

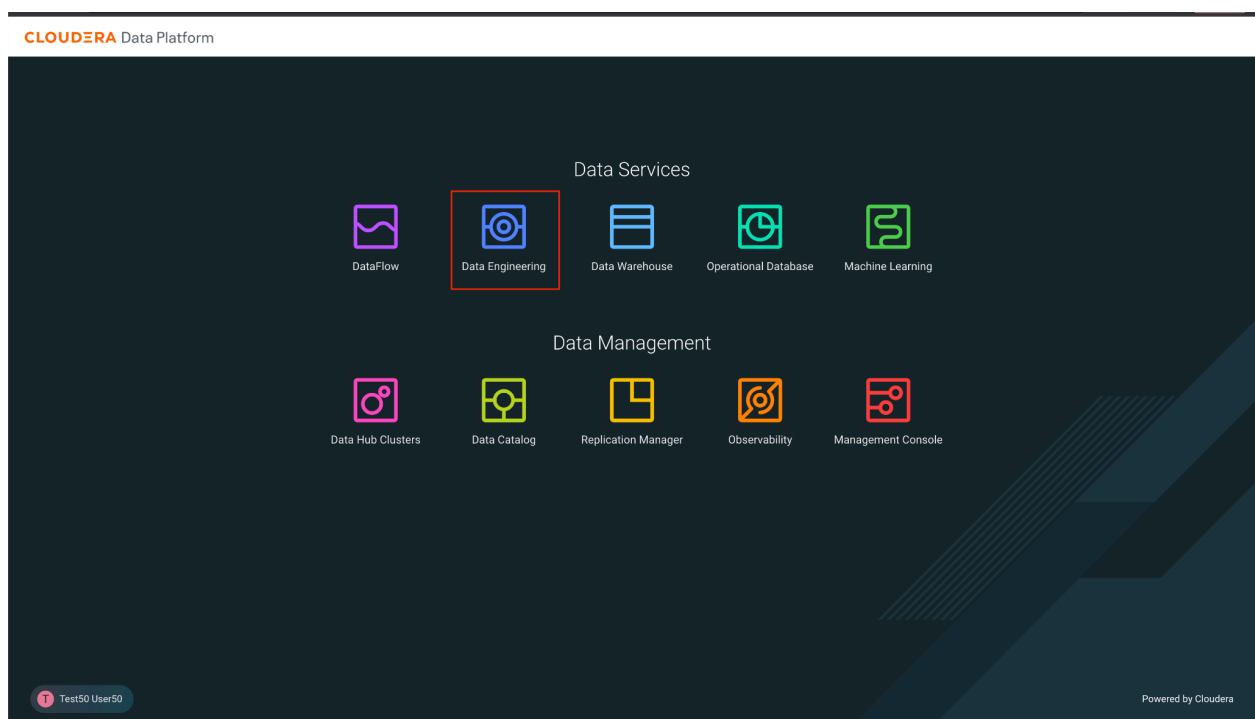
Data Lifecycle en CDP Public Cloud

Data Engineering Lab

Goals:

- Run a data enrichment process
- Run a process to simulate changes to the data
- Configure the execution of a pipeline using low-code/no-code tools

1. Click on DataFlow from CDP PC Home:



2. The Data Engineering Home shows all the actions that can be done, such as Jobs in Spark and pipelines in Airflow, Resources and useful information/documentation. Click on the option **Jobs** from the left menu to create a dataflow in Airflow.

The screenshot shows the Cloudera Data Engineering (CDE) Home dashboard. On the left is a dark sidebar with navigation links: Home, Jobs, Job Runs, Sessions (with a 'Preview' badge), Resources, and Administration. The main content area is titled 'Home' and 'Welcome, Test50'. It features three primary action cards: 'Create' (for jobs and sessions), 'Resources' (for file and python resources), and 'Docs & Downloads' (for references and downloads). Below these is a 'Virtual Clusters' section showing an 'aws ssa-de' cluster with a table of metrics.

CLUSTERA
Data Engineering

Home

Jobs

Job Runs

Sessions **Preview**

Resources

Administration

Help

Test50 User50

1.19.0-b427

Home

Welcome, Test50

Create
Create jobs, orchestrate them or start a session.

- Spark Jobs**
[Create New](#)
[Schedule](#)
[Ad-Hoc Run](#)
- Airflow Pipelines**
[Upload DAG file](#)
[Build a Pipeline](#) **New**
- Sessions** **New**
[Start New](#)

Resources
Create resources for jobs.

