



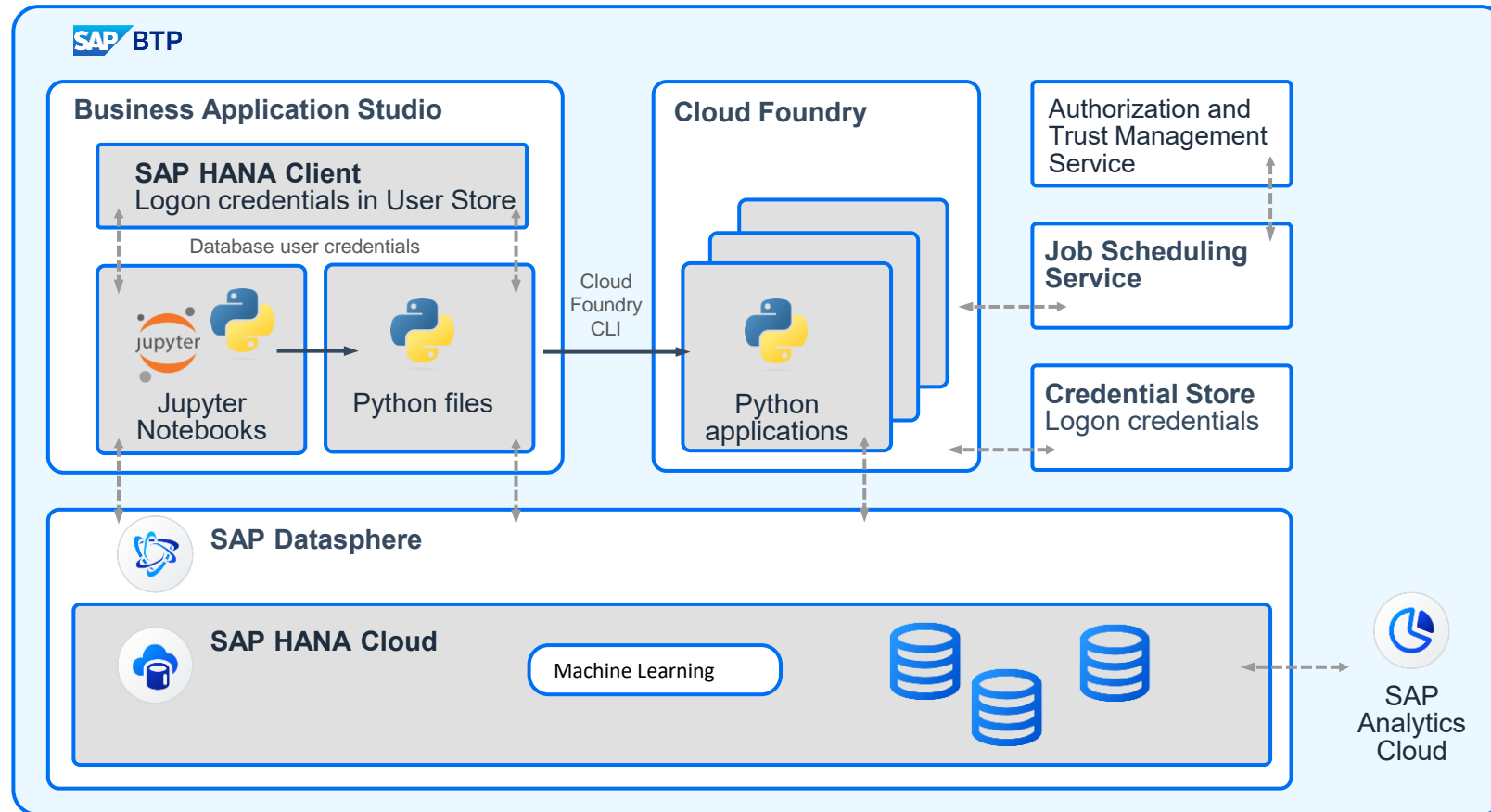
# Scheduling Python on Cloud Foundry With Business Application Studio and no local install

