

1. 一、如何通过HR面

1. 1. HR面的目的

2. 2. HR面的常见问题

1. （1）你对未来3-5年的职业规划
2. （2）如何看待加班(996)?
3. （3）面对大量超过自己承受能力且时间有限的工作时你会怎么办?
4. （4）你之前在上海为什么现在来北京发展?
5. （5）为什么从上一家公司离职?
6. （6）你还有其他公司的Offer吗?
7. （7）如何与HR谈薪资?

2. 二、回答问题的技巧

1. 1. 自我介绍

1. （1）为什么需要自我介绍
2. （2）自我介绍的几个必备要素
3. （3）自我介绍要重点匹配当前岗位的技术栈
4. （4）要在自我介绍中做刻意引导
5. （5）如何准备自我介绍

2. 2. 技术考察

1. （1）多重提问
2. （2）答题法则
3. （3）刻意引导

3. 3. 向面试官提问

3. 三、面试官到底想看什么样的简历?

1. 1. 准备简历模板

2. 2. 准备个人信息

3. 3. 准备专业技能

4. 4. 准备工作经历

5. 5. 准备项目经历

1. （1）切忌流水账写法
2. （2）切忌堆积项目
3. （3）切忌放虚假项目
4. （4）合格的项目经历如何写

6. 6. 其他

1. （1）教育背景
2. （2）注意事项

一、如何通过HR面

HR通常是程序员面试的最后一面,讲道理刷人的几率不大,但是依然有人倒在了这最后一关上,我们会从HR的角度出发来分析如何应对HR面.

1. HR面的目的

HR面往往是把控人才质量的最后一关,与前面的技术面不同,HR面往往侧重员工风险的评估与基本的员工素质。

- **录用风险评估****, **这部分是评估候选人是否具备稳定性,是否会带来额外的管理风险,是否能马上胜任工作,比如频繁的跳槽会带了稳定性的风险,HR会慎重考虑这一点,比如在面试中候选人体现出了「杠精」潜质,HR会担心候选人在工作中会难以与他人协作或者不服从管理,带来管理风险,再比如,虽然国家明确规定在招聘中不得有性别、年龄等歧视,但是一个大龄已婚妇女会有近期产子的可能性,可能会有长期的产假,HR也会做出评估。
- **员工素质评估****, **这部分评估候选人是否具备职场的基本素质,是否有基本的沟通能力,是否有团队精神和合作意识等等,比如一个表现极为内向的候选人,HR可能会对其沟通能力产生怀疑。

所以在与HR交流中要尽量保持踏实稳重、积极乐观的态度,切忌暴露出夸夸其谈、负能量、浮躁等性格缺陷。

2. HR面的常见问题

(1) 你对未来3-5年的职业规划

目的: 这个问题就是考察候选人对未来的规划能力,主要想通过候选人的规划来嗅出候选人对工作的态度、稳定性和对技术的追求.

分析: 一定要在你的回答中体现对技术的追求、对团队的贡献、对工作的态度,不要谈一些假大空的东西,或者薪资、职位这些太过于功利的东西,而且最好体现出你的稳定性,如果是校招生或者工作没几年的新人最好不要涉及创业这种话题,一方面职场新人计划没几年就创业,这种很不切实际,说明候选人没法按实际出发,另一方面说明候选人的稳定性不够.

建议分三部分谈:

1. 首先表示考虑过这个问题(有规划),如何谈一谈自己的现状(结合实际).
2. 接着从工作本身出发,谈谈自己会如何出色完成本职工作,如何对团队贡献、如何帮助带领团队其他成员创造更多的价值、如何帮助团队扩大影响力.

3. 最后从学习出发,谈谈自己会如何精进领域知识、如何通过提升自己专业能力,如何反哺团队.

至于想成为技术leader还是技术专家,就看自己的喜好了.

(2) 如何看待加班(996)?

目的: 考察候选人的抗压能力和责任心

分析: 这个问题几乎是必问的,虽然996ICU事件闹得沸沸扬扬,但是官方的态度很暧昧,只口头批评从没有实际行动,基本上是默许企业违反劳动法的,除了个别外企在国内基本不可能找到不加班的公司,所以在这个面试题中尽量体现出自己愿意牺牲自我时间来帮助团队和企业的意愿就行了,而且强调自己的责任心,如果真的是碰到无意义加班,好好学习怎么用vscode刷LeetCode划水是正道.

