

拉勾教育

— 互联网人实战大学 —

# 《前端高手进阶》

朱德龙 前中兴软创主任工程师

— 拉勾教育出品 —

# 前端高手进阶

开篇词：带你一起筑建前端工程师的“护城河”

## 讲师介绍

拉勾教育

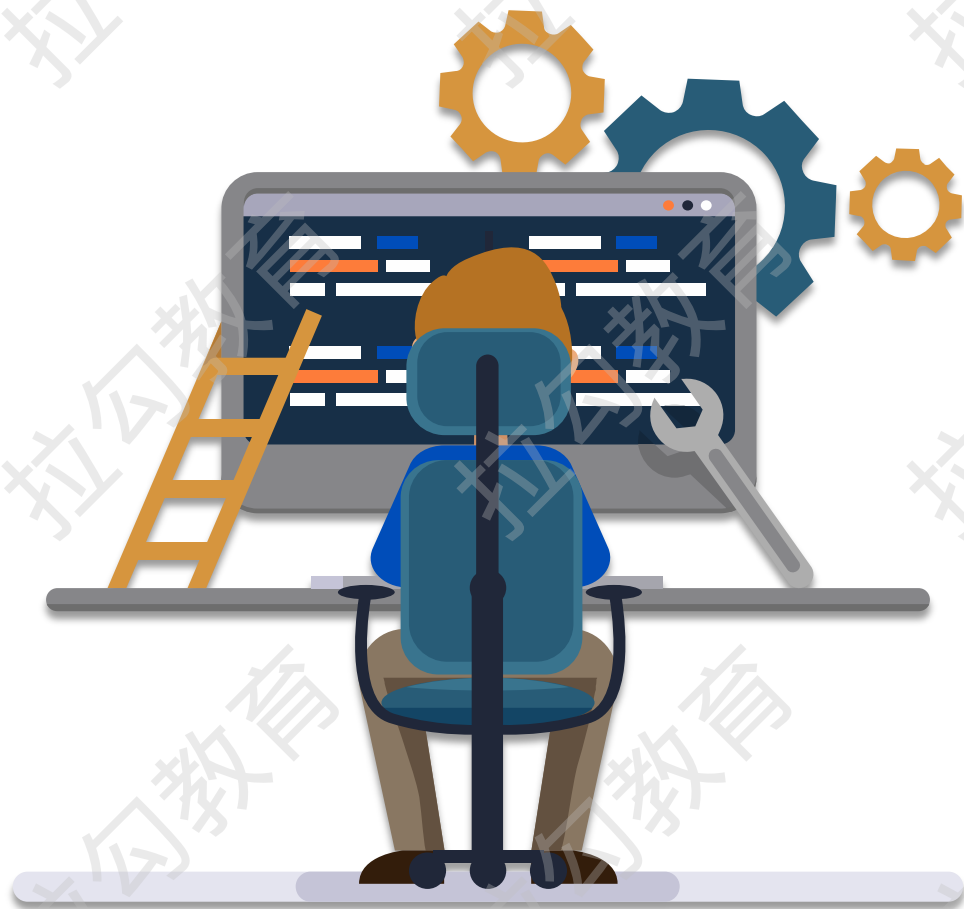
— 互联网人实战大学 —

讲师：朱德龙  
(亚里士朱德)

