by 'ACCOUNTADMIN' 21 hours ago. The 'Stage Files' tab is selected, showing a list of 8 files under the heading 'CSV\_FILES (8 Files)'. The files listed are: shipment\_schedule.csv, maintenance\_check.csv, customer\_complaints.csv, maintenance\_logs.csv, financial\_expenditure.csv, fuel\_cost\_logs.csv, iot\_telemetry.csv, and weather\_observation.csv. Each file entry has a small icon and a link. A red box highlights the list of files. A green circle with the number '8' is located to the right of the list.

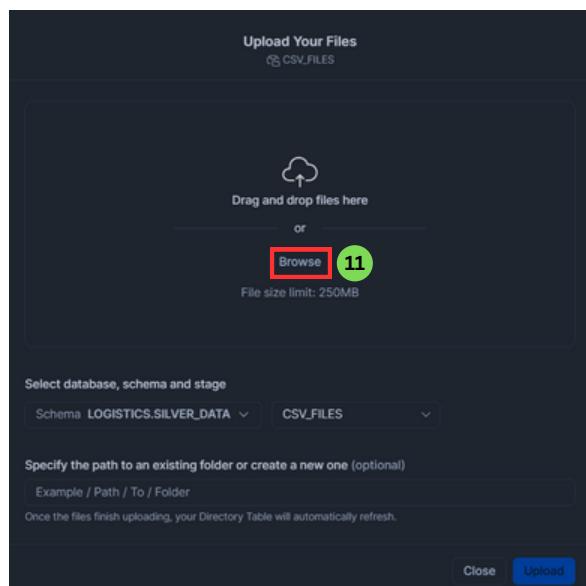
9. Klik pada Tab Database: Database LOGISTICS→ Schema SILVER\_DATA→ Stages → CSV\_FILES



10. Klik tombol + Files untuk menambahkan files ke dalam Stages CSV\_FILES

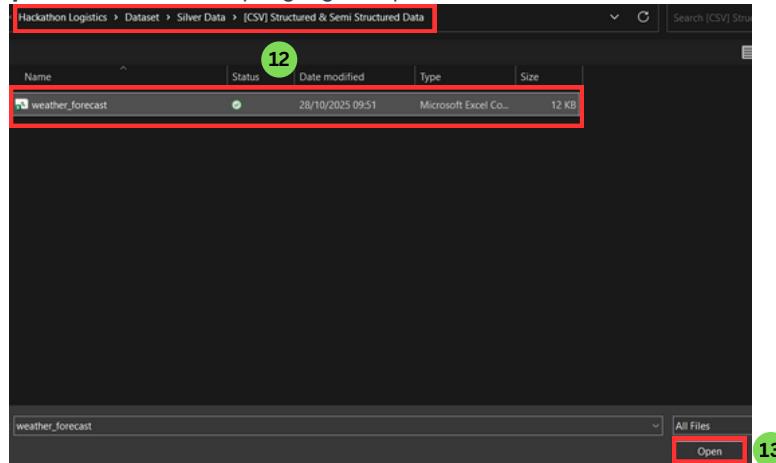


11. Klik Tombol Browse untuk memilih Files yang akan diupload

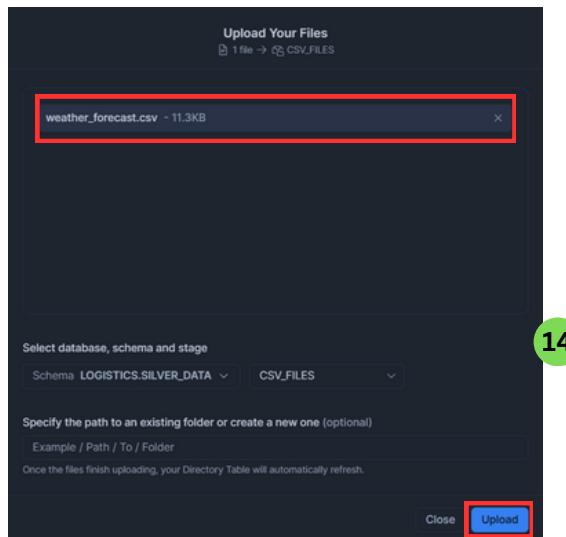


12. Blok semua File CSV yang ada di dalam Folder  
Hackathon Logistics → Dataset → Silver Data → [CSV] Structured & Semi Structured Data

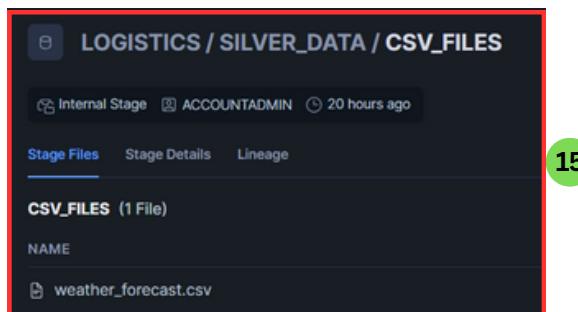
13. Klik tombol Open untuk finalize file yang ingin diupload



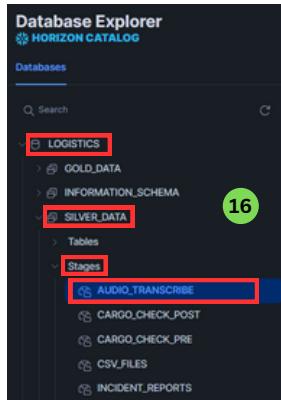
14. Pastikan ada 1 Files yang akan diupload pada step ini, dan Klik Upload untuk finalize proses Upload File ke Stages



15. Pastikan tampilan CSV\_FILES seperti pada gambar di bawah ini



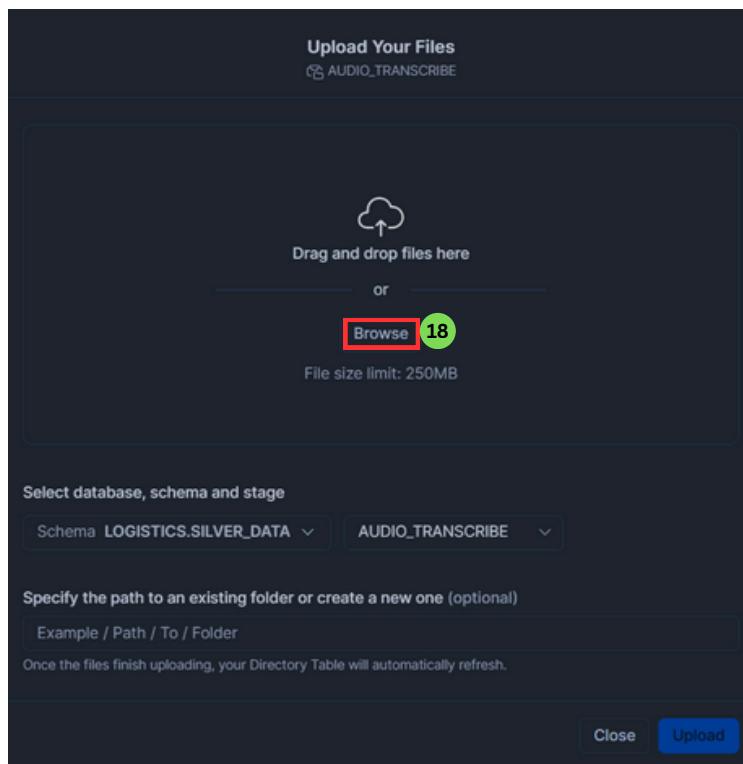
16. Klik pada Tab Database: Database LOGISTICS→ Schema SILVER\_DATA→ Stages → AUDIO\_TRANSCRIBE



17. Klik tombol + Files untuk menambahkan files ke dalam Stages AUDIO\_TRANSCRIBE

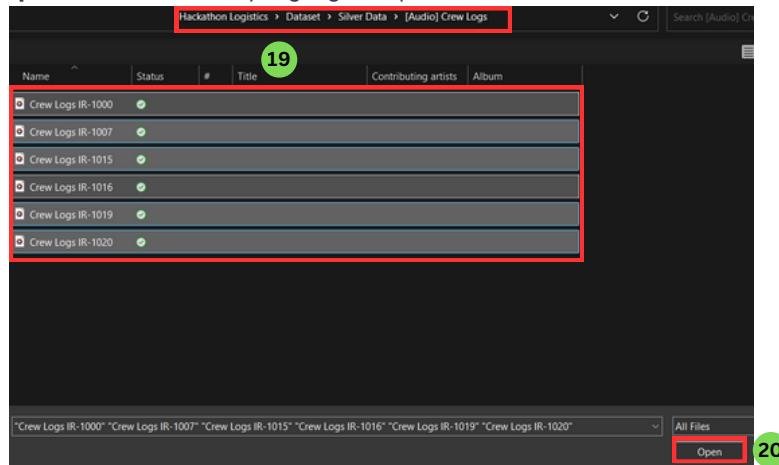


18. Klik Tombol Browse untuk memilih Files yang akan diupload

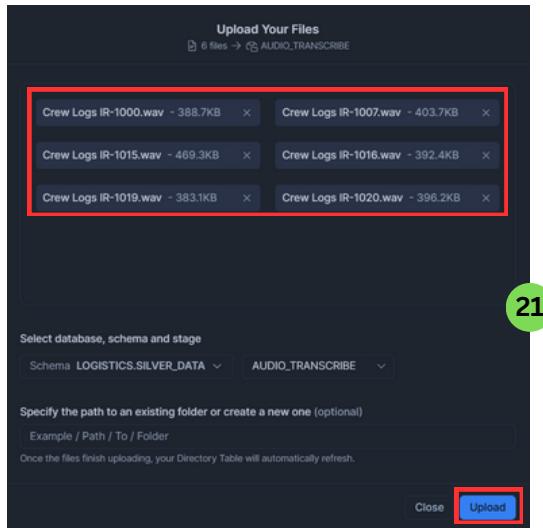


19. Blok semua File CSV yang ada di dalam Folder  
Hackathon Logistics → Dataset → Silver Data → [Audio] Crew Logs

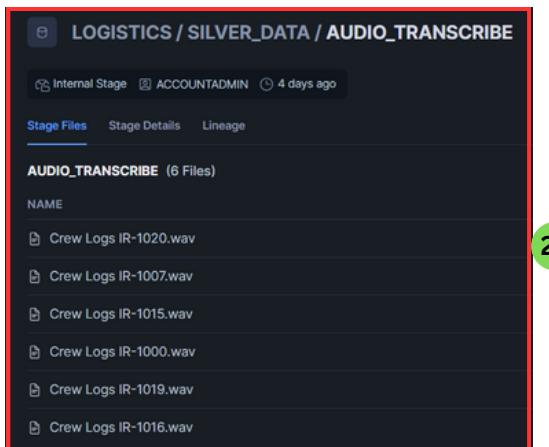
20. Klik tombol Open untuk finalize file yang ingin diupload



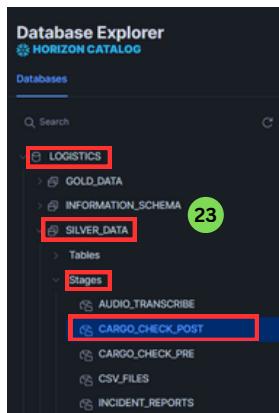
21. Pastikan ada 6 Files yang akan diupload pada step ini, dan Klik Upload untuk finalize proses Upload File ke Stages



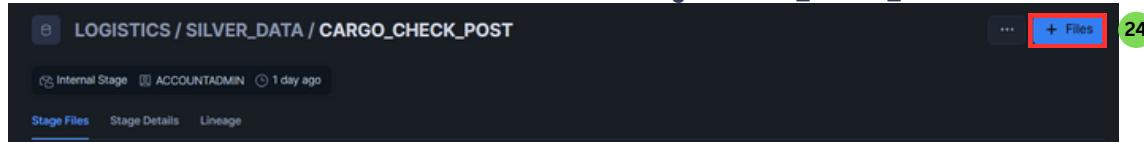
22. Pastikan tampilan AUDIO\_TRANSCRIBE seperti pada gambar di bawah ini



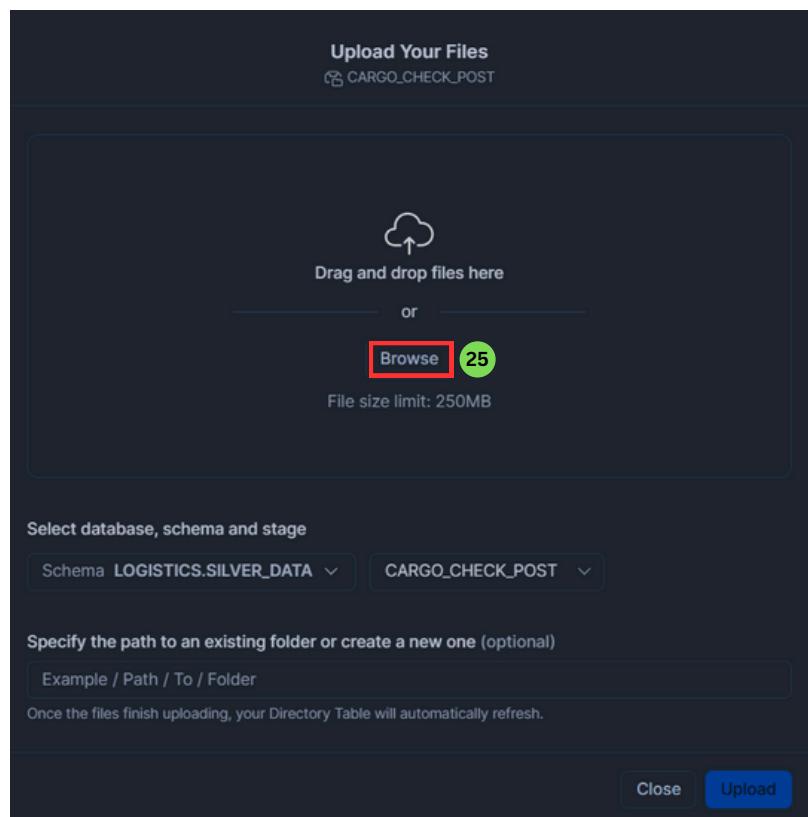
23. Klik pada Tab Database: Database LOGISTICS→ Schema SILVER\_DATA→ Stages → CARGO\_CHECK\_POST



24. Klik tombol + Files untuk menambahkan files ke dalam Stages CARGO\_CHECK\_POST

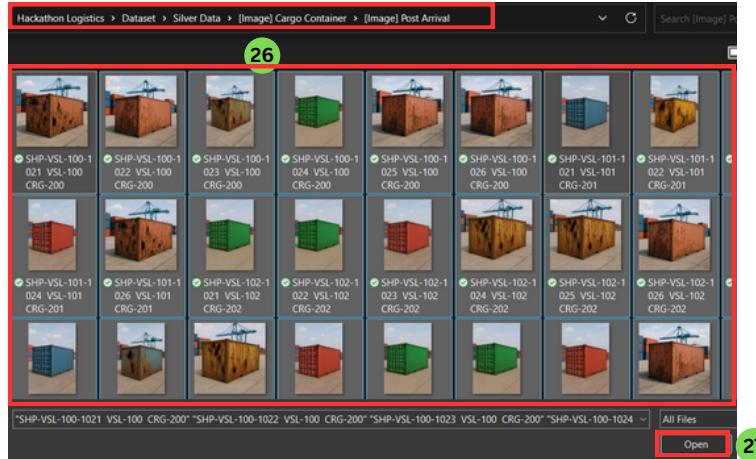


25. Klik Tombol Browse untuk memilih Files yang akan diupload

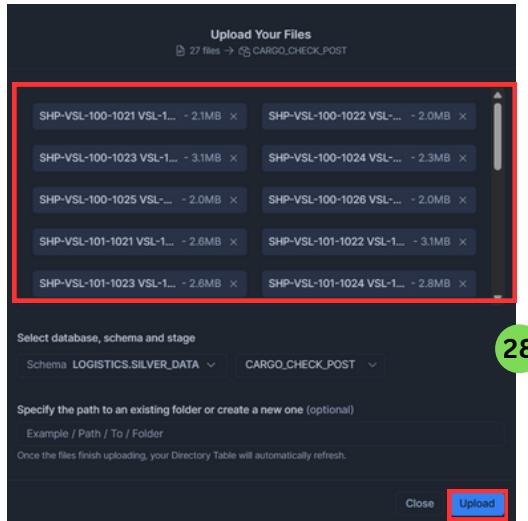


26. Blok semua File CSV yang ada di dalam Folder  
Hackathon Logistics → Dataset → Silver Data → [Image] Cargo Container → [Image] Post Arrival