Andreas Forster, SAP



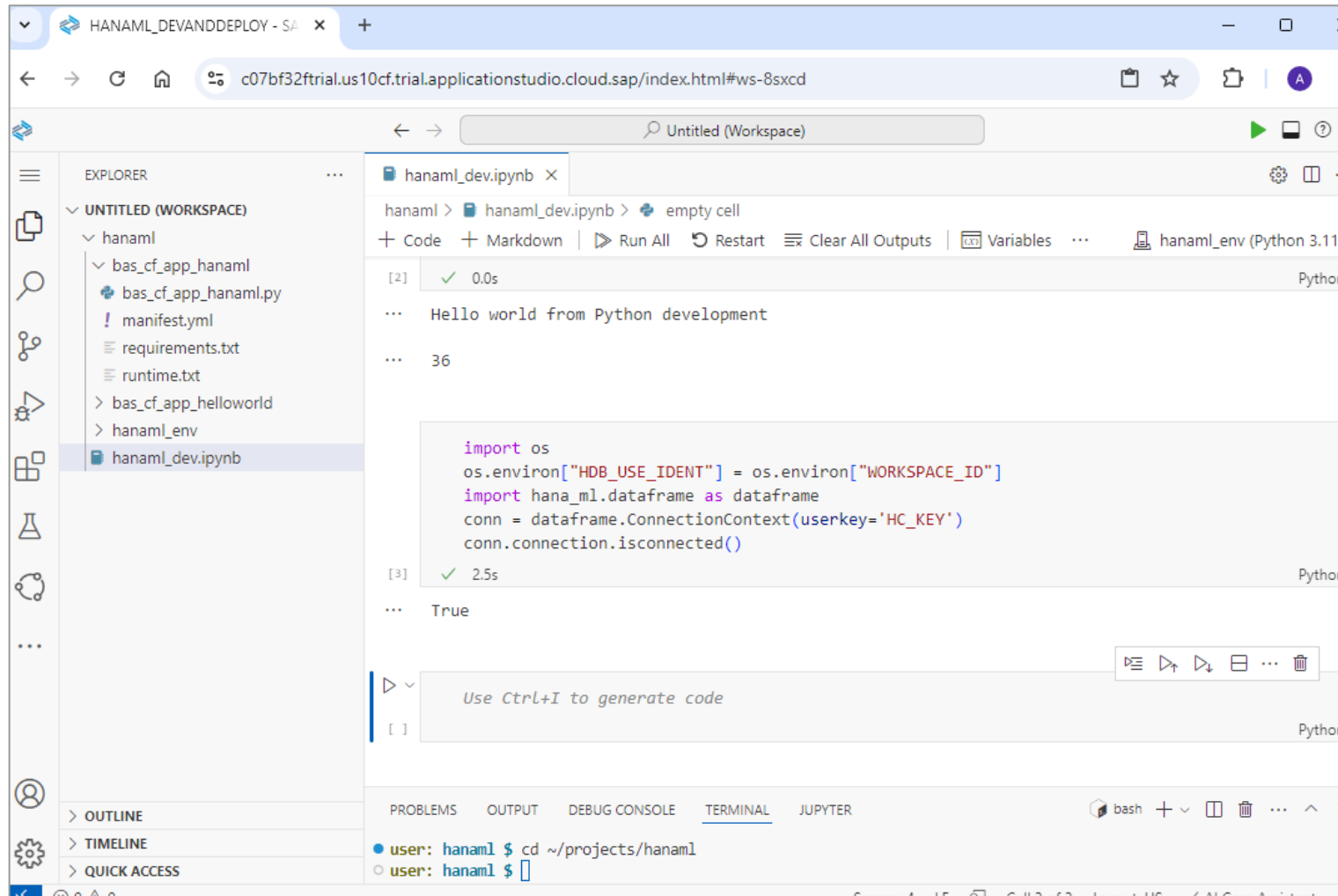
Global Center  
of Excellence  
SAP BTP

# SAP Datasphere Machine Learning with Business Application Studio Architecture



# Jupyter Notebook in Business Application Studio

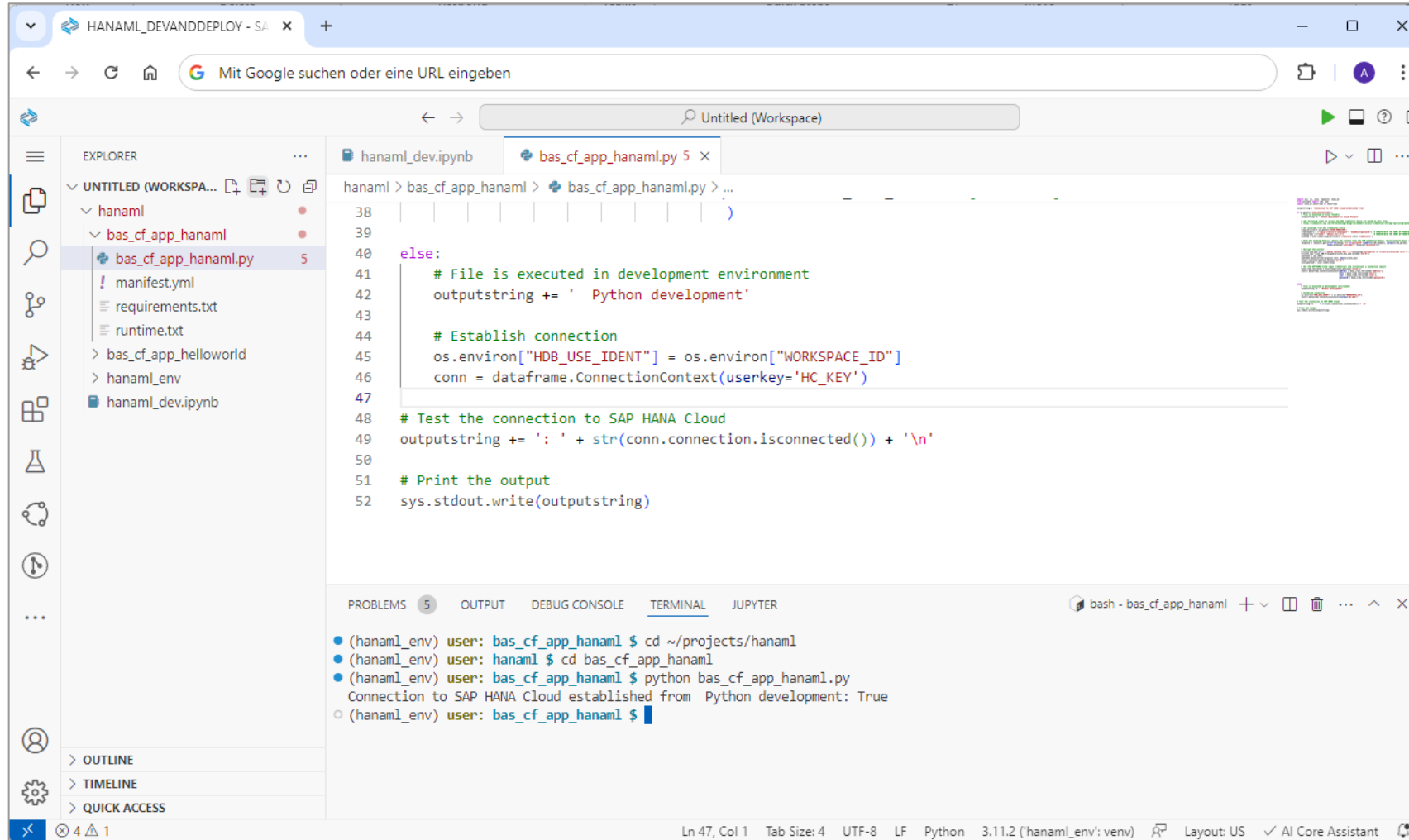
## Logon credentials from the SAP HANA Client's User Store



