KAFKA

Ouvrir un producteur Kafka

Dans un terminal : docker exec -it kafka bash

kafka-console-producer \
--broker-list localhost:9092 \
--topic test-topic

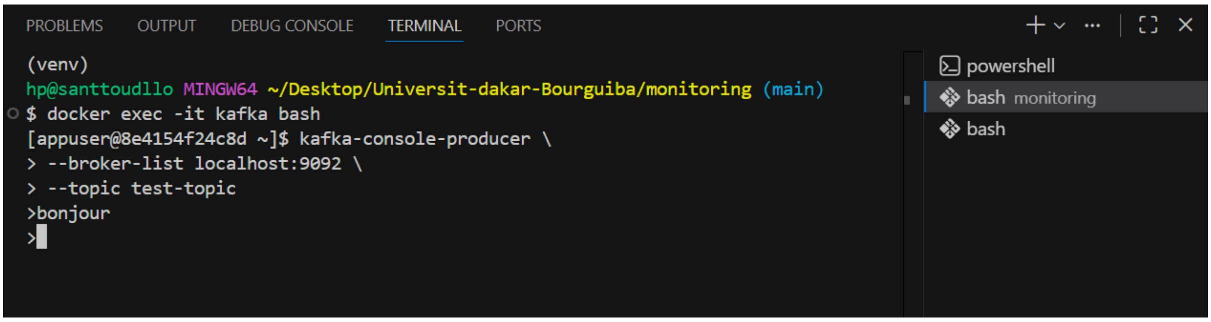
Ouvrir un consumer (lecture)

Dans **un autre terminal** :

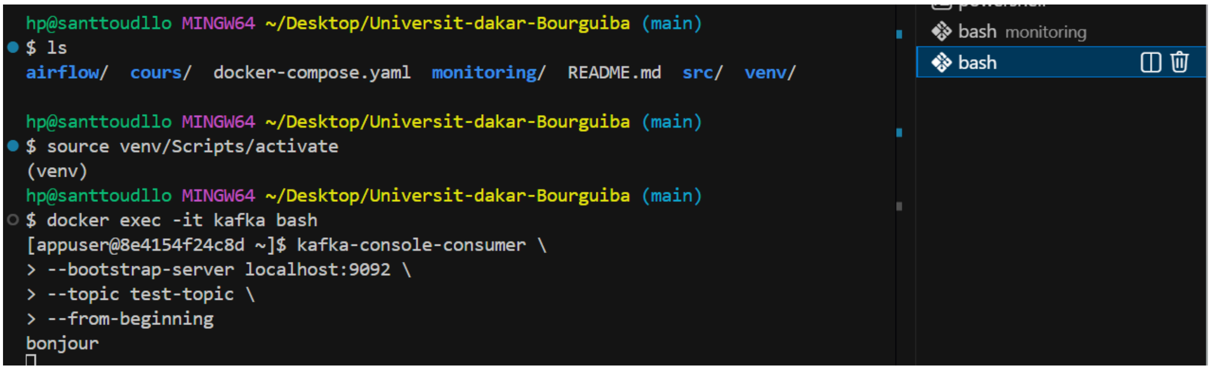
```
docker exec -it kafka bash
```

Puis :

```
kafka-console-consumer \
--bootstrap-server localhost:9092 \
--topic test-topic \
--from-beginning
```



```
(venv)
hp@santtoudllo MINGW64 ~/Desktop/Universit-dakar-Bourguiba/monitoring (main)
$ docker exec -it kafka bash
[appuser@8e4154f24c8d ~]$ kafka-console-producer \
> --broker-list localhost:9092 \
> --topic test-topic
>bonjour
>
```



```
hp@santtoudllo MINGW64 ~/Desktop/Universit-dakar-Bourguiba (main)
$ ls
airflow/  cours/  docker-compose.yaml  monitoring/  README.md  src/  venv/

hp@santtoudllo MINGW64 ~/Desktop/Universit-dakar-Bourguiba (main)
$ source venv/Scripts/activate
(venv)
hp@santtoudllo MINGW64 ~/Desktop/Universit-dakar-Bourguiba (main)
$ docker exec -it kafka bash
[appuser@8e4154f24c8d ~]$ kafka-console-consumer \
> --bootstrap-server localhost:9092 \
> --topic test-topic \
> --from-beginning
bonjour
```

SPARK

Dans le stack, Spark fonctionne en **mode cluster** :

- **Spark Master** → coordonne les calculs
- **Spark Worker** → exécute les tâches
- Les scripts envoient des jobs au cluster

http://localhost:8081

le conteneur Spark : `docker exec -it universit-dakar-bourguiba-spark-master-1 bash`

```
/opt/spark/bin/pyspark
```

exemple : `spark.range(10).show()`

pour faire la data preprocessing exemple avec spark :

Ajoutez un volume dans `docker-compose.yml`.

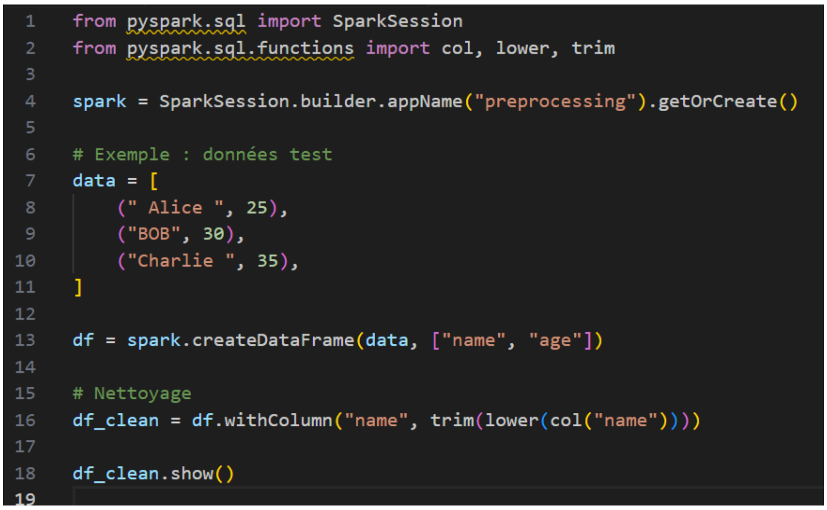
Spark master

```
spark-master:
  volumes:
    - ./spark_jobs:/opt/spark/jobs
```

Spark worker

```
spark-worker:
  volumes:
    - ./spark_jobs:/opt/spark/jobs
```

Exemple de script preprocessing Spark



```python
from pyspark.sql import SparkSession
from pyspark.sql.functions import col, lower, trim

spark = SparkSession.builder.appName("preprocessing").getOrCreate()

# Exemple : données test
data = [
    (" Alice ", 25),
    ("BOB", 30),
    ("Charlie ", 35),
]

df = spark.createDataFrame(data, ["name", "age"])

# Nettoyage
df_clean = df.withColumn("name", trim(lower(col("name"))))

df_clean.show()
```

```
docker exec -it universit-dakar-bourguiba-spark-master-1 \
/opt/spark/bin/spark-submit /opt/spark/jobs/preprocess.py
```