建议:

1. 把加班分为紧急加班和长期加班
2. 对于紧急加班,表示这是每个公司都会遇到的情况,自己愿意牺牲时间帮助公司和团队
3. 对于长期加班,如果是自己长期加班那么会磨练自己的技能,提高自己的效率,如果是团队长期加班,自己会帮助团队找到问题,利用自动化工具或者更高效的协作流程来提高整个团队的效率,帮助大家摆脱加班

当然了,就算你提高了团队效率,还是会被安排更多的任务,加班很多时候仅仅是目的,但是你不能说出来啊,尤其是一些候选人很强硬得表示长期加班不接受,其实可以回答的更委婉,除非你是真的对这个公司没兴趣,如果以进入这个公司为第一目的,还是做个高姿态比较好。

(3) 面对大量超过自己承受能力且时间有限的工作时你会怎么办?

目的: 考察候选人时间管理和处理大量任务的能力,当然也会涉及一定的沟通能力

分析: 程序员的工作内容可能大部分时间并不在写代码上,而是要处理各种会议、需求和沟通,通常都属于工作超负荷的状态,面对上面这种问题不建议以加班的方式来解决,因为主要考察的是你的时间管理能力和沟通能力,这些要素要在回答中体现出来

建议:

1. 将大量任务分解为紧急且重要、重要但不紧急、紧急但不重要、不重要且不紧急,依次完成上述任务,在这里体现出时间管理的能力
2. 与自己的领导沟通将不重要的任务放缓执行或者砍掉,或者派给组内的新人处理,在这里体现出沟通能力

(4) 你之前在上海为什么现在来北京发展?

目的: 考察候选人的稳定性和职业选择

分析: 这个问题一般是上份工作在地异地的情况下大概率出现,HR主要担心候选人异地换工作可能会不稳定,有短期内离职风险,这个时候不建议说"北京互联网公司多,机会多"这种话(合着觉得北京好跳槽?),回答最好要体现出自己的稳定性,比如"女朋友在北京,长期异地,准备来北京一起发展" "家在北京,回北京发展" 等等,潜台词就是以后会在北京发展,不会在多地之间来回摇摆.

(5) 为什么从上一家公司离职?

目的: 考察离职原因,候选人离职风险评估

分析: 这个问题经常会在跳槽的时候问到,这个时候切忌吐槽上一家公司或者自己的上一任老板,尽量从职业发展的角度来回答,凸显自己的稳定性和渴望学习上升的决心,至于一些敏感话题,比如加班太多、薪资太低这种问题也是可以谈的,毕竟你跳槽的诉求就是解决上家公司碰到的问题,但是不能触碰刚才提到的底线问题,切忌吐槽向.

建议:

1. 因为工资低、离家远、加班多、技术含量低等等原因离职
2. 因为离家远花费在路途上的时间过多,不如用来充电,因为加班多导致没有时间充电,无法提高等等

除了不要有负能量和吐槽向,这个部分可以坦诚得说出来

(6) 你还有其他公司的Offer吗?

目的: 评估候选人是否有短时间内入职其他公司的可能性

分析: 很多时候并不是候选人完美符合一个岗位的要求,HR当然想要一个技术更好、要钱更少、技术更匹配的候选人,但是候选人一般都会有这样或者那样的小问题。

比如,你的表现是可以胜任目前的岗位的,但是这个岗位不是很紧急,HR可能把你当做备胎,来找一个性价比更高的候选人。比如,你的表现很好,履历优秀,HR不知道能不能100%拿下你。所以如果你很希望加入这个公司,最好要做到「欲擒故纵」,既要体现自身的市场竞争力,又要给到HR一定的压力。所以,即使你已经拿了全北京互联网公司的offer了,也不要说自己offer多如牛毛,一副满不在乎的样子,这样会给HR造成他入职可能性不大的错觉,因为他的选择太多了。当然,也不要跪在地上舔:"加入公司是我的梦想,我只等这一个offer",放心吧,一定被hr放到备胎人才库中.

建议:

1. 表明自己有三四个已经确认过的offer了(没有offer也要吹,但是不要透露具体公司)
2. 但是第一意向还是本公司,如果薪资差距不大,会优先考虑本公司
3. 再透露出,有一两个offer催得比较急,希望这边快点出结果

(7) 如何与HR谈薪资?

HR与你谈论薪资经常有如下套路:

- HR: 您期望的薪资是多少?
- 你: 25K。

OK, 你已经被HR成功套路。这个时候你的最高价就是25K了, 然后HR会顺着这个价往下砍, 所以你最终的薪资一般都会低于25K。等你接到offer, 你的心里肯定充满了各种“悔恨”: 其实当时报价26、27甚至28、29也是可以的。

正确的回答可以这样, 并且还能够反套路一下HR:

- HR: 您期望的薪资是多少?
- 你: 就我的面试表现, 贵公司最高可以给多少薪水?

