

# 风铃草同学面经

---

- 一、自我介绍
- 二、面试的公司
- 三、面试前准备
- 四、学习方式
- 五、面试问题归纳
- 六、项目经验部分
- 七、面试技巧
- 八、总结

## 一、自我介绍

### 儒猿-石杉架构课

大家好，我叫风铃草，目前在深圳工作，工作6年。在正式分享面经之前，首先感谢一下儒猿技术交流群这个平台给我这次分享的机会，也感谢为平台上做出努力付出的各位大佬。我自己是一个码农，比较喜欢科学类纪录片，喜欢打打羽毛球，偶尔会跟很多程序员一样打打王者但是很菜鸡，哈哈。

---

我大学的专业是软件工程专业，大学期间也喜欢搞搞专业的事情，当时为了能进入校专业社团熬夜打代码还有点让我难忘。毕业后参加了第一份工作，在数据组做开发，平时实际工作中用java搞crud及数据统计报表，另外还跟前端打交道的工作，什么css,js,数据渲染,表格展示什么的。那时候其实心里蛮排斥，因为我自己大学时候做一些小项目就不喜欢前端这方面的东西，自己审美不来事，前端效果也搞不出心里想要的效果，所以也就喜欢搞点“抽象的”东西。

---

后来想想，刚毕业人家可能也不会交给我很多重要的功能开发，一些简单的活和前端无关紧要的就让我来搞。由于自己刚出校门没想清楚自己职业规划，稀里糊涂的参加工作，但是这段工作经验也让我当时

对java开发尤其是java web方向的开发产生了一定误解。让我以为java开发不仅要写后台代码也要写各种前端的东西，因此我有点接受不了这点。

---

后来我就在网上了解到数据开发的方向，加上第一份工作的上司和技术总监在研究hadoop和spark等相关的技术来解决一些公司业务问题，我受到点影响，就决定转方向去做大数据开发，当时心里想，大数据方向应该就单纯一些，不要做那么多杂活，可现在想想，当初自己有点too young too native。

---

可自己转大数据开发后，其实没想象的那么顺心，一直也只是在P2P小公司做一线开发，做着每日几乎重复的工作，做了2、3年后自己慢慢也想着能有机会进入大厂锻炼。可是随着时间流逝，自己距离要去大厂的目标越来越远，心里也越来越慌，期间试了一两次腾讯，因为那时候资历太浅技术也不够，也是无缘大厂，最后18年有缘到了现在的公司平安集团下的一个子公司，在这里得到了很多锻炼，在平安里面的经历也是让我这次跳槽成功的原因之一。

儒猿-右杉架构课

<https://docs.qq.com/pdf/DY3dHR2RDTG9CUU91>

## 二、面试的公司

- 1、虾皮(不通过)
  - 2、腾讯电商(不通过)
  - 3、蘑菇物联 (2轮技术+一轮HR 通过)
  - 4、Aftership(2轮技术+1轮HR 通过)
  - 5、腾讯(5轮技术+1轮HR 已发offer)
  - 6、OPPO(2轮技术+1轮HR 已发offer)
- 

## 三、面试前准备

面试前准备很有必要，我当时面试虾皮和腾讯电商是毫无准备情况下，甚至刚从公司开完会后就视频面试，状态很差，一轮就挂了，吃了很大的亏，后来就在QQ上找老师指导一下，另外大数据岗位其实总