- File**
[Create New](#)
- Python**
[Create New](#)

Docs & Downloads
CDE documentation and tools.

- References**
[API Doc](#)
[Product Doc](#)
[Release Notes](#)
- Downloads**
[CLI Client](#)
[Migration Tool](#) **New**

Virtual Clusters
Autoscaling Spark clusters to run Jobs.

aws **ssa-de**

ssa-de-cluster [View Jobs](#)

Spark 3.2.3

CPU	MEMORY	JOBS
0	0 MB	0

Feedback

3. Here the available tasks are listed. For the purposes of this workshop, two Jobs have been configured:

- **CDE-Table-Update**, generate random changes and enrich table to visualize Lakehouse Time Travel functionality.
- **CDE-Data-Enrichment**, process in Spark (Python) to enrich the data ingested from Kafka and save to a new table.

It is time to create our Job in Airflow. Click on **Create Job**.

The screenshot displays the Cloudera Data Engineering interface. On the left is a dark sidebar with navigation links: Home, Jobs (highlighted), Job Runs, Sessions (with a 'Preview' button), Resources, and Administration. At the bottom of the sidebar are links for Help and the user 'Test50 User50', along with the version '1.19.0-b427'. The main content area is titled 'aws ssa-de-cluster / Jobs'. It features a search bar, filter dropdowns for 'Status' and 'Type', and a 'Create Job' button. Below these is a table with columns: Status, Job, Type, Schedule, Modified On, and Actions. Two jobs are listed: '_CDE-Table-Update' and '_CDE-Data-Enrichment', both of type 'Spark' and schedule 'Ad-Hoc'. The table also shows the modification time for each job. At the bottom right of the table area, there is a pagination control showing 'Items per page: 10' and '1 - 2 of 2'.

Status	Job	Type	Schedule	Modified On	Actions
⌚	_CDE-Table-Update	Spark	Ad-Hoc	May 26, 2023, 12:22:35 PM	⋮
⌚	_CDE-Data-Enrichment	Spark	Ad-Hoc	May 26, 2023, 12:22:21 PM	⋮

4. In the Job creation form, you must enter the following information:

- Job Type: Airflow
- Name: Use the naming <assigned user>-pipeline. Replace <assigned user> with the user assigned to you. For example, user050
- DAG: Editor, to graphically configure the task.

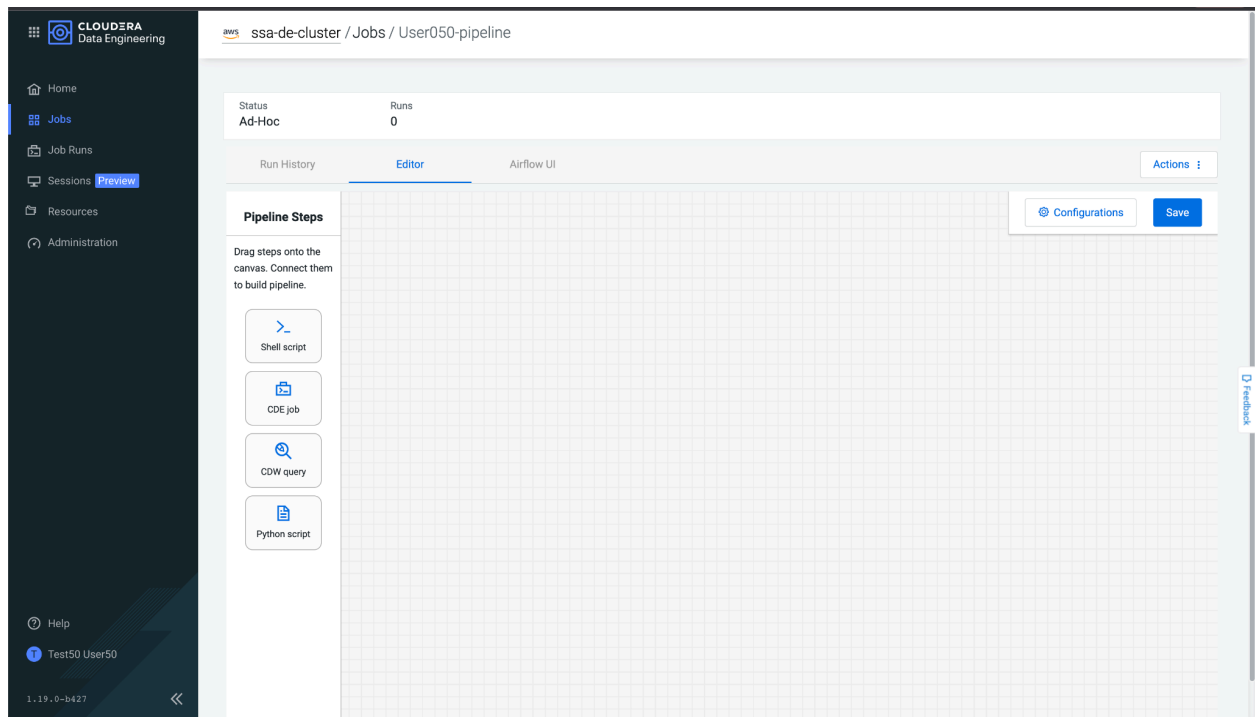
Once entering the values correctly, click on **Create**.

