TP3 – Data Lake & Data Warehouse (Kafka, KSQLDB, Python, SQLite/MySQL)

1. Design du Data Lake

Élément	Description
Emplacement	**Le Data Lake est un dossier local nommé data_lake/ sur la machine.**
Structure	**Les données provenant de Kafka/KSQLDB sont stockées dans des sous-dossiers par topic et date : data_lake/<topic>/<YYYY-MM-DD>/<topic>_<YYYY-MM-DD>.jsonl**
Partitionnement	**Partition par date (ou version) pour faciliter la traçabilité et les chargements incrémentaux.**
Ajout de nouveaux feeds	**Lorsqu'un nouveau flux Kafka est ajouté :** **1) Déclaration dans config/settings.py** **2) Le consumer Kafka crée automatiquement le dossier ;** **3) Le pipeline est réutilisé sans modification du code.**
Stockage Streams & Tables	**Chaque stream et table est stocké sous forme de fichiers JSONL distincts.**
Mode de stockagex	**Streams → append ; Tables agrégées → overwrite ; Données statiques → ignore.**

Structure

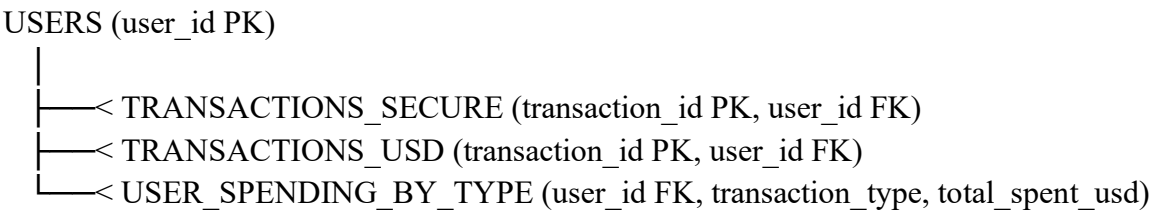
```text
data_lake/
├── config/
│   └── settings.py
├── consumers/
│   ├── kafka_to_datalake.py
│   └── fake_kafka_producer.py
├── jobs/
│   └── sqlite_loader.py
├── pipeline/
│   ├── TRANSACTIONS_SECURE/
│   ├── TRANSACTIONS_USD/
│   ├── TRANSACTIONS_BLACKLISTED/
│   ├── USER_SPENDING_BY_TYPE/
│   └── SPEND_LAST_FIVE_MIN_BY_TYPE/
└── data_warehouse.db
```

