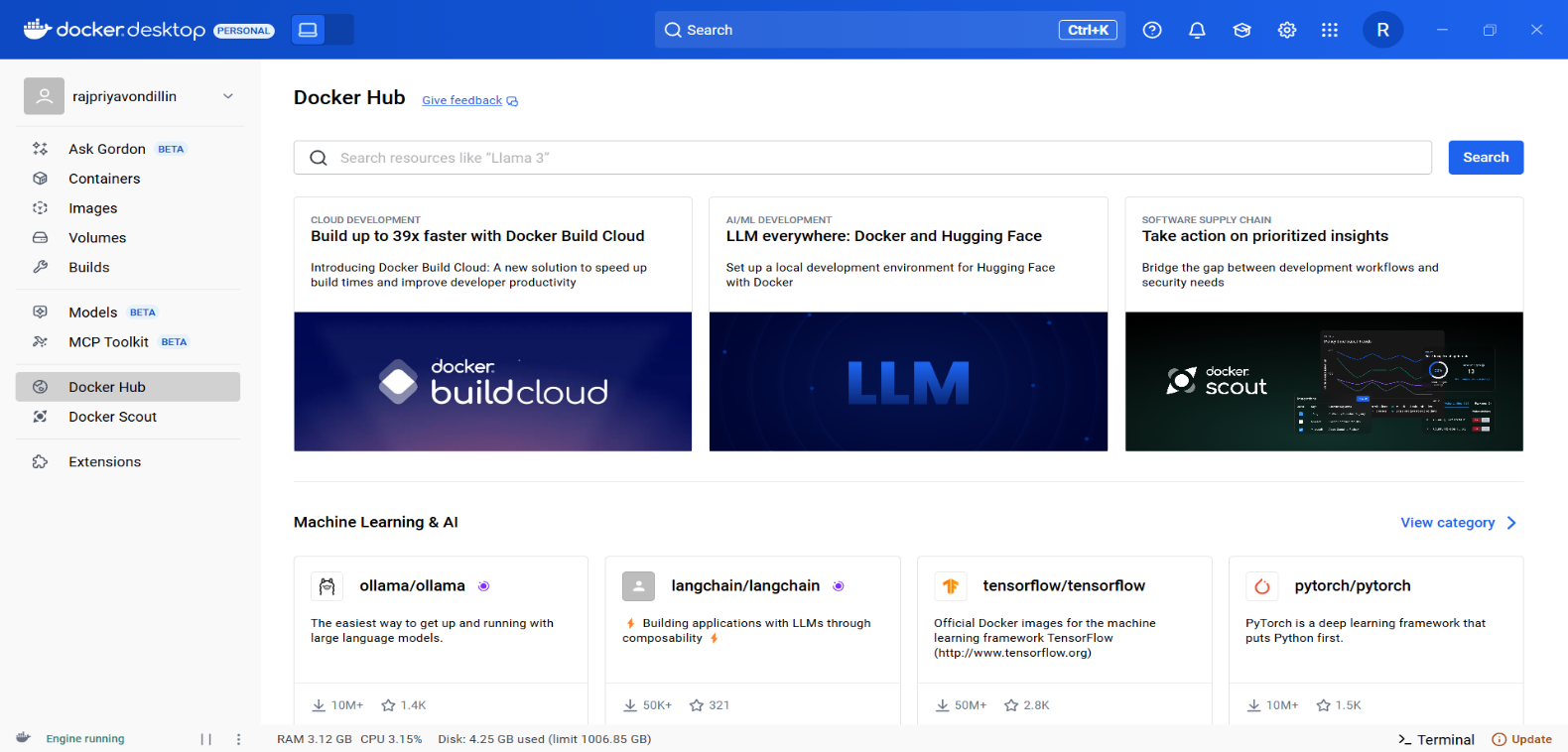


Install Docker - <https://youtu.be/JBEUKrjbWqg?si=Xv4k-04gUxFFF84H> (Reference link)



**CREATING DOCKER HUB ACCOUNT**

* Create a docker hub account using your email Id (It is just like signing up)
* Then restart your PC (optional – but for safer side)
* Once done, you have now created the docker hub account

**Creating the Workspace Manually with the help of yaml file,**

* I have attached the “docker-compose.yaml” file
* Download it in your local folder
* Now at any path in your file manager, create a folder called “airflow-docker”
* **Note down the path**
* Inside the “airflow-docker” folder, put your “docker-compose.yaml” file
* In the same “airflow-docker” folder, create 3 subfolders say,
  + dags
  + logs
  + plugins
* Now open your cmd (Command prompt) and type the following,
* **Open CMD (not PyCharm terminal for this step, unless you’ve mounted Docker inside PyCharm).**  
  Navigate into your folder: (NOTE: Paste your noted “airflow-docker” folder path)

cd C:\Users\rajpr\OneDrive\Documents\HEXAWARE\airflow-docker

* **Run docker-compose** to start Airflow in the background:

