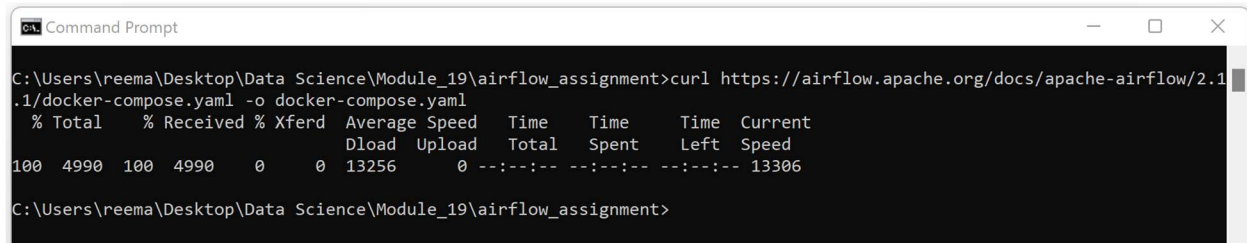


Airflow

1. Pulled the Airflow file using curl command.

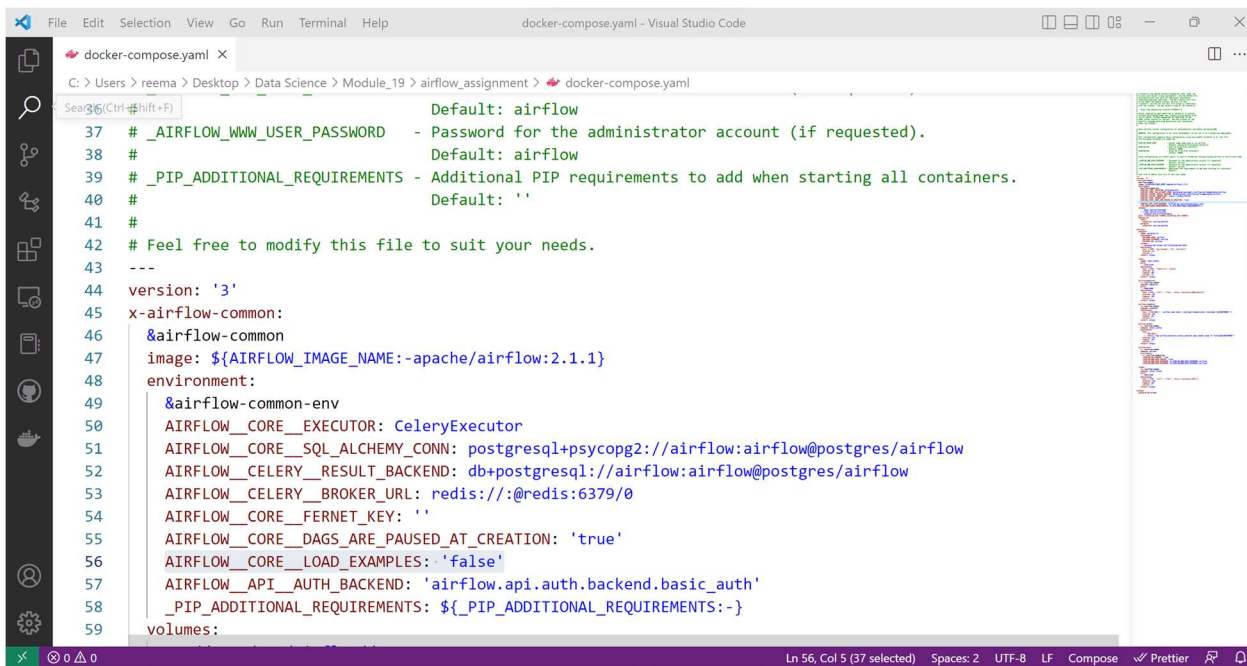


```
Command Prompt

C:\Users\reema\Desktop\Data Science\Module_19\airflow_assignment>curl https://airflow.apache.org/docs/apache-airflow/2.1
.1/docker-compose.yaml -o docker-compose.yaml
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100 4990  100 4990    0     0  13256      0  --:--:-- --:--:-- --:--:-- 13306

C:\Users\reema\Desktop\Data Science\Module_19\airflow_assignment>
```