`source hanaml_env/bin/activate`

# Python application in Business Application Studio

## Logon credentials from the SAP HANA Client's User Store



The screenshot shows the Business Application Studio IDE interface. The Explorer panel on the left displays the project structure with folders 'hanaml' and 'bas\_cf\_app\_hanaml', and files 'manifest.yml', 'requirements.txt', 'runtime.txt', 'bas\_cf\_app\_helloworld', 'hanaml\_env', and 'hanaml\_dev.ipynb'. The main editor shows a Python script 'bas\_cf\_app\_hanaml.py' with the following code:

```
38 |  
39 |  
40 | else:  
41 |     # File is executed in development environment  
42 |     outputstring += ' Python development'  
43 |  
44 |     # Establish connection  
45 |     os.environ["HDB_USE_IDENT"] = os.environ["WORKSPACE_ID"]  
46 |     conn = dataframe.ConnectionContext(userkey='HC_KEY')  
47 |  
48 | # Test the connection to SAP HANA Cloud  
49 | outputstring += ': ' + str(conn.connection.isconnected()) + '\n'  
50 |  
51 | # Print the output  
52 | sys.stdout.write(outputstring)
```

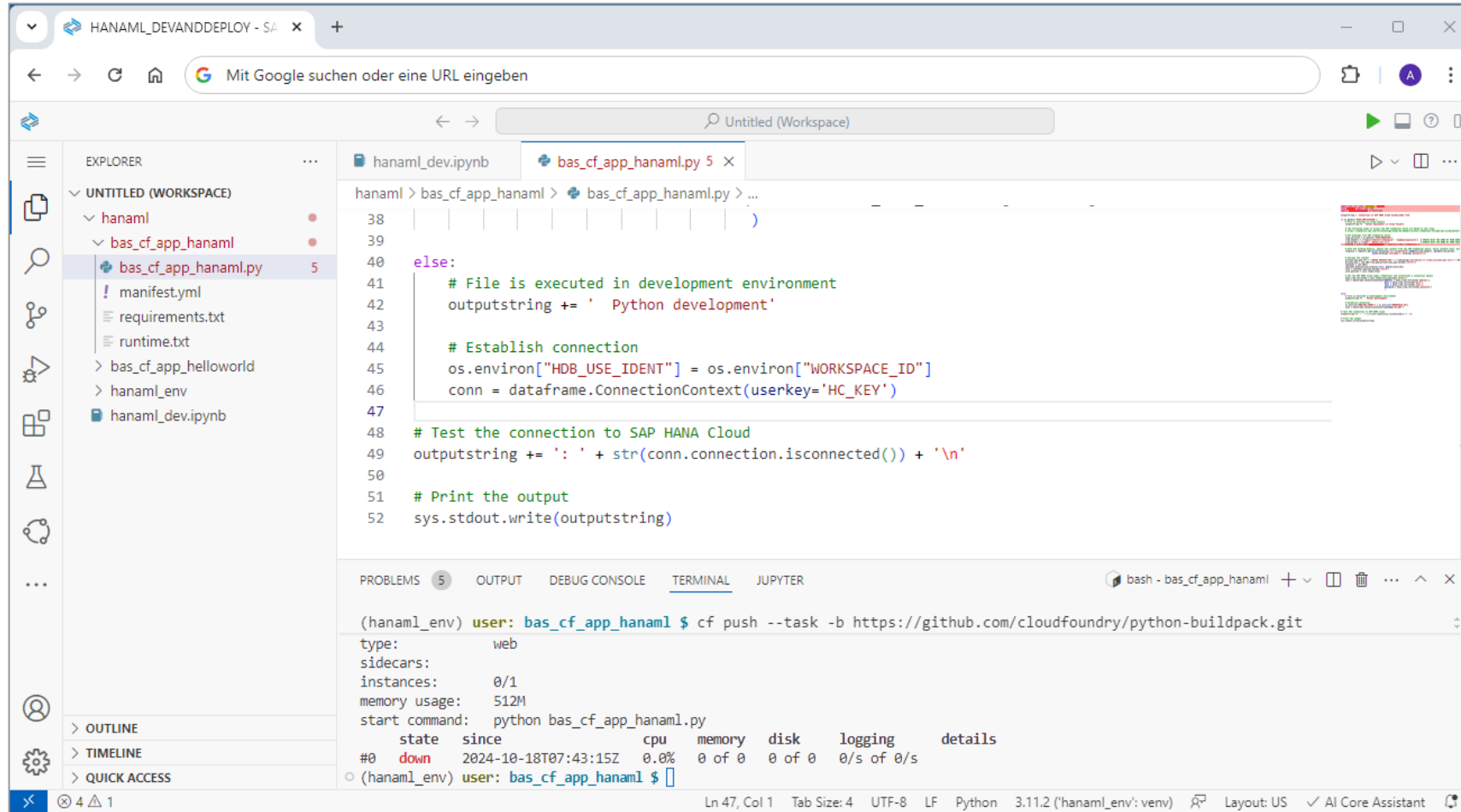
The bottom panel shows the terminal output for the command 'python bas\_cf\_app\_hanaml.py':

```
(hanaml_env) user: bas_cf_app_hanaml $ cd ~/projects/hanaml  
(hanaml_env) user: hanaml $ cd bas_cf_app_hanaml  
(hanaml_env) user: bas_cf_app_hanaml $ python bas_cf_app_hanaml.py  
Connection to SAP HANA Cloud established from Python development: True  
(hanaml_env) user: bas_cf_app_hanaml $
```

```
cd ~/projects/hanaml  
cd bas_cf_app_hanaml  
python bas_cf_app_hanaml.py
```

