

kafka.common.ConsumerRebalanceFailedException

when Consumer Api run I have facing these exception...please anybody help me out

```
Exception in thread "main" kafka.common.ConsumerRebalanceFailedException: 0_VISION-PC-1373267495173-8bbf746b can't rebalance after 4 retries
at kafka.consumer.ZookeeperConsumerConnector$ZKRebalanceListener.syncedRebalance(ZookeeperConsumerConnector.scala:397)
at kafka.consumer.ZookeeperConsumerConnector.kafka$consumer$ZookeeperConsumerConnector$$reinitializeConsumer
(ZookeeperConsumerConnector.scala:681)
at kafka.consumer.ZookeeperConsumerConnector.consume(ZookeeperConsumerConnector.scala:205)
at kafka.javaapi.consumer.ZookeeperConsumerConnector.createMessageStreams(ZookeeperConsumerConnector.scala:77)
at kafka.javaapi.consumer.ZookeeperConsumerConnector.createMessageStreams(ZookeeperConsumerConnector.scala:89)
at Hello.Consumer.run(Consumer.java:40)
at Hello.Consumer.main(Consumer.java:84)
```

my source code for consumer class is....

```
package Hello;
import kafka.consumer.ConsumerConfig;
import kafka.consumer.ConsumerIterator;
import kafka.consumer.KafkaStream;
import kafka.javaapi.consumer.ConsumerConnector;
import kafka.message.Message;

import java.util.HashMap;
import java.util.List;
import java.util.Map;
import java.util.Properties;
import java.util.concurrent.ExecutorService;
import java.util.concurrent.Executors;

public class Consumer
{
    private final ConsumerConnector consumer;
    private final String topic;
    public Consumer(String a_zookeeper, String a_groupId, String a_topic)
    {
        Unknown macro: { consumer = kafka.consumer.Consumer.createJavaConsumerConnector( createConsumerConfig(a_zookeeper, a_groupId)); this.
        topic = a_topic; }

        public void run(int a_numThreads) {
            Map<String, Integer> topicCountMap = new HashMap<String, Integer>();
            topicCountMap.put(topic, new Integer(1));

            topicCountMap.put(topic, new Integer(1));
            /* consumerMap = consumer.createMessageStreams(topicCountMap);
            stream = consumerMap.get(topic).get(0);*/
            System.out.println("before map");
            Map<String, List<KafkaStream<byte[], byte[]>>> consumerMap = new HashMap<String, List<KafkaStream<byte[], byte[]>>>();

            consumerMap=consumer.createMessageStreams(topicCountMap);
            KafkaStream<byte[],byte[]> stream = consumerMap.get(topic).get(0);
            ConsumerIterator<byte[], byte[]> it = stream.iterator();
            while(it.hasNext())
            {
                Unknown macro: { System.out.println(new String(it.next().message())); }

            }
            consumer.shutdown();
        }

        private static ConsumerConfig createConsumerConfig(String a_zookeeper, String a_groupId)
        {
            Unknown macro: { Properties props = new Properties(); props.put("zookeeper.connect", "10.36.76.185}

        public static void main(String[] args) {
            String zooKeeper = "10.36.76.185:2181";
            String groupId = "0";
            String topic = "test";
            int threads = 1;//Integer.parseInt(args[3])

            Consumer consumer = new Consumer(zooKeeper, groupId, topic);
            consumer.run(threads);

            try
            {
                Unknown macro: { Thread.sleep(10000); }

            }
            catch (InterruptedException ie) {
```