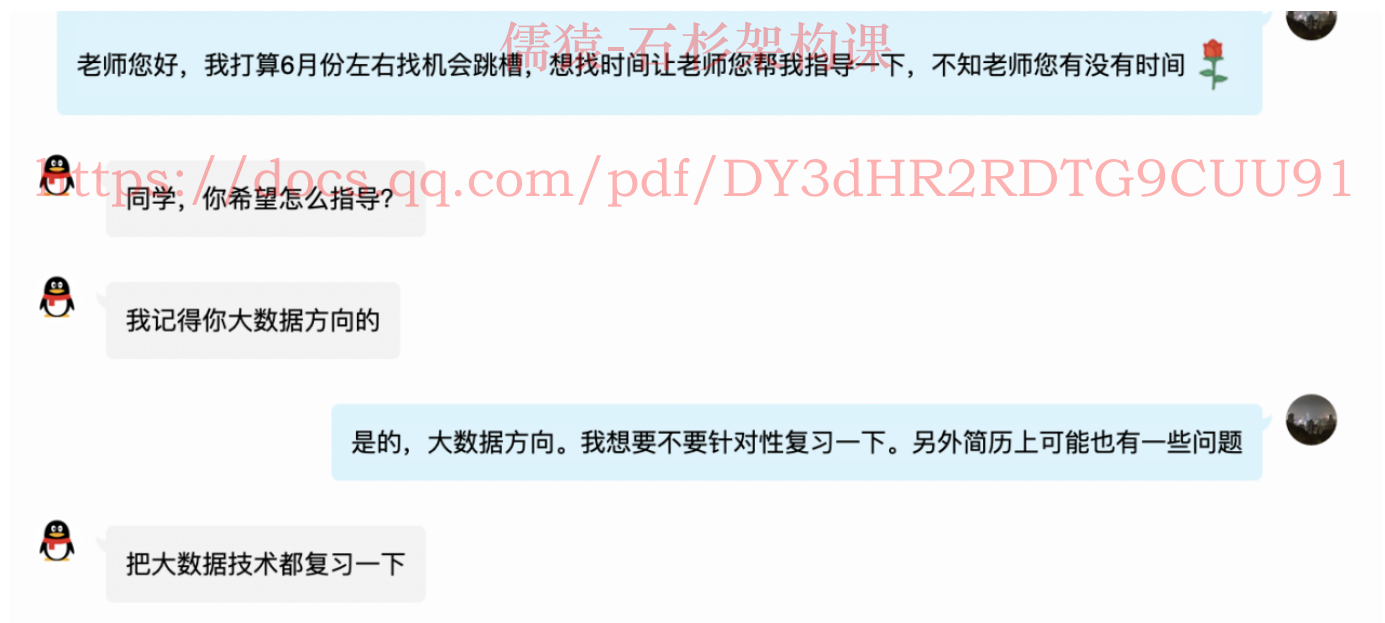
体偏少也不是那么好找，老师告诉我复习很关键，也要大胆出去试一试总结经验。其实我整体复习的时间不长，加起来可能一个月还不到，中间还有各种工作上的事情，总体复习还是比较仓促。但是复习后的面试情况还是要比没准备要好很多。

---

我总体的一个复习计划，具体技术分2个大方向，一个是大数据本身，另一个是Java相关的。大数据本身的技术，主要是根据之前学习的老师hdfs、kafka的自己做的一个笔记和源码结合进行复习。还有一块就是自己平时用到的技术组件（spark/hbase/hive/spark-streaming/sqoop/elastic/db/druid/flink等大数据生态常用技术）在项目中的梳理也很重要。

---

当时找老师请教指导沟通：





复习计划技术方面的一个list，根据自己的情况设置复习内容和完成节点。

用了一个todoist软件来跟踪自己的学习和复习情况。

待复习 4

ooo

○ kafka源码

📅 2月28日

○ hdfs源码

🔗 0/2 📅 3月31日 🗨 1 hdfs

○ spark复习

🔗 0/1 📅 3月31日 18:00 spark

○ java复习

🔗 0/7 📅 3月31日 学习 +1

学习

✕

○ java复习

📅 3月31日

学习

java

⋮

🔗

🚩

🔔

...

子任务 9

评论

活动

○ 集合复习

🔗 0/2 📅 3月17日 🗨 1 java

○ 并发

📅 3月20日 🗨 1

○ 并发集合复习

📅 3月24日 🗨 1

○ IO复习

📅 3月27日 🗨 1 学习 java

⋮ ○ 网络

📅 3月31日 🗨 1

○ 树结构和索引

📅 3月31日 🗨 1

○ JVM

📅 3月31日 🗨 1

儒猿-石杉架构课

<https://docs.qq.com/pdf/DPY3dHR2RDTG9CUU91>

#### 四、学习方式

另外听说大厂面试会问算法，所以我自己挤时间，在地铁上用leetcode刷一刷算法题。虽然我刷算法题很少，面试也不一定用得到原题，但稍微刷了几次，发现自己思路也会宽一些，对自己解题思路还是有一定启发和帮助。