2. Design du Data Warehouse (MySQL / SQLite)

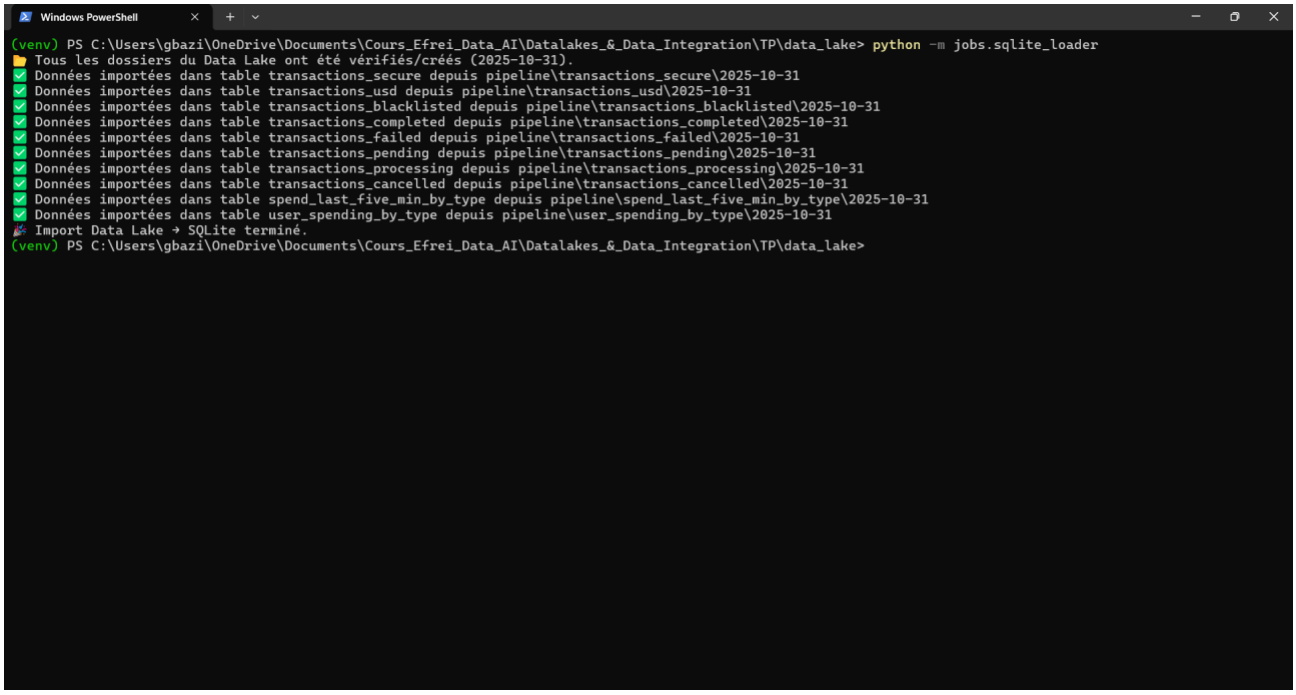
Toutes les tables de KSQLDB sont répliquées dans le Data Warehouse via un job Python (sqlite_loader.py).

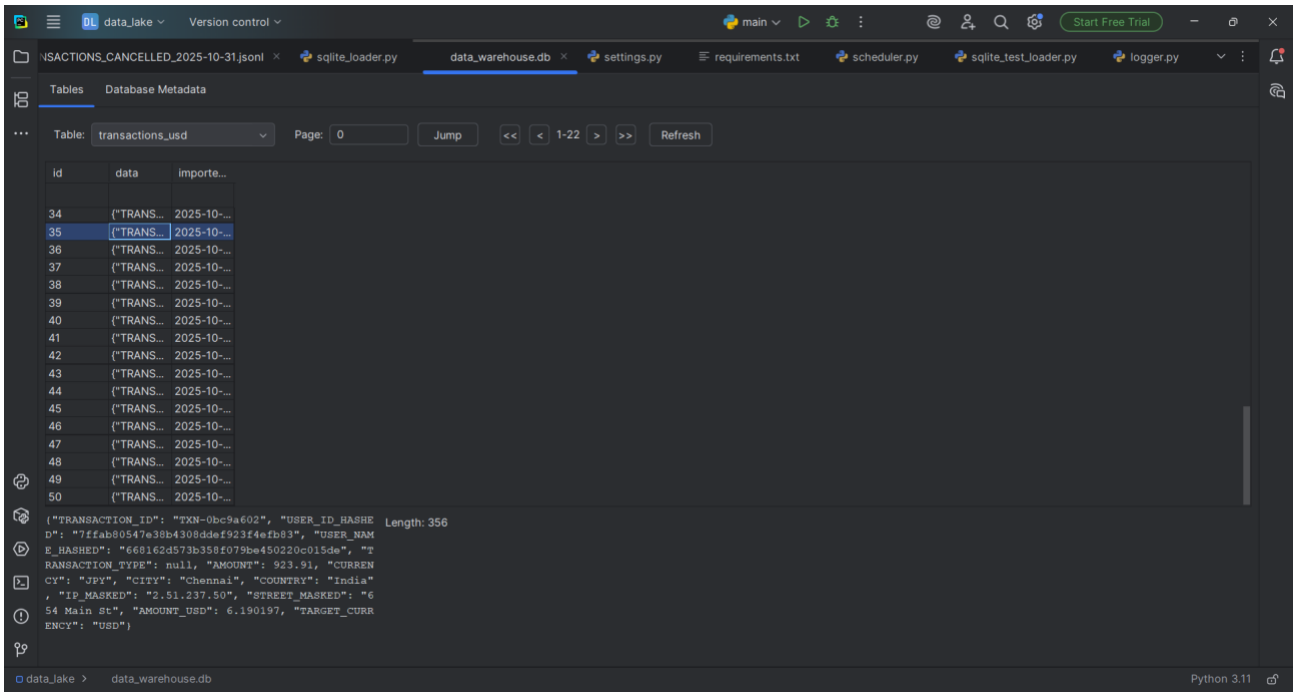
Table	Clé primaire	Clés étrangères	Description
TRANSACTIONS_SECURE	transaction_id	user_id	Transactions brutes anonymisées
TRANSACTIONS_USD	transaction_id	user_id	Transactions converties en USD
USER_SPENDING_BY_TYPE	(user_id, transaction_type)	user_id	Agrégations par type et utilisateur
SPEND_LAST_FIVE_MIN_BY_TYPE	transaction_type	-	Dépense glissante 5 min
USERS	user_id	-	Utilisateurs anonymisés
PERMISSIONS	(user_id, topic_name)	user_id	Gestion des accès par utilisateur

