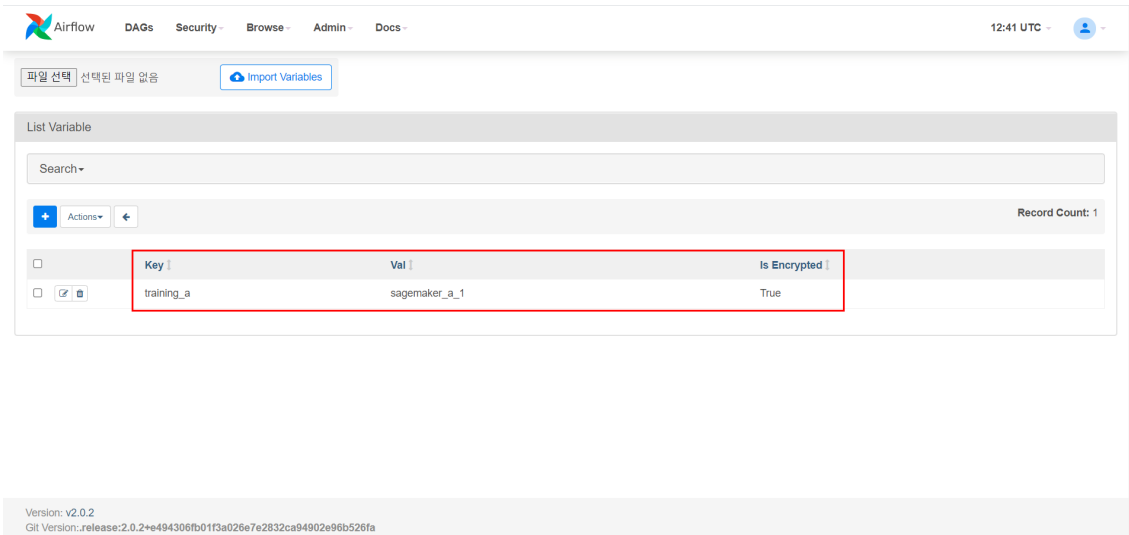


# MWAA 기능 test

MWAA와의 같은 환경의 Lambda 또는 EC2에서 MWAA CLI test

## 1. Variable 변경

### AS-IS Variable



The screenshot shows the AWS MWAA console interface. At the top, there are navigation tabs: Airflow, DAGs, Security, Browse, Admin, and Docs. The main content area is titled 'List Variable'. It features a search bar and a table with columns: Key, Value, and Is Encrypted. The table contains one entry: 'training\_a' with a value of 'sagemaker\_a\_1' and 'Is Encrypted' set to 'True'. The table is highlighted with a red border. The bottom of the console shows the version information: Version: v2.0.2 and Git Version: release:2.0.2+e494306fb01f3a026e7e2832ca94902e96b526fa.

```
# MWAA CLI
variables set {KEY} {VALUE}
# json을 value 값으로 변경하려는 경우
variables set {KEY} --json '{"KEY" : "VALUE", "KEY" : "VALUE"}'
```

```
# Test code
import json
import boto3
import requests
import base64
from pprint import pprint

# Your MWAA Name
resource_id = "cjm-cdf-MwaaEnvironment"

client = boto3.client('mwaa', region_name='us-west-2')
token = client.create_cli_token(Name=resource_id)
url = "https://{0}/aws_mwaa/cli".format(token['WebServerHostname'])
# MWAA CLI to Body
body = """variables set training_a --json '{"Name" : "Romy", "Gender" :
"Female"}'"""
headers = {
    'Authorization' : 'Bearer '+token['CliToken'],
    'Content-Type': 'text/plain'
}
r = requests.post(url, data=body, headers=headers)

mwaa_std_out_message = base64.b64decode(r.json()['stdout']).decode('utf8')
pprint(mwaa_std_out_message)
```

The screenshot shows the Apache Airflow web interface. At the top, there are navigation links: Airflow, DAGs, Security, Browse, Admin, and Docs. The user is logged in at 13:54 UTC. Below the navigation bar, there is a section for '파일 선택' (File Selection) with a button 'Import Variables'. The main content area is titled 'List Variable' and contains a search bar and a table of variables. The table has columns: 'Key', 'Val', and 'Is Encrypted'. A red box highlights the 'Key' and 'Val' columns. The table shows one variable with the key 'training\_a' and the value '{"Name": "Romy", "Gender": "Female"}'. The 'Is Encrypted' column shows 'True'. The footer of the interface displays the version 'v2.0.2' and the Git commit hash '2.0.2+e494306fb01f3a026e7e2832ca94902e96b526fa'.

## 2. 전체 DAG 조회

```
# MWA CLI
# Output 변경 가능 (table, yaml, plain text)
dags report -o json
```

```
# Test code
import json
import boto3
import requests
import base64
from pprint import pprint

# Your MWA Name
resource_id = "cjm-cdf-MwaaEnvironment"

client = boto3.client('mwaa', region_name='us-west-2')
token = client.create_cli_token(Name=resource_id)
url = "https://{0}/aws_mwaa/cli".format(token['WebServerHostname'])
# MWA CLI to Body
body = "dags report -o json"
headers = {
    'Authorization': 'Bearer '+token['CliToken'],
    'Content-Type': 'text/plain'
}
r = requests.post(url, data=body, headers=headers)

mwaa_std_out_message = base64.b64decode(r.json()['stdout']).decode('utf8')
pprint(json.loads(mwaa_std_out_message))
```

```
// result json
[
  {
    'dag_num': '1',
    'dags': ['topic_2_dag_2'],
    'duration': '0:00:00.084525',
    'file': '/topic_2_dag_2.py',
    'task_num': '3'},
  ]
```

```

{
    'dag_num': '1',
    'dags': ['mwaa_emr_2'],
    'duration': '0:00:00.049332',
    'file': '/mwaa_emr_2.py',
    'task_num': '3'},
{
    'dag_num': '1',
    'dags': ['import_config'],
    'duration': '0:00:00.001544',
    'file': '/import_config.py',
    'task_num': '3'
}
]