另外顺带提一些我的学习方式方法：

5

1、学习时间：因为自己身体状态有点不是很好，加上工作经常加班搞得有点疲劳，所以没有一定固定每天学几个小时，但是基本上一有时间，都会去努力学习点东西，努力让自己慢慢有所进步。一般情况我是早上6点半或者7点起来，然后看老师的视频一个多小时，针对容易的会进行一个1.5倍语速播放观看，这个时候会尽量努力集中100%精力跟上老师的思路，早上效果还是不错的。晚上回家如果早的话，也会进行有大概1个小时到1个半小时去学习。正常来说，周末有大块的时间，我一般2天都是在家来持续的看课程，思考吸收，做笔记。

---

2、学习的主要途径：架构课+儒猿公众号专栏+技术网站/公众号+技术官网

3、架构班学习过程：总结起来就是课程+笔记+思考+复习。架构课里面会仔细去学习，遇到不懂的会看2遍或者更多遍，然后做好文字笔记和画图的笔记。文字笔记里面，自己会反复思考技术点和设计思路的考量，从多方面去考虑一些技术设计用处和带来的效果等。这方面的思考会自己在文字笔记里面做好记录。

儒猿-右杉架构课

<https://docs.qq.com/pdf/DY3dHR2RDTG9CUU91>

画图主要是流程和大的架构方面，帮助自己记忆和梳理思路。另外，还有一些难啃的内容，自己也会去反复查看官网和相关技术网站，深入源码仔细体会。另外架构班里面的源码层面，有很多深度确实太深了，需要自己反复去看去思考，比如hdfs中的源码，里面涉及到的细节和底层原理真的好多，虽然在学习老师的hdfs课程之前，自己已经对hdfs使用和原理有了一定程度的掌握。但是在老师课程里面，我还是懂得太少太浅。

---

4、学习的技术跟平时开发的工作结合，在学到的东西，自己大脑里面去思考，已有的技术方面可不可以用到一些新的方案和优化点，比如：学习kafka源码时候，对原理也有更好的理解，里面涉及到那么多性能调优的，我的spark-streaming流应用如何设置性能调优的参数，能让自己的程序性能更好，这些参数背后又是怎么影响我自己的程序性能的。

---

5、多思考，多动手练习实践。这个给我的亲身感受就是，学习转化为自己的东西，得通过实践积累自己的心得可以更好的吸收。比如：狸猫技术窝中的《从0开始带你成为JVM实践高手》，一开始学习没有自己动手去实践，有很多停留理论，但是就是容易忘记。通过动手敲代码，去各种折腾，能理解的更深入一些。

6、关注技术发展趋势和最新技术行业动态，大数据方向我常进的一个网站是infoq,另外也会有一些行业的公众号，也跟一些大佬请教交流，这方面是尽量让自己不要落伍太多，知道自己还有哪些方向需要去跟进和学习。

这里面也顺带分享一下在架构班学习的部分笔记内容(还会随着学习深入逐渐完善更新)。

儒猿-石杉架构课

<https://docs.qq.com/pdf/DY3dHR2RDTG9CUU91>

←

kafka

☰

📁 源码

2020-07-11

📁 架构学习笔记

📁 elastic

📁 flink

📁 hbase

📁 **hdfs**

📁 java

📁 kafka

📁 源码

🔗

kafka生产者-发送过程和内部数据结构

需要指定发送到哪个topic 首先需要选择一个topic的分区，默认是轮询来负载均衡，但是

2021-04-07 9.9 KB

🔗

零拷贝技术详解

以下内容转载自：  
<https://www.jianshu.com/p/028cf0008ca>

2021-01-04 584.5 KB

🔗

kafka 消费者group coordinator

每个consumer group 会选择一个broker作为自己的coordinator

2020-12-19 8.8 KB

🔗

Kafka顺序写数据与pageCache

1. 现代操作系统为了提升IO性能，在程序访问磁盘文件时候，增加了page cache提高

2020-12-19 3.9 KB

🔗

kafka leader和follower如何更新LEO

---

学习架构课的画图笔记

kafka部分笔记:

- 1、《kafka生产者元数据更新流程图》：<https://www.yuque.com/dulante-ikt7c/nwecmo/nzd9nk>
- 2、《kafka源码深度解读-生产者》：<https://www.yuque.com/dulante-ikt7c/nwecmo/ws9bb2>
- 3、《kafka服务端多线程处理模型》：<https://www.yuque.com/dulante-ikt7c/nwecmo/rokzx0>
- 4、《kafka源码解读-消费者coordinator选举连接过程》：<https://www.yuque.com/dulante-ikt7c/nwecmo/bfdh6b>

