





```
//
while (true) {
    ConsumerRecords<String, String> records = consumer.poll(1000); //
    for (ConsumerRecord<String, String> record : records) {
        System.out.printf("Received: key=%s, value=%s, partition=%d, offset=%d%n",
            record.key(), record.value(), record.partition(), record.offset());
    }
}
}
```

poll() نص. إلى الـ البايتية الـ بيانات تحويل تعيد الـ مفسرات للـ تنسيق. الـ مستهلك مجموعة يحدد group.id هن: مرجع. بأي الـ التزام يتم لم إذا الـ أولى الـ رسالة من تبدأ أن يضمن auto.offset.reset و حلقة، في الـ رسائل يستخرج

### الكود تشغيل

1. إطلاق: bin/zookeeper-server-start.sh config/zookeeper.properties مثل و bin/kafka-server-start.sh config/server.properties مشابة أنظمة على
2. bin/kafka-topics.sh --create --topic test-topic --bootstrap-server localhost:9092 --partitions 1 --replication-factor 1. لـ إنشاء test-topic الـ أوامر سطر أداة استخدم: موضوع إنشاء
3. الـ رسائل. لـ إرسال SimpleProducer فئة تنفيذ: الـ منتج تنفيذ
4. استهلكها. يتم الـ رسائل لرؤية SimpleConsumer فئة تنفيذ: الـ مستهلك تنفيذ

### نصائح

- الإنشاجي. للـ كود تسجيلات أو كتلة أضف: الـ أخطاء معالجة
- الـ مستهلك مجموعة لحجم max.poll.records أو الـ منتج إنشاجية لضمان acks مثل خواص بتعديل قم: الـ إعدادات احتياجك. على بناء
- والاستخدام. الـ إعداد يسهل الذي، استخدم في فكر الـ عقدة، للـ تطبيقات: الـ مكثبات

الـ أخطاء، معالجة الـ بيانات، تقسيم مثل محددة ميزات في الـ غوص تريدي كنت إذا أخبرني الـ بدء. على للـ حلول أساسي إعداد هو هذا أكبر! تطبيقي مع الـ تكامل أو