如果经验不够老道的HR可能就会说出一个报价(如25K)来, 然后, 你就可以很开心地顺着这个价慢慢地往上谈了。所以这种情况下, 你最终的薪资肯定是大于25K的。当然, 经验老道的HR会给你一句很官方的套话:

- HR: 您期望的薪资是多少?
- 你: 就我的面试表现, 贵公司最高可以给多少薪水?
- HR: 这个暂且没法确定, 要结合您几轮面试结果和用人部门的意见来综合评定。

虽然薪资很重要, 但是我个人觉得这不是最重要的。我有以下建议:

- 如果你觉得你技术面试效果很好, 可以报一个高一点的薪资, 这样如果HR想要你, 会找你商量的。
- 如果你觉得技术面试效果一般, 但是你比较想进这家公司, 可以报一个折中的薪资。
- 如果你觉得面试效果很好, 但是你不想要进这家公司, 你可以适当“漫天要价”一下。
- 如果你觉得面试效果不好, 但是你想进这家公司, 你可以开一个稍微低一点的工资。

需要注意的是, 面试求职是一个双向选择的过程。面试应该做到不卑不亢, 千万不要因为面试结果不好, 就低声下气地乞求工作, 每个人的工作经历和经验都是不一样的, 技术面试不好, 知道自己的短板针对性地补缺补差就行, 而不是在人事关系上动歪脑筋。

二、回答问题的技巧

技术面试通常至少三轮:

1. 基础面试: 主要考察对岗位和简历中涉及到基础知识部分的提问,包括一部分算法和场景设计的面试题,这一面可能会涉及现场coding.
2. 项目面试: 主要考察简历中涉及的项目,会涉及你项目的相关业务知识、扮演角色、技术取舍、技术攻坚等等.
3. HR面试: 这一面通常是HR把关,主要涉及行为面试,考察候选人是否价值观符合公司要求、工作稳定性如何、沟通协作能力如何等等.

当然,对于初级岗或者校招生会涉及一轮笔试,相当多的公司会在现场面之前进行一轮电话面试,目的是最快速有效地把不符合要求的候选人筛除,对于个别需要跨部门协作的岗位会涉及交叉面试,比如前端候选人会被后端的面试官面试,一些有管理需求的岗位或者重要岗位可能会涉及总监面试或者vp面。

而一个正常的技术面试流程(以项目面为例)分为大致三个部分:

1. 自我介绍
2. 项目(技术)考察
3. 向面试官提问

那么该如何准备技术面试,如何在面试中掌握主动权呢?

1. 自我介绍

几乎所有的面试都是从自我介绍这个环节开始的,所以我们得搞清楚为什么自我介绍通常作为一个面试的开头。

(1) 为什么需要自我介绍

首先,有一个很普遍的问题就是面试官很可能才刚拿到你的简历,他需要在你自我介绍的时候快速浏览你的简历,因为技术面试的面试官很多是一线的员工,面试候选人只是其工作中的一小部分,很多情况下是没有提前看过你的简历的。

其次,自我介绍其实是一个热身,面试官和候选人其实是陌生人,自我介绍不管是面试还是其他情况下,都是两个陌生人彼此交流的起点,也是缓解候选人与面试官之间尴尬的一种热身方式.

最后,自我介绍是展示自我、引出接下来技术面试的引子,是你自己指定技术面试方向的一次机会。

知道了以上原因，我们才能进行准备更好的自我介绍。

（2）自我介绍的几个必备要素

自我介绍归根到底是一个热身运动,因此切忌占用大量的篇幅,上来就把自己从出生的经历到大学像流水账一样吐出来的，往往会被没耐心的面试官打断,而这也暴露了候选人讲话缺乏重点、沟通能力一般的缺点。

但是，一些关键信息是必须体现的，就我个人而言，以下信息是必备的：

- 个人信息: 至少要体现出自己的姓名、岗位和工作年限，应届生则必须要介绍自己的教育背景，如果自己的前东家是个大厂（比如BAT）最好提及，自己的学历是亮点（985或者硕博或者类似于北邮这种CS强校）最好提及，其他的什么有没有女朋友、是不是独生子没人在意，不要占用篇幅。这个部分重点在于「你是谁？」。
- 技术能力: 简要地介绍自己的技术栈,切忌把自己只是简单使用过，写过几个Demo或者看了看文档的所谓「技术栈」也说出来，一旦后面问到算是自找尴尬。这个部分的重点在于「你会什么？」。
- 技能擅长: 重点介绍自己擅长的技术,比如性能优化、高并发、系统架构设计或者是沟通协调能力等等，切忌夸大其词，要实事求是，这是之后考察的重点。这个部分重点自在于「你擅长什么？」。

（3）自我介绍要重点匹配当前岗位的技术栈

你的面试简历可能包含了各种各样的技术栈,但是在自我介绍过程中需要匹配当前岗位的技术要求。就比如你目前面试的是移动端H5前端的开发岗位,就重点在自我介绍中突出自己在移动前端的经验,而此时大篇幅得讲述 自己如何用Node支撑公司的web项目就显得很不明智。

（4）要在自我介绍中做刻意引导

如果你的自我介绍跟流水账一样,没有任何重点,其实面试官也很难办，因为他都没法往下接话...