2. Changed example value (AIRFLOW__CORE__LOAD_EXAMPLES set to false).

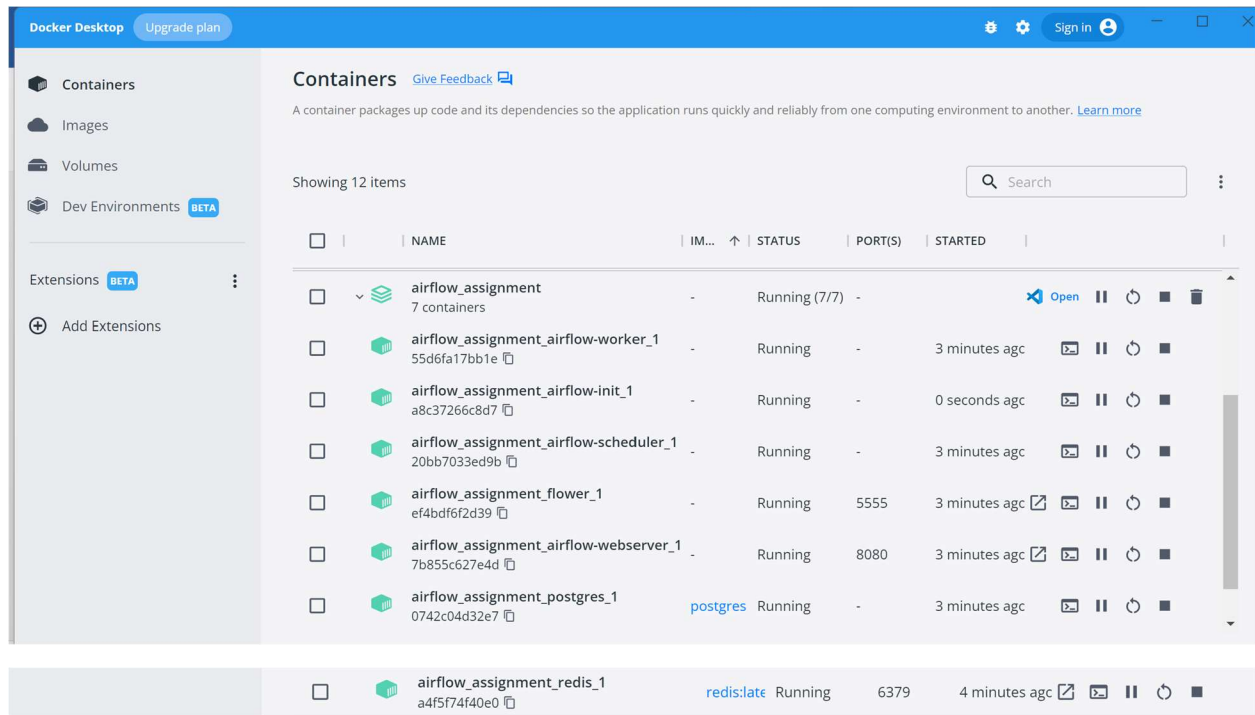


```
File Edit Selection View Go Run Terminal Help docker-compose.yaml - Visual Studio Code

C:\Users\reema\Desktop\Data Science\Module_19\airflow_assignment> docker-compose.yaml

37 # _AIRFLOW_WWW_USER_PASSWORD - Password for the administrator account (if requested).
38 #                               Default: airflow
39 # _PIP_ADDITIONAL_REQUIREMENTS - Additional PIP requirements to add when starting all containers.
40 #                               Default: ''
41 #
42 # Feel free to modify this file to suit your needs.
43 ---
44 version: '3'
45 x-airflow-common:
46   &airflow-common
47   image: ${AIRFLOW_IMAGE_NAME:-apache/airflow:2.1.1}
48   environment:
49     &airflow-common-env
50     AIRFLOW__CORE__EXECUTOR: CeleryExecutor
51     AIRFLOW__CORE__SQL_ALCHEMY_CONN: postgresql+psycopg2://airflow:airflow@postgres/airflow
52     AIRFLOW__CELERY__RESULT_BACKEND: db+postgresql://airflow:airflow@postgres/airflow
53     AIRFLOW__CELERY__BROKER_URL: redis://:@redis:6379/0
54     AIRFLOW__CORE__FERNET_KEY: ''
55     AIRFLOW__CORE__DAGS_ARE_PAUSED_AT_CREATION: 'true'
56     AIRFLOW__CORE__LOAD_EXAMPLES: 'false'
57     AIRFLOW__API__AUTH_BACKEND: 'airflow.api.auth.backend.basic_auth'
58     _PIP_ADDITIONAL_REQUIREMENTS: ${_PIP_ADDITIONAL_REQUIREMENTS:-}
59   volumes:
```