The screenshot shows the Cloudera Data Engineering interface for creating a new job. The left sidebar contains navigation links: Home, Jobs (highlighted), Job Runs, Sessions (with a Preview button), Resources, and Administration. At the bottom of the sidebar are links for Help and a user profile for 'Test50 User50'. The main content area is titled 'aws ssa-de-cluster / Jobs / Create Job'. Under the 'Job Details' section, the 'Job Type' is set to 'Airflow' (selected with a radio button). The 'Name' field contains 'User050-pipeline'. The 'DAG' is set to 'Editor' (selected with a radio button). At the bottom of the form are 'Cancel' and 'Create' buttons. A 'Feedback' link is visible on the right side of the form.

5. IMPORTANT!!!

If the editor page is NOT loading, go to the bottom of this doc for a workaround, if it does, continue as usual

On the Job editing screen, select the Editor tab, and you will see the following canvas to drag the steps of the pipeline that we are going to create. In our case, we are going to create two CDE Jobs and relate them.



6. Let's start with the first Job. Click on the CDE Job button and drag onto the canvas, entering the following settings:

- **title/name:** Data Enrichment
- **Select Job:** select the Job *_CDE-Data-Enrichment*
- Check the checkbox **Override Spark values**. Additional options will appear below.
- **Arguments:** <assigned user>. Use the username assigned to you. For example, user050

The screenshot displays the Cloudera Data Engineering (CDE) interface. On the left is a dark sidebar with navigation links: Home, Jobs, Job Runs, Sessions (highlighted with a 'Preview' button), Resources, and Administration. At the bottom of the sidebar are 'Help' and 'Test50 User50' (with a user icon), and the version '1.19.0-b427'. The main area has a header 'aws ssa-de-cluster / Jobs / User050-pipeline'. Below this is a status bar showing 'Status: Ad-Hoc' and 'Runs: 0'. A tabbed interface shows 'Run History', 'Editor' (active), and 'Airflow UI'. The 'Editor' tab contains a grid canvas with a 'Data Enrichment' job icon placed on it. To the left of the canvas is a 'Pipeline Steps' panel with icons for 'Shell script', 'CDE job', 'CDW query', and 'Python script'. To the right is a configuration panel for the 'Data Enrichment' job, with tabs for 'Configure' and 'Advanced'. The 'Configure' tab is active and shows: 'Select Job' set to '_CDE-Data-Enrichment'; a 'Variables' section with 'Name' and 'Value' input fields; 'Override Spark values' checked; 'Arguments' set to 'user050'; and empty fields for 'Configurations', 'Executors', 'Driver Cores', and 'Executor cores'. A 'Feedback' button is on the far right.