Diagramme Entité-Association (textuel) :



PERMISSIONS (user_id FK, topic_name, access_level)



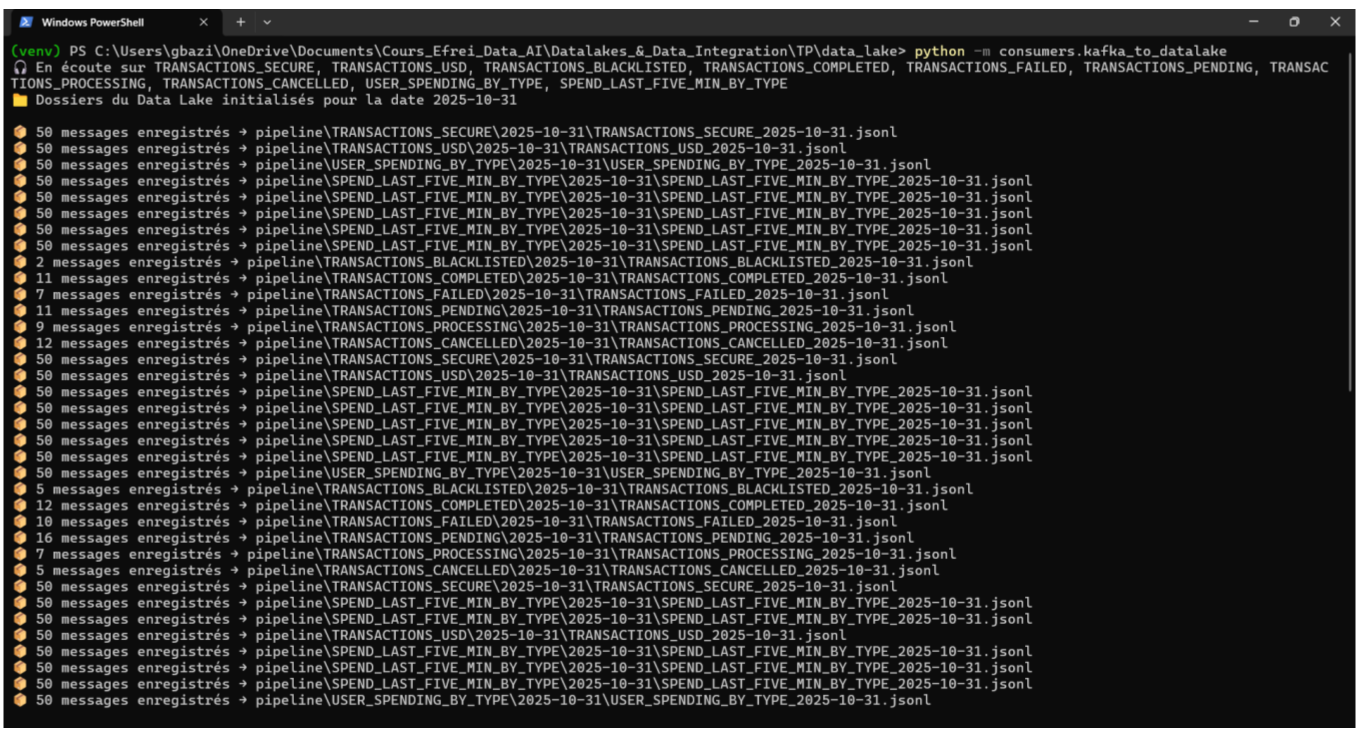


3. Kafka Consumers

Les Kafka Consumers permettent de lire les messages, les écrire dans le Data Lake et charger les données dans le Data Warehouse.

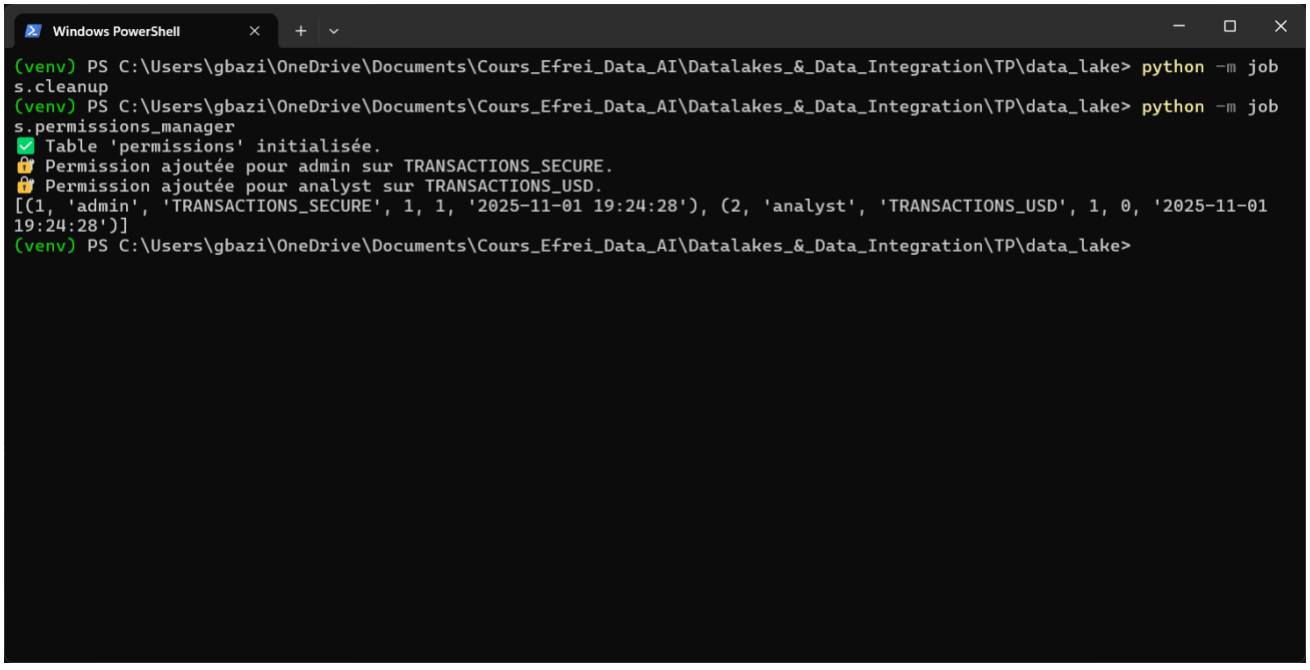
Exemple simplifié :

```
from kafka import KafkaConsumer
import json, os, datetime

consumer = KafkaConsumer('TRANSACTIONS_SECURE',
bootstrap_servers='localhost:9092')
for msg in consumer:
    data = json.loads(msg.value)
    date_str = datetime.date.today().isoformat()
    path = f"data_lake/TRANSACTIONS_SECURE/{date_str}/"
    os.makedirs(path, exist_ok=True)
    with open(f"{path}/data.jsonl", "a") as f:
        f.write(json.dumps(data) + "\n")
```



