



TASK 1 - AIRFLOW

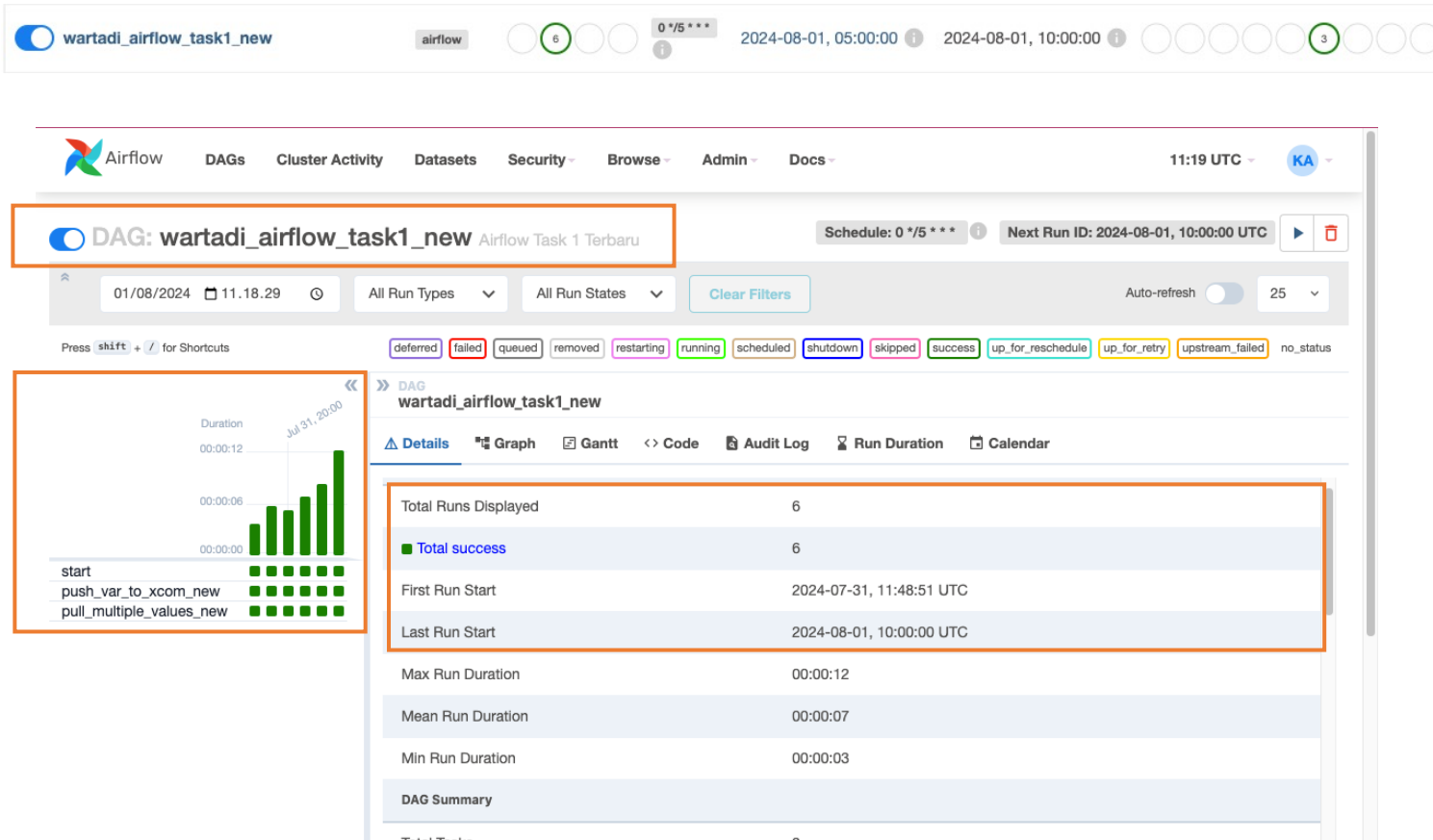
No	Deskripsi	Perintah/Output
1.1	<p>Dalam kasus ini dimulai dengan mencoba menghubungkan computer saya (Wartadis-MacBook-Pro) ke komputer lain dengan alamat IP 34.101.224.54 dan akan login ke komputer tujuan sebagai pengguna raja_rahmanakmaludin. Setelah berhasil terhubung, Anda akan dapat menjalankan perintah pada komputer jarak jauh tersebut seolah-olah Anda sedang menggunakannya secara langsung.</p>	<pre>wartadi@Wartadis-MacBook-Pro alta % ssh raja_rahmanakmaludin@34.101.224.54 raja_rahmanakmaludin@34.101.224.54's password: Linux instance-20240714-035051 6.1.0-23-cloud-amd64 #1 SMP PREEMPT_DYNAMIC Deb ian 6.1.99-1 (2024-07-15) x86_64 The programs included with the Debian GNU/Linux system are free software; the exact distribution terms for each program are described in the individual files in /usr/share/doc/*/copyright. Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law. Last login: Wed Jul 31 11:11:00 2024 from 180.242.68.7 raja_rahmanakmaludin@instance-20240714-035051:~\$</pre>