毕业后进入中兴旗下子公司成为 JSP 全栈工程师  
负责开发 B/S 架构的软件系统  
参与过亿级数据量的复杂系统开发

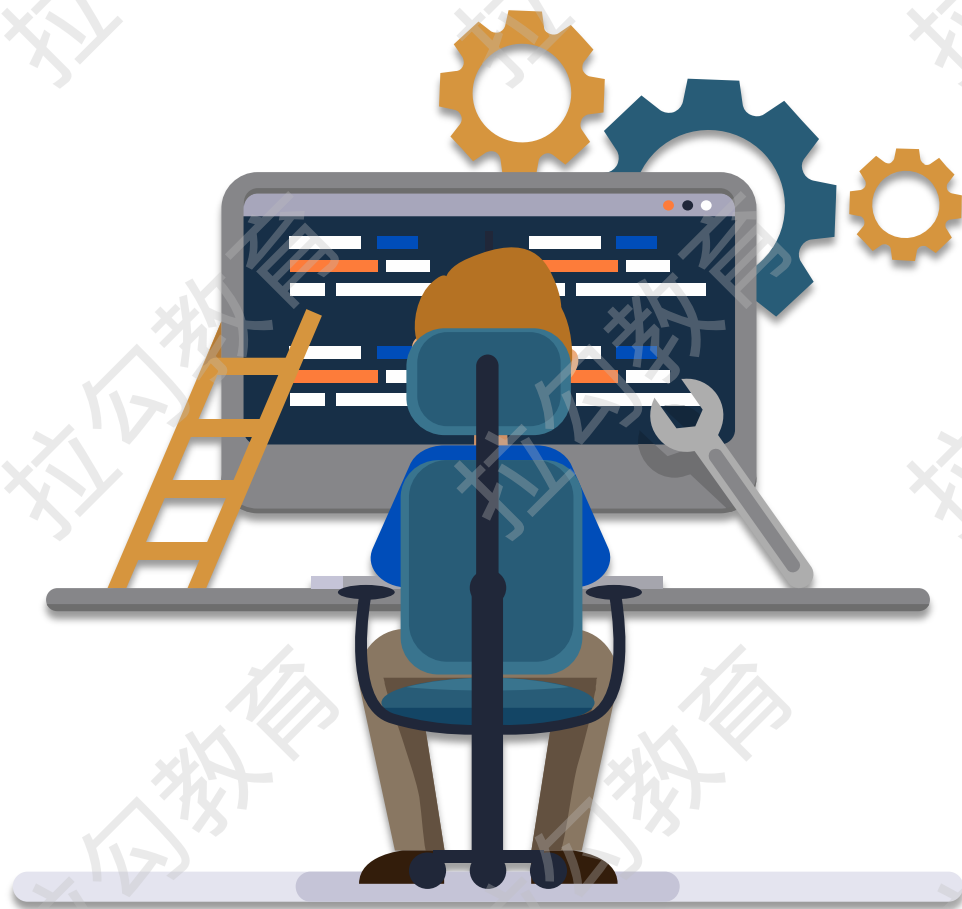
后加入创业公司作为核心骨干开发网上预约平台  
从 0 到 1 搭建并开发过云服务平台的前端项目

