Scenario event driven architecture – transactional

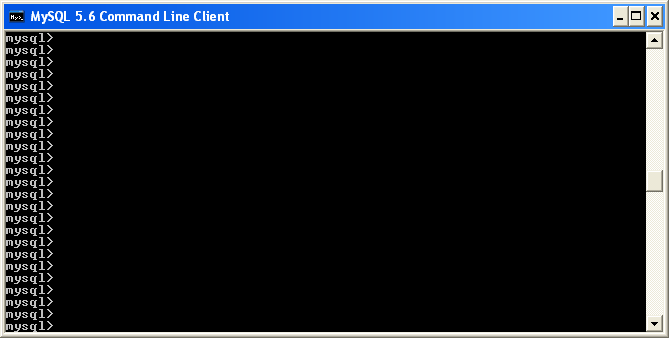
1. START MONGO DB
2. START ZOOK
3. START KAFKA
5. SHOW LISTENER ON THE QUEUE
6. PENDING
7. OK
8. KO
9. START APPLICATION 01

NO DATA IN TABLE

1. SHOW LISTENER ON THE QUEUE
2. PENDING
3. OK
4. KO

mysql> connect bookABattery\_db\_event localhost

1. MYSQL COMMAND LINE



1. LOAD FILE WITH DEMO DATA

mysql>

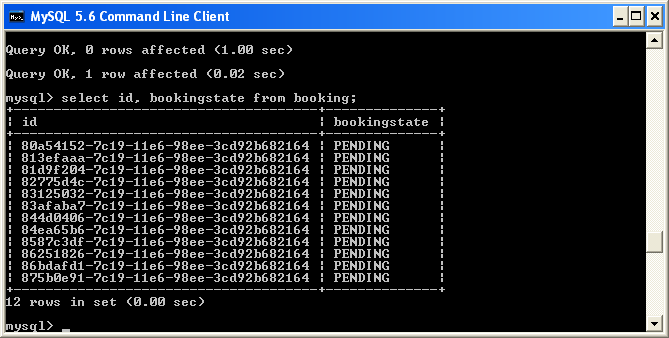
source C:/development/microS\_code2016\_serviceRegistryDiscovery/

01\_bookABattery\_SERVICE/demoLoadBooking.sql;

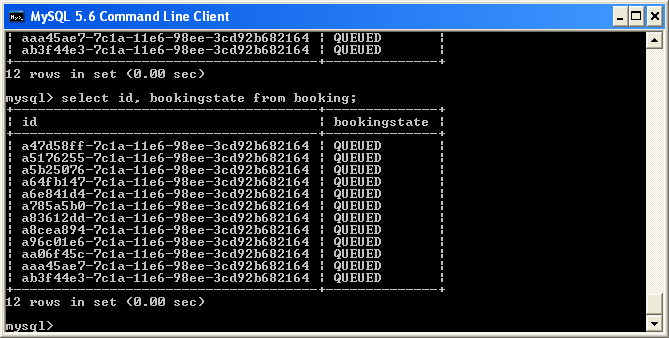
1. SHOW DATA LOADED

mysql>

select id, bookingstate from booking;



1. All in state PENDING
2. After Batch:



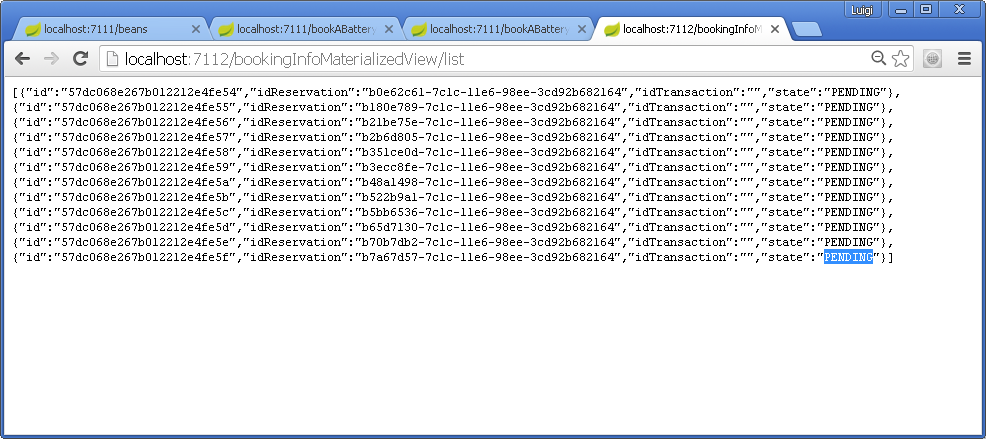
2. RELOLOAD FILE WITH DEMO DATA

mysql>

source C:/development/microS\_code2016\_serviceRegistryDiscovery/

01\_bookABattery\_SERVICE/demoLoadBooking.sql;

1. START MATERIALIZED VIEW
2. CALL MONGO DB REST



1. RELOLOAD FILE WITH DEMO DATA

mysql>

source C:/development/microS\_code2016\_serviceRegistryDiscovery/

01\_bookABattery\_SERVICE/demoLoadBooking.sql;

1. [TRUNCATE DATA MONGO DB]

<http://localhost:7112/bookingInfoMaterializedView/deleteForDemoOnly>

1. START BATTERY MANAGER
2. RELOLOAD FILE WITH DEMO DATA

mysql>

source C:/development/microS\_code2016\_serviceRegistryDiscovery/

01\_bookABattery\_SERVICE/demoLoadBooking.sql;

1. START PUBLISHING AND CONSUMING QUEUE