4. Gouvernance et Sécurité

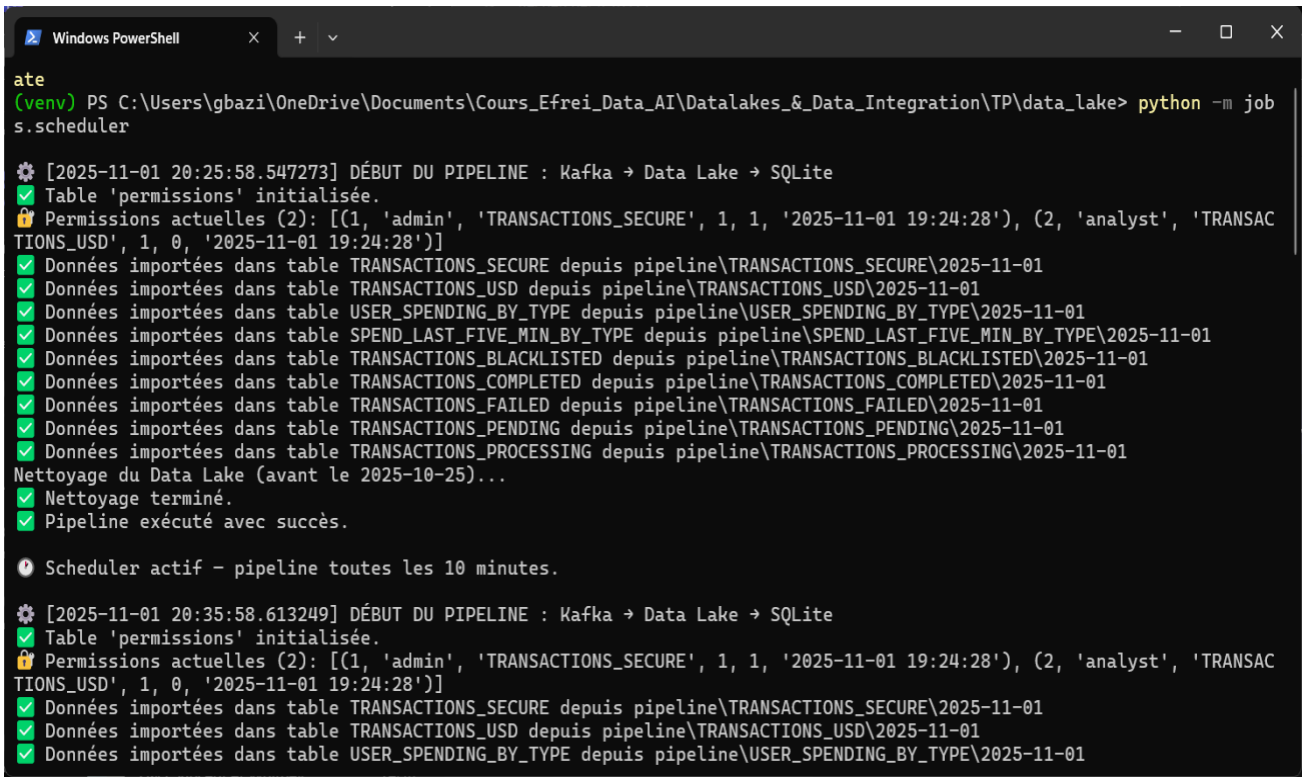
Domaine	Mécanisme	Objectif
Suppression historique	Script Python supprimant les données après 30 jours	Optimisation & conformité RGPD
Gestion des droits	Table permissions (user_id, topic_name, access_level)	Contrôle des accès
Feed Kafka	Déclaration dans settings.py → dossier auto → pipeline réutilisé	Gain de temps
Anonymisation	Masquage KSQLDB (MASK, HASH, CONCAT)	Protection des données sensibles
Traçabilité	Transaction_id + timestamp	Audit & transparence