# Python application in Business Application Studio

## Deploy as application to Cloud Foundry

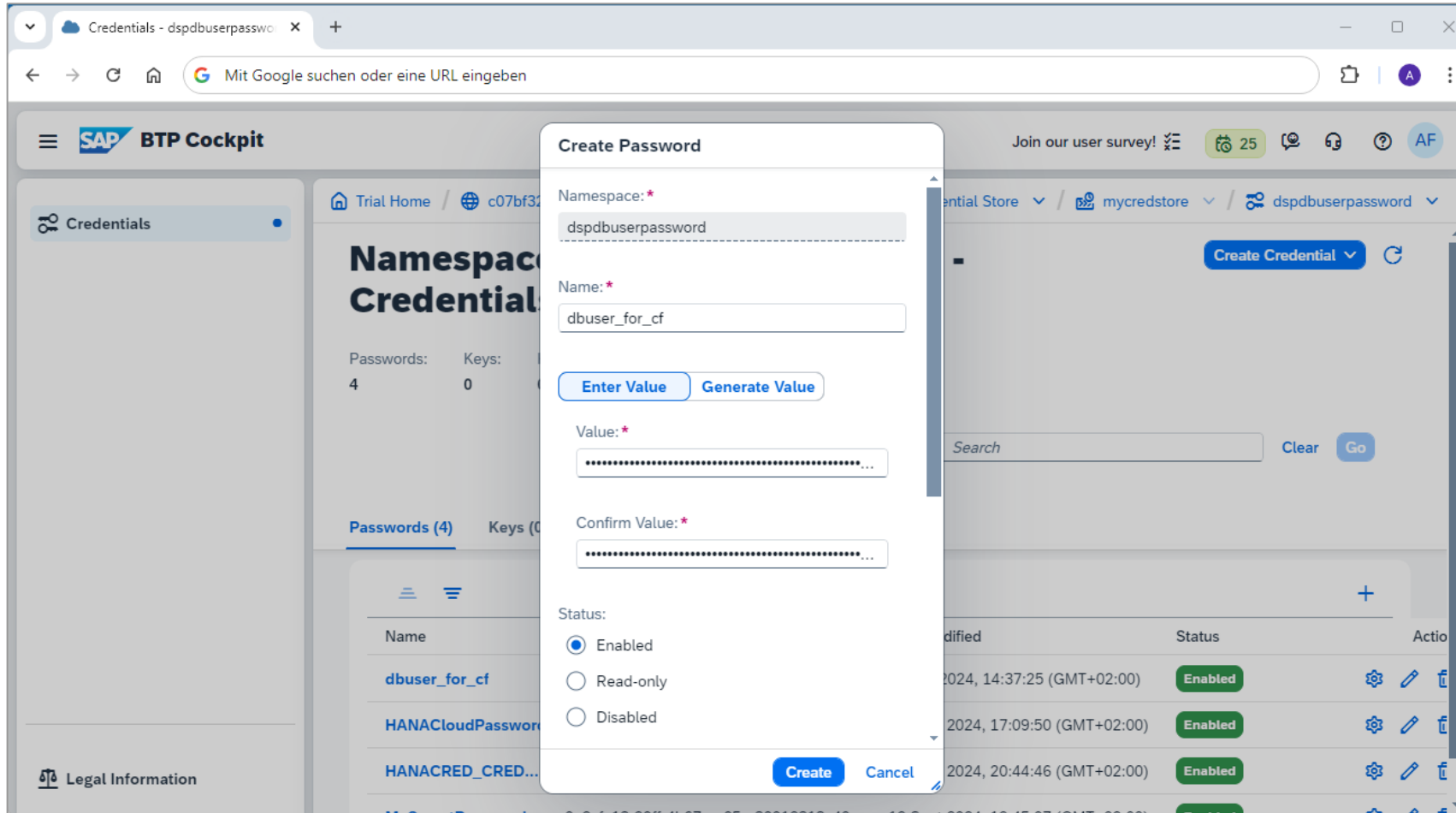


cf login

cf push --task -b <https://github.com/cloudfoundry/python-buildpack.git>

# Python application in Cloud Foundry

## Store database user credentials in Credential Store



The screenshot shows the SAP BTP Cockpit interface. A modal dialog titled "Create Password" is open, allowing the creation of a new password for a credential. The dialog has the following fields and options:

- Namespace:** A text field containing "dspdbuserpassword".