现任创业公司前端负责人  
带领团队完成大数据、重型计算、深度学习等  
业务场景的项目前端开发工作



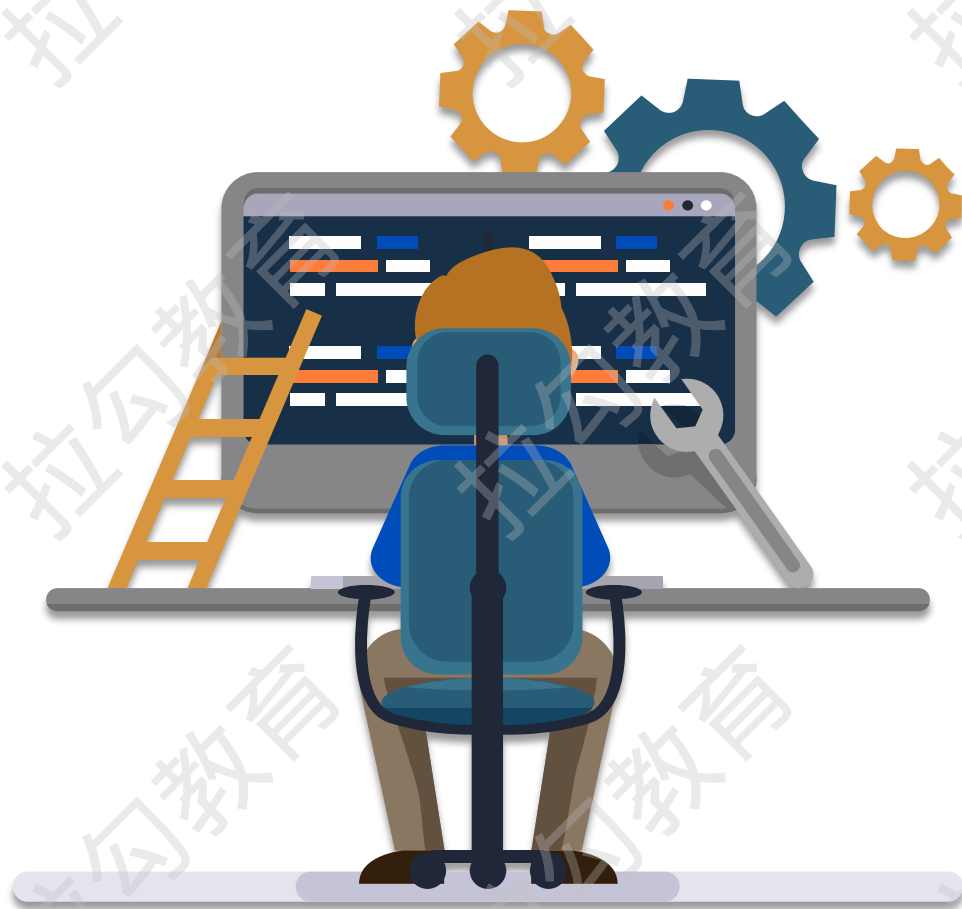
常见误解：

- 工作年限 = 级别（能力）
- 框架/工具使用得越多越好



常见误解：

- 工作年限 = 级别（能力）  
(中年危机！)
- 框架/工具使用得越多越好



常见误解：

- 工作年限 = 级别（能力）  
(中年危机！)
- 框架/工具使用得越多越好  
(使用过多种技术，但都只停留在表面)

## 某人的工作经验简述：

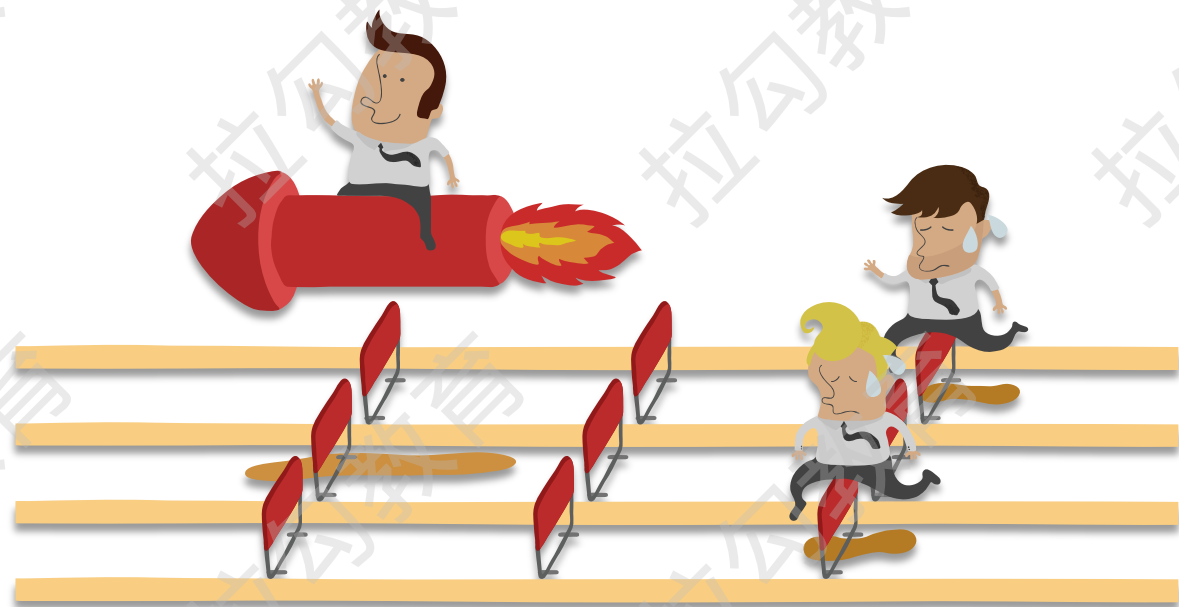
- 使用 zepto.js 和 CSS3 动效开发过一些移动端活动页面
- 使用 React 和 Ant Design 开发过一个医药商城 Web App 的核心模块及后台管理页面
- 熟悉 Git 工具和代码管理流程
- 熟悉 webpack、Gulp.js 等工具的使用
- 自学了 Node.js、Docker、MongoDB，能编写一些小项目

**“熟练” 这种主观性的自我描述建议少用**



前端工程师之间的能力和薪资存在着“贫富分化”的现象

而且随着时间的推进在不断加剧





## 1. 前端知识点太多

前端需要掌握三门语言：

- **HTML 语言**：有上百个用法各异的标签，每个标签还有属性，还有 DOM 提供的 API
- **CSS**：选择器有十几类，属性有上百种，常用的枚举属性 display 有十几种值
- **JavaScript**



## 2. 前端技术更新速度快

- 前端框架有很多种，主流就有 2~3 种
- 构建工具不断更新，有些框架还封装了自己的命令行工具
- 不断出现新技术名词，如 SSR、PWA、Serverless、Flutter 等
- .....