而只要你稍作引导,绝大部分面试官就会接你的话茬,比如「你在自我介绍中重点提及了一个项目，碰到了一些难点，然后被你攻克了，效果如何如何好等等」，如果我是面试官一定会问「你的xx项目的xx难点后来是怎么解决的？」。

面试官的目的是考察候选人的能力，对候选人做出评估，因此需要知道候选人擅长什么，是否匹配岗位，面试官绝大多数情况下很乐意你这种有意无意的引导，这样双方的沟通和评估会很顺利，而不是故意刁难候选人。

（5）如何准备自我介绍

其实最好的方法也是最笨的方法就是把自我介绍写下来，这个自我介绍一定要体现上面提到的几大必备要素，在面试前简单过几遍，能把自我介绍的内容顺利地表达出来即可，切忌跟背课文一样。

自我介绍的时间最好控制在**1-3分钟**之间，这些时间足够面试官把你的简历过一遍了，面试官看完简历后正好接着你的自我介绍进行提问是最舒服的节奏，别上来开始**10分钟**的演讲，面试官等待的时候会很尴尬，这么长的篇幅说明你的自我介绍一定是流水账式的。

2. 技术考察

一个好的技术考察的开始,必须得有自我介绍部分好的铺垫和引导,有一种情况我们经常遇见:

候选人说了一大堆非重点的自我介绍,面试官一时语塞,完全get不到候选人的重点,也不知道候选人擅长什么、有什么亮点项目,然后就在他简历的技术栈中选了本公司也在用的技术,候选人这个时候也开始冒汗,因为这个技术栈并不是他的擅长,回答的也磕磕绊绊,面试官的引导和深入追问也没有达到很好的效果,面试就在这种尴尬的气氛中展开了,面试结束后面试官对候选人的评价是技术不熟练、没有深入理解原理,候选人的感受是,面试官专挑自己不会的问。

所以在前面的部分,一定要做好引导,把面试官的问题引到我们擅长的领域,但是这样还不够,正所谓不打无准备之仗,我们依然需要针对可能出现的问题进行准备。

那么如何准备可能的面试题?

比如你擅长前端的性能优化,在自我介绍的部分已经做好了引导,接下来面试官一定会重点考察你性能优化的能力,很可能会涉及很有深度的问题,即使你擅长这方面的技术,但是如果没有准备也可能临场乱了阵脚。

(1) 多重提问

自我多重提问的意思是,当一个技术问题抛出的时候,你可能面对更深层次的追问。

依旧以前端性能优化为例,面试官可能的提问:

1. 你把这个手机端的白屏时间减少了**150%**以上,是从哪些方面入手优化的?这个问题即使你没做过前端性能优化也能回答个七七八八,无非是组件分割、缓存、**tree shaking**等等,这是第一重比较浅的问题。
2. 我看你用webpack中SplitChunksPlugin这个插件进行分chunk的,你分chunk的取舍是什么?哪些库分在同一个chunk,哪些应该分开你是如何考虑的?如果你提到了

SplitChunksPlugin插件可能会有类似的追问,如果没有实际操作过的候选人这个时候就难以招架了,这个过程一定是需要一定的试错和取舍的。

3. 在分**chunk**的过程中有没有遇到什么坑?怎么解决的?其实**SplitChunksPlugin**这个插件有一个暗坑,那就是**chunkid**自增性导致**id**不固定唯一,很可能一个新依赖就导致**id**全部打乱,使得**http**缓存失效。

以上只是针对**SplitChunksPlugin**插件相关的优化提问,当然也可能从你的性能测试角度、代码层面进行考察,但是思路是类似的。因此不能把自己准备的问题答案停留在一个很浅显的层面,一方面无法展示自己的技术深度,另一方面在面试官的深度体情况下容易丢分,因此在自己的答案后面多进行自我的追问,看一看能不能把问题做的更深入。

(2) 答题法则

很多面试相关的宝典都推荐使用**STAR**法则进行问题的应答,我们不想引入这个额外的概念,基础技术面试的部分老老实实回答面试官的问题即可,通常需要问题运用到这个法则的是项目面,比如让你介绍一下你最得意的项目,回答问题的法则有这几个要点:

- 项目背景: 简要说一下项目的背景,让面试官知道这个项目是做什么的
- 个人角色: 让面试官知道你在这个项目中扮演的角色
- 难点: 让面试官知道你在项目开发过程中碰到的难点
- 解决方案: 针对上面的难点你有哪一些解决方案,是如何结合业务进行取舍的
- 总结沉淀: 在攻克上述的难点后有没有沉淀出一套通用的解决方案,有没有将自己的方案在大部门进行推广等等

重点就在于后面三条,也是最体现你个人综合素质的一部分,我是面试官的话会非常欣赏那种可以发现问题、找到多种方案、能对多种方案进行比对取舍还可以总结沉淀出通用解决方案回馈团队的人。从上述几点可以体现出一个人的技术热情、解决问题的能力 and 总结提高的能力。

(3) 刻意引导

是的,在回答面试官提问的时候也可以做到刻意引导。

举几个简单的例子:

- 除了**Vue**还用**Angular**吗? 这个时候很多候选人就很诚实回答「没有」,其实我们可以回答的更好,把你知道的说出来展示自己的能力才是最重要的,你可以说「我虽然没用过,但是在学习双向绑定原理的时候了解了一下 **Angular**脏检查的原理,在学习**Nestjs**的时候了解了依赖注入的原理,跟**Angular**也是类似的」,面试官一定会接着问你脏检查和依赖注入的问题,虽然你没有用过**Angular**,但是**Angular**的基本原理你都懂,这是很好的加分项,说明候选人有深入理解原理的意愿和触类旁通的能力

- **Vue**如何实现双向绑定的？很多候选人老老实实答了 `object.defineProperty` 如何如何操作,然后就没有了,其实你可以在回答完之后加上一嘴「**Vue 3.0**则选择了更好用的**Proxy**来替代`object.defineProperty`」或者「除了`object.defineProperty`这种数据劫持的方式,观察者模式和脏检查都可以实现双向绑定」,面试官大概率会问「**Proxy**好在哪？」或者「聊聊脏检查」等等,这样下一个问题就会依然在你的可控范围内

我们第一个例子把本来回答不上来的问题,转化为了成功展示自己能力的加分项,第二个例子让自己更多的展示了自己的能力,而且始终使面试官的问题在自己的可控范围内。

3. 向面试官提问

这个部分基本到了面试尾声了,属于做好了不影响大局,但是可能加分,如果做不好很容易踩雷的区域.

首先我们声明几个雷区:

- 切忌问结果: 问了也白问,绝大部分公司规定不会透露结果的,你这样让大家很尴尬
- 切忌问工资: 除了**HR**跟你谈工资的时候,千万别跟技术面试官谈工资,工资是所有公司的高压线,没法谈论
- 切忌问技术问题: 别拿自己不会的技术难题反问面试官,完全没意义,面试官答也不是不答也不是

有几个比较好的提问可供参考:

- 如果我入职这个岗位的话,前三个月你希望我能做到些什么?
- 我的这个岗位的前任是为什么离职的,我什么地方能做的更好?
- 你对这个职位理想人选的要求是什么?

尽量围绕你的岗位进行提问,这可以使得你更快得熟悉你的工作内容,也让面试官看到你对这个岗位的兴趣和热情,重要的是 这些问题对于面试官而言既可以简略回答,也可以详细的给你讲解,如果他很热情得跟你介绍此岗位相关的情况,说明你可能表现得不错,否则的话,你可能不在他的备选名单里,这个时候就需要你早做打算了。

三、面试官到底想看什么样的简历？

面试一直是程序员跳槽时期非常热门的话题,虽然现在已经过了跳槽的旺季,下一轮跳槽季需要到年底才会出现,但是当跳槽季的时候你再看这篇文章可能已经晚了,过冬的粮食永远不是冬天准备的,而是秋收的时候。

简历是你进入面试的敲门砖，也是留给意向公司的第一印象，所以这个很重要，必须在这上面做足了文章，一份优秀的面试简历是整个面试成败的重中之重，我们会详细分析如何准备简历才能保证简历不被刷掉。