```

### 3. 특정 DAG 조회

```

# MWAA CLI
# Output 변경 가능 (table, yaml, plain text)
dags list-jobs -d DAG_ID -o json

```

```

# Test code
import json
import boto3
import requests
import base64
from pprint import pprint

# Your MWAA Name
resource_id = "cjm-cdf-MwaaEnvironment"

client = boto3.client('mwaa', region_name='us-west-2')
token = client.create_cli_token(Name=resource_id)
url = "https://{0}/aws_mwaa/cli".format(token['WebServerHostname'])

# MWAA CLI to Body
body = "dags list-jobs -d import_config -o json"
headers = {
    'Authorization': 'Bearer '+token['CliToken'],
    'Content-Type': 'text/plain'
}

r = requests.post(url, data=body, headers=headers)

mwaa_std_out_message = base64.b64decode(r.json()['stdout']).decode('utf8')
pprint(json.loads(mwaa_std_out_message))

```

```
// result json
[
  {
    'dag_id': 'import_config',
    'end_date': '2021-09-13 14:20:33.075438+00:00',
    'job_type': 'LocalTaskJob',
    'start_date': '2021-09-13 14:20:29.162906+00:00',
    'state': 'success'
  }
]
```

#### 4. 전체 DAG 실행 내역 조회

```
# MWA CLI
# Output 변경 가능 (table, yaml, plain text)
dags list-runs -o json
```

```
# Test code
import json
import boto3
import requests
import base64
from pprint import pprint

# Your MWA Name
resource_id = "cjm-cdf-MwaaEnvironment"

client = boto3.client('mwaa', region_name='us-west-2')
token = client.create_cli_token(Name=resource_id)
url = "https://{0}/aws_mwaa/cli".format(token['webServerHostname'])
# MWA CLI to Body
body = "dags list-runs -o json"
headers = {
    'Authorization': 'Bearer '+token['CliToken'],
    'Content-Type': 'text/plain'
}
r = requests.post(url, data=body, headers=headers)

mwaa_std_out_message = base64.b64decode(r.json()['stdout']).decode('utf8')
pprint(json.loads(mwaa_std_out_message))
```

```
// result json
[
  {
    'dag_id': 'import_config',
    'end_date': '2021-09-13T14:20:33.002278+00:00',
    'execution_date': '2021-09-13T14:20:27.596561+00:00',
    'run_id': 'manual__2021-09-13T14:20:27.596561+00:00',
    'start_date': '2021-09-13T14:20:27.609722+00:00',
    'state': 'success'
  }
]
```

## 5. 특정 DAG 실행 내역 조회

```
# MWA CLI
# Output 변경 가능 (table, yaml, plain text)
dags list-runs -d DAG_ID -o json
```

```
# Test code
import json
import boto3
import requests
import base64
from pprint import pprint

# Your MWA Name
resource_id = "cjm-cdf-MwaaEnvironment"

client = boto3.client('mwaa', region_name='us-west-2')
token = client.create_cli_token(Name=resource_id)
url = "https://{0}/aws_mwaa/cli".format(token['webServerHostname'])
# MWA CLI to Body
body = "dags list-runs -d import_config -o json"
headers = {
    'Authorization': 'Bearer '+token['CliToken'],
    'Content-Type': 'text/plain'
}
r = requests.post(url, data=body, headers=headers)

mwaa_std_out_message = base64.b64decode(r.json()['stdout']).decode('utf8')
pprint(json.loads(mwaa_std_out_message))
```

```
// result json
[
  {
    'dag_id': 'import_config',
    'end_date': '2021-09-13T14:20:33.002278+00:00',
    'execution_date': '2021-09-13T14:20:27.596561+00:00',
    'run_id': 'manual__2021-09-13T14:20:27.596561+00:00',
    'start_date': '2021-09-13T14:20:27.609722+00:00',
    'state': 'success'
  }
]
```

## 6. 특정 DAG 실행 상태 조회

```
# MWA CLI
dags state DAG_ID execution_date
```

```
# Test code
import json
import boto3
import requests
import base64
import ast
```

```
from pprint import pprint

# Your MWA Name
resource_id = "cjm-cdf-MwaaEnvironment"

client = boto3.client('mwaa', region_name='us-west-2')
token = client.create_cli_token(Name=resource_id)
url = "https://{0}/aws_mwaa/cli".format(token['WebServerHostname'])
# MWA CLI to Body
body = "dags state import_config '2021-09-13T14:20:27.596561+00:00'"
headers = {
    'Authorization' : 'Bearer '+token['CliToken'],
    'Content-Type': 'text/plain'
}
r = requests.post(url, data=body, headers=headers)

mwaa_std_out_message = base64.b64decode(r.json()['stdout']).decode('utf8')
status = mwaa_std_out_message.split("\n")[-2]
pprint(status)
```

```
# result
'success'
```