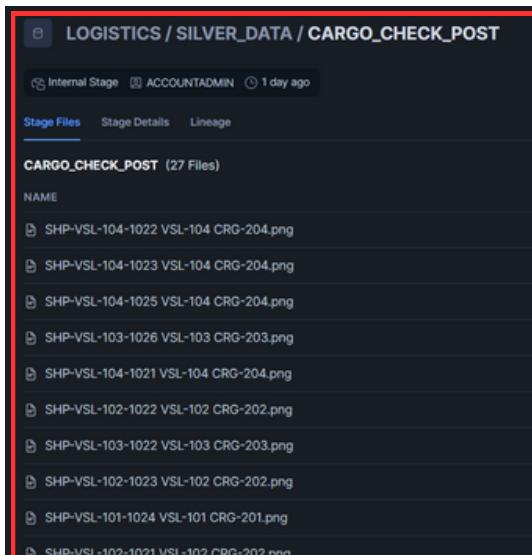
27. Klik tombol Open untuk finalize file yang ingin diupload



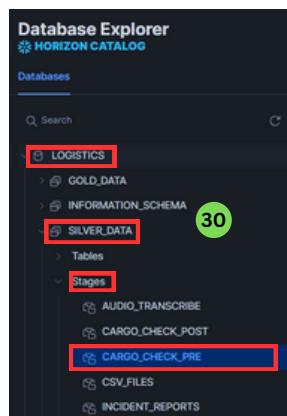
28. Pastikan ada 27 Files yang akan diupload pada step ini, dan Klik Upload untuk finalize proses Upload File ke Stages



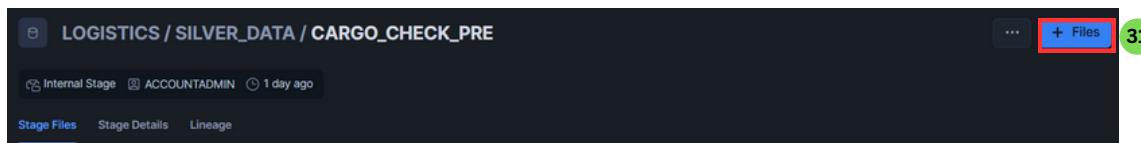
29. Pastikan tampilan CARGO\_CHECK\_POST seperti pada gambar di bawah ini



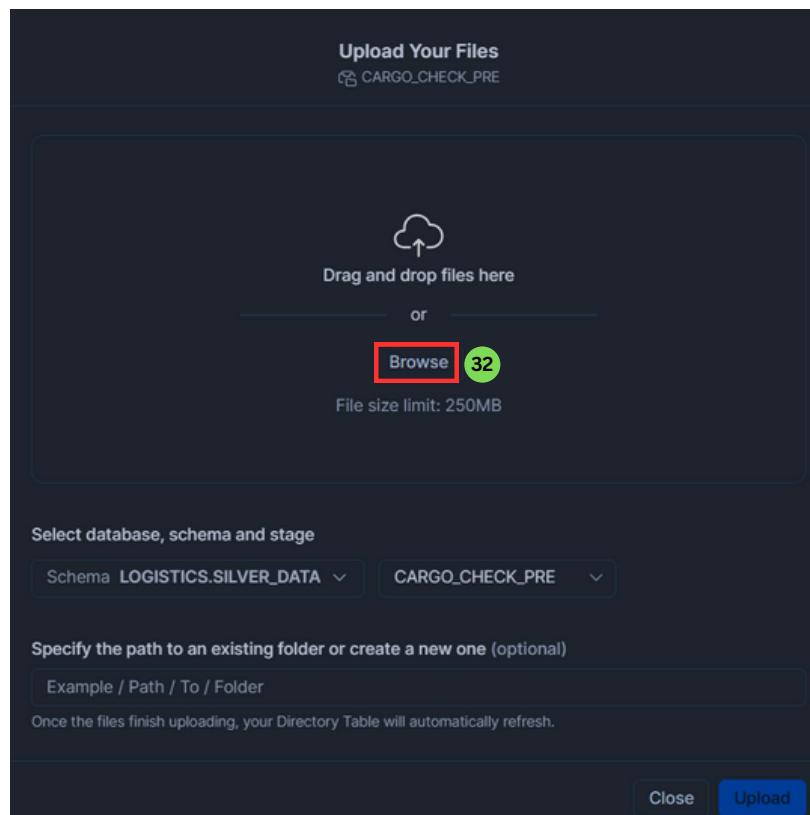
30. Klik pada Tab Database: Database LOGISTICS→ Schema SILVER\_DATA→ Stages → CARGO\_CHECK\_PRE



31. Klik tombol + Files untuk menambahkan files ke dalam Stages CARGO\_CHECK\_PRE



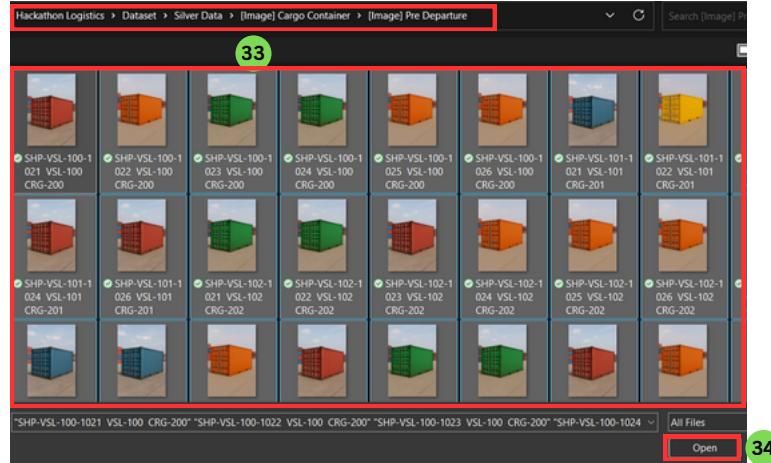
32. Klik Tombol Browse untuk memilih Files yang akan diupload



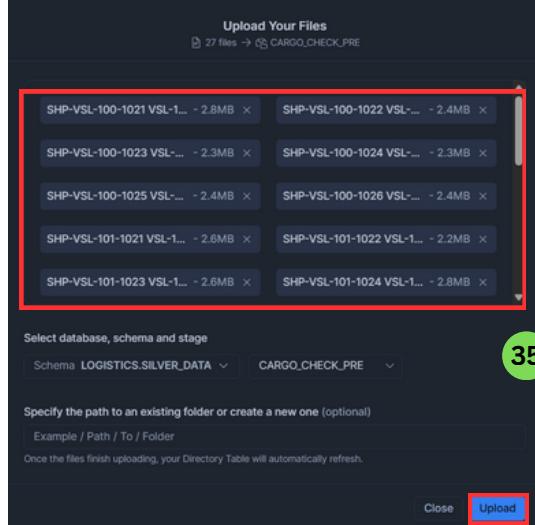
33. Blok semua File CSV yang ada di dalam Folder

Hackathon Logistics → Dataset → Silver Data → [Image] Cargo Container → [Image] Pre Arrival

34. Klik tombol Open untuk finalize file yang ingin diupload



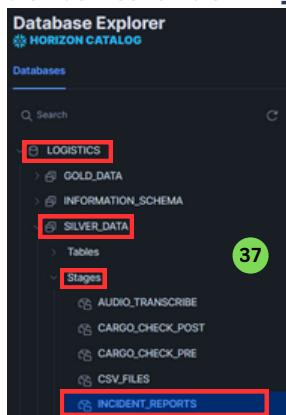
35. Pastikan ada 27 Files yang akan diupload pada step ini, dan Klik Upload untuk finalize proses Upload File ke Stages



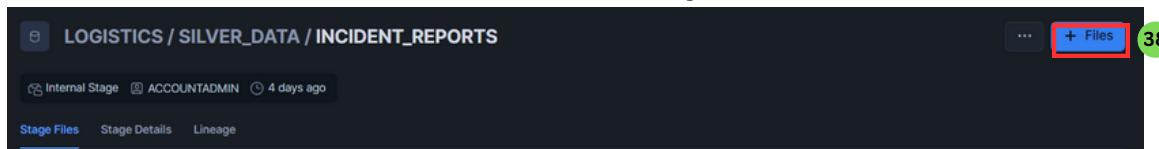
36. Pastikan tampilan CARGO\_CHECK\_PRE seperti pada gambar di bawah ini



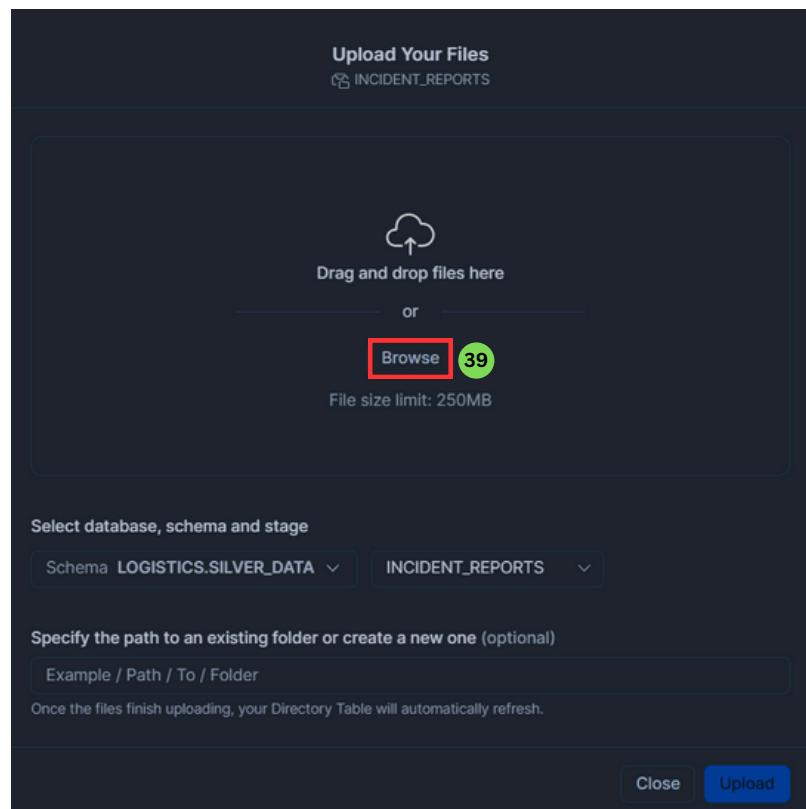
37. Klik pada Tab Database: Database LOGISTICS→ Schema SILVER\_DATA→ Stages → INCIDENT\_REPORTS



38. Klik tombol + Files untuk menambahkan files ke dalam Stages INCIDENT\_REPORTS

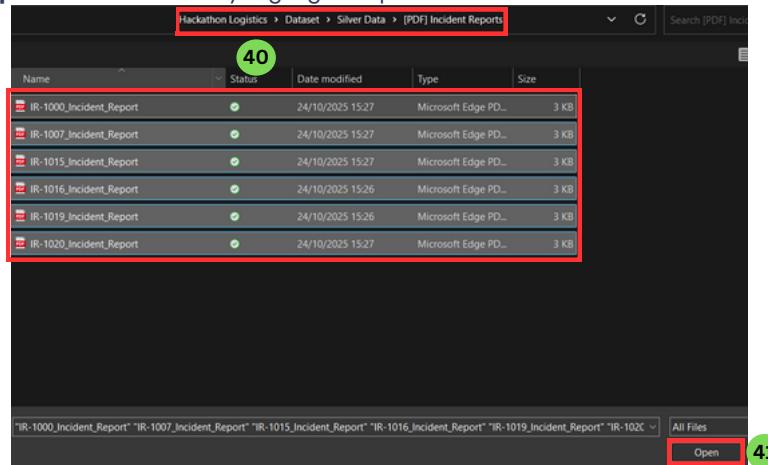


39. Klik Tombol Browse untuk memilih Files yang akan diupload



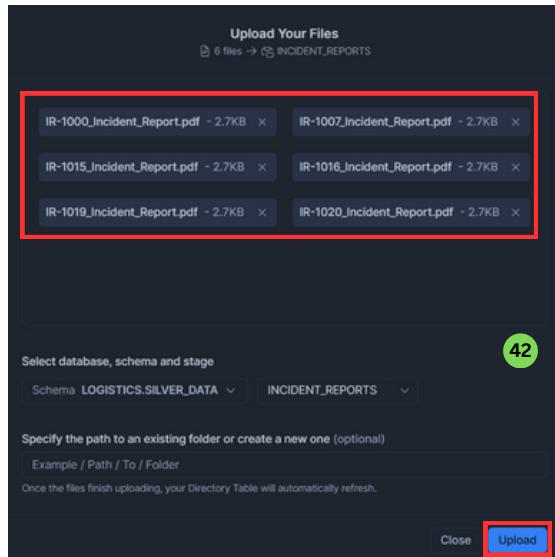
40. Blok semua File CSV yang ada di dalam Folder  
Hackathon Logistics → Dataset → Silver Data → [Image] Cargo Container → [PDF] Incident Reports

41. Klik tombol Open untuk finalize file yang ingin diupload



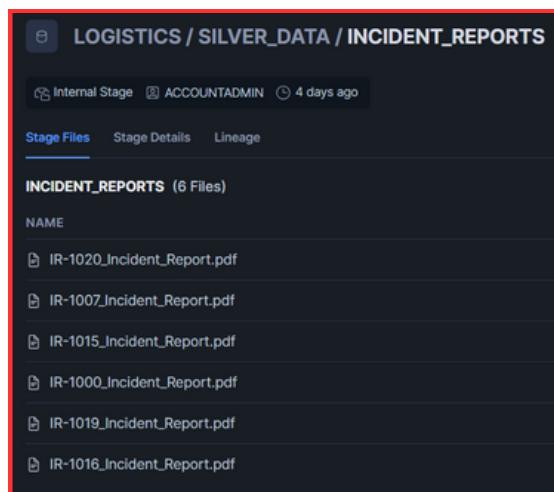
The screenshot shows a file browser window titled "Hackathon Logistics > Dataset > Silver Data > [PDF] Incident Reports". A red box highlights the list of files in the center. The files are: IR-1000\_Incident\_Report, IR-1007\_Incident\_Report, IR-1015\_Incident\_Report, IR-1016\_Incident\_Report, IR-1019\_Incident\_Report, and IR-1020\_Incident\_Report. Each file is a Microsoft Edge PDF, 3 KB in size, and was modified on 24/10/2025 at 15:27 or 15:26. At the bottom right, there is a "Search [PDF] Inci..." input field, a "All Files" dropdown, and a red box around the "Open" button. A green circle with the number 41 is positioned to the right of the "Open" button.