5. Orchestration et Optimisation

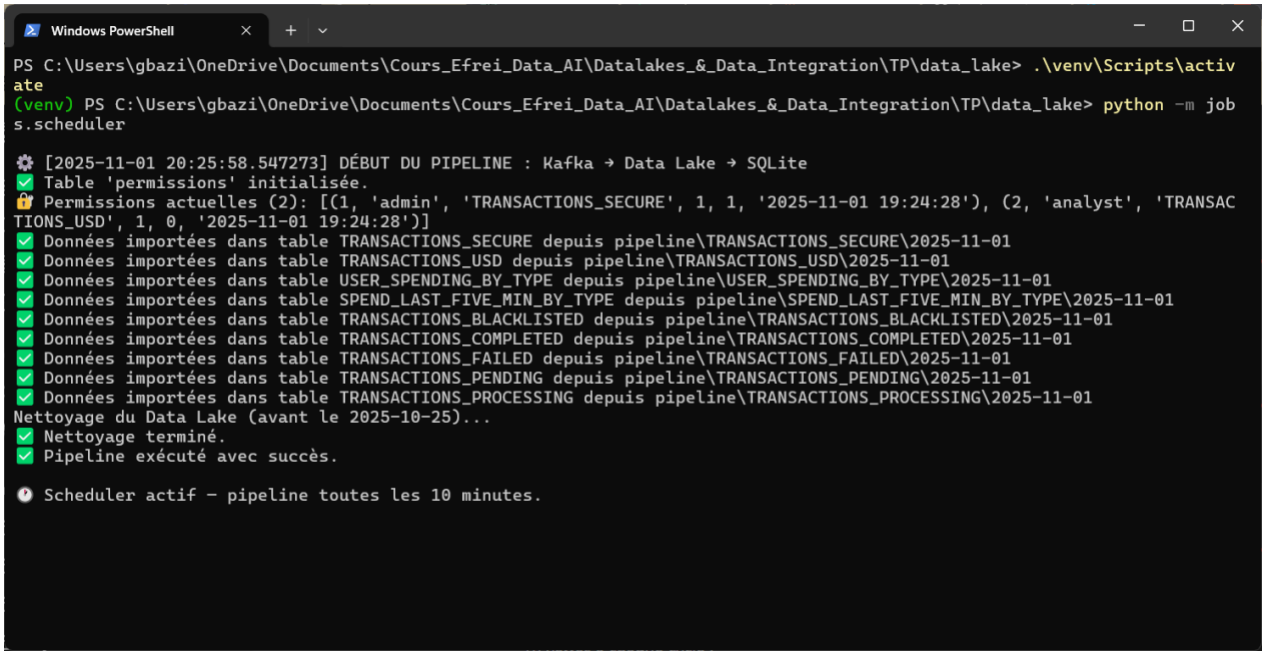
Orchestration automatisée via schedule ou Apache Beam (toutes les 10 minutes).

```
import schedule, os, time

def run_jobs():
    os.system("python consumers/kafka_to_datalake.py")
    os.system("python jobs/sqlite_loader.py")

schedule.every(10).minutes.do(run_jobs)
while True:
    schedule.run_pending()
    time.sleep(60)
```



Synthèse

Domaine	Résumé
Data Lake	Stockage brut JSON partitionné par date, extensible, append/overwrite selon le type de données.
Data Warehouse	Schéma relationnel clair (SQLite/MySQL), clés primaires et étrangères définies.
Kafka Consumers	Automatisation de la collecte et intégration continue.
Sécurité & Gouvernance	Rétention, anonymisation, permissions et logs d'audit.
Orchestration	Automatisation des jobs toutes les 10 minutes via schedule / Apache Beam.