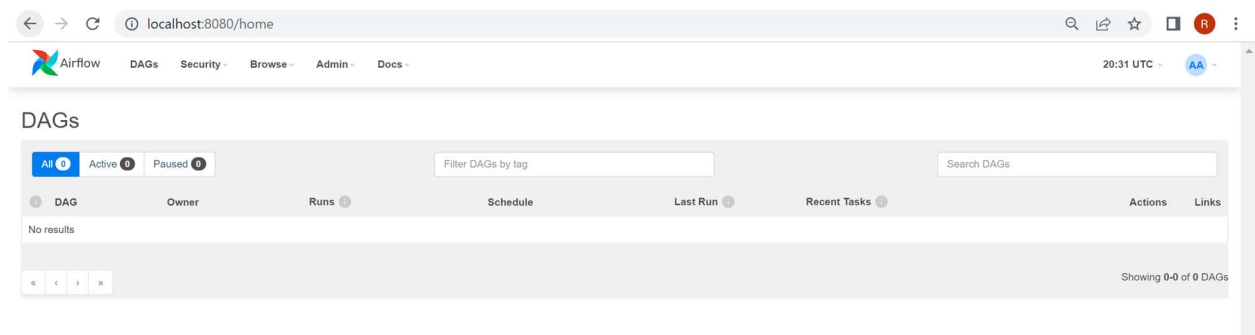
3. Airflow Docker containers.



The screenshot shows the Docker Desktop application. On the left sidebar, the 'Containers' tab is selected. The main area displays a list of 12 containers. The first container, 'airflow_assignment', is expanded to show its sub-containers. The sub-containers include 'airflow_assignment_airflow-worker_1', 'airflow_assignment_airflow-init_1', 'airflow_assignment_airflow-scheduler_1', 'airflow_assignment_flower_1', 'airflow_assignment_airflow-webserver_1', and 'airflow_assignment_postgres_1'. All these sub-containers are in a 'Running' state. Below the main list, another container 'airflow_assignment_redis_1' is visible, also in a 'Running' state.

NAME	IM...	STATUS	PORT(S)	STARTED
airflow_assignment (7 containers)	-	Running (7/7)	-	-
airflow_assignment_airflow-worker_1	-	Running	-	3 minutes ago
airflow_assignment_airflow-init_1	-	Running	-	0 seconds ago
airflow_assignment_airflow-scheduler_1	-	Running	-	3 minutes ago
airflow_assignment_flower_1	-	Running	5555	3 minutes ago
airflow_assignment_airflow-webserver_1	-	Running	8080	3 minutes ago
airflow_assignment_postgres_1	postgres	Running	-	3 minutes ago
airflow_assignment_redis_1	redis:late	Running	6379	4 minutes ago