简历通常有这几部分构成：

1. 基本资料
2. 专业技能
3. 工作经历
4. 项目经历
5. 教育背景

我们会逐一进行分析。

1. 准备简历模板

万事开头难，简历的编写如果从头开始需要浪费很多时间，其实最快速也最聪明的办法就是先找一份还不错的简历模板，之后我们只需要填写信息即可。

简历模板的选择很讲究，有些简历基本不看内容就会被刷掉，这些简历一般会对面试官进行视觉攻击，让简历给面试官的第一印象就是反感。

有两种坑爹的简历模板：

- 一种是经典简历模板，真是堪称『经典』，这种简历模板在我上小学的时候就有了，以现在的眼光看有点不够看了，配色也比较『魔幻』，加上表格类的简历属于low到底端的简历类型，基本上扫一眼就扔了，这种简历只需要3秒钟就能被面试官扔到垃圾堆
- 另一种是设计感十足的简历模板，这种简历设计感十足，这五颜六色的配色一定能亮瞎面试官的双眼，这种花里胡哨的简历同样也是3秒钟沉到垃圾堆底部的简历。

以上两类简历模板堪称面试官杀手，我相信只要你用了上述两类模板，绝对连让面试官看第二眼的兴趣都没有。面试官筛简历要的是高效、清晰、内容突出，不管是HR还是技术面试官都想在最快速的情况下看到有效信息，你眼中所谓的『视觉效果』在别人眼里就是『视觉噪音』或者『视觉垃圾』，严重影响看简历的心情和寻找有效信息的速度。

2. 准备个人信息

个人信息部分主要包括姓名、电话、电子邮箱、求职意向，当然这四个是必填的，其它的都是选填，填好了是加分项，否则很可能减分。

接下来才是重点：

1. **github**：如果准备一个基本没有更新的博客或者没有任何贡献的**github**，那么给面试官一种为了放上去而放上去的感觉，这基本上就是在跟面试官说『这个候选人平时根本没有总结提炼的习惯』，所以如果有长期维护的**github**或者博客一定要放上去，质量好的话会非常有用，如果没有千万别放。
2. 学历：如果你的学历是专科、高中毕业之类的，还写在简历头部强调一遍，这就造成了你是『学渣』的印象，没有公司喜欢学渣的，这又增加了简历被刷的几率，如果是研究生以上学历可以写，突出一下学历优势，本科学历在技术面试领域基本上敲门砖级别的，没必要写。
3. 年龄：如果你是大龄程序员，尤其是你还在求一份低端岗位的时候千万别写，一个大龄程序员在求职一个中低端岗位，说明这些年基本原地踏步，还不能加班，到这里基本上此简历就凉了一半了。
4. 照片：形象优秀的可以贴，尤其是形象优秀的女程序媛，其它的最好不要贴，如果要贴的话，最好是贴那种**PS**过的非常职业的证件照，那种平时搞怪的、光着膀子的生活照，基本就是自杀行为。

3. 准备专业技能

对于程序员的专业技能其实就是技术栈，对于自己的技术栈如何描述是个很难的问题，比如什么算是精通？什么算是了解？什么是熟悉？

关于对技术技能的描述有很多种，有五种的也有三种的，而且每个人对词汇的理解都不一样，我结合相关专家的理解和自己的理解来简单阐述下描述词汇的区别，我们这里只讲三种的了解、熟悉、精通。

- 了解：使用过某一项技术，能在别人指导下完成工作，但不能胜任复杂工作，也不能独立解决问题。
- 熟悉：大量运用过的某一项技术，能独立完成工作，且能独立完成有一定复杂度的工作，在技术的应用层面不会有太大问题，甚至理解一点原理。
- 精通：不仅可以运用某一门技术完成复杂项目，而且理解这项技术背后的原理，可以对此技术进行二次开发，甚至本身就是技术源码的贡献者。

我们就以**Vue**这个框架为例，如果你可以用**vue**写一些简单的页面，独立完成某几个页面的开发，但是无法脱离公司脚手架工作，也无法独立从0完成一个有一定复杂度的项目，只能称之为了解。

如果你有大量运用**vue**的经验，有从0独立完成一定复杂度项目的能力，可以完全脱离脚手架进行开发，且对**vue**的原理有一定的了解，可以称之为熟悉。

如果你用vue完成过复杂度很高的项目，而且非常熟悉vue的原理，是vue源码的主要贡献者，亦或者根据vue源码进行过魔改（比如mpvue），你可以称得上精通。

那么有两个坑是候选人经常犯的，『杂』和『精』，这种两个坑大量集中在应届生和刚毕业每两年的新手身上，其主要特点是『急于表现自我』、『对技术深度与广度出现无知而导致的过度自信』。

首先说说杂，比如你要应聘一个Java后端，老老实实把自己的java技术栈写好就行了，强调一下自己擅长什么即可，最好专精某领域比如『高并发』、『高可用』等等，这个时候一些简历非要给自己加戏，自己会的不会的一股脑往上堆，什么逆向、密码学、图形、驱动、AI都要体现出来，越杂越好，这种简历给人的印象就是个什么都不懂的半吊子。