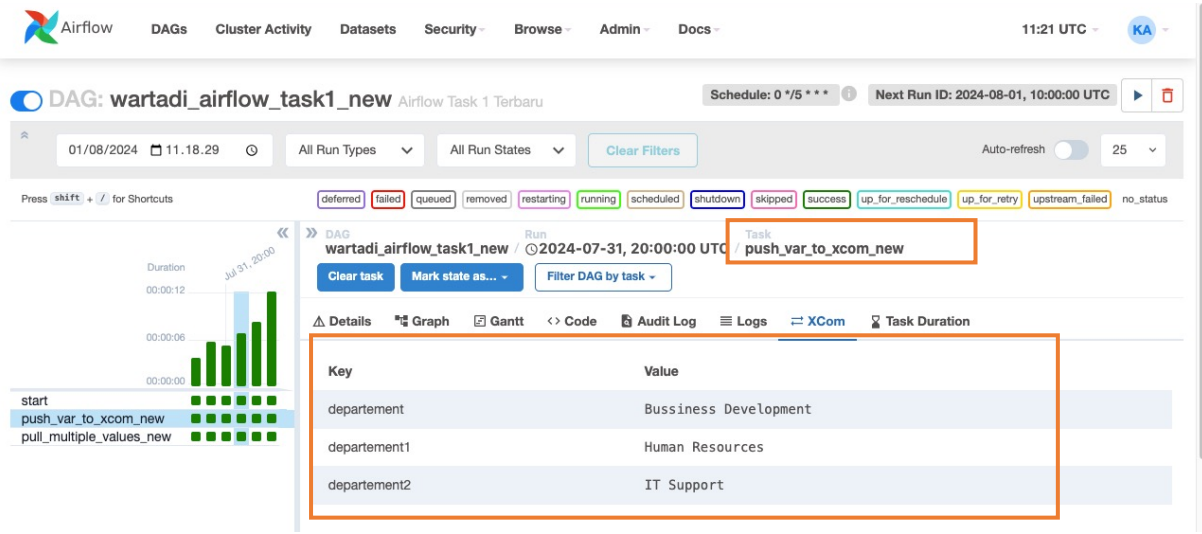
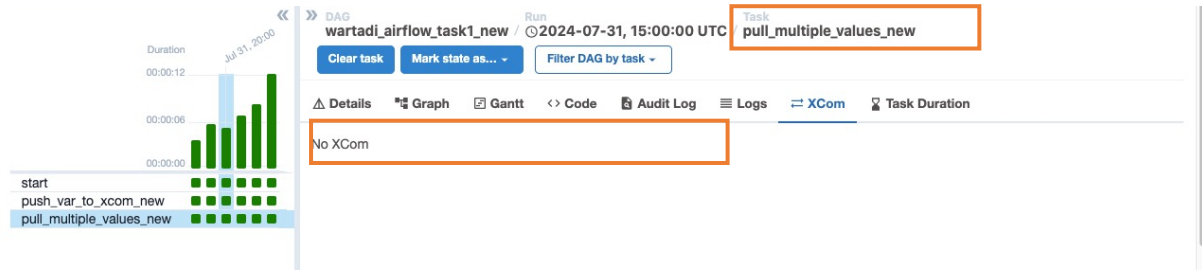
No	Deskripsi	Perintah/Output
1.2	<p>sudo nano wartadi_task1_airflow_new.py</p> <p>Merupakan sebuah perintah di Linux yang digunakan untuk membuat dan mengedit file bernama "wartadi_task1_airflow_new.py" dengan editor teks "nano" yang mana dalam hal ini akan tersimpan pada computer lain yang sudah berhasil kita hubungkan atau kita akses.</p> <p>Dalam hal ini file wartadi_task1_airflow_new.py tersimpan di raja_rahmanakmaludin@instance-20240714-035051:~/airflow-data/docker/dags\$</p>	<pre>raja_rahmanakmaludin@instance-20240714-035051:~/airflow-data/docker/dags\$ ls __pycache__ alterra_connection_kharisma.py alterra_connection_yovina.py alterra_hasda_task-1.py alterra_hasda_task-2.py alterra_hasda_task2.py alterra_hook_kharisma.py alterra_hook_yovina.py alterra_nurhasanahdarus_task1.py alterra_tes_connection_khairullah.py connection_dimas_task2.py connection_example.py connection_example_dimas.py connection_example_fitri.py dag_github_data.py dimas-alltask1-airflow.py dimas-alltask2-airflow.py dimas-task1-airflow.py etl_github_data.py farhan_task2.py farhan_task_1.py farhan_test.py fitri_airflowtask1.py fitri_airflowtask2.py get_var_example.py hello_world.py hello_world_operator.py hook_example.py hook_example_dimas.py hook_example_fitri.py ingestion.py integrate_all.py integrate_all_part2.py integrate_all_part3.py kharisma_airflow_task1.py kharisma_airflow_task2.py loop_print_var_example.py rais-alltask1-airflow.py rais-alltask2-airflow.py sample.csv wartadi_task1_airflow.py wartadi_task1_airflow_new.py wartadi_task2_airflow.py xcom_example_decorator.py xcom_example_native.py yovina_airflow_task1.py yovina_airflow_task2.py zola_task1_airflow.py zola_task2_airflow.py</pre>