---

## 儒猿-石杉架构课

hdfs部分笔记:

- 1、《hdfs架构图》：<https://www.yuque.com/dulante-ikt7c/nwecmo/hhu9xi>
- 2、《hdfs NameNode启动流程图》：<https://www.yuque.com/dulante-ikt7c/nwecmo/fnl9cu>
- 3、《hdfs 联邦机制架构图》：<https://www.yuque.com/dulante-ikt7c/nwecmo/xqe7sb>
- 4、《DataNode下线block副本复制过程》：<https://www.yuque.com/dulante-ikt7c/nwecmo/qau2me>
- 5、《HDFS思维导图》：<https://www.yuque.com/dulante-ikt7c/nwecmo/exbx7k>

---

## 五、面试问题归纳

(由于自己面试过程没做非常详细的记录，事后梳理有很多记忆的东西也忘了蛮多，这里我就没有对面试过程中的问题给出自己当时一一的回答，实在很抱歉，问题归纳部分大家参考一下就好):

基础知识:

java相关: 常用数据结构、并发集合、多线程和并发包、IO相关



数据结构：红黑树、二叉树、B+树、堆等

算法相关：这个每个公司可能问得不一样，可以刷刷leetcode来拓宽自己思路

---

hbase相关：

架构原理

数据读写过程，如果落地保存

rowkey如何分布、物理模型、预写日志原理、memstore数据结构

如何优化（读写优化、大批量数据如何优化、表设计优化、设计原则）

Lsm-tree原理、底层数据存储的格式

使用优化方面

---

儒猿-石杉架构课

---

elastic相关：

<https://docs.qq.com/pdf/DY3dHR2RDTG9CUU91>  
数据读写详细流程、translog作用、索引作用

使用场景，索引如何分片

深度分页问题

优化方面经验

---

Spark相关

数据倾斜怎么解决？内存模型？shuffle详细过程，有何问题？

大小表join怎么解决？各个进程角色的作用？

提交任务详细过程？

rdd转换详细过程？

线上程序资源如何分配？优化经验？

如何保证spark-streaming精准一次性处理？大数据情况落地怎么考量和设计？

如何定位生产问题并解决？

---

Kafka:

架构原理和优势特点

生产消息详细过程

broker存储消息流程粘包拆包问题如何解决

消费者如何维护offset、rebalance等等。

---

Zookeeper:

架构原理

使用场景?

leader选举过程?

session如何存储?

zk内存数据结构如何存储?

儒猿-石杉架构课

<https://docs.qq.com/pdf/DY3dHR2RDTG9CUU91>

---

数仓及Hive:

hive基本原理概念, 优化经验, 现场写sql实现统计,与spark-sql区别。

数仓分层的原理、数仓如何建设? 数据质量和数据治理问题

实时数仓设计方案

OLAP相关:

方案和具体olap组件的技术原理(比如: druid或clickhouse), 优势和缺点, 优化手段? OLAP解决公司什么业务问题? 如何实施及效果?

---

## 六、项目经验部分

架构+业务流程+成效, 根据具体项目经验介绍, 要体现深度和难度的地方, 带来什么价值。

## 6.1 具体面试案例：

由于面试后有一些技术问题和细节也没能回忆起来，请各位见谅。

这里就其中腾讯面试过程，因为这次面试流程经历时间最长，记忆稍微深刻一些，给大家做一个介绍：

---

一面：

自我介绍，项目介绍，项目的架构和业务流程，总体问一些项目中涉及的大的技术方向，自己承担什么角色和职责，一些大数据框架的使用场景，spark用来干嘛，hbase如何使用等等，总体就是对项目的业务流程做了一个比较详细的了解。最后手动实现一道编程题：一个计算模块的程序，根据输入的数字进行计算(面试官说简单点就直接根据数字输入进行打印输出),在10个数字到来之前按照先到先计算(打印输出)的方式进行，到超过10个数字后按照整体排序最大的数字的进行依次计算(打印输出)，如果恢复到10个数字以下那么继续按照数字正常输入的顺序进行打印。程序可以支持多线程并发计算。到了交流阶段大概五分钟，问了一下部门做什么的，我心里想这一环节问一个问题差不多，留着后面轮的面试多问一两个问题。这一面比较简单，只要算法题过了，项目能把控住问题基本上不大。