7. Configure the second Job. Click on the CDE Job button and drag onto the canvas, entering the following settings:

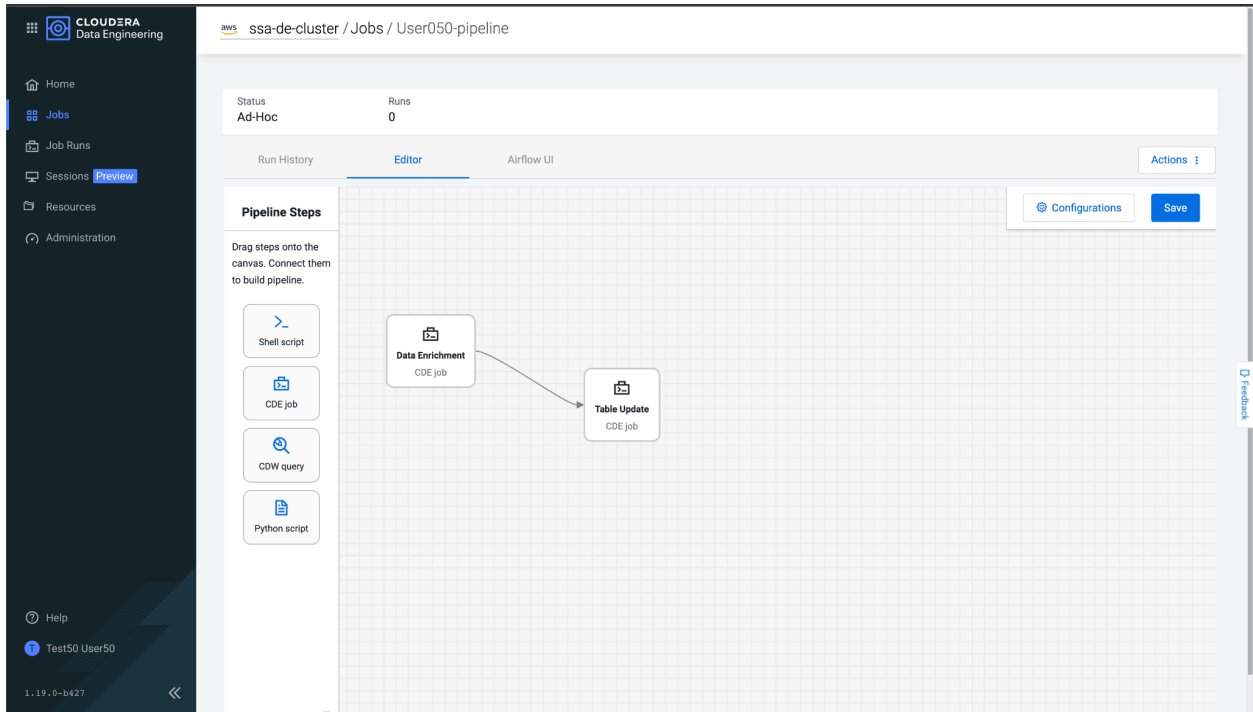
- **title/name:** Table Update
- **Select Job:** select the Job_ *CDE-Table-Update*
- Check the checkbox **Override Spark values**. Additional options will appear below.
- **Arguments:** <assigned user>. Use the username assigned to you. For example, user050

The screenshot displays the Cloudera Data Engineering (CDE) interface. On the left is a dark sidebar with navigation links: Home, Jobs, Job Runs, Sessions (with a 'Preview' button), Resources, and Administration. At the bottom of the sidebar, it shows 'Test50 User50' and the version '1.19.0-b427'. The main area is titled 'aws ssa-de-cluster / Jobs / User050-pipeline'. It features a status bar with 'Status: Ad-Hoc' and 'Runs: 0'. Below this are tabs for 'Run History', 'Editor' (which is active), and 'Airflow UI'. An 'Actions' menu is visible in the top right. The 'Editor' tab shows a canvas with a grid. On the left of the canvas is a 'Pipeline Steps' panel with icons for 'Shell script', 'CDE job', 'CDW query', and 'Python script'. Two 'CDE job' steps are on the canvas: 'Data Enrichment' and 'Table Update'. The 'Table Update' step is selected, and its configuration panel is open on the right. This panel has tabs for 'Configure' and 'Advanced'. Under 'Configure', the 'Select Job' dropdown is set to '_CDE-Table-Update'. There is a 'Variables' section with a table for Name and Value. The 'Override Spark values' checkbox is checked. The 'Arguments' field contains 'user050'. There are also fields for 'Configurations', 'Executors', 'Driver Cores', and 'Executor cores'. A 'Feedback' button is located on the far right edge of the configuration panel.