Command wartadi_task1_airflow_new.py



No	Deskripsi	Input Program
2	<p>Dalam hal ini terdiri dari 3 task antara lain :</p> <ol style="list-style-type: none">1. start2. push_var_to_xcom_new3. pull_multiple_values_new <p>Kode ini membuat DAG yang berjalan setiap 5 jam, menyimpan beberapa nilai ke XCom, dan kemudian mengambil nilai-nilai tersebut dari XCom.</p> <p>Code ini yang akan menjadi acuan task program yang akan berjalan pada Airflow http://34.101.224.54:8080/.</p>	<pre>GNU nano 7.2 wartadi_task1_airflo from datetime import datetime from airflow import DAG from airflow.operators.python_operator import PythonOperator from airflow.operators.empty import EmptyOperator # Number 1 Create DAG that run in every 5 hours. dag=DAG('wartadi_airflow_task1_new', description='Airflow Task 1 Terbaru', schedule_interval='0 */5 * * *', start_date=datetime(2023, 10, 21), catchup=False) start = EmptyOperator(task_id='start', dag=dag,) # ti = task instance # Number 2 Suppose we define a new task that push a variable to xcom. def push_var_to_xcom_new(ti=None): ti.xcom_push(key='departement', value='Bussiness Development') ti.xcom_push(key='departement1', value='Human Resources') ti.xcom_push(key='departement2', value='IT Support') #Number 3 How to pull multiple values at once? def pull_multiple_values_new(ti=None): departement = ti.xcom_pull(task_ids='push_var_department', key='departement') departement1 = ti.xcom_pull(task_ids='push_var_department', key='departement1') departement2 = ti.xcom_pull(task_ids='push_var_department', key='departement2') print(f'print departement variable from xcom: {departement}, {departement1}, {departement2}') push_var_to_xcom_new = PythonOperator(task_id = 'push_var_to_xcom_new', python_callable = push_var_to_xcom_new) pull_multiple_values_new = PythonOperator(task_id = 'pull_multiple_values_new', python_callable = pull_multiple_values_new) start >> push_var_to_xcom_new >> pull_multiple_values_new</pre>