---

二面：

自我介绍，项目经验介绍，这个环节面试官比较好奇我们报表项目的整体的技术方案和业务流程，在这个环节做了详细的沟通和交流，同时问了很多为什么，比如为何要存储分表，数据量大你们为何要这么处理，有没有什么碰到什么难点怎么优化等。最后也编程实现了一道算法题，具体有点不太记得了。总结：在项目和技术方案上，自己一定要提前心中有数，把控到位，而且要结合自己公司的业务场景来阐明。

---

三面：

自我介绍，项目经验简单介绍，这个环节面试官对我们OLAP方案的实施特别感兴趣，整个环节围绕druid的技术来沟通。

大的方向围绕这么几个：

1、为何要druid，采用druid的框架的原因是什么？为何不用别的OLAP框架比如：clickhouse？

2、druid架构原理，有哪些角色，为何性能超高，有哪些特点和优缺点；3：我们公司是如何使用druid，有没有做一些工具和上层的抽象？如何结合业务来实施？druid引入使用后的成效？

关于OLAP整个环节，基本上占据了90%的面试时间，虽然面试过程中问题很多，但是我心里一点不害怕。因为druid正是从我手头预研引入使用的。我从公司碰到的业务痛点问题，结合我们部门自己的实际情况，druid组件本身也研究和使用的都比较深入，尤其在业务理解和使用心得，遇到的困难都一清二楚都很有体会，所以整个环节很顺利。

---

四面（面试委员会）：

时间比较短，面试官上来没有开场白和自我介绍环节，直接就是问问题。问了一些简单的java问题，然后问我大数据组件中，哪个比较擅长，我当时脱口而出是kafka。因为在架构班里面把老师的kafka课程学习完了，心想应该有把握。没想到这个环节问的源码级别非常细致，但正好有一些细节没有复习到位，把自己坑了。整体问的包含2大块，一块是kafka原理方面，一块是kafka源码方面。其实这2大块内容如果认真跟着老师学习了kafka的课程，加上自己思考，应该都不难。

---

这里就不细致给大家说面试OK的内容，就给大家分享一下没有回答好的问题：

1、kafka元数据里面的数据结构，面试官要想知道的是Metadata对象中的Cluster的数据结构，我当时只回答了Metadata本身却没有回答上Cluster内部的结构，后悔自己当时源码没有看得非常仔细。

2、kafka怎么提高服务端IO方面的性能，单纯从磁盘方面来考虑的话。这一面让我感触很深，面试官的强悍实力和问问题的细节程度让我有些震撼到了，这一面让我感觉到有很强的心里压力。但是总体来说因为跟着老师kafka课程学习过来，大部分技术问题都抗下来了，让我感觉还算勉强过了这一面。

---

五面(经理)：

自我介绍，履历经历和技术栈情况了解。项目流程方面，团队配合分工，工作心得等。然后就是技术方面主要问了对数仓的理解，数仓怎么建设，大数据存储组件，大数据spark有哪些优化的经验，hbase怎

么使用。数据库mysql的索引数据结构，为何用B+树做索引？

---

单元测试如何做的，最后是一道场景题：如何设计一个K/V 存储系统，只要单机版的就行。这些问题常规的都还OK，总体感觉就是各方面都会问一下。但是在这里面也是吃了亏，主要是2个问题：单元测试如何做的？k/v存储系统如何设计？

---

因为自己平时写大数据的项目，比如：开发好的spark程序，都是直接用spark-shell自己在开发集群上去一步步测试调试看结果对不对，没问题后再部署到测试环境，不像java系统那样，写test case直接运行测试用例，这方面经验太少，我就根据自己大学学习java程序的一个记忆说用junit做单元测试，但是面试官不是要这么简单一句话，一步步问我如何实施和测试覆盖率等细节问题，这个问题紧逼让我紧张了，也让我在后面一个K/V存储系统设计问题上心态和思路受到一定影响。

第二个问题：K/V存储系统，当时回答的也是比较片面，没有从整个大的架构思路去回答，应该是先从架构层面框住大的方向，然后才从各个大模块依据面试过程中的情况进行细说。K/V存储系统的问题，在我面试结束后，立马就想起来应该如何设计。