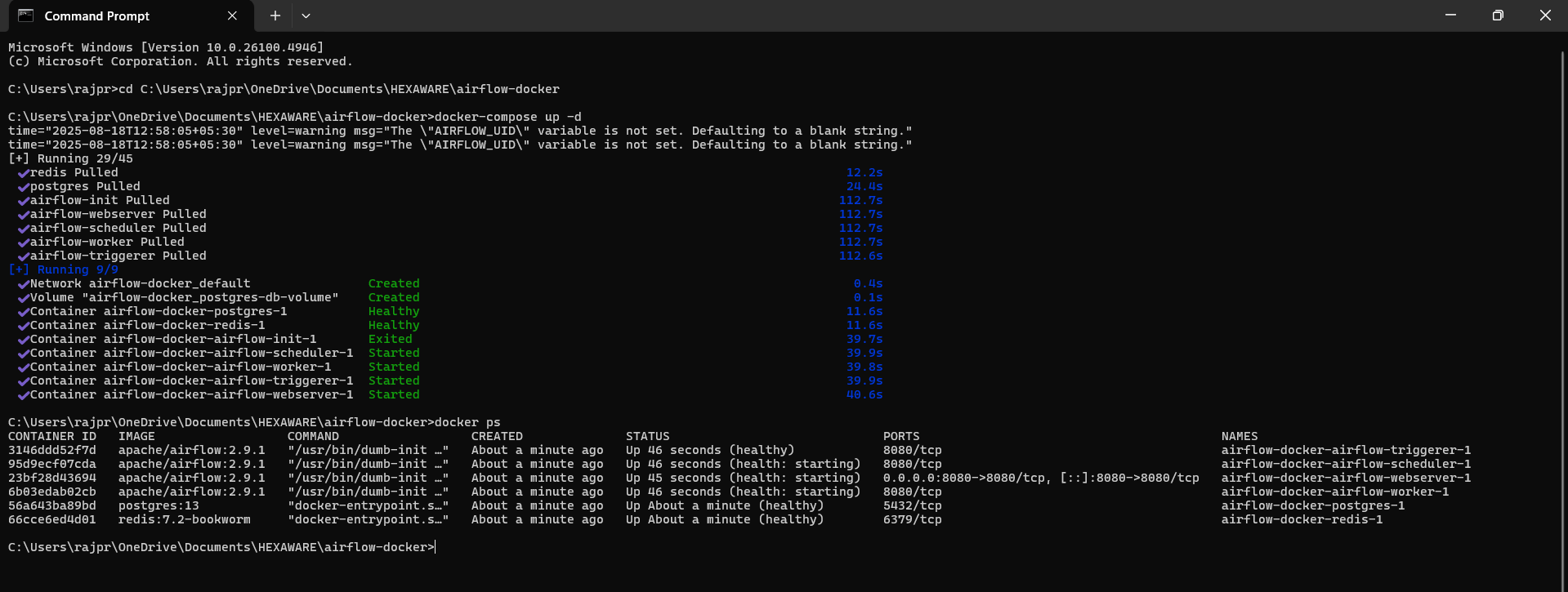
docker-compose up -d

Wait a few minutes (first time it will pull several Docker images, so it may take a while).

* Once running, check containers:

docker ps

* You should see services like airflow-webserver, airflow-scheduler, airflow-worker, etc.
* You must see your cmd something like this,



* Open your browser and go to: <https://locall.host/8080/#google_vignette>

(Airflow UI will be available there).

* Now click on to the Go to localhost: 8080, a green colour button
* Refresh and wait for sometimes it will surely open
* Once done, type the username and password,
* By default, in the **official Airflow Docker Compose setup**, the login is:

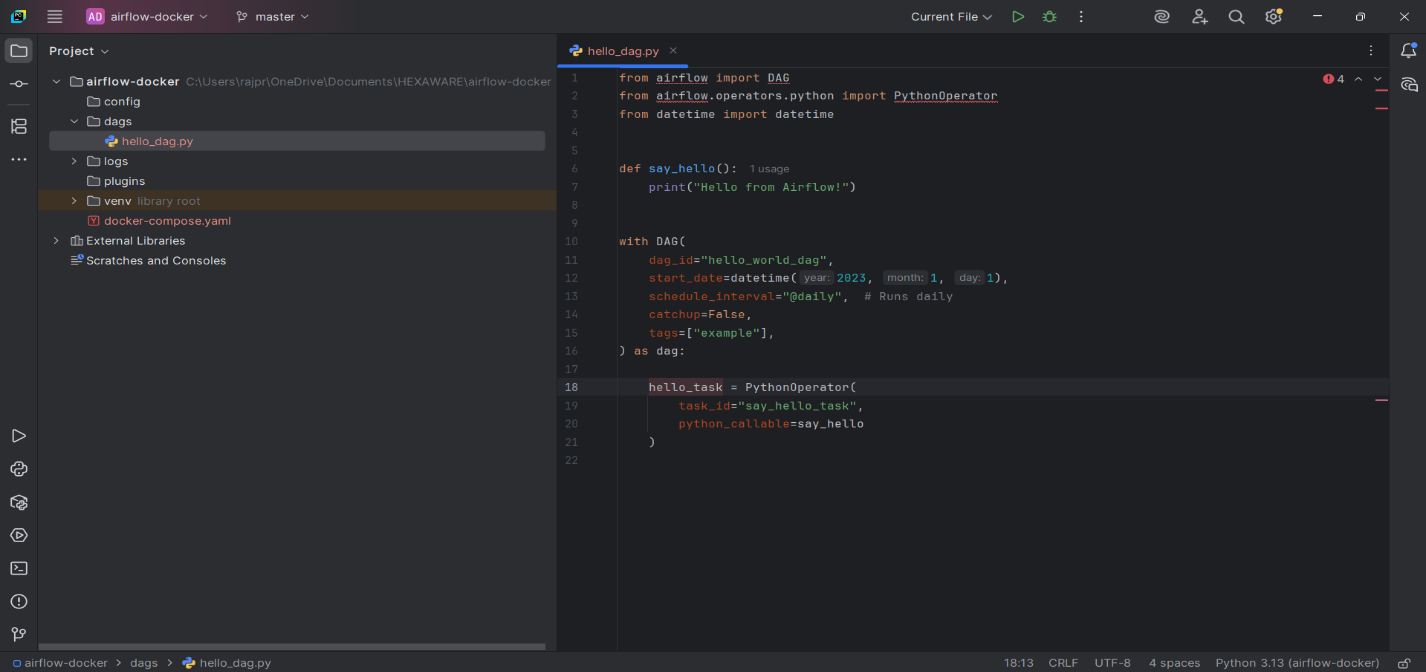
**Username:** airflow

**Password:** airflow

First try with airflow / airflow.

* By doing the above steps, 3 steps have been completed
  + Step 1: Install Airflow Locally (Using Virtualenv)
    - Create and activate a virtual environment
    - Install Airflow (with local executor + SQLite for dev)
  + Step 2: Initialize Airflow
  + Step 3: Start Airflow UI and Scheduler

**TO PERFORM STEP 4, WHICH IS TO CREATE A SIMPLE DAG**

* Open PyCharm and Navigate to File -> Open “airflow-docker” folder
* In the left side of contents in PyCharm, you will find all the subfolders inside this “airflow-docker” folder
* Set up your python interpreter
* Inside the dags subfolder in PyCharm Right Click -> new -> python file -> name it as “hello\_dag.py”
* And enter this sample code,
* from airflow import DAG  
  from airflow.operators.python import PythonOperator  
  from datetime import datetime  
    
    
  def say\_hello():  
   print("Hello from Airflow!")  
    
    
  with DAG(  
   dag\_id="hello\_world\_dag",  
   start\_date=datetime(2023, 1, 1),  
   schedule\_interval="@daily", # Runs daily  
   catchup=False,  
   tags=["example"],  
  ) as dag:  
    
   hello\_task = PythonOperator(  
   task\_id="say\_hello\_task",  
   python\_callable=say\_hello  
   )
* Finally, you will see something like this

✅ Important: When using Docker, you don’t run DAGs locally. Airflow inside the container runs them automatically.

* Go to your browser → <https://locall.host/8080/#google_vignette>
* Refresh the UI → you will see “hello\_world\_dag”
* Click **Trigger DAG** ▶️ Airflow will execute the DAG inside the container.

**✅ Key takeaway:**

* **Do not run DAG Python files directly in PyCharm** if you’re using Docker.
* PyCharm is only for **editing** the DAG code.
* Dockerized Airflow is the one that **executes** the DAGs.

**Here’s what happens when you Trigger DAG and what the output looks like:**

**1. DAG Status in Airflow UI**

* In the **DAGs** list, the hello\_world\_dag row will change:
  + Status will show **running** (green arrow spinning).
  + Once finished, it will turn **green** → indicates successful execution.

**2. Task Instance Status**

* Click on the DAG name → you’ll see the **graph view / tree view**.
* Your single task say\_hello\_task will appear as a **node**.
* Node colours indicate status:
  + **Green** → success
  + **Yellow** → running
  + **Red** → failed

**3. Logs / Output**

* Click on the task → **View Log**.
* You will see something like:

[2025-08-18 12:35:00,123] {python.py:123} INFO - Executing <PythonOperator: say\_hello\_task> on ...

[2025-08-18 12:35:00,125] {python.py:126} INFO - Hello from Airflow!

[2025-08-18 12:35:00,130] {taskinstance.py:1450} INFO - Marking task as SUCCESS

* The line Hello from Airflow! is exactly from your print() in the DAG. ✅

**4. Summary**

* DAG triggered → task runs inside Docker container → output goes to **Airflow logs**.
* You **don’t see the output in PyCharm** because the Python process runs inside Docker.