- Name:** A text field containing "dbuser\_for\_cf".
- Value:** A masked text field (dots) for the password.
- Confirm Value:** A masked text field (dots) for confirming the password.
- Status:** Radio buttons for "Enabled" (selected), "Read-only", and "Disabled".
- Buttons:** "Enter Value", "Generate Value", "Create", and "Cancel".

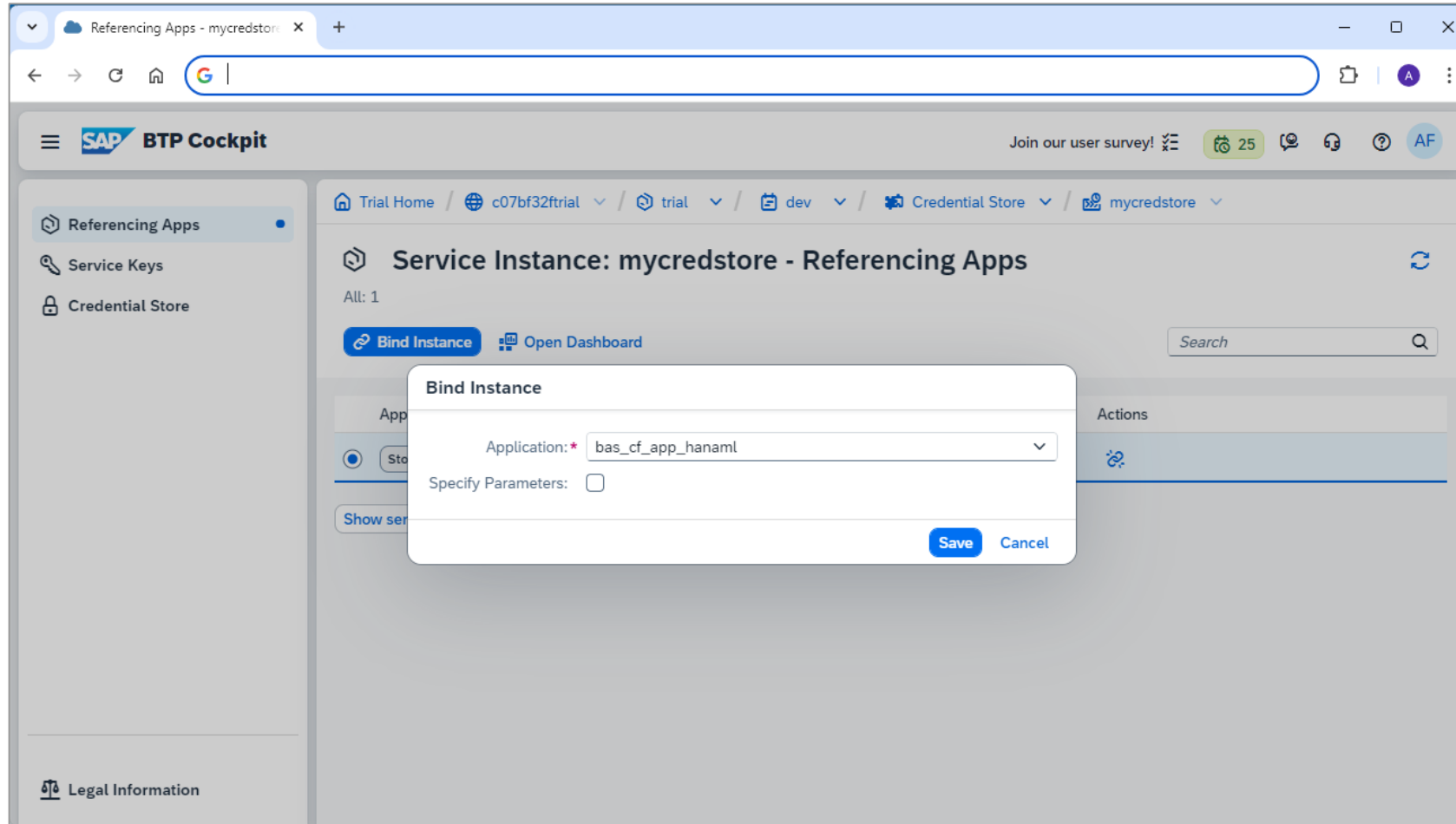
The background shows the "Credentials" section of the SAP BTP Cockpit, with a table listing existing credentials. The table has columns for Name, Status, and Action.