4. Logged in to Airflow.

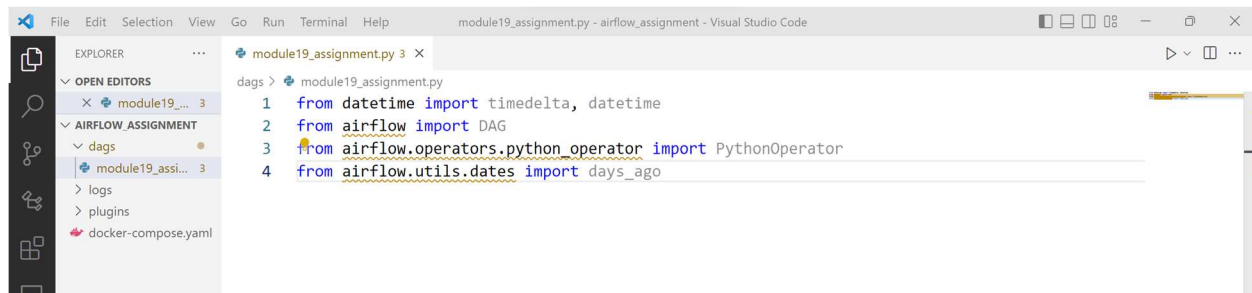


The screenshot shows the Airflow web interface at the URL 'localhost:8080/home'. The 'DAGs' tab is selected in the top navigation bar. The main content area displays a table for DAGs. The table has columns for 'DAG', 'Owner', 'Runs', 'Schedule', 'Last Run', 'Recent Tasks', 'Actions', and 'Links'. The table is currently empty, showing 'No results'. The status filters at the top indicate 'All' (1), 'Active' (0), and 'Paused' (0) DAGs.