42. Pastikan ada 6 Files yang akan diupload pada step ini, dan Klik Upload untuk finalize proses Upload File ke Stages



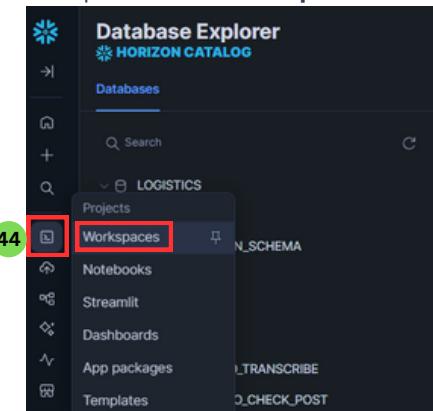
The screenshot shows the "Upload Your Files" interface. A red box highlights the list of files being uploaded: IR-1000\_Incident\_Report.pdf, IR-1007\_Incident\_Report.pdf, IR-1015\_Incident\_Report.pdf, IR-1016\_Incident\_Report.pdf, IR-1019\_Incident\_Report.pdf, and IR-1020\_Incident\_Report.pdf. Each file is 2.7KB in size. Below the file list, there are dropdown menus for "Schema" (LOGISTICS.SILVER\_DATA) and "Stage" (INCIDENT\_REPORTS). There is also a text input field for specifying a path to an existing folder or creating a new one. At the bottom right, there is a "Close" button and a red box around the "Upload" button. A green circle with the number 43 is positioned to the right of the "Upload" button.

43. Pastikan tampilan INCIDENT\_REPORTS seperti pada gambar di bawah ini

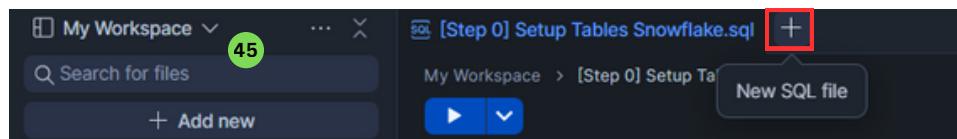


The screenshot shows the "LOGISTICS / SILVER\_DATA / INCIDENT\_REPORTS" stage details page. A red box highlights the list of files under the heading "INCIDENT\_REPORTS (6 Files)". The files listed are: IR-1020\_Incident\_Report.pdf, IR-1007\_Incident\_Report.pdf, IR-1015\_Incident\_Report.pdf, IR-1000\_Incident\_Report.pdf, IR-1019\_Incident\_Report.pdf, and IR-1016\_Incident\_Report.pdf. The stage details show it is an Internal Stage created by ACCOUNTADMIN 4 days ago. There are tabs for "Stage Files", "Stage Details", and "Lineage". A green circle with the number 43 is positioned to the right of the list of files.

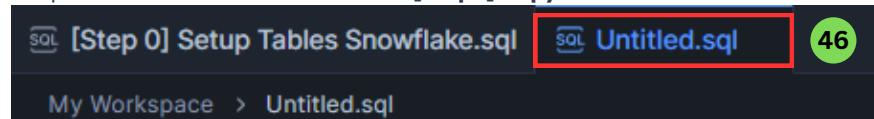
44. Klik pada tombol **Projects & Workspaces** untuk membuka Tab Worksheets



45. Klik pada tombol + untuk membuat SQL File baru



46. Double Klik pada nama File SQL dan ubah ke [Step 1] Copy Data From CSV



47. Copy dan Paste Query SQL dari [Step 1] Copy Data From CSV.sql di github repo ke Worksheet “[Step 1] Copy Data From CSV”

48. Run All di Worksheet yang sudah terisi dengan Query SQL, dengan Klik Tombol Panah ke bawah & Klik Tombol Run All

Query SQL ini berfungsi untuk loading data dari semua File CSV ke dalam Tables

The screenshot shows the 'Step 1' worksheet with the 'Run all' command highlighted by a red box and circled with a green marker labeled '48'. The command is defined as 'Ctrl + Shift + Enter'.

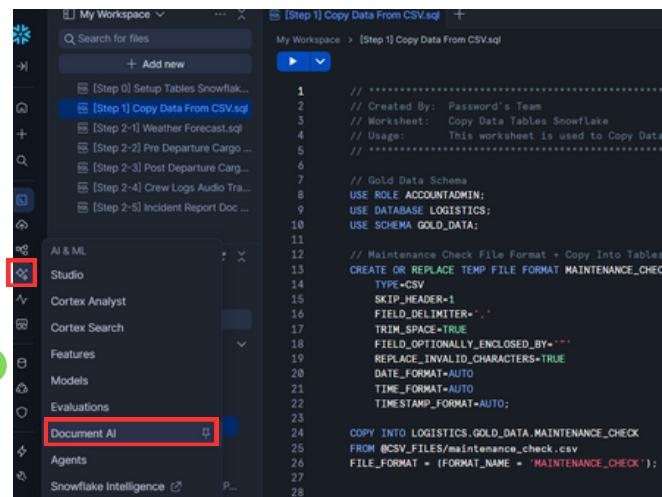
```
1 Run all
2 Ctrl + Shift + Enter
3 By: Password's Team
4 // Usage: This worksheet is used to Copy Data from CSV into Tables
5 // *****
6
7 // Gold Data Schema
8 USE ROLE ACCOUNTADMIN;
9 USE DATABASE LOGISTICS;
10 USE SCHEMA GOLD_DATA;
11
12 // Maintenance Check File Format + Copy Into Tables
13 CREATE OR REPLACE TEMP FILE FORMAT MAINTENANCE_CHECK
14   TYPE=CSV
15   SKIP_HEADER=1
16   FIELD_DELIMITER=','
17   TRIM_SPACE=TRUE
18   FIELD_OPTIONALLY_ENCLOSED_BY=''''
19   REPLACE_INVALID_CHARACTERS=TRUE
20   DATE_FORMAT=AUTO
21   TIME_FORMAT=AUTO
22   TIMESTAMP_FORMAT=AUTO;
```

## 4. Create and Training our Document AI Model

Saatnya mulai membuat Model Document AI

Pada tahap ke-4 ini, buat akan mulai dalam pembuatan Model yang akan buat gunakan dalam Use Case kali ini, yaitu DocumentAI

1. Dari tampilan Workspaces, Klik Tab AI & ML pada bagian Navigation Bar di sebelah kiri dan Klik Tombol Document AI untuk membuka tampilan Document AI



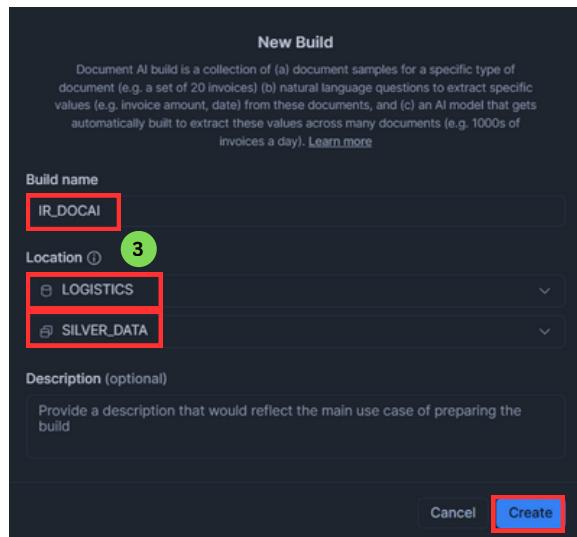
```
1 // ****
2 // Created By: Password's Team
3 // Worksheet: Copy Data Tables Snowflake
4 // Usage: This worksheet is used to Copy Data
5 // ****
6
7 // Gold Data Schema
8 USE ROLE ACCOUNTADMIN;
9 USE DATABASE LOGISTICS;
10 USE SCHEMA GOLD_DATA;
11
12 // Maintenance Check File Format + Copy Into Tables
13 CREATE OR REPLACE TEMP FILE FORMAT MAINTENANCE_CHECK
14   TYPE=CSV
15   SKIP_HEADER=1
16   FIELD_DELIMITER=','
17   TRIM_SPACE=True
18   FIELD_OPTIONALLY_ENCLOSED_BY=''''
19   REPLACE_INVALID_CHARACTERS=True
20   DATE_FORMAT=Auto
21   TIME_FORMAT=Auto
22   TIMESTAMP_FORMAT=Auto;
23
24 COPY INTO LOGISTICS.GOLD_DATA.MAINTENANCE_CHECK
25 FROM @CSV_FILES/maintenance_check.csv
26 FILE_FORMAT = (FORMAT_NAME = 'MAINTENANCE_CHECK');
```

2. Langkah selanjutnya Klik Tombol + Build untuk memulai tahap pembuatan Model Document AI



3. Pastikan isi pada Build name & Location sesuai dengan yang ada di gambar di bawah ini

4. Klik tombol Create untuk memulai pembuatan Model Document AI



New Build

Document AI build is a collection of (a) document samples for a specific type of document (e.g. a set of 20 invoices) (b) natural language questions to extract specific values (e.g. invoice amount, date) from these documents, and (c) an AI model that gets automatically built to extract these values across many documents (e.g. 1000s of invoices a day). [Learn more](#)

Build name

IR\_DOCAI

Location

LOGISTICS

SILVER\_DATA

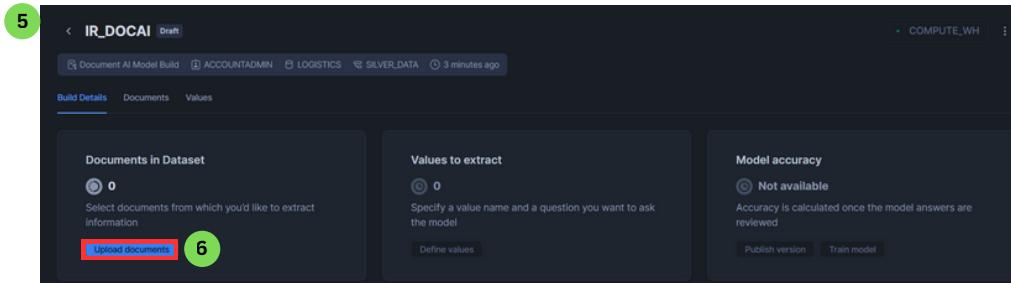
Description (optional)

Provide a description that would reflect the main use case of preparing the build

Create

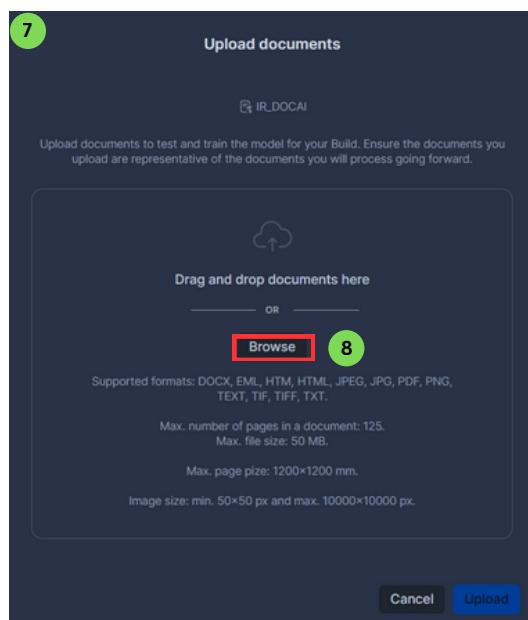
5. Setelah **Model Document AI** sukses terbuat, akan muncul tampilan seperti di bawah ini

6. Selanjutnya **klik Tombol Upload Documents** untuk **meng-upload File Incident Reports**



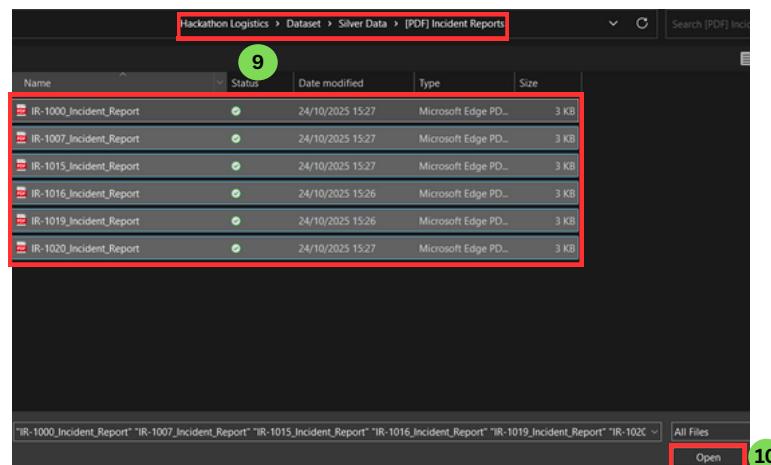
7. Tampilan di bawah ini akan muncul ketika buat **Klik Tombol Upload Documents**

8. kita bisa **Drag and Drop file** di dalam **Folder Incident Reports** atau **Browse Files** dengan menekan **tombol Browse**

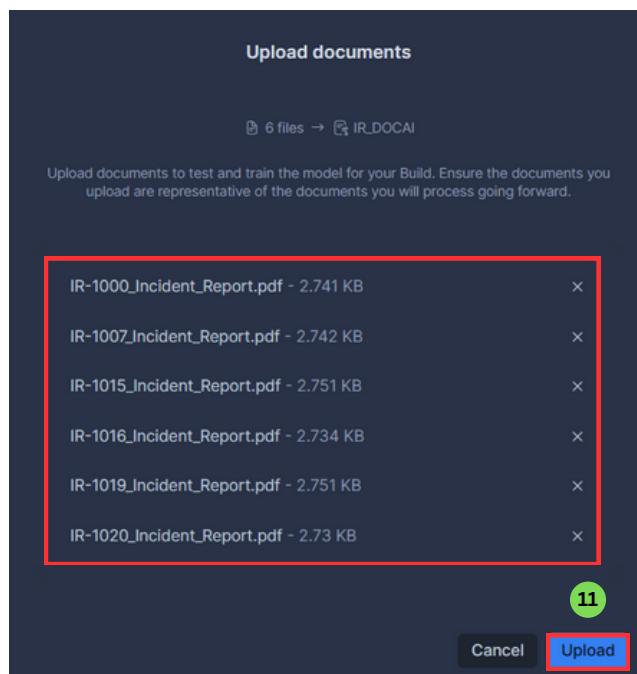


9. Blok semua File CSV yang ada di dalam Folder  
Hackathon Logistics → Dataset → Silver Data → [PDF] Incident Reports

10. Klik tombol Open untuk finalize file yang ingin diupload

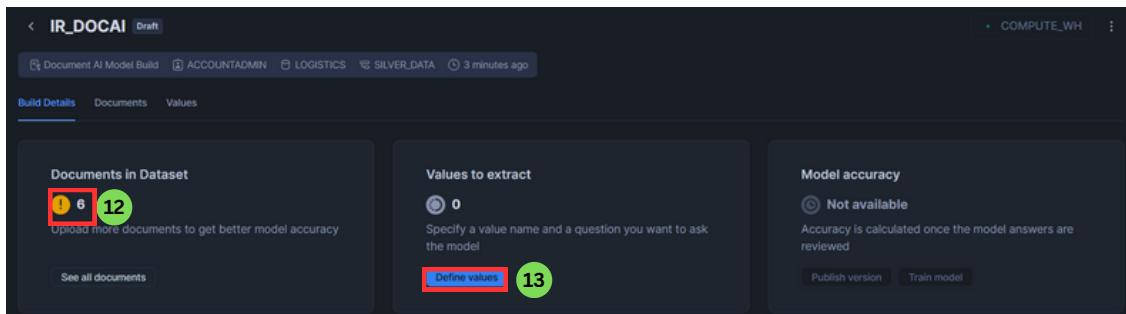


11. Pastikan ada 6 Files yang akan diupload pada step ini, dan Klik Upload untuk finalize proses Upload File ke Stages



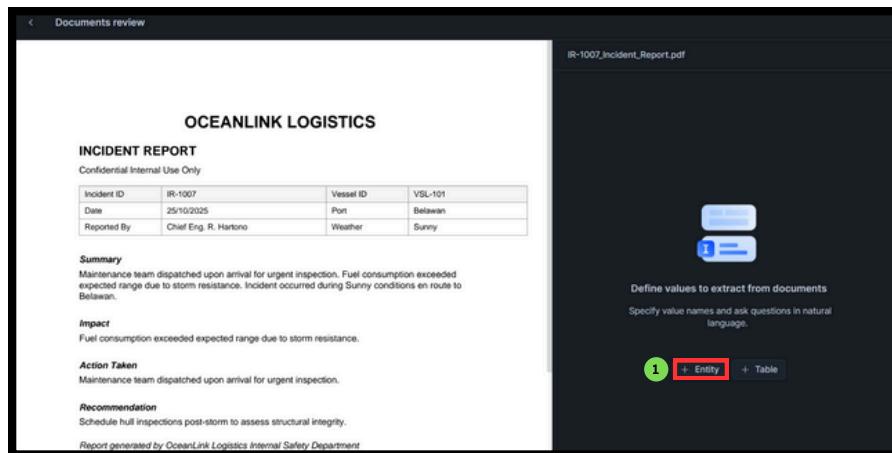
12. Setelah proses **Upload File Incident Reports** selesai, maka **Jumlah Documents in Dataset** akan berubah dari **0 → 6**