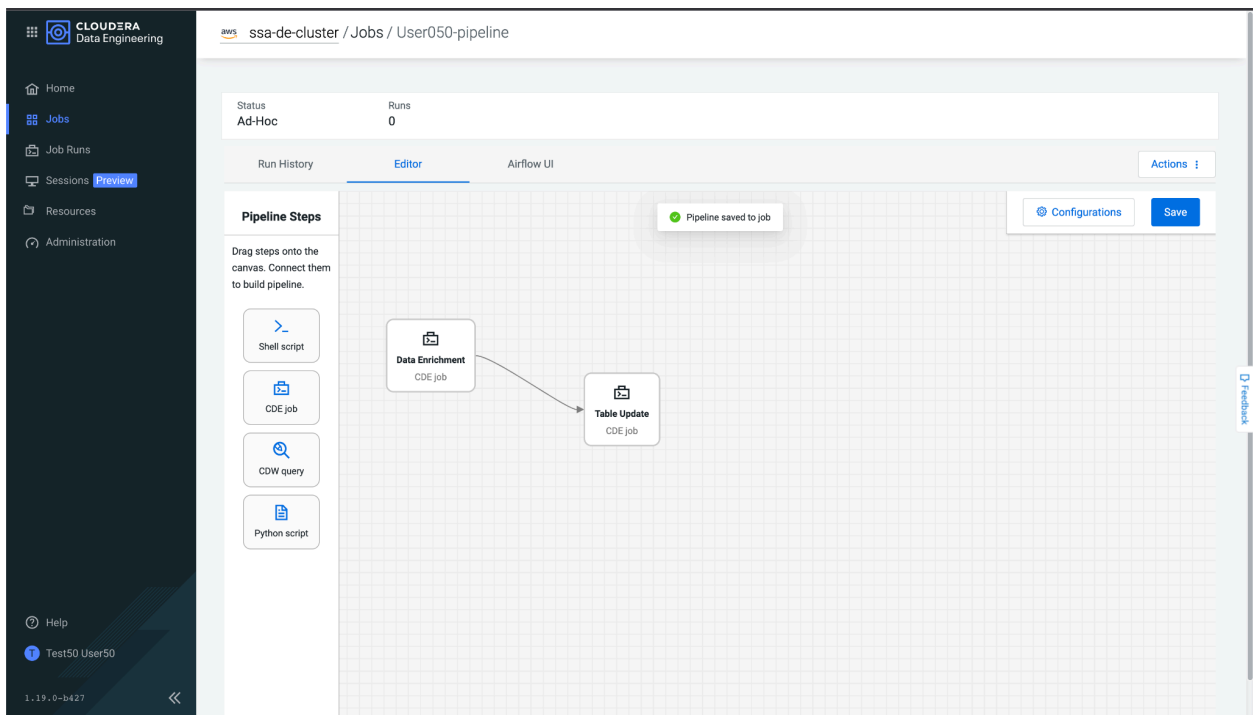
8. To set up the execution sequence, bind **Data Enrichment** with **Table Update**. For that, click on the right connector of the job of **Data Enrichment** and drag to the left connector of **Table Update**.

The screenshot shows the Cloudera Data Engineering interface. On the left is a sidebar with navigation links: Home, Jobs, Job Runs, Sessions (with a 'Preview' button), Resources, and Administration. Below these are 'Help' and 'Test50 User50'. The main area is titled 'aws ssa-de-cluster / Jobs / User050-pipeline'. It has tabs for 'Run History', 'Editor' (selected), and 'Airflow UI'. At the top right of the editor are 'Configurations' and 'Save' buttons. On the left of the editor is a 'Pipeline Steps' panel with instructions: 'Drag steps onto the canvas. Connect them to build pipeline.' It lists four step types: Shell script, CDE job, CDW query, and Python script. On the canvas, there are two job nodes: 'Data Enrichment CDE job' and 'Table Update CDE job'. The 'Data Enrichment' node has a red box around its right connector, indicating it is the target for the next step.

This screenshot shows the same Cloudera Data Engineering interface as the previous one, but with the pipeline updated. The 'Data Enrichment CDE job' and 'Table Update CDE job' are now connected by a line, indicating that the execution sequence has been established. The 'Table Update' node is highlighted with a green border. All other interface elements, including the sidebar, top navigation, and 'Pipeline Steps' panel, remain the same.



9. Once the Jobs have been joined, click on **Save** to save the settings made. You should see a message indicating **Pipeline saved to job**.



10. The time has come to run the pipeline. On the upper right side of the canvas, click **Actions** -> **Run Now**.

CloudERA Data Engineering

aws ssa-de-cluster / Jobs / User050-pipeline

Status: Ad-Hoc, Runs: 0

Run History | Editor | Airflow UI

Pipeline Steps

Drag steps onto the canvas. Connect them to build pipeline.