Name	Status	Action
dbuser_for_cf	Enabled	[Settings] [Edit] [Delete]
HANACloudPassword	Enabled	[Settings] [Edit] [Delete]
HANACRED_CRED...	Enabled	[Settings] [Edit] [Delete]

```
{"address": "REPLACEWITHYOURHANASERVER", "port": 443, "user": "REPLACEWITHYOURUSER",  
"password": "REPLACEWITHYOURPASSWORD"}
```

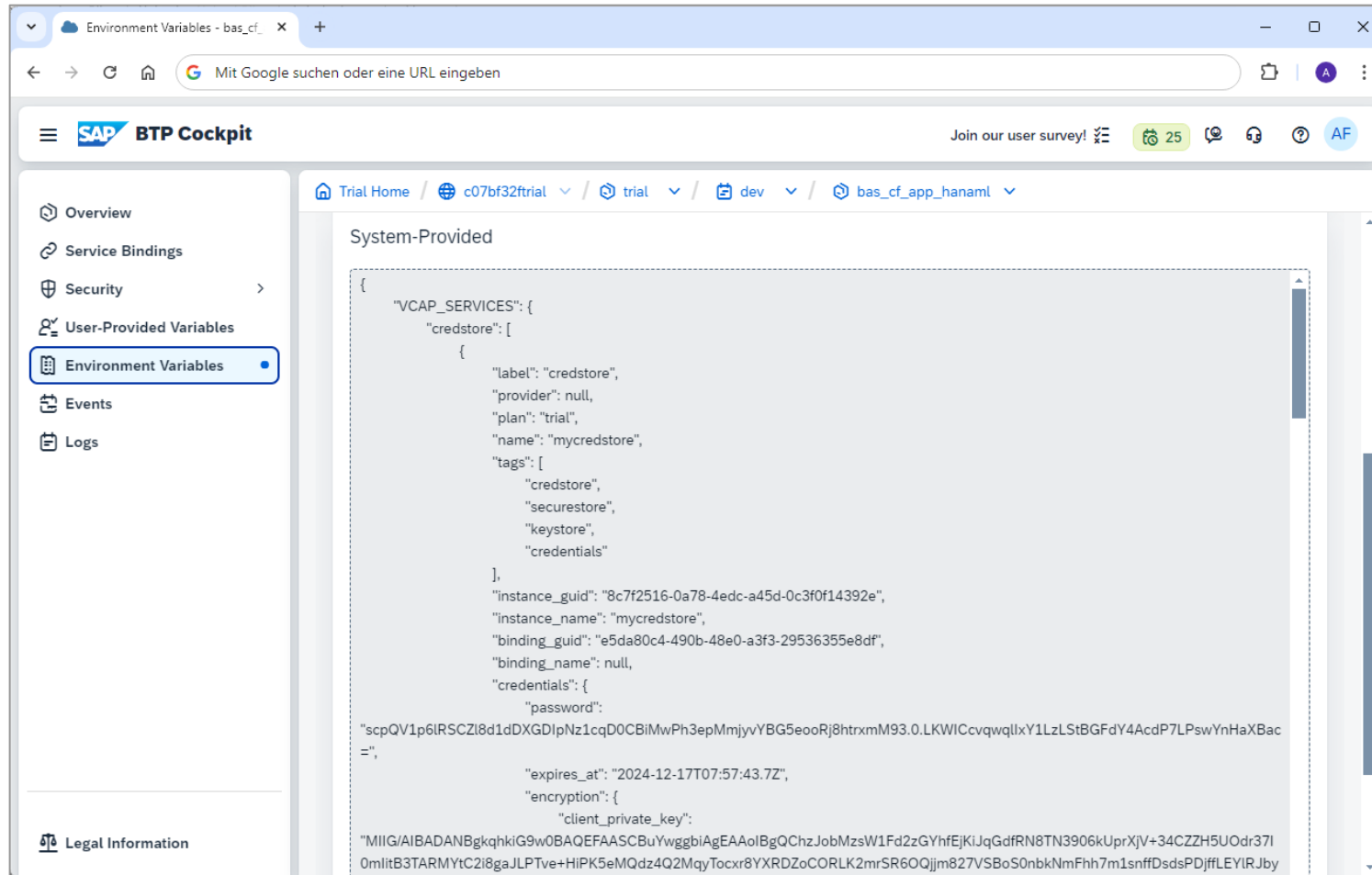
# Python application in Cloud Foundry

## Bind instance of Credential Store to the Cloud Foundry Python application



# Python application in Cloud Foundry

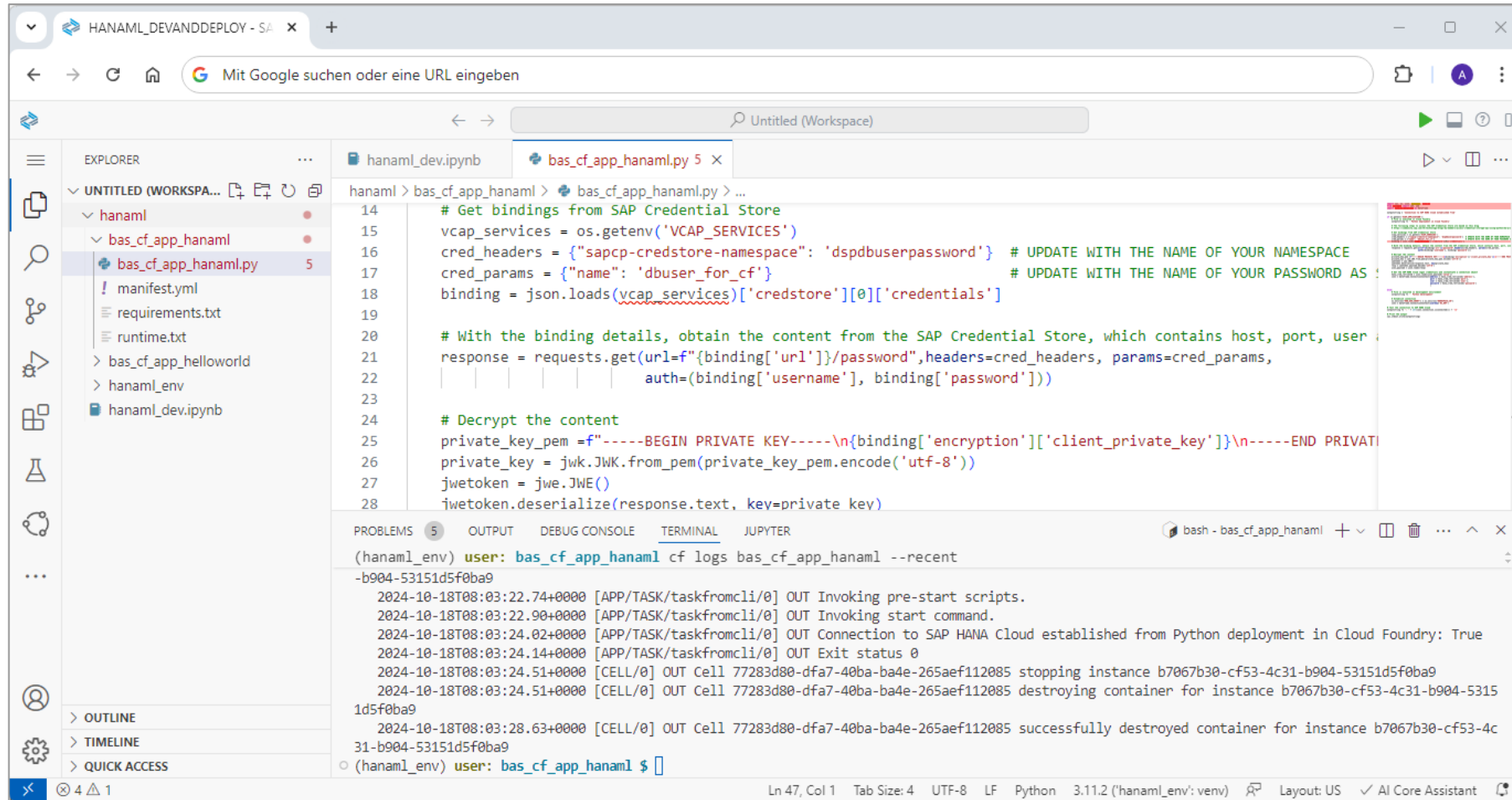
## Environment variable of the application now allow access to the Credential Store





# Python application in Cloud Foundry

## Run application via Cloud Foundry CLI



```
hanaml > bas_cf_app_hanaml > bas_cf_app_hanaml.py > ...
14 # Get bindings from SAP Credential Store
15 vcap_services = os.getenv('VCAP_SERVICES')
16 cred_headers = {"sapcp-credstore-namespace": 'dspdbuserpassword'} # UPDATE WITH THE NAME OF YOUR NAMESPACE
17 cred_params = {"name": 'dbuser_for_cf'} # UPDATE WITH THE NAME OF YOUR PASSWORD AS
18 binding = json.loads(vcap_services)['credstore'][0]['credentials']
19
20 # With the binding details, obtain the content from the SAP Credential Store, which contains host, port, user
21 response = requests.get(url=f"{binding['url']}/password", headers=cred_headers, params=cred_params,
22                          auth=(binding['username'], binding['password']))