再说说精，一个刚毕业的应届生，出来简历就各种精通，精通Java、精通Java虚拟机、精通spring全家桶、精通kafka等等，请放心，这种简历是不会没头没脑投过来了，这种在大学里就精通各种的天才早被他的各种学长介绍进了大厂或者外企做某某Star重点培养了，往往看到的这种也是半吊子。

4. 准备工作经历

工作经历本身不用花太多笔墨去写，面试官主要想看的就是每段工作经历的持续时间、在不同公司担任的职责如何、是否有大厂的工作经验等等。

那么什么简历在这里给面试官减分呢？

- 频繁跳槽：比如三年换了四家公司，每个公司呆的时长不要超过一年
- 常年初级岗：比如工作五六年之后依然在完成一些简单的项目开发
- 末流公司经历：在技术招聘届，大厂的优先级最高比如BAT、TMD甚至微软、谷歌等外企，知名度独角兽其次，比如商汤、旷视等等，一般的互联网公司排在第三，就是工作中小型的互联网公司一般大家叫不上名字，排在最后的就是外包和传统企业的经历

所以，如果你有频繁跳槽的经历怎么办？在本公司老老实实等到满一年再跳槽。

如果常年初级岗怎么办？想办法晋升或者参与一些业界知名项目，再或者写一个有一定复杂度的私人项目。

如果有末流公司经历怎么办？如果是很久以前的末流公司经验可以直接不写，也没人在乎你很早之前的工作经历，如果你现在就在末流公司，赶紧想办法跳槽，去不了大厂，去非知名的互联网公司也算是胜利大逃亡了。

5. 准备项目经历

项目经历不管对于社招还是校招都是重中之重，很多时候成败就在于项目经历这块，一个普通本科可以通过优秀的项目经历逆袭985，一个小厂的员工也可以获得大厂的面试机会。

但是必须要说一下项目经历的编写很讲究，这是为后面面试部分铺路的绝佳机会，也是直接让你的简历扑街的重点沦陷区域。

先说容易让简历扑街的几个坑位。

（1）切忌流水账写法

项目经历流水账写法是绝大多数简历的通病，通篇下来就讲了一件事『我干了啥』。

大部分简历却是这样的：

用Vue、Vuex、Vue-router、axios等技术开发电商网站的前端部分，主要负责首页、店铺详情、商品详情、商品列表、订单详情、订单中心等相关页面的开发工作，与设计师与后端配合，可要高度还原设计稿。

这个描述有什么问题？其实看似也没啥问题，但是这种流水账写法太多了，完全无法突出自己的优势展现自己的能力。项目经历是考察重点，面试官想知道候选人在一次项目经历中扮演的角色、负责的模块、碰到的问题、解决的思路、达成的效果以及最后的总结与沉淀。

而上面的描述只显示了『我干了啥』，所以这种项目描述几乎是没意义的，因为对于面试官而言他看不到有效信息，没有有效信息的项目描述基本就没价值了，如果这个时候你还没有大厂经历或者名校背书，基本上也就凉了。

（2）切忌堆积项目

堆积项目这种现象往往出现在没有什么优秀项目经历的简历身上，候选人企图以数量优势掩盖质量的劣势，其实往往适得其反，项目经历的一栏最好放2-3个项目，非常优秀的项目可能放一个就足够了，举个极端例子如果有一天尤雨溪写简历，其实只需要在项目经历那些一行『Vue.js作者』就行了，当然，他并不需要投简历。

有一些项目切忌放上去：

- **demo级项目**：很多简历居然还在放一些仿xx官网的demo，这是十足的减分项，有一些则是东拼西凑抄了一些框架的源码搞了个玩具项目，没有任何价值。
- **烂大街的项目**：这种以vue技术栈的为最，由于视频网站的某门课程流行，导致大量的仿饿了么、仿qq音乐、仿美团、仿去哪儿，同样Java的同学也是仿电商网

站、仿大众点评等等，十份简历5份一模一样的项目，你是面试官会怎么想。

- 低质量的开源项目：一个大原则就是低star的尽量别放（除非是高质量代码的冷门项目），长期弃坑的也不要放，不要为了凑数量把低质量的项目暴露出来，好好藏着。

如果只放两个项目，最好的搭配是一个公司内部挑大梁的项目和一个社区内的开源项目，后者之所以可以占据一席之地，是因为通过你的开源项目，面试官可以通过commit完整看到你的创造过程，比如工程化建设、commit规范、代码规范、协作方式、代码能力、沟通能力等等，这甚至比面试都有用，没有比开源项目更能展示你综合素质的东西了。