13. Proses selanjutnya, kita akan **Define Value** yang ingin kita **Extract**, dengan **Klik Tombol Define Values**



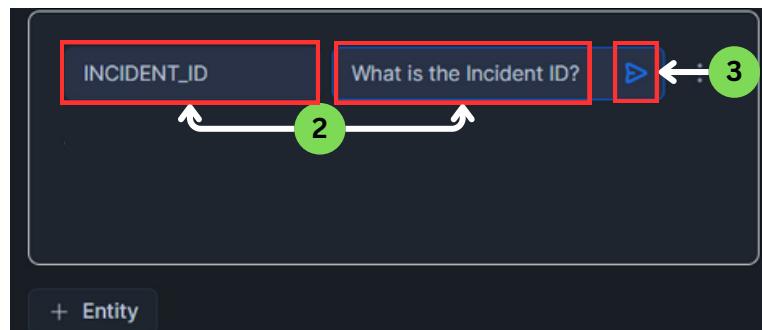
## Saatnya kita melatih Model yang sudah kita buat dengan 6 File dari Incident Reports

1. Kita akan memulai dengan **Menambahkan Value** dengan **Klik Tombol + Value** di bagian kanan untuk menambahkan **Value yang akan kita extract**



2. **Value pertama** yang akan kita **extract** yaitu **INCIDENT\_ID** dengan pertanyaan "**What is the Incident ID?**"

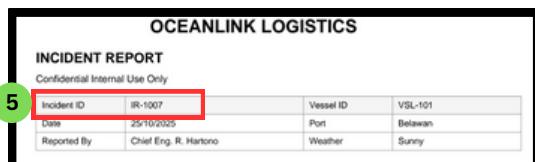
3. Setelah memasukkan **Value & Pertanyaan** yang akan kita gunakan, **Klik Tombol ▶** di samping pertanyaan yang sudah kita masukkan untuk mendapatkan **Nilai** dari **Document yang digunakan**



4. **Nilai dan Confidence Model** dari pertanyaan yang kita berikan, akan muncul di bagian bawah **INCIDENT\_ID & Pertanyaan** yang kita masukkan

5. **kita** perlu memastikan bahwa **Nilai** yang dihasilkan oleh **Model** sudah sesuai dengan **Document** yang ada disampingnya

6. Jika sesuai, maka **Klik Tombol ✓** di bagian pojok kanan **Nilai**



7. Ulangi Semua proses dari Langkah 1 – 6 untuk Setiap Value | Pertanyaan berikut:

No	Value	Pertanyaan
2	DATE	What is the Incident Date?
3	REPORTED_BY	Who reported the Incident?
4	VESSEL_ID	What is the Vessel ID?
5	PORT	Where is the Incident Port?
6	WEATHER	What is the weather when the incident occurred?
7	SUMMARY	What is the summary of Incident Reports?
8	IMPACT	What is the Incident Impact?
9	ACTION_TAKEN	What is the Action Taken regarding the Incident?
10	RECOMMENDATION	What is the recommendation after the Incident?

## OCEANLINK LOGISTICS

### INCIDENT REPORT

Confidential Internal Use Only

Incident ID	IR-1007	Vessel ID	VSL-101
2 Date	25/10/2025	4 Port	Belawan
3 Reported By	Chief Eng. R. Hartono	5 Weather	Sunny

6

**Summary**

7 Maintenance team dispatched upon arrival for urgent inspection. Fuel consumption exceeded expected range due to storm resistance. Incident occurred during Sunny conditions en route to Belawan.

**Impact**

8 Fuel consumption exceeded expected range due to storm resistance.

**Action Taken**

9 Maintenance team dispatched upon arrival for urgent inspection.

**Recommendation**

10 Schedule hull inspections post-storm to assess structural integrity.

Report generated by OceanLink Logistics Internal Safety Department

8. Jika kita sudah memasukkan **Semua Value | Questions (10)**, maka Tampilan kita akan seperti pada gambar berikut

9. Ketika semua **Value | Questions** sudah benar dan sesuai dengan yang ada di gambar, dapat dilanjutkan dengan **Klik Tombol Accept all and review next** pada tombol di pojok bawah kanan

OCEANLINK LOGISTICS  
INCIDENT REPORT  
Confidential Internal Use Only

Incident ID	IR-1007	Vessel ID	VSL-101
Date	25/10/2025	Port	Belawan
Reported By	Chief Eng. R. Hartono	Weather	Sunny

**Summary**  
Maintenance team dispatched upon arrival for urgent inspection. Fuel consumption exceeded expected range due to storm resistance. Incident occurred during Sunny conditions en route to Belawan.

**Impact**  
Fuel consumption exceeded expected range due to storm resistance.

**Action Taken**  
Maintenance team dispatched upon arrival for urgent inspection.

**Recommendation**  
Schedule hull inspections post-storm to assess structural integrity.

Report generated by OceanLink Logistics Internal Safety Department

INCIDENT\_ID : What is the Incident ID? IR-1007  
DATE : What is the Incident Date? 25/10/2025  
REPORTED\_BY : Who reported the Incident? Chief Eng. R. Hartono  
VESSEL\_ID : What is the Vessel ID? VSL-101

Accept all and review next

12. Kita akan kembali ke **Tab Build Details**, dengan **Klik Tab Build Details**

13. Pastikan bahwa **Values to extract** berubah dari **0 menjadi 5** dan ada **Tanda ✓** di sebelah kiri angka

14 Selanjutnya kita perlu **mengakses Proses Training Model** untuk meningkatkan **Model Accuracy** dengan **Klik Tombol Train Model** di sebelah kanan pada bagian **Model Accuracy**

IR\_DOCAI Draft

Build Details Documents Values

Documents in Dataset: 6

Values to extract: 10

Model accuracy: 0.95

Train model

15. Pada saat kita **Klik Tombol Train Model**, akan muncul tampilan seperti pada gambar berikut

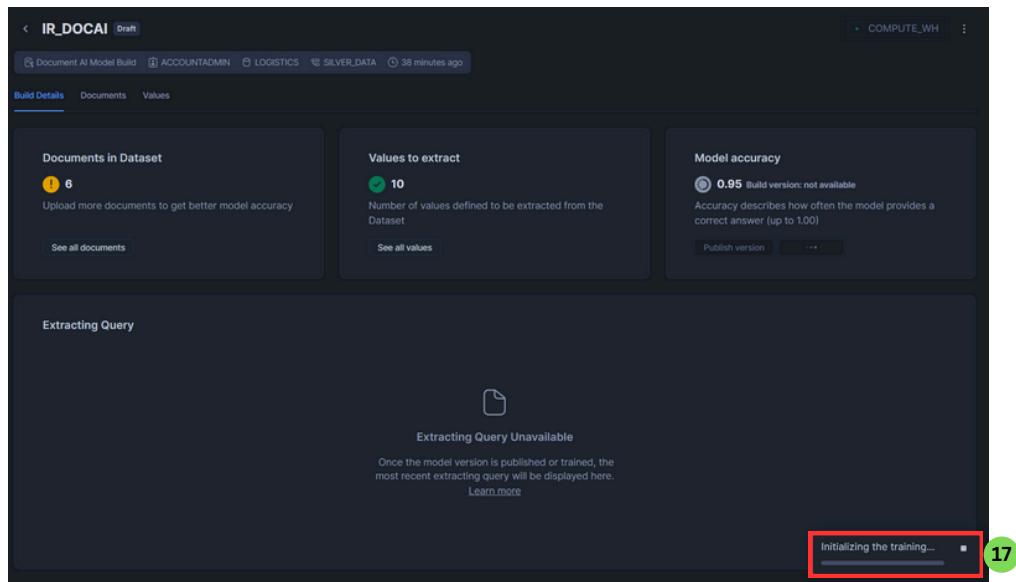
16. **Klik pada Tombol Start Training** untuk memulai proses **Training Model Document AI**

Start training

Training the model might take some time. Note that training creates a new Build version and extracting queries.

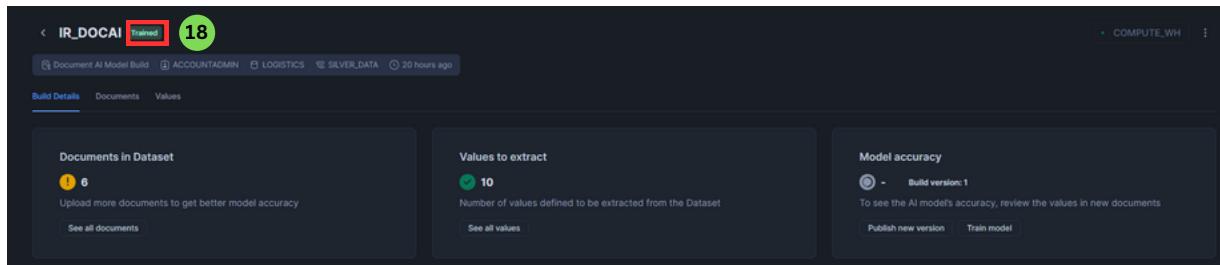
Cancel Start Training

17. Akan muncul sebuah **tampilan kecil di bagian pojok kanan bawah** untuk menunjukkan **Progress** dari **Proses Training Model**



Notes: Proses ini akan memakan waktu yang cukup lama (~ 20 menit)

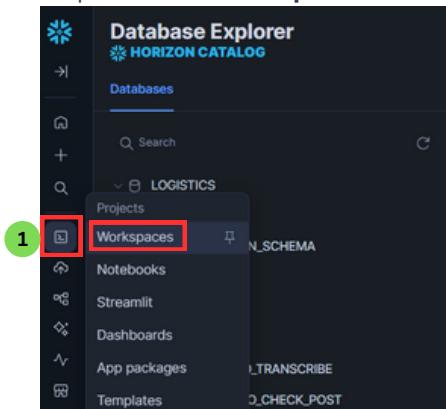
18. Setelah Proses Training selesai, akan muncul "**Trained**" di bagian kanan nama model sesuai dengan gambar di bawah ini



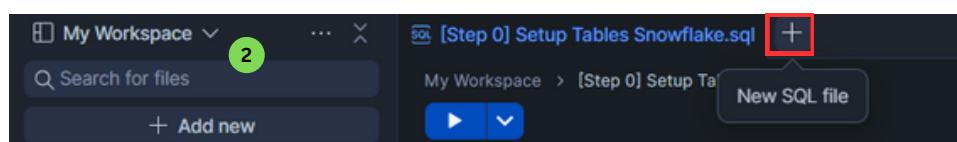
## 5. Transforming Data

Saatnya melakukan Transformation Data

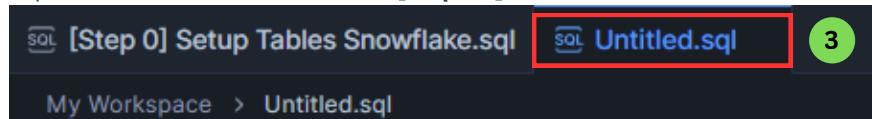
1. Klik pada tombol **Projects** & Klik pada tombol **Workspaces** untuk membuka **Tab Worksheets**



2. Klik pada tombol + untuk membuat **SQL File baru**



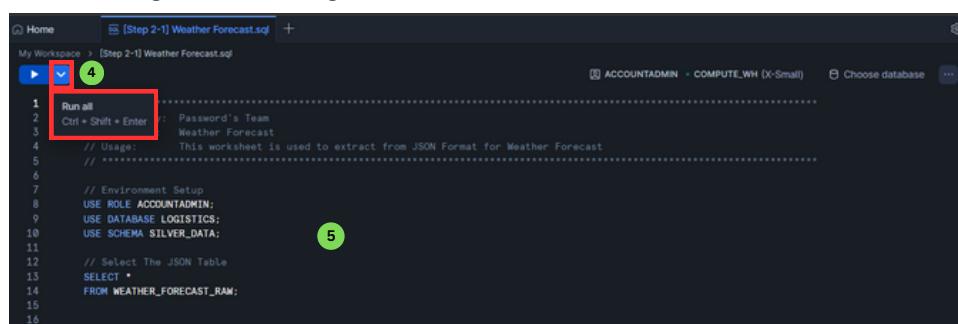
3. Double Klik pada nama **File SQL** dan ubah ke **[Step 2-1] Weather Forecast**



4. Copy dan Paste Query SQL dari **[Step 2-1] Weather Forecast.sql** di github repo ke **Worksheet** “[Step 2-1] Weather Forecast”

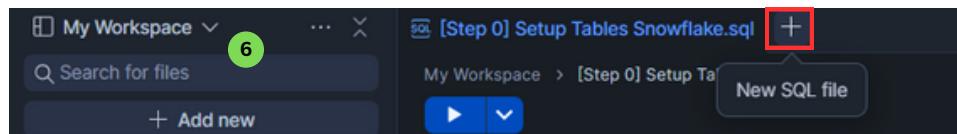
5. Run All di **Worksheet** yang sudah terisi dengan Query SQL, dengan **Klik Tombol Panah ke bawah & Klik Tombol Run All**

Query SQL ini berfungsi untuk loading data dari semua File CSV ke dalam Tables

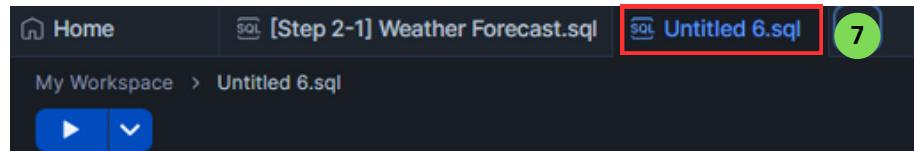


```
1 Run all
2 Ctrl + Shift + Enter
3 // Usage: This worksheet is used to extract from JSON Format for Weather Forecast
4 // *****
5
6
7 // Environment Setup
8 USE ROLE ACCOUNTADMIN;
9 USE DATABASE LOGISTICS;
10 USE SCHEMA SILVER_DATA;
11
12 // Select The JSON Table
13 SELECT *
14 FROM WEATHER_FORECAST_RAW;
```

6. Klik pada tombol + untuk membuat **SQL File baru**



7. Double Klik pada nama **File SQL** dan ubah ke **[Step 2-2] Pre Departure Cargo Container Condition Check**



8. Copy dan Paste Query SQL dari **[Step 2-2] Pre Departure Cargo Container Condition Check.sql** di github repo ke **Worksheet** "**[Step 2-2] Pre Departure Cargo Container Condition Check**"

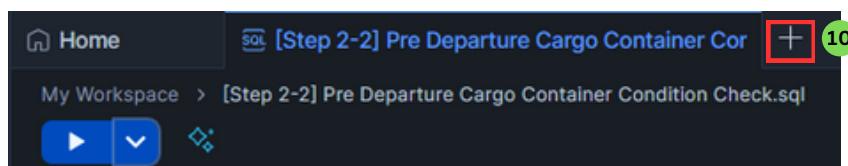
9. Run All di **Worksheet** yang sudah terisi dengan Query SQL, dengan **Klik Tombol Panah ke bawah & Klik Tombol Run All**

