注意事项？

不好理解的概念让chat通俗易懂地解释

举例子，比喻更容易理解概念

可以作为学习笔记，通过问题形式记录

无论大小，都可以记录

可以先把问题记录，写在其它笔记软件上

学任何东西都可以在这里对自己提问

要单独做笔记的可以单独记录

充当随记的功能，记录一些有价值的信息，里面包含了自己的一些思考等

都要以提问题的形式：是什么、怎么样、为什么、凭什么…

以问号结尾

锻炼自己问问题的能力，能够提出问题，有自己的想法，如果要单独做笔记可以单独去做，不影响这个

什么是消息队列？

消息： 跨进程传递的数据

队列：一种先进先出的数据结构

什么是中间件？

中间件是软件

提供系统软件和应用软件之间的连接

提供软件各部件之间连接

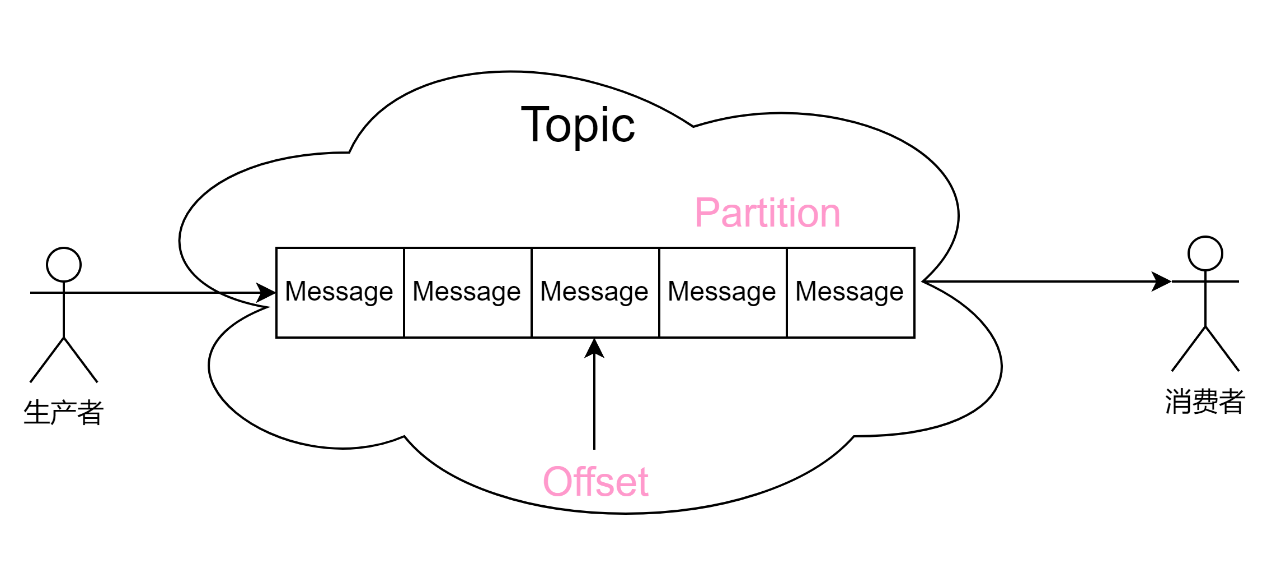
kafaka的partition是什么？

1 kafka 有 topic，生产者和消费者根据topic来确定往哪里发送消息，接受消息

2 topic下面有不同的partition，一个partition代表一个队列

3 offset 表示在partition里元素的偏移量

partition是一个数据结构



消费组用什么作用？

消费者A 属于 消费组1

消费者B 属于 消费组1

一个消息进来之后，如果消费者A消费了，消费者B就不会消费，隶属与同一个消费组的消费者同一个一个消息只会消费一次

Elasticsearch中的文档有什么特点?

1. 不同文档的字段数量和名称可以不一样。文档可以有不同结构
2. 不同文档中同名的字段，类型必须一样。相同字段必须是相同类型

AO/PO/OO分别代表什么？

面向切面

面向过程

面向对象

如何构建自己的笔记系统？

1. 改变认知：就是要开多个窗口，存放不同的信息。
2. 信息是有价值的，不同价值的信息放在不同窗口里面。

习惯开多窗口，在不同窗口间进行切换

不要怕窗口开的多，说明分类很齐全，各种笔记软件结合用，不同价值信息放在不同的窗口里面

不同软件平台，不同窗口，配合起来使用

低价值信息 – 放进流水账里

一些疑问，问题 – 放进多问为什么

单独做的笔记 – 放进CSDN

|  |  |  |  |
| --- | --- | --- | --- |
|  | 低价值信息 | 中等价值信息 | 高价值信息 |
| 放在哪里 | 流水账.md | 多问为什么.docx | CSDN |
| 内容 |  | 随记、疑惑、学习心得  以提问题的形式记录 | 单独做的笔记思考 |
| 是否同步 | 不用同步 | git同步 | 云同步 |

要对信息有一个分类，然后记录在不同的文件、软件、窗口里面，进行分流

可以不断改变这套打法，目前的打法是这样

核心：把遇到的信息进行分类归纳，放在不同的地方

关键：习惯开很多个窗口，进行信息流程，窗口多开一点，更好做各种笔记等

窗口开的越多越好，习惯多开窗口，复杂地做笔记

要习惯开很多窗口，在不同窗口进行切换，这样可以更好地进行信息流转，任务栏的窗口越多越好，窗口开了就不要随便关，就打开就是，把常用的软件固定到任务栏。

信息有价值

窗口越多越好

窗口打开了就不要关闭

任务栏软件铺满最好

要适应复杂，不要太洁净，不然不好做复杂的事情

操作习惯改变

面对复杂工程、笔记等才可以游刃有余

主动变得复杂

什么是消息？

是传输信息的载体，是生产和消费数据的最小单位

Topic、消息、生产者、消费者、生产组、消费组之间的关系？

1. 一个Topic可以有多条消息，一条消息只能属于一个Topic
2. 一个生产者可以生产多个Topic的消息
3. 一个消费者只能订阅和消费一个Topic的消息，一个Topic里的消息可以被多个不同消费组里的消费者消费

RocketMQ里的基本概念和Kafka里的基本概念总结？

RocketMQ：消息、Topic、标签、队列、分片、消息标识、消费组

Kafka： 消息、Topic、 、分区、 、 、消费组

Redis 是如何保证lua脚本原子性？

1 Redis 是单线程架构，在执行lua脚本的时候不会执行其它命令

2 Redis 在执行lua脚本里面的命令时，如果其中有一个失败，就会回滚到执行前的状态。