DAG	Owner	Runs	Schedule	Last Run	Recent Tasks	Actions	Links
No results							

5. Setting up 'module19_assignment.py file'

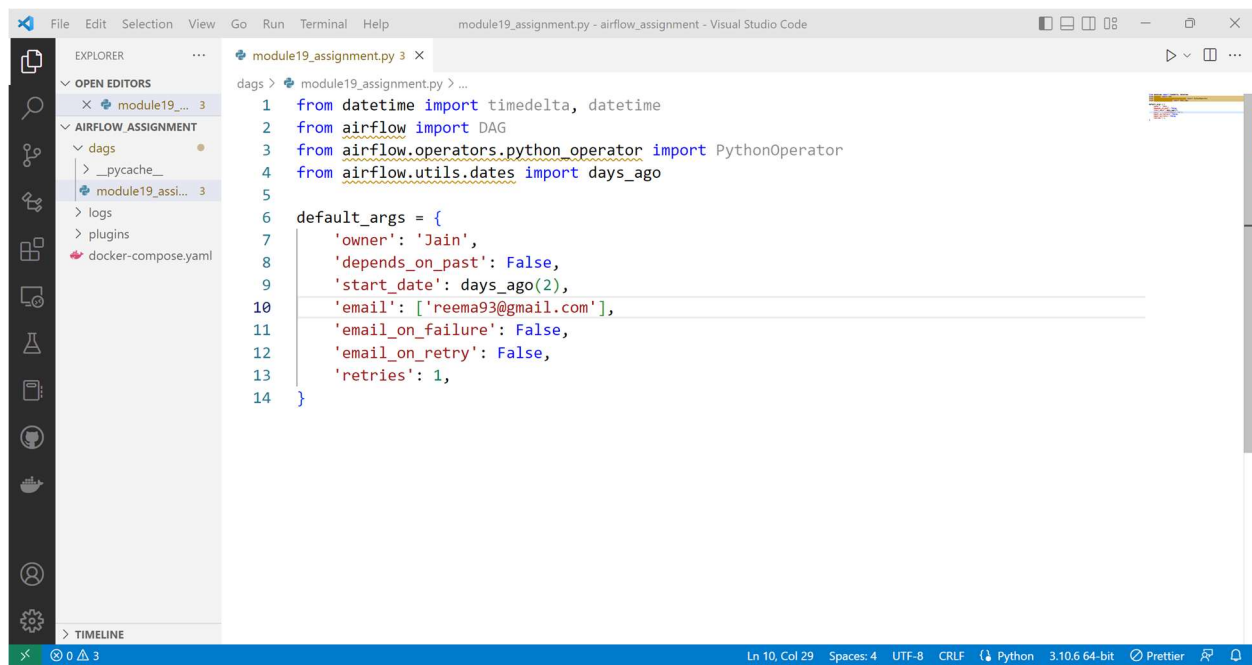
a) Imported the required *libraries*



The screenshot shows the Visual Studio Code editor with the file 'module19_assignment.py' open. The Explorer sidebar on the left shows the project structure with 'AIRFLOW_ASSIGNMENT' containing 'dags' and 'module19_assignment.py'. The code editor displays the following imports:

```
1 from datetime import timedelta, datetime
2 from airflow import DAG
3 from airflow.operators.python_operator import PythonOperator
4 from airflow.utils.dates import days_ago
```

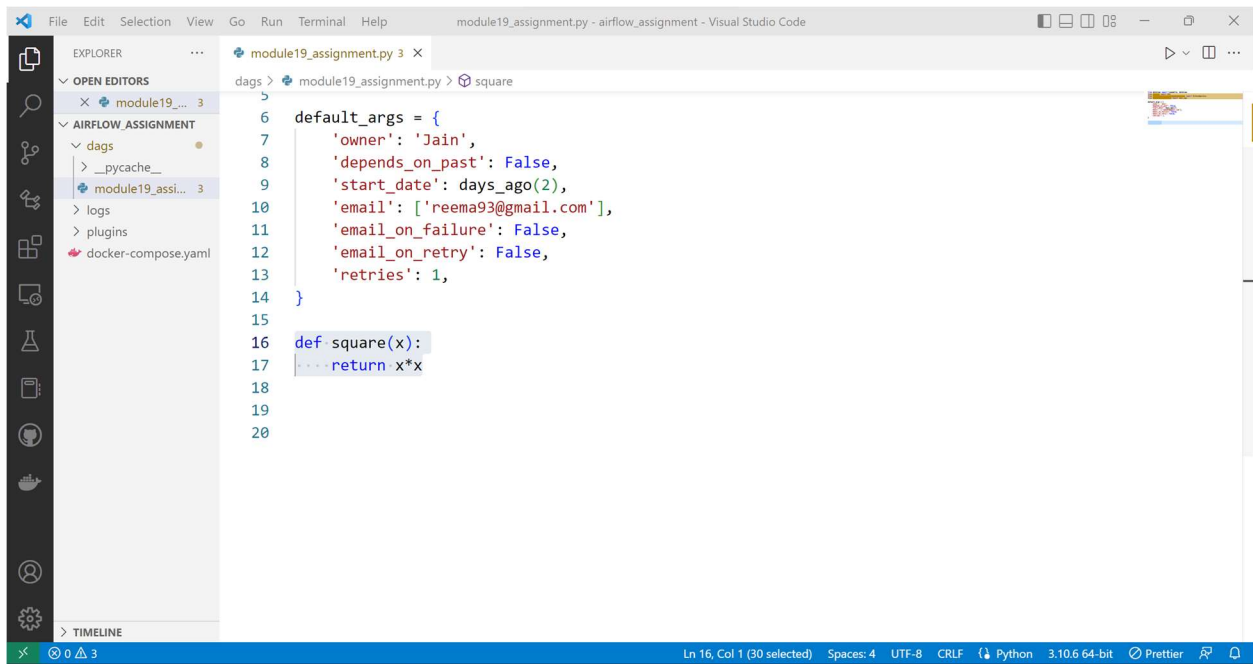
b) Set up DAG correctly, including your last name and email.