Query SQL ini berfungsi untuk loading data dari semua File CSV ke dalam Tables

The screenshot shows the Snowflake interface with the title bar '[Step 2-2] Pre Departure Cargo Container Cor'. A red box highlights the 'Run all' button in the toolbar. A green circle with the number '8' is placed over the 'Run all' button. Another green circle with the number '9' is placed over the status bar which shows 'ACCOUNTADMIN COMPUTE\_WH (X-Small) LOGISTICS SILVER\_DATA'.

```
1 Run all
2 Ctrl + Shift + Enter // Password's Team
3 // Pre Departure Cargo Container Condition Check
4 // Usage: This worksheet is used to check Cargo Container Condition Pre Departure
5 //
6
7 // Environment: Setup
8 USE ROLE ACCOUNTADMIN;
9 USE DATABASE LOGISTICS;
10 USE SCHEMA SILVER_DATA;
11
12 // Listing All Container Files
13 LIST @CARGO_CHECK_PRE;
14
15
16 // Converting From List of Files into a String of Content
17 INSERT INTO CARGO_CHECK_PRE_PATH
18 (SELECT TO_FILE('CARGO_CHECK_PRE', RELATIVE_PATH) AS CARGO_CHECK_PRE,
19 RELATIVE_PATH AS RELATIVE_PATH
20 FROM DIRECTORY(@CARGO_CHECK_PRE));
```

10. Klik pada tombol + untuk membuat **SQL File baru**



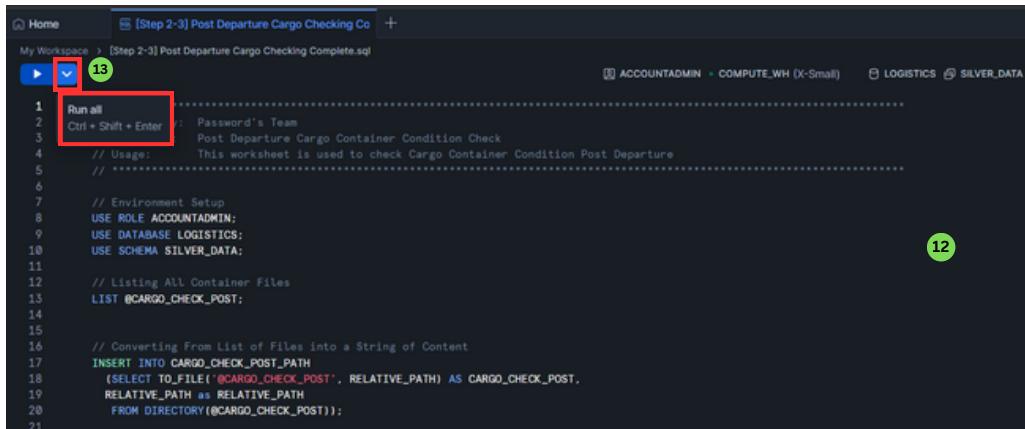
11. Double Klik pada nama **File SQL** dan ubah ke **[Step 2-3] Post Departure Cargo Checking Complete**



12. Copy dan Paste Query SQL dari **[Step 2-3] Post Departure Cargo Checking Complete.sql** di github repo ke Worksheet “[Step 2-3] Post Departure Cargo Checking Complete”

13. Run All di **Worksheet** yang sudah terisi dengan Query SQL, dengan **Klik Tombol Panah ke bawah & Klik Tombol Run All**

Query SQL ini berfungsi untuk loading data dari semua File CSV ke dalam Tables

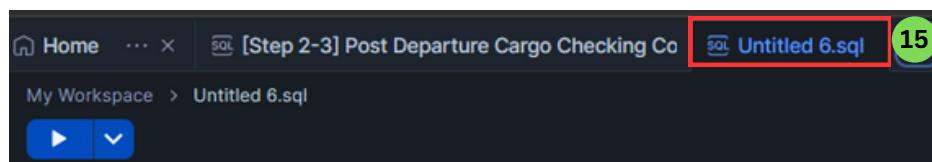


```
1 Run all
2 Ctrl + Shift + Enter
3 // Usage:
4 // This worksheet is used to check Cargo Container Condition Post Departure
5 //
6
7 // Environment Setup
8 USE ROLE ACCOUNTADMIN;
9 USE DATABASE LOGISTICS;
10 USE SCHEMA SILVER_DATA;
11
12 // Listing All Container Files
13 LIST @CARGO_CHECK_POST;
14
15
16 // Converting From List of Files into a String of Content
17 INSERT INTO CARGO_CHECK_POST_PATH
18 (SELECT TO_FILE(@CARGO_CHECK_POST, RELATIVE_PATH) AS CARGO_CHECK_POST,
19 RELATIVE_PATH AS RELATIVE_PATH
20 FROM DIRECTORY(@CARGO_CHECK_POST));
21
```

14. Klik pada tombol + untuk membuat **SQL File baru**



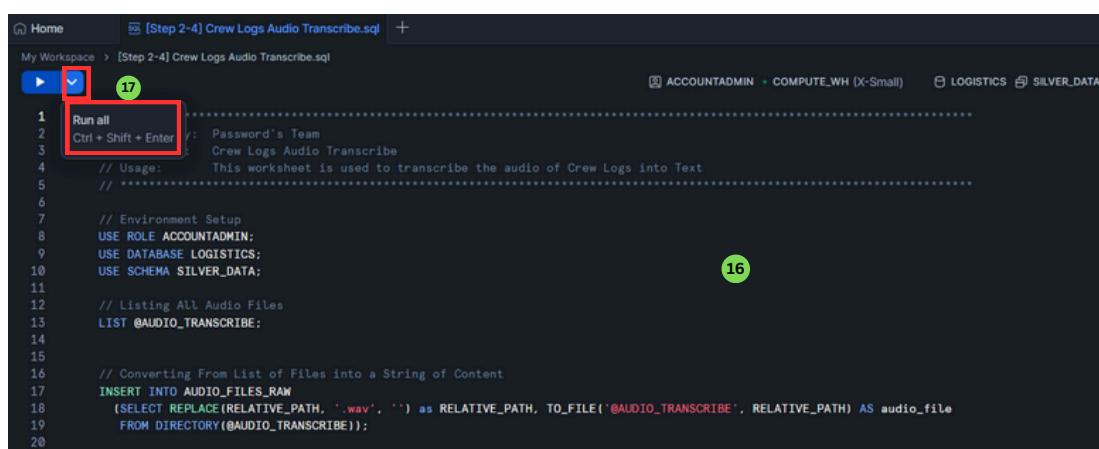
15. Double Klik pada nama **File SQL** dan ubah ke **[Step 2-4] Crew Logs Audio Transcribe**



16. Copy dan Paste Query SQL dari **[Step 2-4] Crew Logs Audio Transcribe.sql** di github repo ke Worksheet “[Step 2-4] Crew Logs Audio Transcribe”

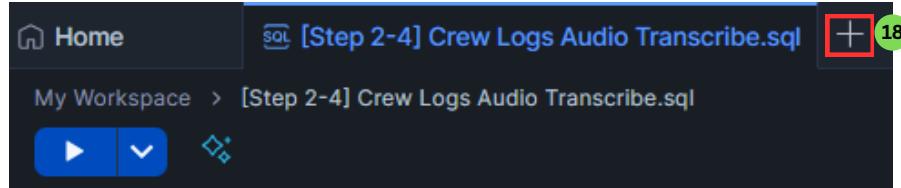
17. Run All di **Worksheet** yang sudah terisi dengan Query SQL, dengan **Klik Tombol Panah ke bawah & Klik Tombol Run All**

Query SQL ini berfungsi untuk loading data dari semua File CSV ke dalam Tables

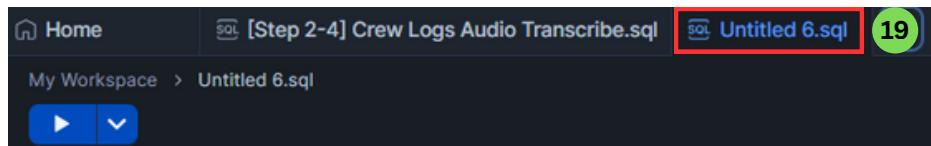


```
1 Run all
2 Ctrl + Shift + Enter
3 // Usage:
4 // This worksheet is used to transcribe the audio of Crew Logs into Text
5 //
6
7 // Environment Setup
8 USE ROLE ACCOUNTADMIN;
9 USE DATABASE LOGISTICS;
10 USE SCHEMA SILVER_DATA;
11
12 // Listing All Audio Files
13 LIST @AUDIO_TRANSCRIBE;
14
15
16 // Converting From List of Files into a String of Content
17 INSERT INTO AUDIO_FILES_RAW
18 (SELECT REPLACE(RELATIVE_PATH, '.wav', '') AS RELATIVE_PATH, TO_FILE('@AUDIO_TRANSCRIBE', RELATIVE_PATH) AS audio_file
19 FROM DIRECTORY(@AUDIO_TRANSCRIBE));
20
```

18. Klik pada tombol + untuk membuat SQL File baru



19. Double Klik pada nama File SQL dan ubah ke **[Step 2-5] Incident Report Doc AI**



20. Copy dan Paste Query SQL dari **[Step 2-5] Incident Report Doc AI.sql** di github repo ke **Worksheet "Step 2-5] Incident Report Doc AI"**

21. Run All di **Worksheet** yang sudah terisi dengan Query SQL, dengan **Klik Tombol Panah ke bawah & Klik Tombol Run All**

Query SQL ini berfungsi untuk loading data dari semua File CSV ke dalam Tables

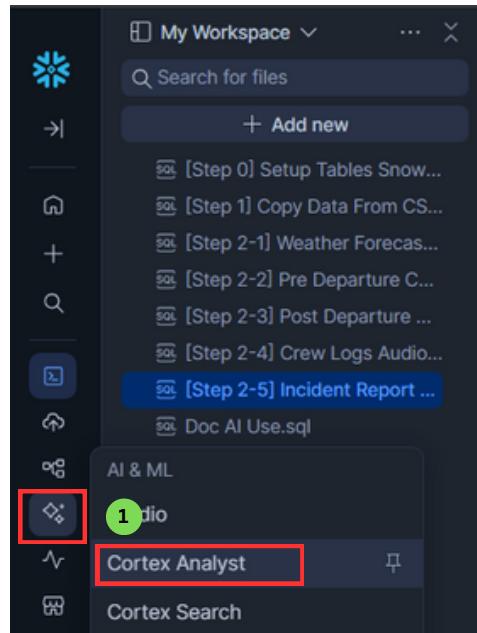
A screenshot of a worksheet interface. The top navigation bar shows 'Home' and a tab labeled '[Step 2-5] Incident Report Doc AI.sql'. Below the navigation bar, there are two blue navigation buttons. A red-bordered box highlights the first button with the number '21'. The main area contains a query script with several numbered lines. Line 1 has a red-bordered box around it. Line 21 has a green-bordered box around it. Line 20 is also highlighted with a green circle.

```
1 Run all
2 Ctrl + Shift + Enter // Password's Team
3 // Usage: Incident Report Doc AI
4 // This worksheet is used to extract data from Incident Reports PDF into Table Format
5 //
6
7 // Environment Setup
8 USE ROLE ACCOUNTADMIN;
9 USE DATABASE LOGISTICS;
10 USE SCHEMA SILVER_DATA;
11
12 // Listing All Incident Reports
13 LIST @INCIDENT_REPORTS;
14
15
16 // Extracting Data From Incident Reports
17 INSERT INTO INCIDENT_REPORTS_RAW
18 SELECT IR_DOCAI!PREDICT(GET_PRESIGNED_URL(@INCIDENT_REPORTS, RELATIVE_PATH)) AS ir_object
19 FROM DIRECTORY(@INCIDENT_REPORTS);
20
21
```

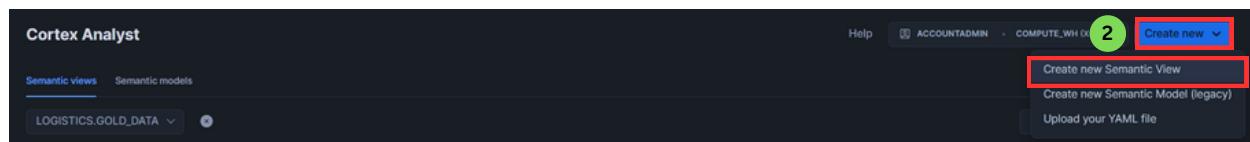
## 6. Cortex Analyst

Saatnya membuat Model Cortex Analyst

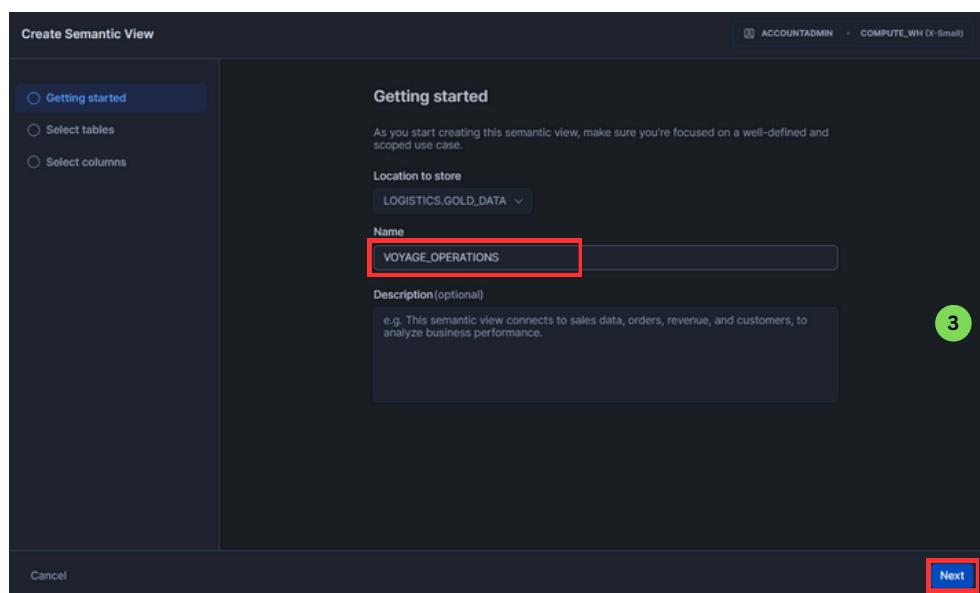
1. Klik Tab AI & ML dan setelah itu **Klik Cortex Analyst** untuk membuka tab **Cortex Analyst** untuk membuat **Semantic Model View**



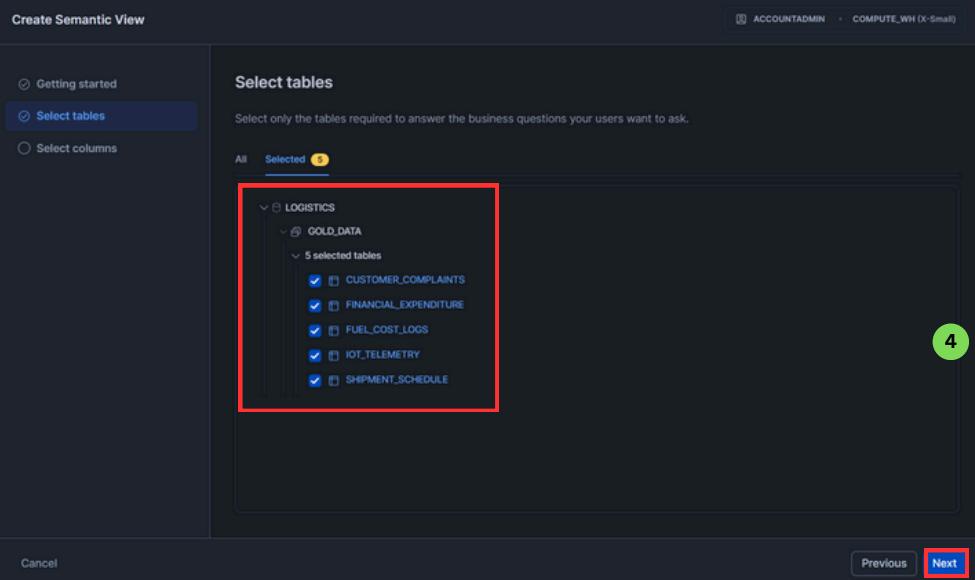
2. Klik tombol **Create new** dan Klik tombol **Create new Semantic View** untuk memulai pembuatan **Semantic View Model**



3. Buat Semantic View yang pertama yaitu **VOYAGE\_OPERATIONS** & Klik tombol **next**



4. Pilih 5 Tables sesuai dengan gambar di bawah ini



Create Semantic View

Select tables

Select only the tables required to answer the business questions your users want to ask.