No	Deskripsi	Airflow														
2.1	<p>Dari perintah sebelumnya kita menuju ke http://34.101.224.54:8080/.</p> <p>Perintah yang sudah disimpan pada file wartadi_task1_airflow_new.py dimana DAG diberikan nama "wartadi_airflow_task1_new" sukses dan berjalan tanpa ada error dan konteks ini program sudah berjalan 6 kali sesuai schedule yang sudah ditentukan</p> <p>Untuk tugas dalam DAG ini meliputi</p> <ol style="list-style-type: none">1. start2. push_var_to_xcom_new3. put_multiple_values_new	 <p>The screenshot displays the Airflow web interface for the DAG <code>wartadi_airflow_task1_new</code>. At the top, a status bar shows the DAG is running with 6 successful runs out of 5 scheduled. The main interface includes a navigation bar with options like DAGs, Cluster Activity, Datasets, Security, Browse, Admin, and Docs. The DAG details section shows the schedule <code>0 * / 5 * * *</code> and the next run ID <code>2024-08-01, 10:00:00 UTC</code>. A bar chart on the left visualizes the run durations, with the highest bar reaching approximately 00:00:12. On the right, a table summarizes the DAG's performance:</p> <table><tr><td>Total Runs Displayed</td><td>6</td></tr><tr><td>Total success</td><td>6</td></tr><tr><td>First Run Start</td><td>2024-07-31, 11:48:51 UTC</td></tr><tr><td>Last Run Start</td><td>2024-08-01, 10:00:00 UTC</td></tr><tr><td>Max Run Duration</td><td>00:00:12</td></tr><tr><td>Mean Run Duration</td><td>00:00:07</td></tr><tr><td>Min Run Duration</td><td>00:00:03</td></tr></table> <p>The DAG Summary section is also visible at the bottom.</p>	Total Runs Displayed	6	Total success	6	First Run Start	2024-07-31, 11:48:51 UTC	Last Run Start	2024-08-01, 10:00:00 UTC	Max Run Duration	00:00:12	Mean Run Duration	00:00:07	Min Run Duration	00:00:03
Total Runs Displayed	6															
Total success	6															
First Run Start	2024-07-31, 11:48:51 UTC															
Last Run Start	2024-08-01, 10:00:00 UTC															
Max Run Duration	00:00:12															
Mean Run Duration	00:00:07															
Min Run Duration	00:00:03															

No	Deskripsi	Airflow								
2.2	<p>Terlihat bahwa pada saat Task push_var_to_xcom_new berjalan terdapat 3 key dan 3 values</p> <p>Kemudian terlihat sudah tidak ada key dan values ketika task pull_multiple_values_new berjalan</p>	<div><p>The screenshot shows the Airflow DAG 'wartadi_airflow_task1_new' at 20:00 UTC. The task 'push_var_to_xcom_new' is highlighted with an orange box. Below the task name, a table displays XCom values:</p><table><tr><th>Key</th><th>Value</th></tr><tr><td>departement</td><td>Bussiness Development</td></tr><tr><td>departement1</td><td>Human Resources</td></tr><tr><td>departement2</td><td>IT Support</td></tr></table></div> <div><p>The screenshot shows the same DAG at 15:00 UTC. The task 'pull_multiple_values_new' is highlighted with an orange box. Below the task name, the XCom section shows 'No XCom'.</p></div>	Key	Value	departement	Bussiness Development	departement1	Human Resources	departement2	IT Support
Key	Value									
departement	Bussiness Development									
departement1	Human Resources									
departement2	IT Support									

THANK YOU 😊