23
24 # Decrypt the content
25 private_key_pem = f"-----BEGIN PRIVATE KEY-----\n{binding['encryption']['client_private_key']}\n-----END PRIVATE KEY-----"
26 private_key = jwk.JWK.from_pem(private_key_pem.encode('utf-8'))
27 jwtoken = jwe.JWE()
28 jwtoken.deserialize(response.text, key=private_key)
```

```
(hanaml_env) user: bas_cf_app_hanaml cf logs bas_cf_app_hanaml --recent
-b904-53151d5f0ba9
2024-10-18T08:03:22.74+0000 [APP/TASK/taskfromcli/0] OUT Invoking pre-start scripts.
2024-10-18T08:03:22.90+0000 [APP/TASK/taskfromcli/0] OUT Invoking start command.
2024-10-18T08:03:24.02+0000 [APP/TASK/taskfromcli/0] OUT Connection to SAP HANA Cloud established from Python deployment in Cloud Foundry: True
2024-10-18T08:03:24.14+0000 [APP/TASK/taskfromcli/0] OUT Exit status 0
2024-10-18T08:03:24.51+0000 [CELL/0] OUT Cell 77283d80-dfa7-40ba-ba4e-265aef112085 stopping instance b7067b30-cf53-4c31-b904-53151d5f0ba9
2024-10-18T08:03:24.51+0000 [CELL/0] OUT Cell 77283d80-dfa7-40ba-ba4e-265aef112085 destroying container for instance b7067b30-cf53-4c31-b904-53151d5f0ba9
2024-10-18T08:03:28.63+0000 [CELL/0] OUT Cell 77283d80-dfa7-40ba-ba4e-265aef112085 successfully destroyed container for instance b7067b30-cf53-4c31-b904-53151d5f0ba9
(hanaml_env) user: bas_cf_app_hanaml $
```

```
cf run-task bas_cf_app_hanaml --command "python bas_cf_app_hanaml.py" --name taskfromcli
cf logs bas_cf_app_hanaml --recent
```