The screenshot shows the Visual Studio Code editor with the file 'module19_assignment.py' open. The code editor displays the DAG setup:

```
1 from datetime import timedelta, datetime
2 from airflow import DAG
3 from airflow.operators.python_operator import PythonOperator
4 from airflow.utils.dates import days_ago
5
6 default_args = {
7     'owner': 'Jain',
8     'depends_on_past': False,
9     'start_date': days_ago(2),
10    'email': ['reema93@gmail.com'],
11    'email_on_failure': False,
12    'email_on_retry': False,
13    'retries': 1,
14 }
```

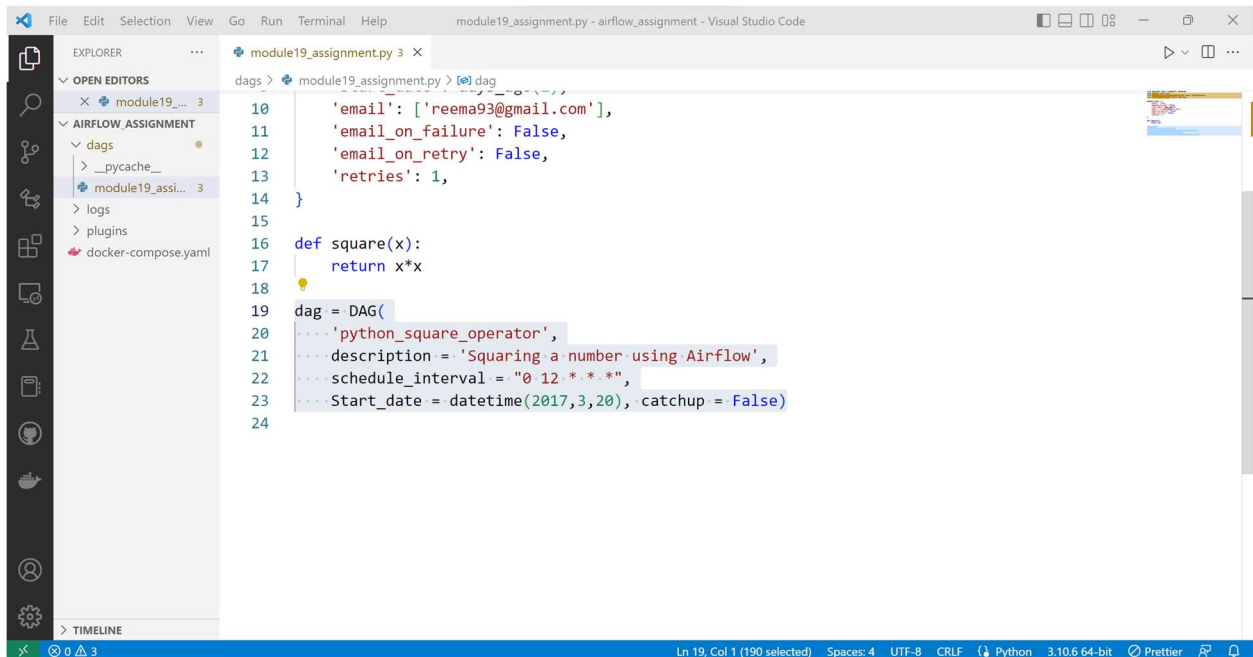
c) Defined the square() function correctly.



The screenshot shows the Visual Studio Code editor with the file `module19_assignment.py` open. The left sidebar shows the Explorer view with the file structure. The main editor area shows the following code:

```
6 default_args = {
7     'owner': 'Jain',
8     'depends_on_past': False,
9     'start_date': days_ago(2),
10    'email': ['reema93@gmail.com'],
11    'email_on_failure': False,
12    'email_on_retry': False,
13    'retries': 1,
14 }
15
16 def square(x):
17     return x*x
18
19
20
```