Shell script, CDE job, CDW query, Python script

Data Enrichment CDE job

Table Update CDE job

Actions: Run Now, Delete

Feedback

11. You should see the pipeline execution screen, indicating that the execution has been initialized.

CloudERA Data Engineering

aws ssa-de-cluster / Jobs / User050-pipeline

Status: Ad-Hoc, Runs: 0

Run History | Editor | Airflow UI

Duration

Search by Run Id

Status	Run ID	Duration	User	Start Time ↓	Actions
⬢	7		user050	May 26, 2023, 1:32:09 PM	⋮

Items per page: 10 1 - 1 of 1

Feedback

12. Click on the Airflow UI tab to see the execution detail of each step in the pipeline. The configured Data Enrichment and Table Update jobs are listed at the bottom left. The colors indicate the status of each job. Make sure the radio button **Auto-refresh** is enabled to automatically display the status of jobs.

The screenshot shows the Cloudera Data Engineering interface. The sidebar on the left contains navigation links: Home, Jobs, Job Runs, Sessions (highlighted with a 'Preview' button), Resources, and Administration. The main content area is titled 'aws ssa-de-cluster / Jobs / User050-pipeline'. It displays the pipeline status as 'Ad-Hoc' with '0' runs. Below this, there are tabs for 'Run History', 'Editor', and 'Airflow UI' (which is selected). The 'Airflow UI' tab shows the DAG for 'User050_pipeline'. At the top right of the DAG view, it says 'Schedule: None' and 'Next Run: None'. Below the DAG view, there is a 'Clear Filters' button and a list of job run statuses: (deferred), (failed), (queued), (running), (scheduled), (skipped), (success), (up_for_reschedule), (up_for_retry), (upstream_failed), and (no_status). The 'Auto-refresh' toggle is enabled. At the bottom left, a table lists the jobs: 'Data_Enrichment' and 'Table_Update'. On the right side, there is a 'DAG Details' section with a 'DAG Runs Summary' table. The summary table shows 'Total Runs Displayed' as 1, 'Total running' as 1, 'First Run Start' as 2023-05-26, 18:32:10 UTC, 'Last Run Start' as 2023-05-26, 18:32:10 UTC, and 'Max Run Duration' as 00:00:21.

DAG Runs Summary	
Total Runs Displayed	1
Total running	1
First Run Start	2023-05-26, 18:32:10 UTC
Last Run Start	2023-05-26, 18:32:10 UTC
Max Run Duration	00:00:21

13. You can see more information about the execution by clicking on the view **Graph**. Hovering the mouse over the Job name displays specific information for each step in the pipeline. Make sure the pipeline status is Success, which indicates that the entire pipeline was able to run without issue.

The screenshot displays the Cloudera Data Engineering interface for the 'User050_pipeline' DAG. The interface includes a sidebar with navigation options like Home, Jobs, Job Runs, Sessions, Resources, and Administration. The main panel shows the pipeline's status as 'SUCCESS' and provides various views (Grid, Graph, Calendar, etc.). A tooltip for the 'Data_Enrichment' task is visible, showing its status as 'success' and providing details such as Task ID, Run ID, Operator, and Duration. The DAG diagram shows two tasks: 'Data_Enrichment' and 'Table_Update'.

Cloudera Data Engineering

aws ssa-de-cluster / Jobs / User050-pipeline

Status: Ad-Hoc | Runs: 1

Run History | Editor | **Airflow UI** | Actions

DAG: User050_pipeline success Schedule: None | Next Run: None

Grid | **Graph** | Calendar | Task Duration | Task Tries | Landing Times | Gantt | Details | <> Code

Audit Log

2023-05-26T18:32:11Z | Runs: 25 | Run | cde-run-job-operator

Task Details:

- Status: success
- Task Id: Data_Enrichment
- Run: 2023-05-26, 18:36:24 UTC
- Run Id: cde-job-run-7
- Operator: CdeRunJobOperator
- Duration: 1Min 11.6765sec