（3）切忌放虚假项目

一个项目做没做过只要有经验的面试官一问便知，如果你真的靠假项目忽悠过了面试，那这个公司八成也有问题，人才把关不过硬，你可以想象你的队友都是什么水平，在这种公司大成长价值也不大。好，如果你说实在没项目可写了，我只能造假了，那么你应该想一下这多层追问。

比如你说你优化了一个前端项目的首屏性能，降低了白屏时间，那么面试官对这个性能优化问题会进行深挖，来考察候选人的实际水平：

1. 你的性能优化指标是怎么确定的？平均下来时间减短了多少？
2. 你的性能是如何测试的？有两种主流的性能测试方法你是怎么选的？
3. 你是根据哪些指标进行针对性优化的？
4. 除了你说的这些优化方法还有没有想过通过xx来解决？
5. 你的这个优化方法在实际操作中碰到过什么问题吗？有没有进一步做过测试？
6. 我们假设这么一种情况，比如xxxx，你会这么进行优化？

面试官多层追问的逻辑是这样的：了解背景 -> 了解方案 -> 深挖方案 -> 模拟场景

首先得了解你性能优化的指标如何，接着需要了解你是怎么测试的指标、再怎么进行针对性优化的，再接着提出一些其它解决方案考察你对优化场景的知识储备和方案决策能力，最后再模拟一个其它的业务场景，来考察你的技能迁移能力，看看是否是对某块领域有一定的了解，而不是只针对某个项目。

如果要真的在面试现场对答如流，那么一定是在某一块领域有一定知识储备的人，不是随随便便搞个项目就能蒙混过关的。

（4）合格的项目经历如何写

合格的项目经历必须要有以下几点：

- 项目概述

- 个人职责
- 项目难点
- 工作成果

如果你不怕字太多，还可以选择性加入解决方案、选型思路等等，但是由于篇幅限制和为面试铺垫就不太建议写得太多。

项目概述的目的是让面试官理解项目，不是每个人面试官都做过你的那种项目，所以需要有一个简述方便面试官理解。

个人职责就是告诉面试官你在本项目中扮演的角色，是领导者？主导者？还是跟随者，你负责了哪些模块，承担了多大的工作量，以此来评估你在团队中的作用。

项目难点的目的在于让面试官看到你碰到的技术难题，方便后续面试对项目进行一系列讨论。

工作成果就很明显了，面试官需要看到你在做了上述工作到底达成了什么成绩，这个时候最好以数据说话，比如访问量、白屏时间等等。

这个时候也切忌展开长篇大论，把技术细节一个个写上去，甚至还写了心路历程的都是大忌，一方面篇幅太大会造成视觉混乱，另一方面面试官想看到的是『简』历，不是技术总结，面试官要面对上百份简历没那么时间来看你长篇大论，长篇大论大可以在面试中展开。

最好的方法就是一行文字简单得说清楚即可，反正项目面的时候一定会问到，到时候好好把你准备的内容讲给面试官，掌握面试的主动权就是从项目经历这一栏中开始

6. 其他

（1）教育背景

应届生可以写得更详细一点，比如绩点排名怎么样，有没有突出的科目，社招就不要写太多了，简单的入学时间、学校、专业即可，而且写你的最高学历即可，没必要从初中就开始写学历流水账，没有人看的。

（2）注意事项

- **自我评价不建议写**：技术面试几乎没人看你的自我评价，连面试技术问题都嫌『talk is cheap show me the code』，你的自我评价除了占篇幅没啥用处，充其量算是面试官的干扰信息。
- **简历封面千万别搞**：这都是一些简历制作网站骗用户付费的伎俩，不仅是互联网行业，其它行业我也没见过要简历封面这种无用操作的。

- **证书不建议写：**应届生可以酌情考虑弄个六级证书什么的，对于社招而言，列一堆证书甚至是减分项，国内的各种证你也懂的，是有多不自信才沦落到靠一堆证书来证明自己的价值。
- **千万别用技能图表：**首先用90分、80分来评价自己的技术本身就没有什么说服力，也不可能这么精准，而且什么是90分、什么是80根本就没有一个公论，所以用一般的比较通用的熟悉、精通描述即可，千万别加戏，面试官或者HR没那么多闲工夫去理解你的图表，老老实实按最通用高效的方式描述自己的技术栈。
- **简历最好一页：**程序员又不是设计师有时候需要作品呈现，如果你的简历超过一页那么一定是出问题了，要么项目、技术栈描述太多太杂占据大量篇幅，要么加了一堆图表或者图画来加戏，当然往往是犯前一个错误的更多。