

Übungsaufgabe: Transformation der abgerufenen Daten

Voraussetzungen

- Docker und Docker Compose sind installiert und das Airflow-Setup aus Sprint 1 läuft

User Story 3: Transformation der abgerufenen Daten

Beschreibung: Als Entwickler möchte ich die aus dem MiniIO-Bucket abgerufenen Daten in ein standardisiertes Format transformieren, um sie später in einer Datenbank zu speichern.

Akzeptanzkriterien:

- Die Datei wird erfolgreich transformiert und in einem neuen Bucket gespeichert.

Schritte zur Umsetzung:

1. **Demo DAGs deaktivieren:** Deaktivieren Sie beim Start von Airflow die Demo DAGs.
<https://stackoverflow.com/questions/43410836/how-to-remove-default-example-dags-in-airflow>
2. **DAG erstellen:** Öffnen Sie die Datei `transformation_dag_exercise.py` und lösen Sie die Aufgaben. Kopieren Sie die Datei in das Verzeichnis `dev-environment/dags/`. Der DAG erscheint nach einigen Sekunden im Web-Interface.
3. **io.StringIO:**

Recherchieren Sie die Dokumentation von `io.StringIO` und erklären Sie in eigenen Worten, wie es funktioniert und warum wir es hier verwenden um die Funktion `read_csv` zu unterstützen.
4. **DAG ausführen:** Aktivieren Sie den DAG und führen Sie ihn aus. Überprüfen Sie die Logs im Web-Interface, ob die Datei erfolgreich transformiert wurde.

Ressourcen:

Airflow

- [Airflow S3 Hook](#)
- [Airflow DAG Context Manager](#)
- [Airflow S3 Load File](#)
- [Airflow DAG Documentation](#)

Python

- `io.StringIO`
- `pandas read_csv`
- `pandas to_csv`
- `pandas round`
- `pandas to_datetime`