All Selected 5

LOGISTICS

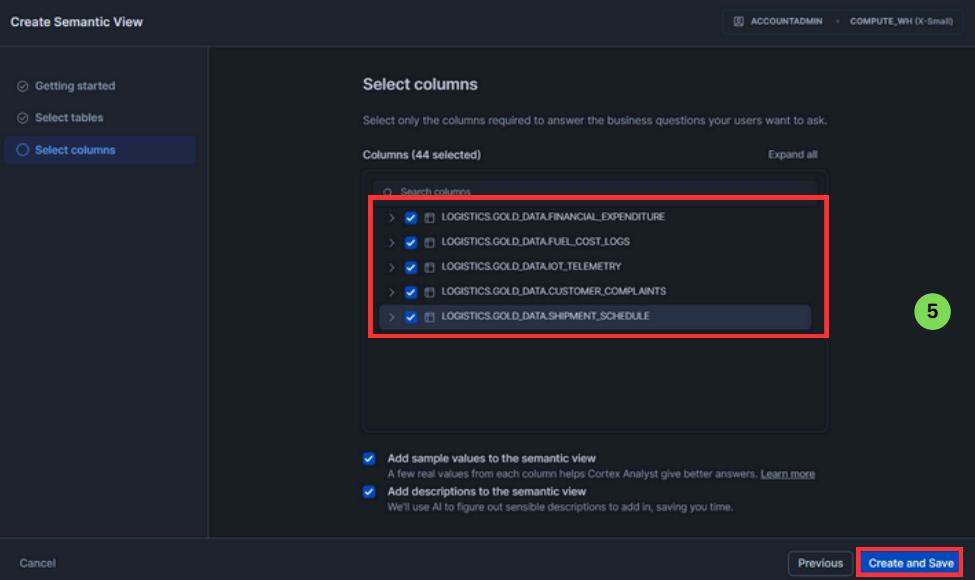
GOLD\_DATA

5 selected tables

- CUSTOMER\_COMPLAINTS
- FINANCIAL\_EXPENDITURE
- FUEL\_COST\_LOGS
- IOT\_TELEMETRY
- SHIPMENT\_SCHEDULE

Cancel Previous Next

5. Pilih semua kolom di semua Tabel seperti gambar di bawah ini



Create Semantic View

Select columns

Select only the columns required to answer the business questions your users want to ask.

Columns (44 selected) Expand all

Search columns

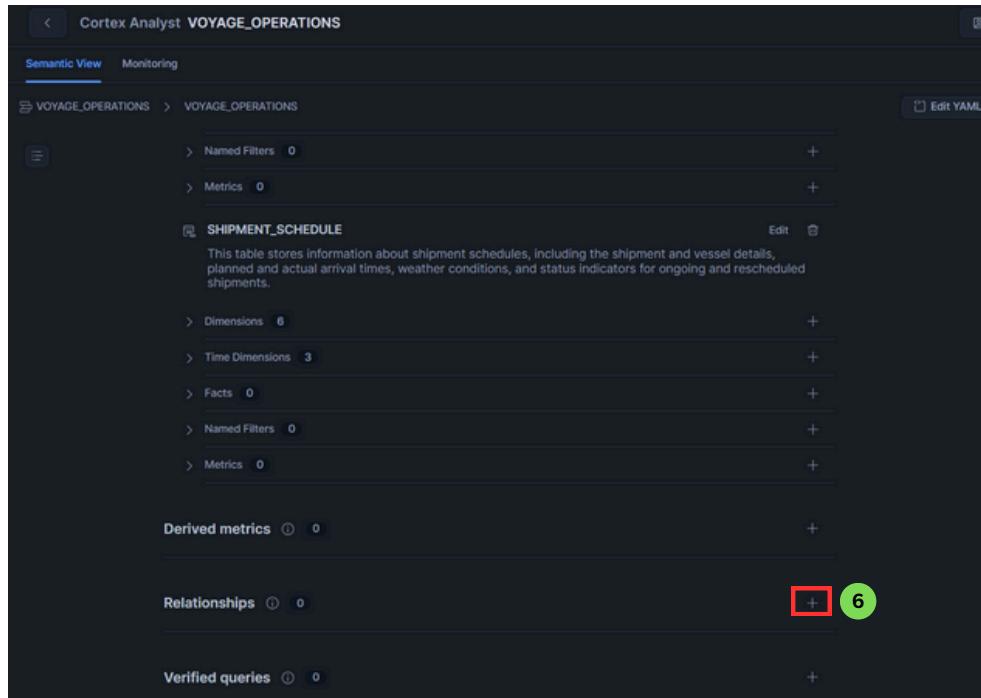
- >  LOGISTICS.GOLD\_DATA.FINANCIAL\_EXPENDITURE
- >  LOGISTICS.GOLD\_DATA.FUEL\_COST\_LOGS
- >  LOGISTICS.GOLD\_DATA.IOT\_TELEMETRY
- >  LOGISTICS.GOLD\_DATA.CUSTOMER\_COMPLAINTS
- >  LOGISTICS.GOLD\_DATA.SHIPMENT\_SCHEDULE

Add sample values to the semantic view  
A few real values from each column helps Cortex Analyst give better answers. [Learn more](#)

Add descriptions to the semantic view  
We'll use AI to figure out sensible descriptions to add in, saving you time.

Cancel Previous Create and Save

6. Klik Tombol + pada Relationships untuk menghubungkan antar tabel



Cortex Analyst VOYAGE\_OPERATIONS

Semantic View Monitoring

VOYAGE\_OPERATIONS > VOYAGE\_OPERATIONS

Named Filters 0 Metrics 0

**SHIPMENT\_SCHEDULE**  
This table stores information about shipment schedules, including the shipment and vessel details, planned and actual arrival times, weather conditions, and status indicators for ongoing and rescheduled shipments.

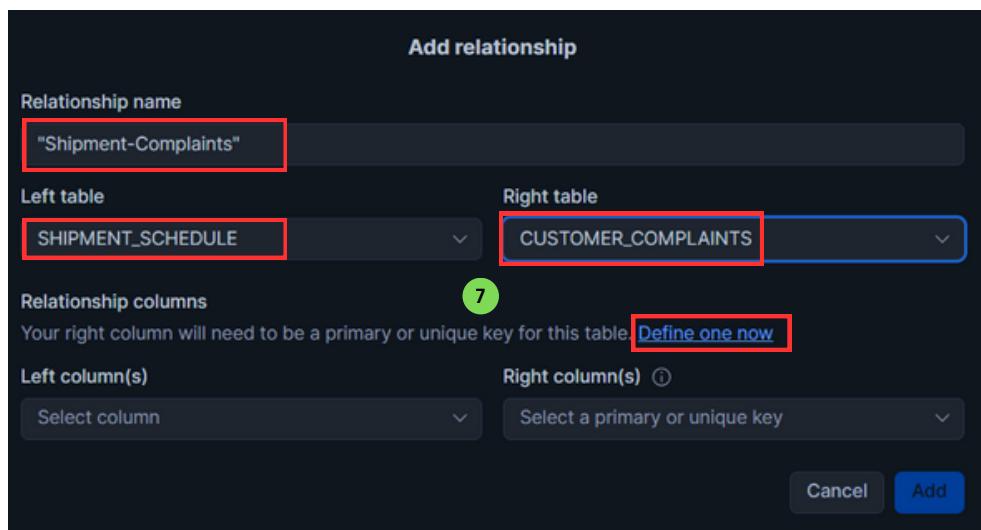
Dimensions 6 Time Dimensions 3 Facts 0 Named Filters 0 Metrics 0

Derived metrics 0

Relationships 0 **[+]** **6**

Verified queries 0

7. Masukkan "Shipment-Complaints" pada relationship name dan pilih SHIPMENT\_SCHEDULE untuk left table dan CUSTOMER\_COMPLAINTS untuk right table & Klik Define one now (untuk establish primary key)



Add relationship

Relationship name  
"Shipment-Complaints"

Left table  
SHIPMENT\_SCHEDULE

Right table  
CUSTOMER\_COMPLAINTS

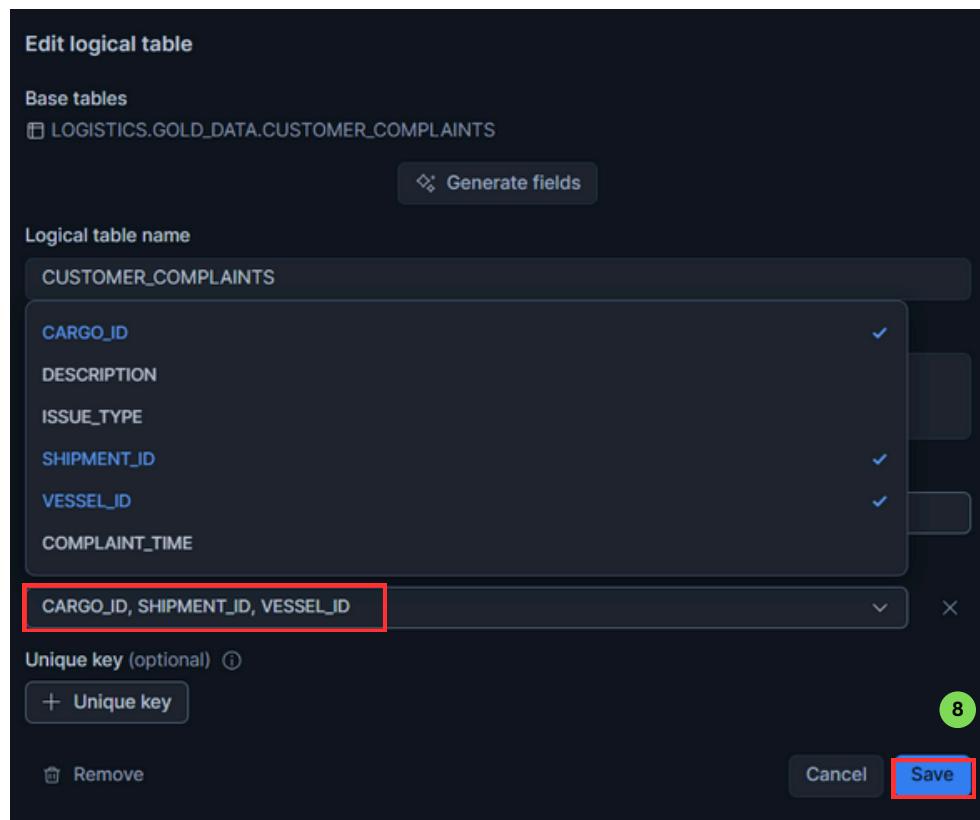
Relationship columns  
Your right column will need to be a primary or unique key for this table. **Define one now**

Left column(s)  
Select column

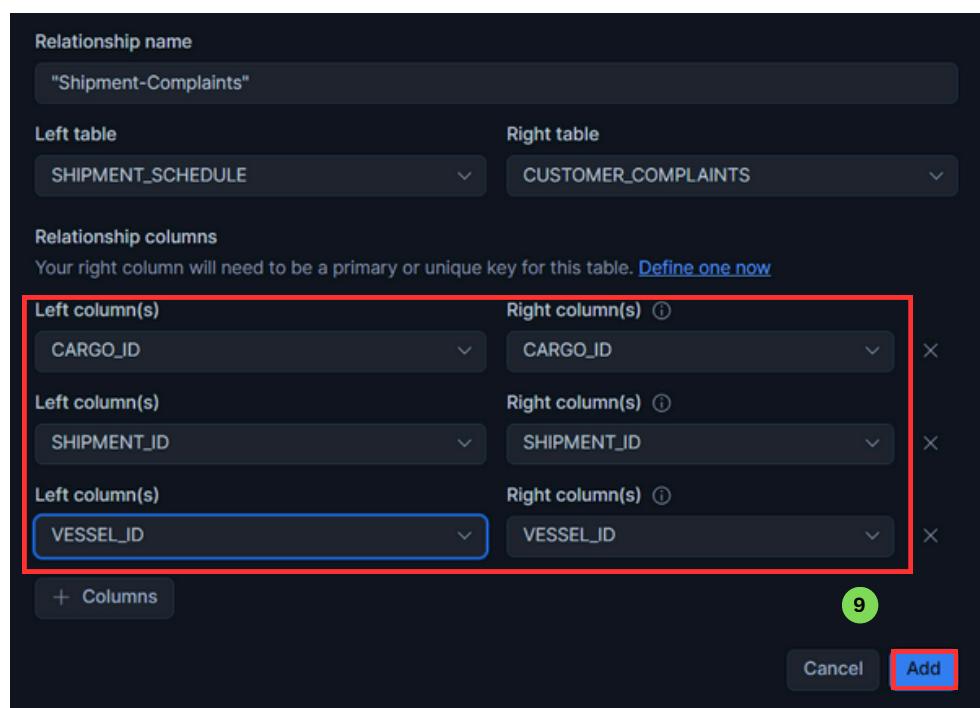
Right column(s)  
Select a primary or unique key

Cancel Add

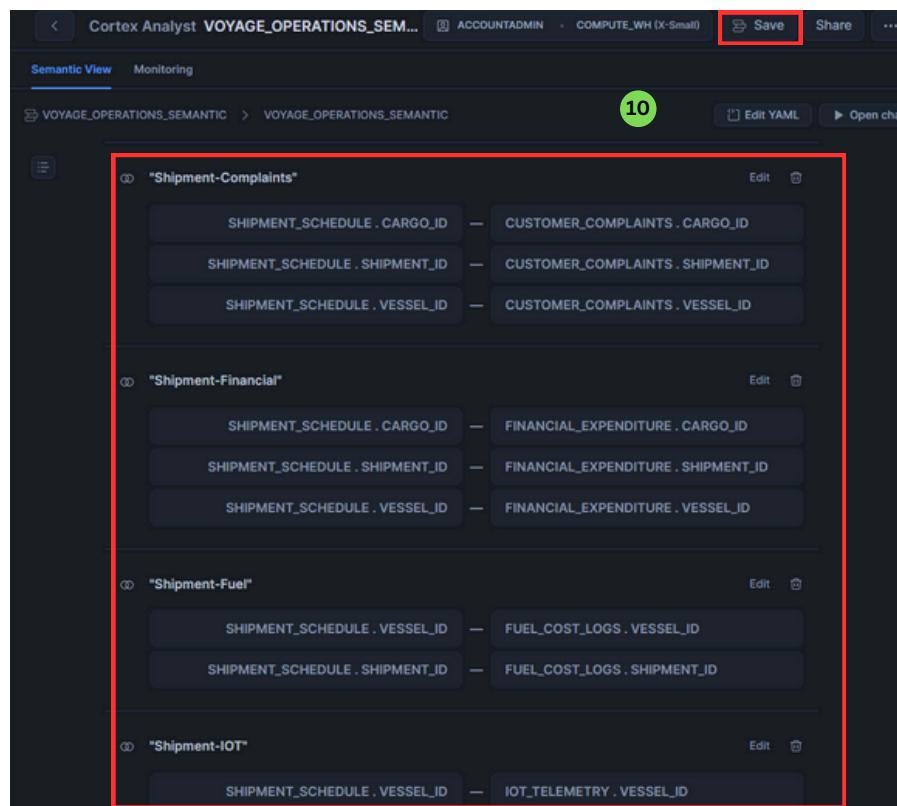
8. Pilih **CARGO\_ID, SHIPMENT\_ID, dan VESSEL\_ID** untuk menjadi Primary Key pada table **Customer Complaints** & Klik Save