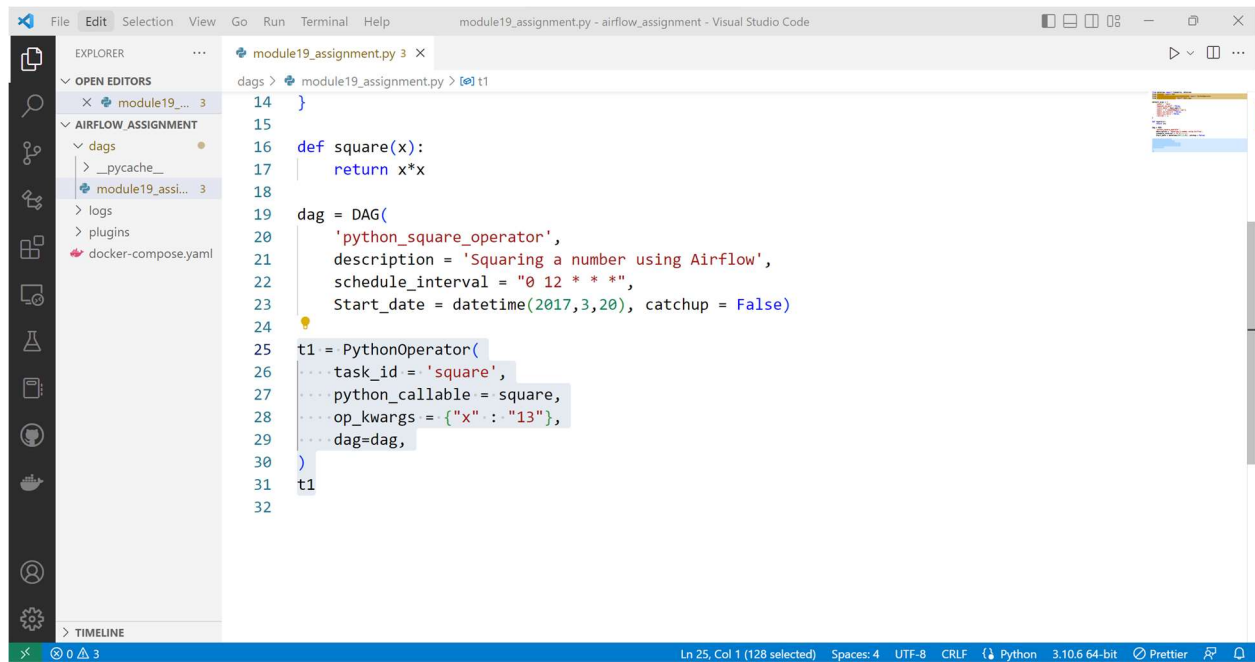
d) Defined the DAG object.



The screenshot shows the Visual Studio Code editor with the file `module19_assignment.py` open. The left sidebar shows the Explorer view with the file structure. The main editor area shows the following code:

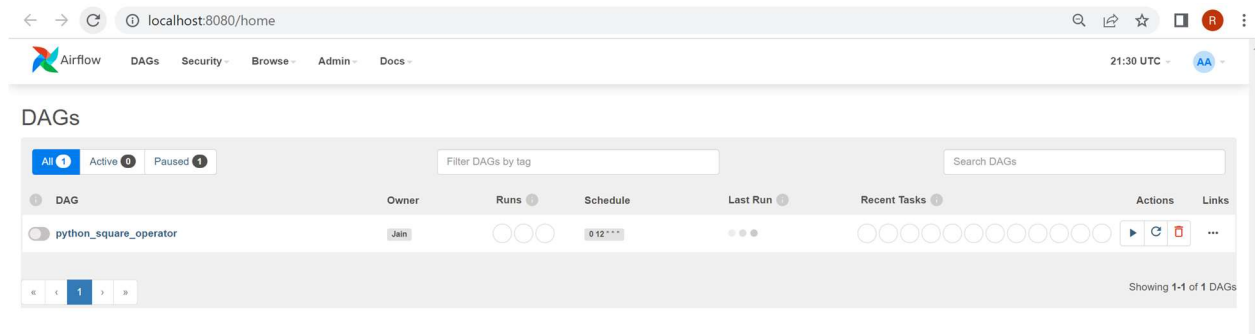
```
10    'email': ['reema93@gmail.com'],
11    'email_on_failure': False,
12    'email_on_retry': False,
13    'retries': 1,
14 }
15
16 def square(x):
17     return x*x
18
19 dag = DAG(
20     'python_square_operator',
21     description = 'Squaring a number using Airflow',
22     schedule_interval = "0 12 * * *",
23     start_date = datetime(2017,3,20), catchup = False)
24
```

e) Defined the DAG Task correctly.



```
14 }
15
16 def square(x):
17     return x*x
18
19 dag = DAG(
20     'python_square_operator',
21     description = 'Squaring a number using Airflow',
22     schedule_interval = "0 12 * * *",
23     Start_date = datetime(2017,3,20), catchup = False)
24
25 t1 = PythonOperator(
26     task_id = 'square',
27     python_callable = square,
28     op_kwargs = {"x": "13"},
29     dag=dag,
30 )
31 t1
32
```

