**Kafak接口文档**

**本接口是针对kafak开发的，主要可实现producer发送消息与consumer接收消息数据。**

**接口列表：  
kafka\_open  
kafka\_send  
kafka\_poll  
kafka\_commit  
kafka\_close**

**相关接口：  
kafka\_open**

**描述：**   
 连接kafka server服务器。

**语法：** kafka\_open(filename或fileObject;topic,...;partitionSize)

**备注：**

连接server服务器,将kafka属性参数properties存放到以后缀名为”xxx. properties”的配置文件中。支持单机模式与群集模式，它们的配置类似。群集连接服务器最好是leader所在的设备。(测试用flower设备，有时收或发消息时间比较长)  
**选项：**  @c 群集选项，支持对群集的操作

|  |  |
| --- | --- |
|  |  |

**参数：**topic 查询具体的某个topic或多个topic.  
filename 以后缀名为”. properties”的属性参数文件，其中涉及发送时的键、值编码与接收时的解码，且它们需要对应。  
fileObject file文件对象。  
partitionSize topic的分区数量，在群集下创建topic时才有效，不支持修改其大小。

**返回值：**   
hd连接对象。

**示例：**

|  |  |  |
| --- | --- | --- |
|  | **A** |  |
| 1 | =kafka\_subscribe("D: /kafka.properties"; "topic-test") | //用配置文件kafka.properties，topic为topic-test，连接kafka server |
| 2 | =kafka\_subscribe(file("D: /kafka.properties"); "topic-test";"String","byte[]") | //以file对象方式连接kafka server，其余同上 |
|  |  |  |
| 4 | =kafka\_close(A1) | 关闭连接 |

**D: /kafka.properties内容为：**

##produce

bootstrap.servers=192.168.0.76:9092

producer.type=sync

request.required.acks=1

serializer.class=kafka.serializer.DefaultEncoder

key.serializer=org.apache.kafka.common.serialization.StringSerializer

value.serializer=org.apache.kafka.common.serialization.StringSerializer

##consume

group.id=test

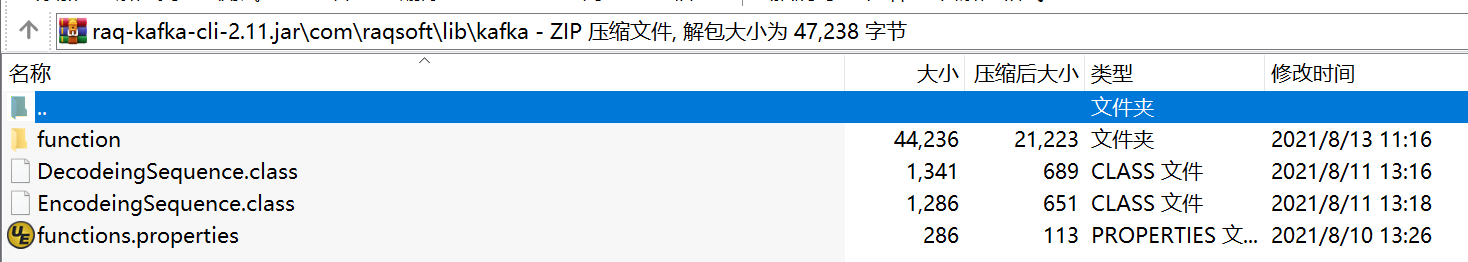
zookeeper.session.timeout.ms=1000

zookeeper.sync.time.ms=200

auto.commit.interval.ms=500

auto.offset.reset=earliest

key.deserializer=org.apache.kafka.common.serialization.StringDeserializer

value.deserializer=org.apache.kafka.common.serialization.StringDeserializer  
  
**自定义编码、解码使用说明：**1。将包含编码、解码的jar文件放到exlib\kafkacl目录下，如类似下图所示：  
  
编码文件：EncodeingSequence.class,解码文件：DecodeingSequence.class,   
对应的消息体对象为序列Sequnce。

