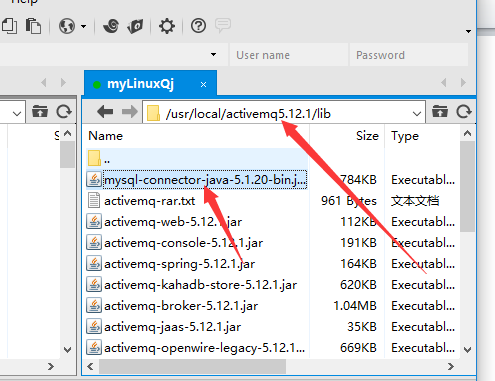


# 1、JDBC消息持久化

## 1、本实例是采用的mysql的方式，所以需要引入MySQL的驱动，将它放到activemq 的lib包下



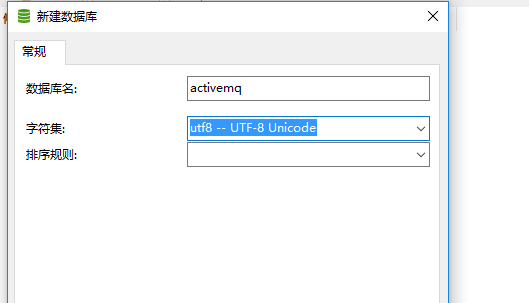
## 2、activemq 配置jdbc 数据存储的数据源

|  |
| --- |
| <persistenceAdapter>  <!-- <kahaDB directory="${activemq.data}/kahadb"/> -->  <jdbcPersistenceAdapter dataSource="#mysql-ds" />  </persistenceAdapter> |

## 3、</broker>结点之后，增加数据源的配置配置数据源，这里需要注意是dbcp2（和视频上不一样的）

|  |
| --- |
| <bean id="mysql-ds" class="org.apache.commons.dbcp2.BasicDataSource" destroy-method="close">  <property name="driverClassName" value="com.mysql.jdbc.Driver"/>  <property name="url" value="jdbc:mysql://localhost/activemq?useUnicode=true&amp;characterEncoding=UTF-8"/>  <property name="username" value="root"/>  <property name="password" value="147094"/>  </bean> |

## 4、创建数据库activemq

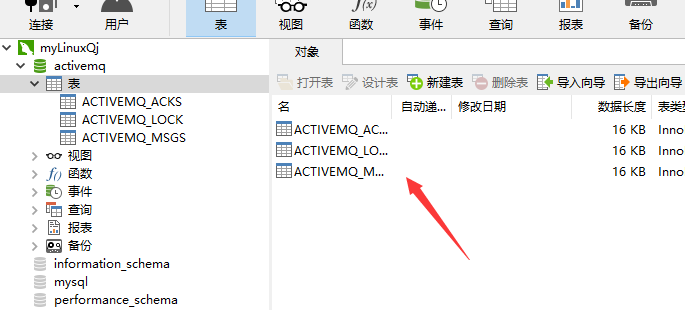


## 5、重新启动activemq ，数据库中会自动帮我们创建三张表

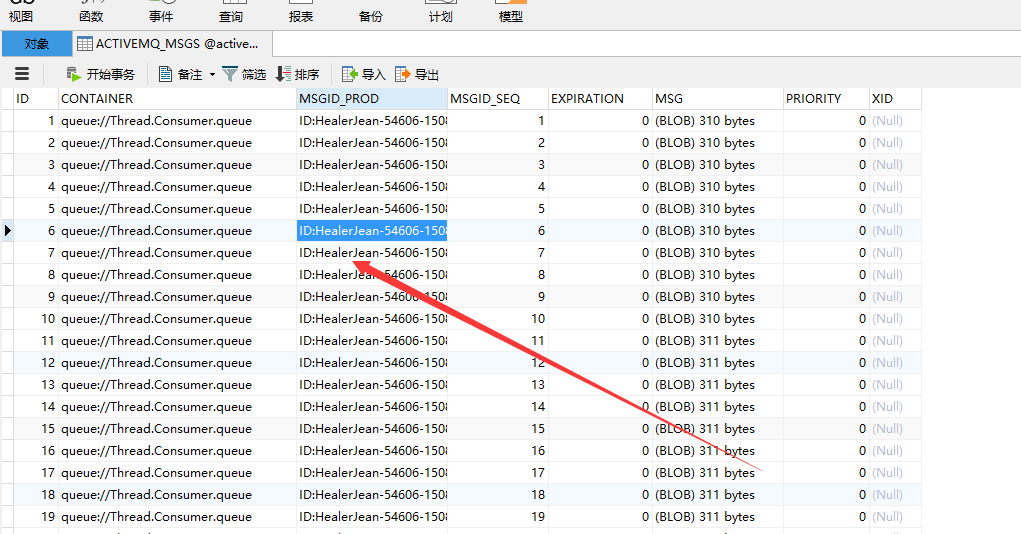
activemq\_acks：ActiveMQ的签收信息。

activemq\_lock:ActiveMQ的锁信息。

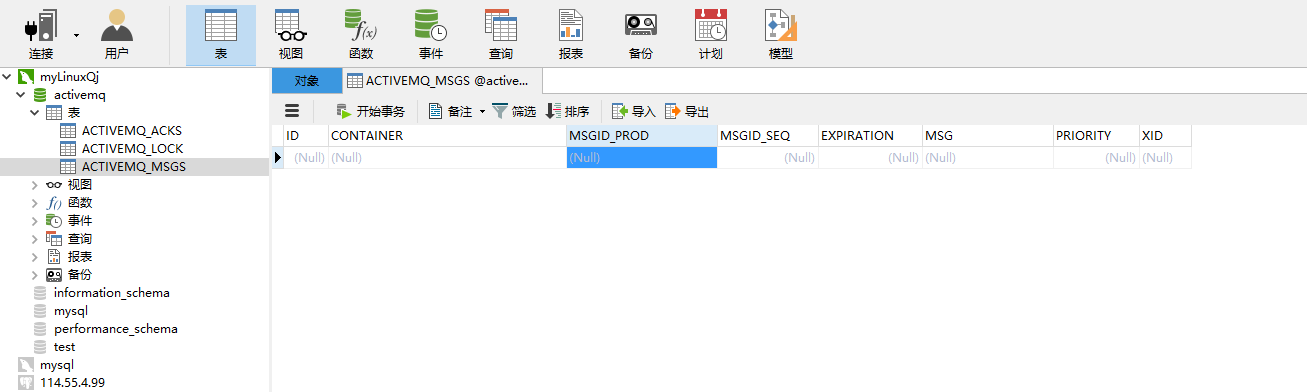
activemq\_msgs:ActiveMQ的消息的信息



## 6、运行一个生产者，队列queue



## 7、运行消费者，会发现上面的消失了



## 8、持久化的topic，运行之后会发现里面是没有数据的，所以需要先注册，也就是先运行一个topic的消费者。 然后才能有数据，但是这个里面的消息是不会消失的。因为这个是持久的

# 二、jdbc高速缓存，

### 比如，上面的持久化topic先放到缓存中，提供给java程序执行，然后再慢慢放入到数据库中

## 1、配置

|  |
| --- |
| <!-- <persistenceAdapter>  <kahaDB directory="${activemq.data}/kahadb"/>  <jdbcPersistenceAdapter dataSource="#mysql-ds" />  </persistenceAdapter>  -->  <persistenceFactory>  <!-- 指定数据文件的路径及数据源 -->  <journalPersistenceAdapterFactory  dataDirectory="activemq-data"  dataSource="#mysql-ds"/>  </persistenceFactory> |