9. Pastikan Relationship Columns untuk left & right columns sesuai dengan di gambar & Klik tombol add



10. Ulangi langkah 6 – 9 sehingga sesuai dengan gambar di bawah ini & Klik Tombol Save



The screenshot shows the Cortex Analyst Semantic View interface. At the top, it displays the title "Cortex Analyst VOYAGE\_OPERATIONS\_SEM..." and the user "ACCOUNTADMIN - COMPUTE\_WH (X-Small)". On the far right, there are buttons for "Save" (highlighted with a red box), "Share", and more options. A green circle with the number "10" is positioned in the top right corner. The main area contains four semantic models listed vertically:

- "Shipment-Complaints"**:
  - SHIPMENT\_SCHEDULE . CARGO\_ID — CUSTOMER\_COMPLAINTS . CARGO\_ID
  - SHIPMENT\_SCHEDULE . SHIPMENT\_ID — CUSTOMER\_COMPLAINTS . SHIPMENT\_ID
  - SHIPMENT\_SCHEDULE . VESSEL\_ID — CUSTOMER\_COMPLAINTS . VESSEL\_ID
- "Shipment-Financial"**:
  - SHIPMENT\_SCHEDULE . CARGO\_ID — FINANCIAL\_EXPENDITURE . CARGO\_ID
  - SHIPMENT\_SCHEDULE . SHIPMENT\_ID — FINANCIAL\_EXPENDITURE . SHIPMENT\_ID
  - SHIPMENT\_SCHEDULE . VESSEL\_ID — FINANCIAL\_EXPENDITURE . VESSEL\_ID
- "Shipment-Fuel"**:
  - SHIPMENT\_SCHEDULE . VESSEL\_ID — FUEL\_COST\_LOGS . VESSEL\_ID
  - SHIPMENT\_SCHEDULE . SHIPMENT\_ID — FUEL\_COST\_LOGS . SHIPMENT\_ID
- "Shipment-IOT"**:
  - SHIPMENT\_SCHEDULE . VESSEL\_ID — IOT\_TELEMETRY . VESSEL\_ID

11. Ulangi Langkah 1 – 10 untuk Cortex Analyst dengan title Weather Observations agar sesuai dengan gambar di bawah ini untuk sisi Relationship & Logical Tables

**WEATHER\_OBSERVATIONS\_SEMANTIC**

Custom Instructions (1) Edit

Logical tables (3) +

**SHIPMENT\_SCHEDULE** Edit (3)

This table stores information about shipment schedules, including the unique identifier for each shipment, the vessel and cargo involved, the planned departure and arrival times, the actual arrival time, weather conditions, and a flag indicating whether the shipment is ongoing.

> Dimensions (6) +  
> Time Dimensions (3) +  
> Facts (0) +  
> Named Filters (0) +  
> Metrics (0) +

**WEATHER\_FORECAST** Edit (1)

This table stores historical weather forecast data, capturing various conditions and measurements at specific points in time. It includes details on forecasted weather conditions, temperature, wave height, weather risk index, and wind speed, all tied to a specific timestamp.

> Dimensions (1) +  
> Time Dimensions (1) +  
> Facts (4) +  
> Named Filters (0) +  
> Metrics (0) +

**WEATHER\_OBSERVATION**

Edit (1)

This table stores weather observation data collected from various vessels. Each record represents a single observation, including the vessel's ID, the timestamp of the observation, and various weather-related metrics such as weather condition, wind speed, wave height, temperature, and a calculated weather risk index.

> Dimensions (2) +  
> Time Dimensions (1) +  
> Facts (4) +  
> Named Filters (0) +  
> Metrics (0) +

**Relationships** (2) +

**"Weather Forecast - Shipment"** Edit (1)

SHIPMENT\_SCHEDULE . DEPARTURE\_TIME — WEATHER\_FORECAST . TIMESTAMP

**"Weather Observation - Shipment"** Edit (1)

SHIPMENT\_SCHEDULE . VESSEL\_ID — WEATHER\_OBSERVATION . VESSEL\_ID

12. Ulangi Langkah 1 – 10 untuk Cortex Analyst dengan title Asset Health agar sesuai dengan gambar di bawah ini untuk sisi Relationship & Logical Tables

**ASSET\_HEALTH\_SEMANTIC**

Custom Instructions (1) Edit

**Logical tables** (7) +

- CARGO\_CHECK\_POST**

This table stores information about the inspection results of cargo shipments, including any defects or conditions noted during the inspection, and links the inspection results to the specific shipment, vessel, and cargo being transported.

  - > Dimensions 5
  - > Time Dimensions 0
  - > Facts 0
  - > Named Filters 0
  - > Metrics 0
- CARGO\_CHECK\_PRE**

This table stores pre-shipment inspection data for cargo, including any defects or conditions noted during the inspection, as well as identifiers for the shipment, vessel, and cargo being inspected.

  - > Dimensions 5
  - > Time Dimensions 0
  - > Facts 0
  - > Named Filters 0
  - > Metrics 0

**INCIDENT\_REPORTS**

This table stores information about incidents that have occurred on vessels, including details about the incident, the actions taken, and the impact of the incident. It also captures information about the vessel and the person who reported the incident, as well as any recommendations for future prevention.

- > Dimensions 9
- > Time Dimensions 1
- > Facts 0
- > Named Filters 0
- > Metrics 0

**MAINTENANCE\_CHECK**

This table stores maintenance check records for vessels, tracking various parameters such as vibration levels, temperature, and oil pressure, along with the condition of the machine and any recommended maintenance actions, all tied to a specific shipment, vessel, and cargo.

- > Dimensions 6
- > Time Dimensions 1
- > Facts 3
- > Named Filters 0
- > Metrics 0

**MAINTENANCE\_LOGS**

This table stores records of maintenance activities performed on vessels, including the vessel ID, timestamp of the maintenance, type of issue addressed, and the severity level of the issue.

- > Dimensions 2
- > Time Dimensions 1
- > Facts 1
- > Named Filters 0
- > Metrics 0

**SHIPMENT\_SCHEDULE**

This table stores information about shipment schedules, including the unique identifier for each shipment, the vessel and cargo involved, the planned departure and arrival times, the actual arrival time, weather conditions, and a flag indicating whether the shipment is ongoing.

- > Dimensions 6
- > Time Dimensions 3
- > Facts 0
- > Named Filters 0
- > Metrics 0

**TRANSCRIBED\_TEXT**

This table stores transcribed text data associated with specific incidents, where each Incident is uniquely identified by an incident ID and the corresponding transcribed text is stored in a separate column.

- > Dimensions 0
- > Time Dimensions 0
- > Facts 0
- > Named Filters 0
- > Metrics 0

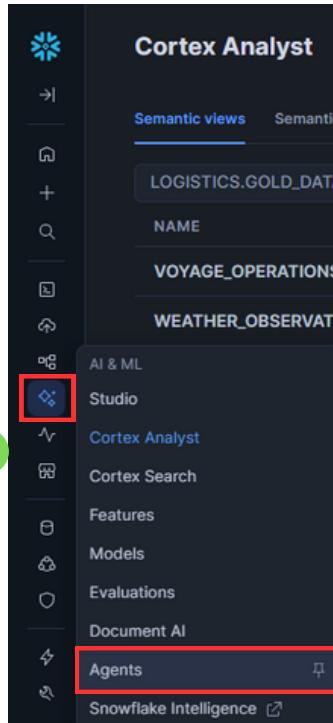
**Relationships** (8) +

- "Incident-Crew Logs"**
  - INCIDENT\_REPORTS . INCIDENT\_ID — TRANSCRIBED\_TEXT . INCIDENT\_ID
- "Shipment-Cargo\_Post"**
  - SHIPMENT\_SCHEDULE . CARGO\_ID — CARGO\_CHECK\_POST . CARGO\_ID
  - SHIPMENT\_SCHEDULE . SHIPMENT\_ID — CARGO\_CHECK\_POST . SHIPMENT\_ID
  - SHIPMENT\_SCHEDULE . VESSEL\_ID — CARGO\_CHECK\_POST . VESSEL\_ID
- "Shipment-Cargo\_Pre"**
  - SHIPMENT\_SCHEDULE . CARGO\_ID — CARGO\_CHECK\_PRE . CARGO\_ID
  - SHIPMENT\_SCHEDULE . SHIPMENT\_ID — CARGO\_CHECK\_PRE . SHIPMENT\_ID
  - SHIPMENT\_SCHEDULE . VESSEL\_ID — CARGO\_CHECK\_PRE . VESSEL\_ID
- "Shipment-Incident"**
  - SHIPMENT\_SCHEDULE . VESSEL\_ID — INCIDENT\_REPORTS . VESSEL\_ID
- "Shipment-Maintenance Checks"**
  - SHIPMENT\_SCHEDULE . CARGO\_ID — MAINTENANCE\_CHECK . CARGO\_ID
  - SHIPMENT\_SCHEDULE . SHIPMENT\_ID — MAINTENANCE\_CHECK . SHIPMENT\_ID
  - SHIPMENT\_SCHEDULE . VESSEL\_ID — MAINTENANCE\_CHECK . VESSEL\_ID
- "Shipment-Maintenance Logs"**
  - SHIPMENT\_SCHEDULE . VESSEL\_ID — MAINTENANCE\_LOGS . VESSEL\_ID

## 6. Cortex Agents

Saatnya membuat Cortex Agents

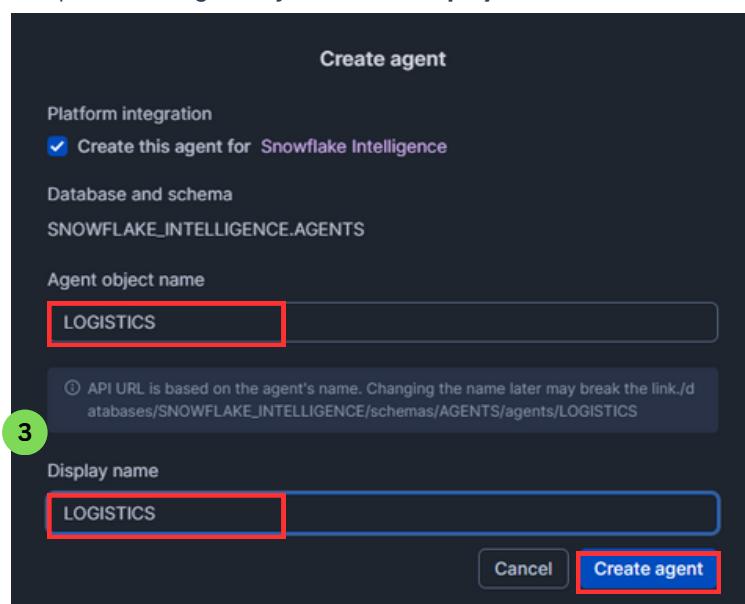
1. Klik Tab AI & ML dan setelah itu **Klik Agents** untuk **membuka tab Cortex Agent** untuk **membuat Cortex Agent Snowflake**



2. Kita akan memulai pembuatan **Cortex Agents** dengan **Klik tombol Create agent**



3. Masukkan **LOGISTICS** pada **Field Agent object name & Display name** dan **Klik tombol Create agent**



Create agent

Platform integration

Create this agent for Snowflake Intelligence

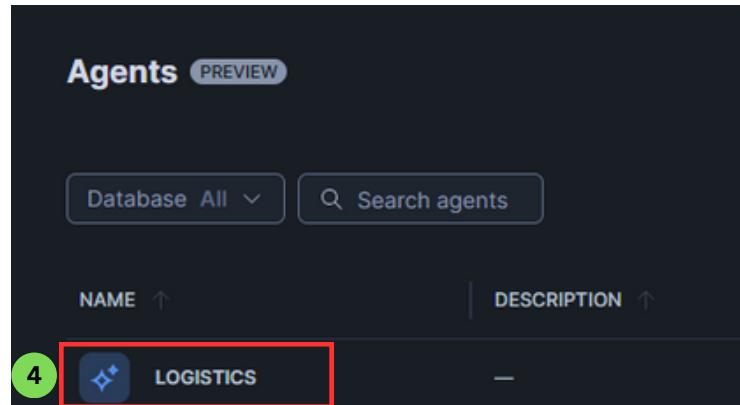
Database and schema

SNOWFLAKE\_INTELLIGENCE.AGENTS

Agent object name

Display name

4. Klik Agent LOGISTICS untuk melakukan konfigurasi Model Cortex Agents



The screenshot shows the 'Agents' interface with a dark theme. At the top, there is a 'PREVIEW' button. Below it are two search/filter boxes: 'Database All' and 'Search agents'. A table lists agents with columns 'NAME' and 'DESCRIPTION'. The 'LOGISTICS' agent is highlighted with a red box and a green number '4' to its left. The table header has arrows indicating sorting for both columns.

5. Klik tombol Edit untuk memulai konfigurasi Model Cortex Agents

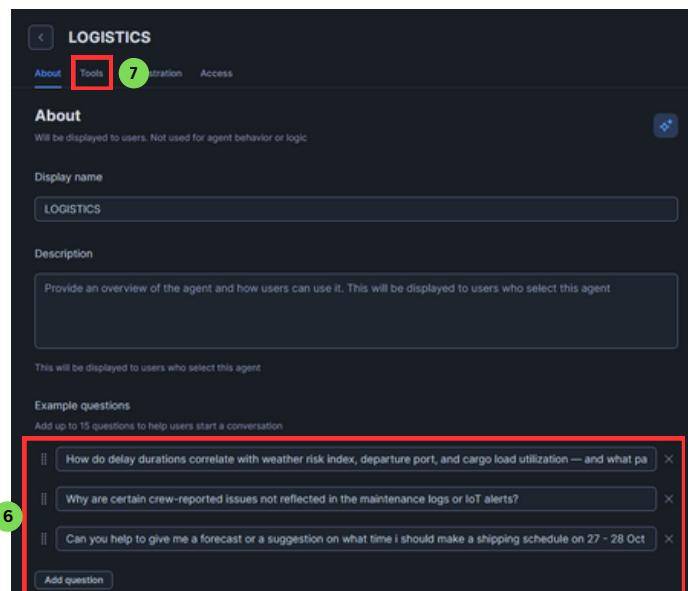


The screenshot shows the 'LOGISTICS' agent details page. At the top, there is a back arrow, the agent name 'LOGISTICS', and a date 'Oct 30, 2025'. Below that are tabs for 'Overview' and 'Monitoring'. On the right side, there is an 'Edit' button, a 'Preview in Snowflake Intelligence' dropdown, and a help icon. A green number '5' is placed next to the 'Edit' button.

6. Masukkan 3 Example Questions & Klik Tombol Add Questions untuk menambahkan ketiga pertanyaan

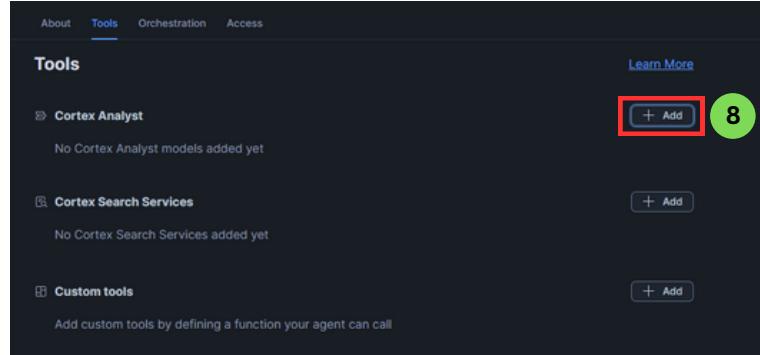
1. How do delay durations correlate with weather risk index, departure port, and cargo load utilization — and what patterns explain why certain vessels face longer disruptions than others?
2. Why are certain crew-reported issues not reflected in the maintenance logs or IoT alerts?
3. Can you help to give me a forecast or a suggestion on what time I should make a shipping schedule on 27 - 28 Oct 2025 based on past weather data and the forecasted weather data — give me the reasons as to why you think its the optimal window, based on past and the forecasted weather data

7. Klik Tools untuk melanjutkan Konfigurasi Cortex Agents



The screenshot shows the 'Tools' configuration page for the 'LOGISTICS' agent. It includes sections for 'About' (with a green number '7' next to the 'Registration' tab), 'Example questions' (with a red box and a green number '6' next to the input area), and 'Add question' (a button at the bottom). Three example questions are listed in the input field, each preceded by a green number (1, 2, 3).