在xxx.properties中配置自定义的解码：  
key.serializer=org.apache.kafka.common.serialization.StringSerializer   
value.serializer=com.raqsoft.lib.kafka.EncodeingSequence

key.deserializer=org.apache.kafka.common.serialization.StringDeserializer

value.deserializer=com.raqsoft.lib.kafka.DecodeingSequence

**kafka\_close()**

**描述：**   
 关闭kafka连接。

**语法：** kafka\_close(hd)

**备注：**

关闭kafka连接，释放资源  
**选项：**

**参数：**hd kafka连接对象。

**返回值：**   
无。

**示例：**

|  |  |  |
| --- | --- | --- |
|  | **A** |  |
| **1** | =kafka\_subscribe("D: /kafka.properties";"topic-test") | //连接server |
| **2** | =kafka\_close(A1) | //关闭连接 |

**kafka\_send()**

**描述：**   
 producer发送消息，根据服务器配置，支持单机模式或群集模式。

**语法：** kafka\_send(hd, [key,] value)

kafka\_send(hd, partition, key, value)

**备注：**

单机模式时，参数key可以省略。  
群集模式时，partition, key, value都需要填写。

选项：

**参数：**hd 连接对象key 消息的关键字。  
value 消息的值。  
partition 群集下的分区索引号，以0开始的整数。

**返回值：**   
无。

**示例：**

|  |  |  |
| --- | --- | --- |
|  | **A** |  |
| **1** | =kafka\_subscribe("D:/kafka.properties";"topic-test") | //单机模式连接server |
| **2** | =kafka\_send(A1, "A100"，"apple") | //发送消息 |
| **3** | =kafka\_close(A1) | //关闭连接 |
| **4** | =kafka\_subscribe@c("D:/kafka-cluster.properties";"topic-cluster") | //群集模式连接server |
| **5** | =kafka\_send(A1,1,"A100"，"apple") | //发送消息 |
| **6** | =kafka\_close(A4) | //关闭连接 |

**kafka\_poll()**

**描述：**   
 获取consumer消息记录, 返回序表。

**语法：** kafka\_poll(hd, timeout,[ partition1, partition2,…])

**备注：**

获取当前topic下的记录。群集模式下的topic只能是单个，无参数partition时，则遍历当前topic下所有的partition  
选项：  
 @c 游标查询

**参数：**hd 连接对象timeout poll循环间隔时间，缺省为1000ms  
partition 群集模式下topic的分区。可以是整数，可整数数组。如果群集模式下，无参数partition时，则遍历当前topic所有的下partition。

**返回值：**   
序表或游标。

**示例：**

|  |  |  |
| --- | --- | --- |
|  | **A** |  |
| **1** | =kafka\_subscribe("D: /kafka.properties";"topic-test") | //连接server |
| **2** | =kafka\_poll@c(A1) | //以游标方式获取数据 |
| **3** | =A2.skip(10) | //跳过100条记录 |
| **4** | =A2.fetch(100) | //获取100条记录 |
| **5** | =kafka\_close(A1) | //关闭 |

**kafka\_commit()**

**描述：**   
 手动提交当前consumer offset的位置。

**语法：** kafka\_commit(hd)

**备注：**

缺省时为同步提交当前consumer offset的位置。  
选项：  
 @a 异步提交

**参数：**hd consumer对象

**返回值：**   
无返回值。

**示例：**

|  |  |  |
| --- | --- | --- |
|  | **A** |  |
| **1** | =kafka\_subscribe("D: /kafka.properties";"topic-test") | //连接server |
| **2** | =kafka\_poll (A1) | //获取数据 |
| **3** | =kafka\_commit(A1) | //同步当前consumer offset |
| **4** | =kafka\_close(A1) | //关闭连接 |

**依赖包文件：**

**kafka\_2.11-1.1.0.jar  
kafka-clients-1.1.0.jar**

**slf4j-api-1.7.25.jar**

**slf4j-log4j12-1.7.25.jar**

**log4j-1.2.17.jar**