UTC: Started: 2023-05-26, 18:33:29 | Ended: 2023-05-26, 18:34:40

Find Task... | Update

Auto-refresh

DAG Diagram: Data_Enrichment → Table_Update

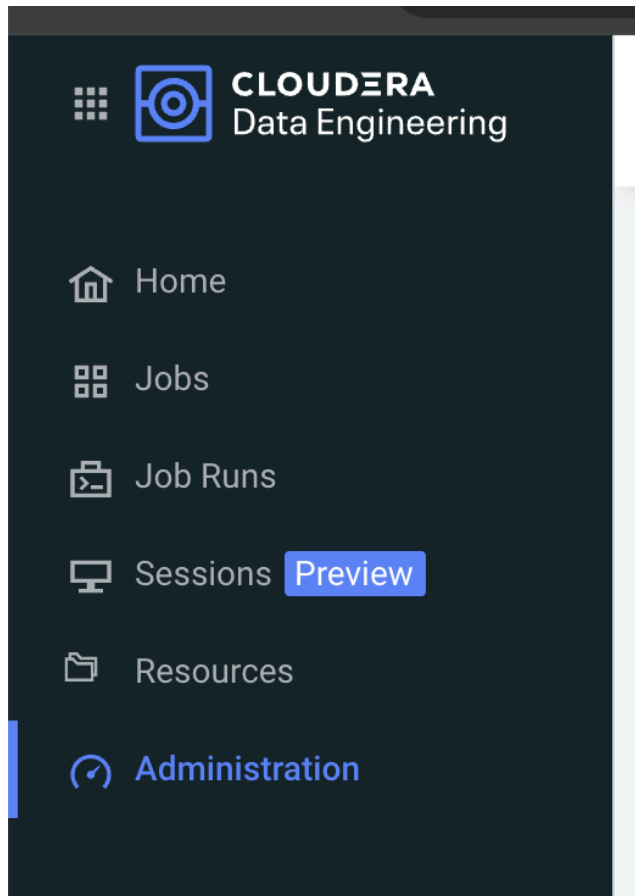
The execution status appears next to the name of the pipeline (marked in red). If it is green and indicates **Success**, it means that the execution was successful.

The screenshot displays the Cloudera Data Engineering (CDE) interface. On the left is a dark sidebar with navigation links: Home, Jobs, Job Runs, Sessions (with a 'Preview' button), Resources, and Administration. The main content area shows the 'User050_pipeline' DAG. At the top, the status is 'Ad-Hoc' and there is '1' run. Below this, the 'Airflow UI' tab is active. The DAG title 'User050_pipeline' is followed by a green 'success' status indicator (highlighted with a red box in the original image), 'Schedule: None', and 'Next Run: None'. A toolbar offers various views: Grid, Graph (selected), Calendar, Task Duration, Task Times, Landing Times, Gantt, Details, and Code. An 'Audit Log' link is also present. A filter bar shows a date '2023-05-26T18:32:11Z', a dropdown for 'Runs' set to '25', and a dropdown for 'Run' set to 'cde-job-run-7'. A 'Layout' button is on the right. Below the filter bar, a tooltip for 'Status: success' is visible, containing details: Task Id: Table_Update, Run: 2023-05-26, 18:36:36 UTC, Run Id: cde-job-run-7, Operator: CdeRunJobOperator, Duration: 1 Min 1.533Sec, and UTC timestamps for Start and End. The DAG graph shows two tasks: 'Data_Enrichment' and 'Table_Update'. A 'Feedback' button is on the far right. The bottom left corner shows '1.19.0-b427' and a back arrow.

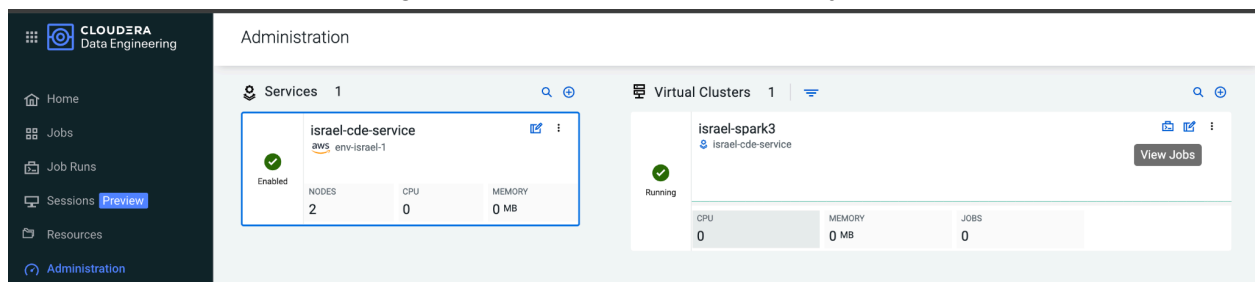
End of Lab 2

Workaround for Editor page not loading:

