1. **阿里MQ支持几种协议：**

支持 HTTP 协议：支持 RESTful 风格 HTTP 协议完成收发消息,可以解决跨语言使用 MQ 问题。

* 支持 MQTT 协议：支持主动推送模型，多级 Topic 模型支持一次触达 1000万+ 终端，可广泛应用于物联网和社交即时通信场景。
* 支持 TCP 协议：区别于 HTTP 简单的接入方式，提供更为专业、可靠、稳定的 TCP 协议的 SDK 接入。

**5、生产者与Topic的对应关系：**

1:N，即一个 Topic 只能绑定一个 Producer ID, 但是同一个 Producer ID 可以对应多个 Topic

**6、消费者与Topic对应关系：**

N:N，同一个 Consumer ID 可以订阅多个 Topic，同一个 Topic 也可以对应多个 Consumer ID

**8、MQ支持发送哪几种消息：**

普通消息、事务消息、延时消息、定时消息、

**9、Message ID 、Message Key**

Message ID：消息的全局唯一标识，由 MQ 系统自动生成，唯一标识某条消息。

Message Key：消息的业务标识，由消息生产者（Producer）设置，唯一标识某个业务逻辑。

**10、Tag与Topic是什么关系，在消息中起什么作用：**

Topic是一级消息类型，Tag就是二级消息类型。tag是消息标签、消息类型，用来区分某个MQ的Topic下的消息分类，*不同的Topic之间的消息没有必然的联系，而Tag则用来区分同一个Topic下相互关联的消息，比如全集和子集的关系，流程先后的关系。*

注：

发送消息时，每条消息必须指明消息类型Tag

消费者如需订阅某Topic下所有类型的消息，Tag用 \* 符号表示

消费者如需订阅某Topic下某一种类型的消息，请明确标明Tag

消费者如需订阅某Topic下多种类型的消息，请在多个Tag之间用 || 分隔

同一个消费者多次订阅某Topic下的不同Tag，后者会覆盖前者

**11、MQ消息默认最多保留3天**

**12、重置消费置点是什么？**

以时间轴为坐标，在消息持久化存储的时间范围内（默认3天），重新设置消息订阅者对其订阅 Topic 的消费进度，设置完成后订阅者将接收设定时间点之后由消息发布者发送到 MQ 服务端的消息

**13、消费轨迹是什么？**

在一条消息从发布者发出到订阅者消费处理过程中，由各个相关节点的时间、地点等数据汇聚而成的完整链路信息。通过消息轨迹，用户能清晰定位消息从发布者发出，经由 MQ 服务端，投递给消息订阅者的完整链路，方便定位排查问题。

**14、消息过滤是什么？**

订阅者可以根据消息标签（Tag）对消息进行过滤，确保订阅者最终只接收被过滤后的消息类型。消息过滤在 MQ 服务端完成

**15、集群消费和广播消费**

**集群：**MQ 约定使用相同 Consumer ID 的订阅者属于同一个集群，同一个集群下的订阅者消费逻辑必须完全一致（包括 Tag 的使用），这些订阅者在逻辑上可以认为是一个消费节点。

**集群消费：**当使用集群消费模式时，MQ 认为任意一条消息只需要被集群内的任意一个消费者处理即可。

一个 Consumer ID 所标识的所有 Consumer 平均分摊消费消息。例如某个 Topic 有 9 条消息，一个 Consumer ID 有 3 个 Consumer 实例，那么在集群消费模式下每个实例平均分摊，只消费其中的 3 条消息。

**广播消费：**当使用广播消费模式时，MQ 会将每条消息推送给集群内所有注册过的客户端，保证消息至少被每台机器消费一次。

一个 Consumer ID 所标识的所有 Consumer 都会各自消费某条消息一次。例如某个 Topic 有 9 条消息，一个 Consumer ID 有 3 个 Consumer 实例，那么在广播消费模式下每个实例都会各自消费 9 条消息。

**16、延时消息/定时消息最多可设置多少天？当消费者存在消息堆积时，定时/延时消息是否能严格按照配置的时间进行消费？**

40天；消费者存在消息堆积时 无法严格按照配置时间消费

**17、重试消息对哪种消息方式有效？默认每条消息最多可重试多少次？每次的间隔时间是多少？如果自定义配置重试次数，那么大于默认重试次数的时间间隔是多少？**

重试消息对集群方式有效；

默认最多16次，分别是10s 30s 1m 2m 3m 4m 5 m 6m 7m 8m 9m 10m 20m 30m 1h 2h

自定义配置：大于默认的时间间隔为2h

**18、在事务消息中，半消息指的是什么？消息回查指的是什么？**

* 事务消息：MQ 提供类似 X/Open XA 的分布事务功能，通过 MQ 事务消息能达到分布式事务的最终一致。
* 半消息：暂不能投递的消息，发送方已经将消息成功发送到了 MQ 服务端，但是服务端未收到生产者对该消息的二次确认，此时该消息被标记成“暂不能投递”状态，处于该种状态下的消息即半消息。
* 消息回查：由于网络闪断、生产者应用重启等原因，导致某条事务消息的二次确认丢失，MQ 服务端通过扫描发现某条消息长期处于“半消息”时，需要主动向消息生产者询问该消息的最终状态（Commit 或是 Rollback），该过程即消息回查。

**19、事务消息发送完成本地事务后，可在 execute 方法中返回如下三种状态，分别是什么？**

* TransactionStatus.CommitTransaction 提交事务，允许订阅方消费该消息。
* TransactionStatus.RollbackTransaction 回滚事务，消息将被丢弃不允许消费。
* TransactionStatus.Unknow 暂时无法判断状态，期待固定时间以后 MQ Server 向发送方进行消息回查。

**20、默认单 Topic 发送消息上限为每秒 多少条，最高可申请扩展至多少 以上？**

上限5000条，可扩展至10w以上

**21、允许海量消息堆积，单个 Topic 可堆积100亿+条消息，系统高流量压力下依然可靠**

**22、同一网络内，消息传输网络时延在10毫秒之内，性能测试下，网卡可被打满。**

**23、默认单条消息大小最大支持多少？**

默认单条消息大小最大支持 256KB，华北2 地域支持 4MB 大消息。

**24、MQ 可以应用但不局限于以下业务场景：**

* 一对多，多对多异步解耦，基于发布订阅模型，对分布式应用进行异步解耦，增加应用的水平扩展能力。
* 削峰填谷，大促等流量洪流突然来袭时，MQ 可以缓冲突发流量，避免下游订阅系统因突发流量崩溃。
* 日志监控，作为重要日志的监控通信管道，将应用日志监控对系统性能影响降到最低。
* 消息推送，为社交应用和物联网应用提供点对点推送，一对多广播式推送的能力。
* 金融报文，发送金融报文，实现金融准实时的报文传输，可靠安全。
* 电信信令，将电信信令封装成消息，传递到各个控制终端，实现准实时控制和信息传递。

**25、消息队列管理控制台域名是什么?**

<https://ons.console.aliyun.com>