6. Screenshot of the Airflow UI showing that DAG is configured correctly.



7. Screenshot of the log showing that the DAG ran successfully.

The screenshot displays the Apache Airflow web interface in a browser. The address bar shows the URL: `localhost:8080/log?dag_id=python_square_operator&task_id=square&execution_date=2022-12-08T21%3A35%3A43.847690%2B00%3A00`. The interface includes a top navigation bar with links for DAGs, Security, Browse, Admin, and Docs. The main content area is titled "Log by attempts" and shows a single attempt (1) of the task. The log text is as follows:

```
*** Reading local file: /opt/airflow/logs/python_square_operator/square/2022-12-08T21:35:43.847690+00:00/1.log
[2022-12-08 21:35:44,845] [taskinstance.py:896] INFO - Dependencies all met for <TaskInstance: python_square_operator.square 2022-12-08T21:35:43.847690+00:00 [queued]>
[2022-12-08 21:35:44,865] [taskinstance.py:896] INFO - Dependencies all met for <TaskInstance: python_square_operator.square 2022-12-08T21:35:43.847690+00:00 [queued]>
[2022-12-08 21:35:44,865] [taskinstance.py:1087] INFO -
-----
[2022-12-08 21:35:44,865] [taskinstance.py:1088] INFO - Starting attempt 1 of 2
[2022-12-08 21:35:44,865] [taskinstance.py:1089] INFO -
-----
[2022-12-08 21:35:44,874] [taskinstance.py:1107] INFO - Executing <Task(PythonOperator): square> on 2022-12-08T21:35:43.847690+00:00
[2022-12-08 21:35:44,878] [standard_task_runner.py:52] INFO - Started process 5574 to run task
[2022-12-08 21:35:44,881] [standard_task_runner.py:76] INFO - Running: ['***', 'tasks', 'run', 'python_square_operator', 'square', '2022-12-08T21:35:43.847690+00:00', '--job-id', '7', '--pool', 'default_pool', '--raw', '--subdir',
[2022-12-08 21:35:44,882] [standard_task_runner.py:77] INFO - Job 7: Subtask square
[2022-12-08 21:35:44,920] [logging_mixin.py:104] INFO - Running <TaskInstance: python_square_operator.square 2022-12-08T21:35:43.847690+00:00 [running]> on host 55d6fa17bb1e
[2022-12-08 21:35:44,975] [taskinstance.py:1302] INFO - Exporting the following env vars:
AIRFLOW_CTX_DAG_EMAIL=reema93@gmail.com
AIRFLOW_CTX_DAG_OWNER=Jain
AIRFLOW_CTX_DAG_ID=python_square_operator
AIRFLOW_CTX_TASK_ID=square
AIRFLOW_CTX_EXECUTION_DATE=2022-12-08T21:35:43.847690+00:00
AIRFLOW_CTX_DAG_RUN_ID>manual_2022-12-08T21:35:43.847690+00:00
[2022-12-08 21:35:44,977] [python.py:151] INFO - Done. Returned value was: 169
[2022-12-08 21:35:45,004] [taskinstance.py:1211] INFO - Marking task as SUCCESS. dag_id=python_square_operator, task_id=square, execution_date=20221208T213543, start_date=20221208T213544, end_date=20221208T213545
[2022-12-08 21:35:45,032] [taskinstance.py:1265] INFO - 0 downstream tasks scheduled from follow-on schedule check
[2022-12-08 21:35:45,055] [local_task_job.py:151] INFO - Task exited with return code 0
```

At the bottom of the interface, the version information is displayed: Version: v2.1.1 and Git Version: .dev0+a71f9d6f25e3255ac33755da2590c308062dbd1.dirty.