8. Klik tombol + Add pada Cortex Analyst

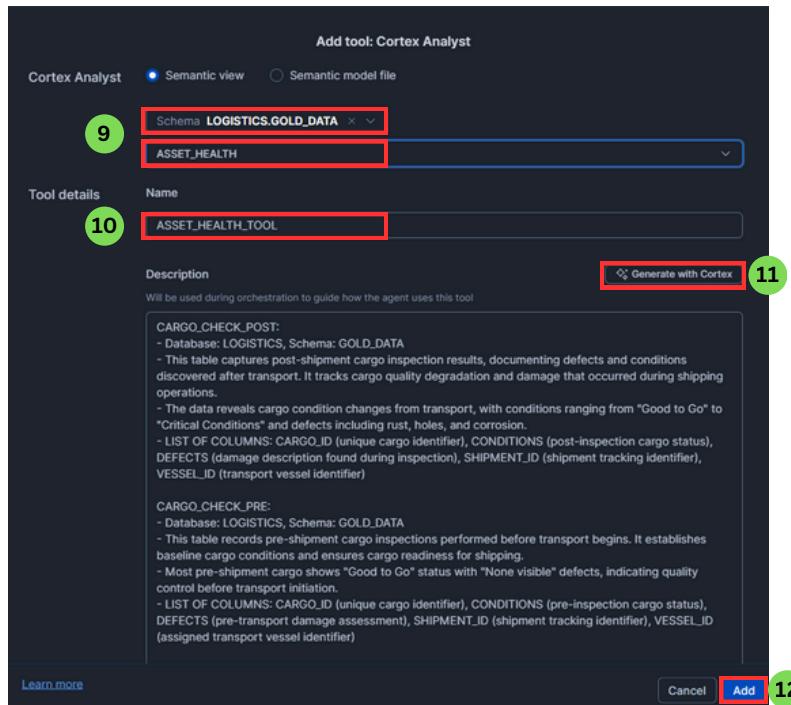


9. Pastikan Schema yang digunakan adalah LOGISTICS.GOLD\_DATA dan memilih Model ASSET\_HEALTH

10. Masukkan ASSET\_HEALTH\_TOOL pada nama Tool Details

11. Klik tombol Generate with Cortex tuntuk memberikan deskripsi yang dapat digunakan oleh Cortex Agents

12. Klik Add untuk menambahkan Cortex Analyst yang pertama



Add tool: Cortex Analyst

Cortex Analyst  Semantic view  Semantic model file

9 Schema LOGISTICS.GOLD\_DATA

10 ASSET\_HEALTH

Name ASSET\_HEALTH\_TOOL

Description Will be used during orchestration to guide how the agent uses this tool

Generate with Cortex

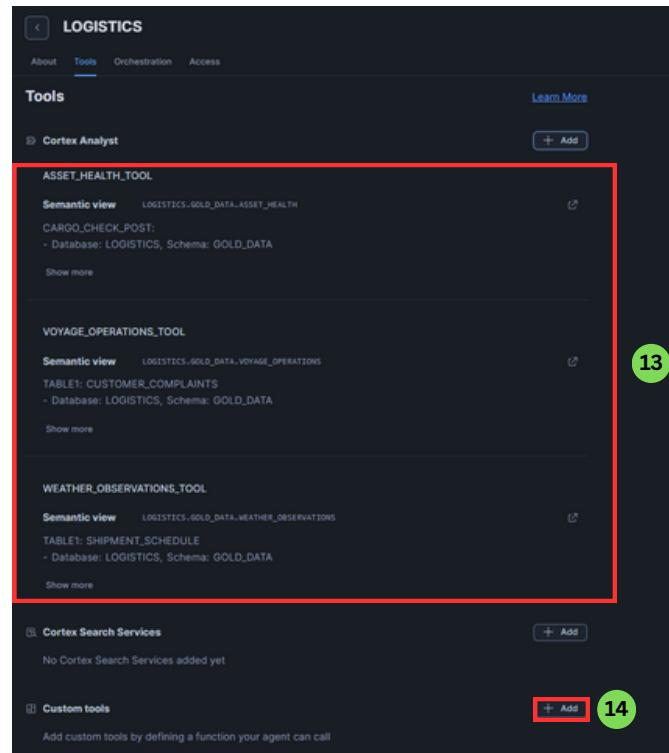
CARGO\_CHECK\_POST:  
- Database: LOGISTICS, Schema: GOLD.DATA  
- This table captures post-shipment cargo inspection results, documenting defects and conditions discovered after transport. It tracks cargo quality degradation and damage that occurred during shipping operations.  
- The data reveals cargo condition changes from transport, with conditions ranging from "Good to Go" to "Critical Conditions" and defects including rust, holes, and corrosion.  
- LIST OF COLUMNS: CARGO\_ID (unique cargo identifier), CONDITIONS (post-inspection cargo status), DEFECTS (damage description found during inspection), SHIPMENT\_ID (shipment tracking identifier), VESSEL\_ID (transport vessel identifier)

CARGO\_CHECK\_PRE:  
- Database: LOGISTICS, Schema: GOLD.DATA  
- This table records pre-shipment cargo inspections performed before transport begins. It establishes baseline cargo conditions and ensures cargo readiness for shipping.  
- Most pre-shipment cargo shows "Good to Go" status with "None visible" defects, indicating quality control before transport initiation.  
- LIST OF COLUMNS: CARGO\_ID (unique cargo identifier), CONDITIONS (pre-inspection cargo status), DEFECTS (pre-transport damage assessment), SHIPMENT\_ID (shipment tracking identifier), VESSEL\_ID (assigned transport vessel identifier)

Learn more Cancel Add

13. Lakukan Langkah 8 – 12 untuk Cortex Analyst agar sesuai dengan gambar di bawah ini

14. Klik tombol + Add pada Custom Tools



The screenshot shows the 'Tools' section of the Cortex Analyst interface. It lists three semantic views:

- ASSET\_HEALTH\_TOOL**: Semantic view LOGISTICS.GOLD\_DATA..ASSET\_HEALTH. CARGO\_CHECK\_POST: - Database: LOGISTICS, Schema: GOLD\_DATA.
- VOYAGE\_OPERATIONS\_TOOL**: Semantic view LOGISTICS.GOLD\_DATA.VOYAGE\_OPERATIONS. TABLE1: CUSTOMER\_COMPLAINTS: - Database: LOGISTICS, Schema: GOLD\_DATA.
- WEATHER\_OBSERVATIONS\_TOOL**: Semantic view LOGISTICS.GOLD\_DATA.WEATHER\_OBSERVATIONS. TABLE1: SHIPMENT\_SCHEDULE: - Database: LOGISTICS, Schema: GOLD\_DATA.

Below these, there are sections for 'Cortex Search Services' (No Cortex Search Services added yet) and 'Custom tools' (Add custom tools by defining a function your agent can call). A green circle labeled '13' is positioned to the right of the semantic view list, and another green circle labeled '14' is positioned below the 'Custom tools' section.

15. Pastikan sesuai dengan di gambar bawah ini untuk semua field

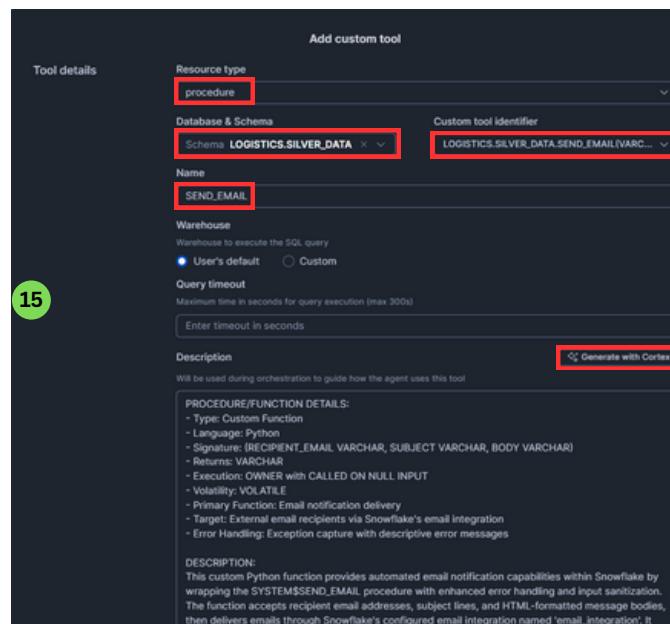
**Resource type** : procedure

**Database & Schema** : LOGISTICS.SILVER\_DATA

**Tools** : LOGISTICS.SILVER\_DATA.SEND\_EMAIL

**Name** : SEND\_EMAIL

dan Klik Generate with Cortex



The screenshot shows the 'Add custom tool' dialog box. In the 'Tool details' section, the 'Resource type' is set to 'procedure', 'Database & Schema' is set to 'Schema: LOGISTICS.SILVER\_DATA', and the 'Name' is set to 'SEND\_EMAIL'. A green circle labeled '15' is positioned to the left of the dialog. In the 'Description' section, there is a note: 'Will be used during orchestration to guide how the agent uses this tool' followed by a 'Generate with Cortex' button. The 'PROCEDURE/FUNCTION DETAILS:' section contains the following Python code:

```
PROEDURE/FUNCTION DETAILS:
- Type: Custom Function
- Language: Python
- Signature: (RECIPIENT_EMAIL VARCHAR, SUBJECT VARCHAR, BODY VARCHAR)
- Returns: VARCHAR
- Execution: OWNER with CALLED ON NULL INPUT
- Volatility: VOLATILE
- Primary Function: Email notification delivery
- Target: External email recipients via Snowflake's email integration
- Error Handling: Exception capture with descriptive error messages

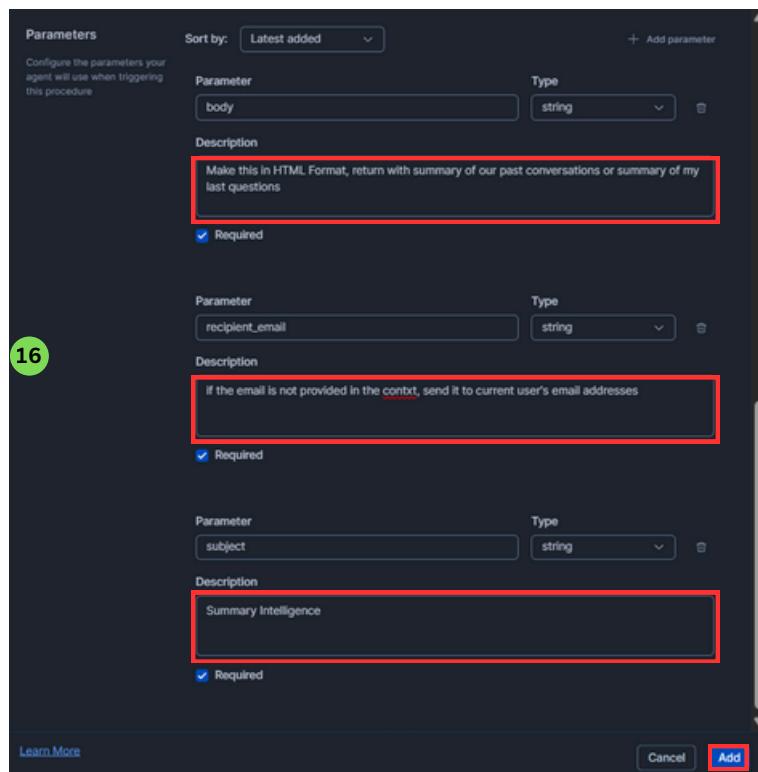
DESCRIPTION:
This custom Python function provides automated email notification capabilities within Snowflake by wrapping the SYSTEM$SEND_EMAIL procedure with enhanced error handling and input sanitization. The function accepts recipient email addresses, subject lines, and HTML-formatted message bodies, then delivers emails through Snowflake's configured email integration named 'email_integration'. It
```

16. Pastikan sesuai dengan di gambar bawah ini untuk semua field  
**body** : Make this in HTML Format, return with summary of our past conversations or summary of my last questions

**recipient\_email** : if the email is not provided in the context, send it to current user's email addresses

**subject** : Summary Intelligence

dan Klik Add untuk Finalize Custom Tools



The screenshot shows a 'Parameters' configuration interface with three entries:

- body**: Type: string. Description: Make this in HTML Format, return with summary of our past conversations or summary of my last questions.
- recipient\_email**: Type: string. Description: If the email is not provided in the context, send it to current user's email addresses.
- subject**: Type: string. Description: Summary Intelligence.

A green circle with the number 16 is overlaid on the left side of the first parameter entry. At the bottom right of the interface are 'Cancel' and 'Add' buttons, with 'Add' being highlighted by a red border.

17. Klik Tombol **Orchestration** untuk membuka Tab Orchestration

18. Masukkan **Orchestration Instructions:**

**"There are 3 Cortex Analyst and 1 Cortex Search in this tools**

**Cortex Analyst:**

**1. Voyage Operations: to know day to day operations of voyage operations or shipping operations**

**2. Weather Observations: to know about the weather conditions on the shipping time and the forecasted**

**3. Asset Health: to know about the conditions of the ship, container, the results of maintenance, etc regarding the logistics asset**

**Custom Tools:**

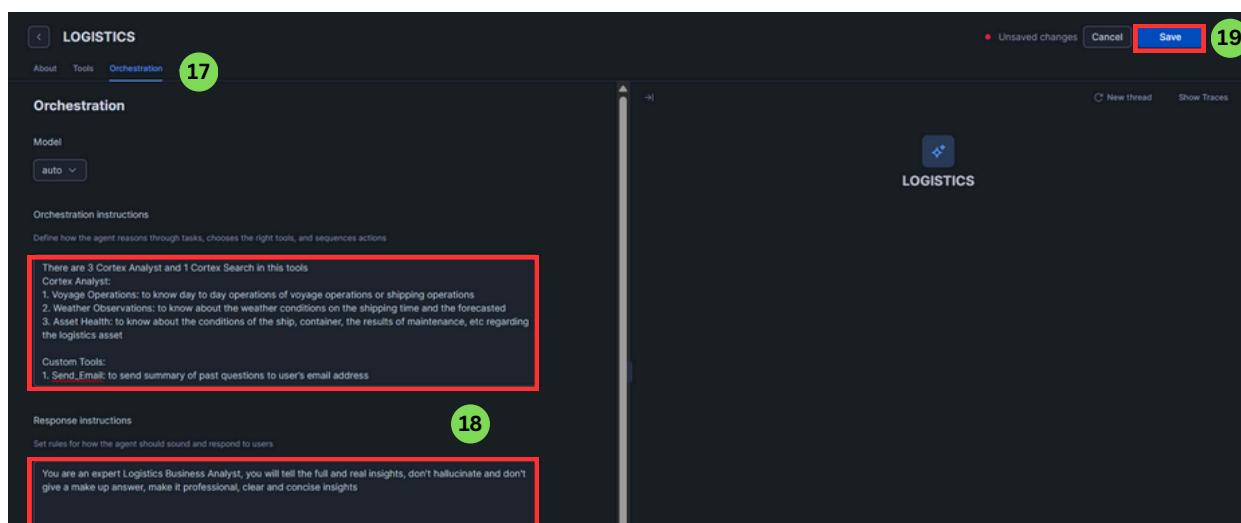
**1. Send\_Email: to send summary of past questions to user's email address"**

dan

**Response Instructions:**

**"You are an expert Logistics Business Analyst, you will tell the full and real insights, don't hallucinate and don't give a make up answer, make it professional, clear and concise insights"**

19. Klik Save untuk finalize Cortex Agents



Selamat, kita berhasil menyelesaikan Use Case Setup Hackathon ini!

Sampai jumpa di Use Case Setup Hackathon Snowflake berikutnya! ✨😊