Windows PowerShell

```
(venv) PS C:\Users\gbazi\OneDrive\Documents\Cours_Efrei_Data_AI\Datalakes_&_Data_Integration\TP\data_lake> python -m consumers.kafka_to_datalake
🎧 En écoute sur TRANSACTIONS_SECURE, TRANSACTIONS_USD, TRANSACTIONS_BLACKLISTED, TRANSACTIONS_COMPLETED, TRANSACTIONS_FAILED, TRANSACTIONS_PENDING, TRANSAC
TIONS_PROCESSING, TRANSACTIONS_CANCELLED, USER_SPENDING_BY_TYPE, SPEND_LAST_FIVE_MIN_BY_TYPE
📁 Dossiers du Data Lake initialisés pour la date 2025-10-31

📦 50 messages enregistrés → pipeline\TRANSACTIONS_SECURE\2025-10-31\TRANSACTIONS_SECURE_2025-10-31.jsonl
📦 50 messages enregistrés → pipeline\TRANSACTIONS_USD\2025-10-31\TRANSACTIONS_USD_2025-10-31.jsonl
📦 50 messages enregistrés → pipeline\USER_SPENDING_BY_TYPE\2025-10-31\USER_SPENDING_BY_TYPE_2025-10-31.jsonl
📦 50 messages enregistrés → pipeline\SPEND_LAST_FIVE_MIN_BY_TYPE\2025-10-31\SPEND_LAST_FIVE_MIN_BY_TYPE_2025-10-31.jsonl
📦 50 messages enregistrés → pipeline\SPEND_LAST_FIVE_MIN_BY_TYPE\2025-10-31\SPEND_LAST_FIVE_MIN_BY_TYPE_2025-10-31.jsonl
📦 50 messages enregistrés → pipeline\SPEND_LAST_FIVE_MIN_BY_TYPE\2025-10-31\SPEND_LAST_FIVE_MIN_BY_TYPE_2025-10-31.jsonl
📦 50 messages enregistrés → pipeline\SPEND_LAST_FIVE_MIN_BY_TYPE\2025-10-31\SPEND_LAST_FIVE_MIN_BY_TYPE_2025-10-31.jsonl
📦 50 messages enregistrés → pipeline\SPEND_LAST_FIVE_MIN_BY_TYPE\2025-10-31\SPEND_LAST_FIVE_MIN_BY_TYPE_2025-10-31.jsonl
📦 2 messages enregistrés → pipeline\TRANSACTIONS_BLACKLISTED\2025-10-31\TRANSACTIONS_BLACKLISTED_2025-10-31.jsonl
📦 11 messages enregistrés → pipeline\TRANSACTIONS_COMPLETED\2025-10-31\TRANSACTIONS_COMPLETED_2025-10-31.jsonl
📦 7 messages enregistrés → pipeline\TRANSACTIONS_FAILED\2025-10-31\TRANSACTIONS_FAILED_2025-10-31.jsonl
📦 11 messages enregistrés → pipeline\TRANSACTIONS_PENDING\2025-10-31\TRANSACTIONS_PENDING_2025-10-31.jsonl
📦 9 messages enregistrés → pipeline\TRANSACTIONS_PROCESSING\2025-10-31\TRANSACTIONS_PROCESSING_2025-10-31.jsonl
📦 12 messages enregistrés → pipeline\TRANSACTIONS_CANCELLED\2025-10-31\TRANSACTIONS_CANCELLED_2025-10-31.jsonl
📦 50 messages enregistrés → pipeline\TRANSACTIONS_SECURE\2025-10-31\TRANSACTIONS_SECURE_2025-10-31.jsonl
📦 50 messages enregistrés → pipeline\TRANSACTIONS_USD\2025-10-31\TRANSACTIONS_USD_2025-10-31.jsonl