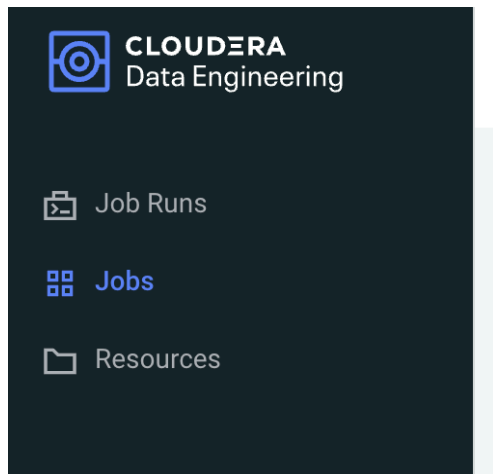
Click on Administration:



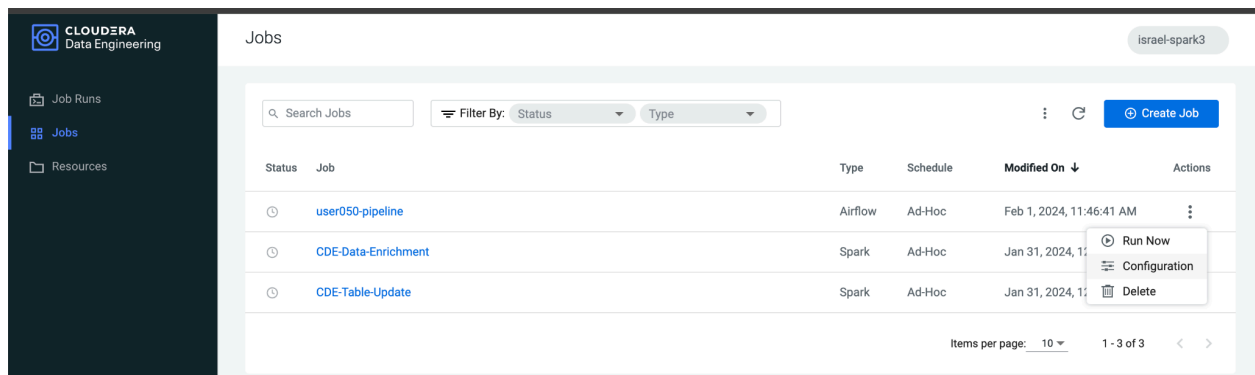
Click on the briefcase on the right side to view the spark cluster jobs:



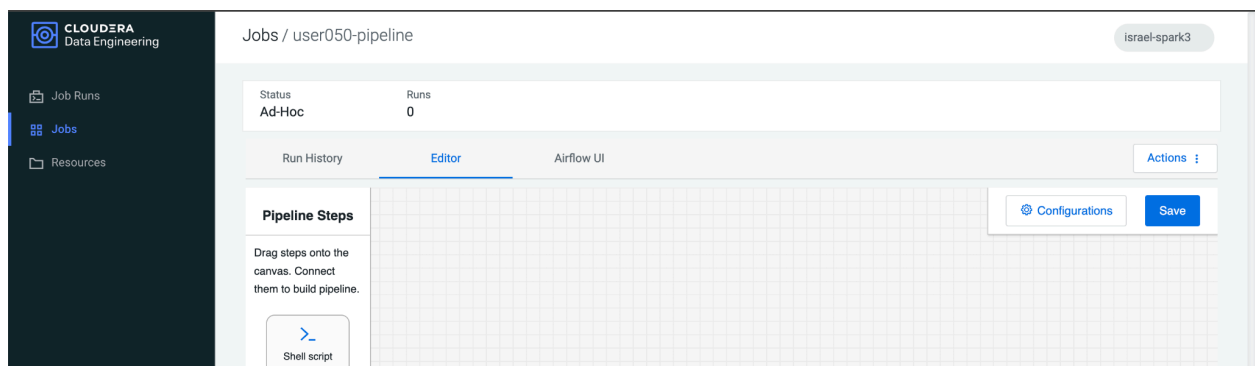
Click on Jobs:



Open Configuration for the previously created <userid>-pipeline job



Click on the Editor tab



The Editor canvas should load properly now, continue with step 6 above