# Python application in Cloud Foundry

## Schedule application with Job Scheduling Service

Job Scheduling Service Dashboard

Service Instance: jobschedulerinstance / Task: bas\_cf\_app\_hanaml\_task / Schedule: f078040c-2425-4491-bfa2-9ac87f5c59a7

Schedule Run Logs

Schedule Pattern: Recurring Schedule (Repeat Interval) - 1 hour Total Runs: 44

Download run logs

View 10 rows 1 of 5 Pages

Runlog ID	Scheduled time (UTC)	Scheduled to run at (UTC)	Execution time (UTC)	Completion time (UTC)	Status/State	Skipped runs
47e3f502-b7fc-4146-901a-fcd79d37a282	Oct 18, 2024, 7:15:24 AM	2024-10-18 08:15:24	Oct 18, 2024, 8:14:56 AM	Oct 18, 2024, 8:14:56 AM	SCHEDULED/SCHEDULED	0
0f34946d-b4c0-4f42-bba9-1c5ac861c4bb	Oct 18, 2024, 6:15:24 AM	2024-10-18 07:15:24	Oct 18, 2024, 7:15:24 AM	Oct 18, 2024, 7:16:02 AM	COMPLETED/SUCCESS	0
9b55f373-f60e-45eb-9a06-16852b78eaaa	Oct 18, 2024, 5:15:24 AM	2024-10-18 06:15:24	Oct 18, 2024, 6:15:24 AM	Oct 18, 2024, 6:16:02 AM	COMPLETED/SUCCESS	0
cbc85c17-0308-48e9-b27f-983975131bb1	Oct 18, 2024, 4:15:24 AM	2024-10-18 05:15:24	Oct 18, 2024, 5:15:24 AM	Oct 18, 2024, 5:15:52 AM	COMPLETED/SUCCESS	0
10d9ce60-5c49-4911-92c2-824030c87d54	Oct 18, 2024, 3:15:24 AM	2024-10-18 04:15:24	Oct 18, 2024, 4:15:24 AM	Oct 18, 2024, 4:15:52 AM	COMPLETED/SUCCESS	0
d29b3744-99e9-4571-812c-f91ebfc825bd	Oct 18, 2024, 2:15:24 AM	2024-10-18 03:15:24	Oct 18, 2024, 3:15:24 AM	Oct 18, 2024, 3:15:52 AM	COMPLETED/SUCCESS	0
9422ed7b-2a73-4571-8f4d-965485d02363	Oct 18, 2024, 1:15:24 AM	2024-10-18 02:15:24	Oct 18, 2024, 2:15:24 AM	Oct 18, 2024, 2:15:52 AM	COMPLETED/SUCCESS	0
cdb5bc8c-4b86-4ea5-b9db-64962afb8f5e	Oct 18, 2024, 12:15:24 AM	2024-10-18 01:15:24	Oct 18, 2024, 1:15:24 AM	Oct 18, 2024, 1:15:52 AM	COMPLETED/SUCCESS	0
0137a06a-cf2d-4754-9c05-fdd742640d90	Oct 17, 2024, 11:15:24 PM	2024-10-18 00:15:24	Oct 18, 2024, 12:15:24 AM	Oct 18, 2024, 12:16:01 AM	COMPLETED/SUCCESS	0
8ff909d5-2bfb-47b8-badf-1b4e8545911b	Oct 17, 2024, 10:15:24 PM	2024-10-17 23:15:24	Oct 17, 2024, 11:15:24 PM	Oct 17, 2024, 11:15:52 PM	COMPLETED/SUCCESS	0

Useful Links

Legal Information

<https://community.sap.com/t5/technology-blogs-by-sap/scheduling-python-code-on-cloud-foundry/ba-p/13503697>

# Thank you.

**Andreas Forster**  
Chief AI/ML Expert  
Global Center of Excellence  
[andreas.forster@sap.com](mailto:andreas.forster@sap.com)