📦 50 messages enregistrés → pipeline\SPEND_LAST_FIVE_MIN_BY_TYPE\2025-10-31\SPEND_LAST_FIVE_MIN_BY_TYPE_2025-10-31.jsonl
📦 50 messages enregistrés → pipeline\SPEND_LAST_FIVE_MIN_BY_TYPE\2025-10-31\SPEND_LAST_FIVE_MIN_BY_TYPE_2025-10-31.jsonl
📦 50 messages enregistrés → pipeline\SPEND_LAST_FIVE_MIN_BY_TYPE\2025-10-31\SPEND_LAST_FIVE_MIN_BY_TYPE_2025-10-31.jsonl
📦 50 messages enregistrés → pipeline\SPEND_LAST_FIVE_MIN_BY_TYPE\2025-10-31\SPEND_LAST_FIVE_MIN_BY_TYPE_2025-10-31.jsonl
📦 50 messages enregistrés → pipeline\SPEND_LAST_FIVE_MIN_BY_TYPE\2025-10-31\SPEND_LAST_FIVE_MIN_BY_TYPE_2025-10-31.jsonl
📦 50 messages enregistrés → pipeline\USER_SPENDING_BY_TYPE\2025-10-31\USER_SPENDING_BY_TYPE_2025-10-31.jsonl
📦 5 messages enregistrés → pipeline\TRANSACTIONS_BLACKLISTED\2025-10-31\TRANSACTIONS_BLACKLISTED_2025-10-31.jsonl
📦 12 messages enregistrés → pipeline\TRANSACTIONS_COMPLETED\2025-10-31\TRANSACTIONS_COMPLETED_2025-10-31.jsonl
📦 10 messages enregistrés → pipeline\TRANSACTIONS_FAILED\2025-10-31\TRANSACTIONS_FAILED_2025-10-31.jsonl
📦 16 messages enregistrés → pipeline\TRANSACTIONS_PENDING\2025-10-31\TRANSACTIONS_PENDING_2025-10-31.jsonl
📦 7 messages enregistrés → pipeline\TRANSACTIONS_PROCESSING\2025-10-31\TRANSACTIONS_PROCESSING_2025-10-31.jsonl
📦 5 messages enregistrés → pipeline\TRANSACTIONS_CANCELLED\2025-10-31\TRANSACTIONS_CANCELLED_2025-10-31.jsonl
📦 50 messages enregistrés → pipeline\TRANSACTIONS_SECURE\2025-10-31\TRANSACTIONS_SECURE_2025-10-31.jsonl
📦 50 messages enregistrés → pipeline\SPEND_LAST_FIVE_MIN_BY_TYPE\2025-10-31\SPEND_LAST_FIVE_MIN_BY_TYPE_2025-10-31.jsonl
📦 50 messages enregistrés → pipeline\SPEND_LAST_FIVE_MIN_BY_TYPE\2025-10-31\SPEND_LAST_FIVE_MIN_BY_TYPE_2025-10-31.jsonl
📦 50 messages enregistrés → pipeline\TRANSACTIONS_USD\2025-10-31\TRANSACTIONS_USD_2025-10-31.jsonl
📦 50 messages enregistrés → pipeline\SPEND_LAST_FIVE_MIN_BY_TYPE\2025-10-31\SPEND_LAST_FIVE_MIN_BY_TYPE_2025-10-31.jsonl
📦 50 messages enregistrés → pipeline\SPEND_LAST_FIVE_MIN_BY_TYPE\2025-10-31\SPEND_LAST_FIVE_MIN_BY_TYPE_2025-10-31.jsonl
📦 50 messages enregistrés → pipeline\SPEND_LAST_FIVE_MIN_BY_TYPE\2025-10-31\SPEND_LAST_FIVE_MIN_BY_TYPE_2025-10-31.jsonl
📦 50 messages enregistrés → pipeline\USER_SPENDING_BY_TYPE\2025-10-31\USER_SPENDING_BY_TYPE_2025-10-31.jsonl
```