---

腾讯这边面试给我的总体感受就是，面试流程很规范，会人性化与求职者沟通面试时间，整个五轮技术面试都是在视频上，晚上下班后回到家进行的。但同时流程也比较慢，每一轮都差不多要一周。另外面试过程中发现，对方是从整体上来考察一个人是否合格入选，有些地方没有答好自己总结好经验，不断去从面试中发现自己的不足。

---

## 七、面试技巧

1、面试前准备：因为我是视频面试，一般都是提前10到15分钟提前进入，然后调整好视频角度，声音大小，关好窗户保持安静的环境不影响接下来双方正常沟通，同时把自己开场白自我介绍和项目经历心里自己默念一遍。这个环节感觉还是挺有用，对自己内心会有一个自信心增强的感觉。

---

2、面试过程中：控制好自己的语速，时刻关注留意面试官的表情和反应。在开始的自我介绍环节，尽量不要太长，适当突出一下自己的技术特长和项目履历当中闪光的地方或者与众不同的地方。这方面介绍其实对后续接下来具体技术面试沟通，有可能会产生很大的影响。因为面试官有时候没有提前看简历，通过现场口述表达过程中听到擅长的技术点，很可能会给后面沟通埋下伏笔。

---

另外项目介绍环节，这方面自己可以多突出重点和自己擅长的地方，多结合自己的平时工作中自己的好的方面的心得和收获，有些时候会跟面试官产生共鸣。具体技术沟通方面，有时候可能会碰到自己不会的问题，这时候自己不能慌，要尽量想办法缓解一下尴尬的气氛扭转局面，这个时候要胆大心细，注意观察面试官表情和语气。

## 儒猿-石杉架构课

比如：问一个具体源码的实现或者底层数据结构，自己也确实不知道，但是自己心里有一个还挺有把握的实现思路或者方案，这时候你试探性的口吻问一下面试官，自己这方面确实没了解到，但是有个实现思路可以跟你说一下么？这种一般是对有耐心的面试官有点效果，但是也不一定百试百灵。如果有些面试官比较直接的那种，可以适当卖惨，这个也是没办法的事情，只能之后自己巩固复习了。

---

整个交流沟通过程中，碰到有机会转移到自己擅长的技术领域内来，要多找这样的机会，这样才能掌握主动。最后面试自己提问环节，这方面可以根据自己之前面试的情况的了解，适当提几个自己关心的问题，适当与面试官互动一下也可以。我自己一般也会提前设计好几个问题。在面试中发现，如果跟面试官互动或者互相探讨过程比较多，一般给对方比较深刻印象，可以适当根据情况来一些技术方面的互动和沟通，这对面试还挺有帮助。

---

3、面试巩固：一场面试下来，有得有失，要及时总结梳理，对于不满意的部分，可以自己着重记录及时查漏补缺，比如我一开始面试其实自我介绍时候，有些废话，没有突出自己的亮点和优势，我事后自己梳理，将自我介绍精简概述，也突出一些重点。另外还有一些HR可能会问的问题，比如：为何离职，

自己职业规划，对自己的定位啊，过往工作经历中的亮点成绩等，这些问题也要想清楚，有时候回答不好还挺影响hr这一关的体验分。

---

## 八、总结

- 1、面试前一定要准备和复习，针对自己的项目经历做好梳理和细节把控
  - 2、学习没有捷径，需要自己不断思考和总结。
  - 3、将爱好和工作结合起来是一件幸福的事情，比如我们喜欢搞技术的，将自己的技术应用起来并给公司带来价值，是很了不起的事情。有了爱好和价值体现，会让自己持续的努力下去，也会让自己更有动力学习。
- 

- 4、鼓励的话，但不是打鸡血。学习和成长有时候是孤独的，有时候客观原因会延缓自己进步的进程，但努力一点一滴学习进步，总有一天会有一个收成的时候。

- 5、真诚感谢的话，感谢石杉老师，让我在学习和成长道路上有了一个更有方向和深度的方式，也感谢儒猿和架构班里面所有同学大佬们，让学习不那么孤独，也有你们的帮助和指导，谢谢！

- 6、最后祝福，祝大家都能实现自己的理想，工作顺利，生活快乐，家庭美满。