**REAL TIME PROCESSING APPLICATION**

BUILD COMMAND(DOCKERFILE)

docker build -t crypto-realtime-app .

RUNNING THE CONTAINER after build command

docker run -it -p 8501:8501 --name cryptorpa crypto-realtime-app

OPENING IT ON TEW TERMINAL

docker exec -it cryptorpa bash

ZOOKEEPER

$KAFKA\_HOME/bin/zookeeper-server-start.sh $KAFKA\_HOME/config/zookeeper.properties

KAFKA

$KAFKA\_HOME/bin/kafka-server-start.sh $KAFKA\_HOME/config/server.properties

---------------------------------------------------------------------------------------------------------------------------------------

TESTING THE PIPELINE (creating a test topic on a new terminal)

producer

/opt/kafka/bin/kafka-console-producer.sh --topic test-topic --bootstrap-server localhost:9092

consumer

/opt/kafka/bin/kafka-console-consumer.sh --topic test-topic --from-beginning --bootstrap-server localhost:9092

**wget https://repo1.maven.org/maven2/org/xerial/sqlite-jdbc/3.36.0.3/sqlite-jdbc-3.36.0.3.jar -P /app**

Download Kafka Connector JAR

**wget https://repo1.maven.org/maven2/org/apache/spark/spark-sql-kafka-0-10\_2.12/3.5.1/spark-sql-kafka-0-10\_2.12-3.5.1.jar -P /app**

download the Kafka client dependency:

**wget https://repo1.maven.org/maven2/org/apache/kafka/kafka-clients/3.6.1/kafka-clients-3.6.1.jar -P /app**

**# Download Kafka clients**

**wget https://repo1.maven.org/maven2/org/apache/kafka/kafka-clients/3.5.1/kafka-clients-3.5.1.jar**

**# Download Spark Kafka Token Provider**

**wget <https://repo1.maven.org/maven2/org/apache/spark/spark-token-provider-kafka-0-10_2.12/3.5.1/spark-token-provider-kafka-0-10_2.12-3.5.1.jar>**

**wget https://repo1.maven.org/maven2/org/apache/commons/commons-pool2/2.11.1/commons-pool2-2.11.1.jar**

**Python3 src/producer.py**

**Python3 src/consumer.py**

**spark-submit \**

**--jars /app/sqlite-jdbc-3.36.0.3.jar,/app/spark-sql-kafka-0-10\_2.12-3.5.1.jar,/app/kafka-clients-3.5.1.jar,/app/spark-token-provider-kafka-0-10\_2.12-3.5.1.jar,/app/commons-pool2-2.11.1.jar \**

**src/spark\_processor.py**

**python3 src/batch\_scheduler.py**

**Streamlit run src/dashboard.py**