## 2. 前端技术更新速度快



别更新了,学不动了

## 2. 前端技术更新速度快

企图通过量变引起质变的方式并不可取，时间精力是一方面

更重要的是，囫圇吞枣的学习方式缺乏对前端知识系统的认知

只是把知识点堆砌在一起而没有形成自己的认知框架

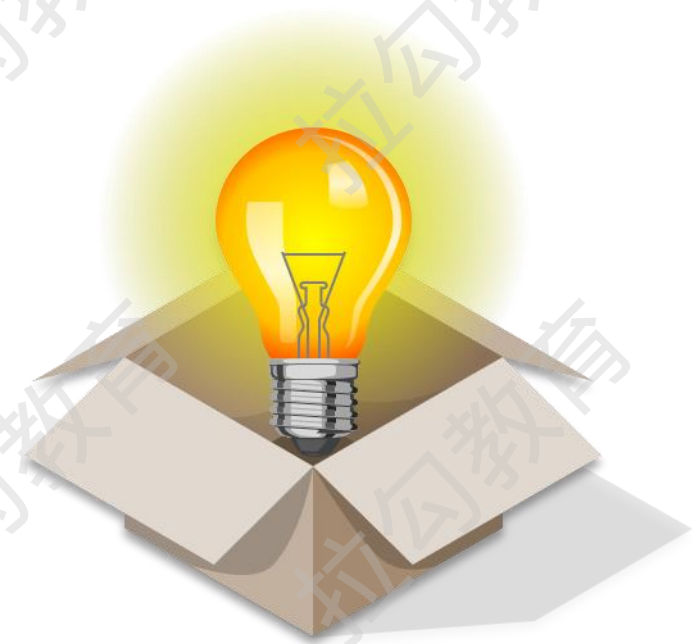


## 3. 前端应用场景越来越复杂

不同开发场景下的实现目标、API 及调试工具又各有差异

比如微信小程序和钉钉小程序就提供了各自独立的开发工具

- **浏览器环境**下为 JavaScript 提供了 BOM 和 DOM
- **Node.js 环境**下提供的是 fs、net 等模块



# 如何破局？

拉勾教育

— 互联网人实战大学 —

## 1. 建立合理的知识结构

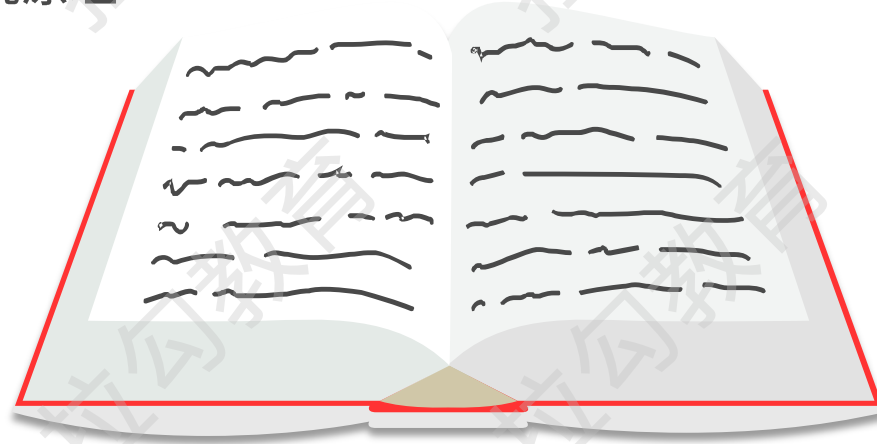
### 知识框架的可扩展性

在尽量少的调整自身已有知识结构的情况下，可以不断将新的技术知识吸纳进来

### 知识点的完备性

每个知识点不应停留在只会调用接口函数的程度，而是深入其实现原理

然后能加以运用，从而构建更复杂更具通用性的项目



## 2. 培养可复用的工作能力

除了最基本的能看懂文档、调用接口的编程能力之外

还需要着重提升以下能力：

- **探究能力**，深度探究技术背后的原理，并且能结合实践灵活运用
- **解构能力**，能够分析和分解复杂问题，并一步步解决
- **归纳能力**，建立知识点之间的联系，并找到其共性，从而达到举一反三的目的



深入讲解前端核心知识点的底层原理以及相关实战经验

希望借此帮助你建立起系统的知识框架

提升综合技术能力

## 模块一：前端核心基础知识

- 1 你真的熟悉 HTML 标签吗？
- 2 如何高效操作 DOM 元素？
- 3 3 个使用场景助你用好 DOM 事件
- 4 掌握 CSS 精髓：布局
- 5 如何管理你的 CSS 代码？
- 加餐1 手写 CSS 预处理
- 6 浏览器如何渲染页面？
- 7 关于 JavaScript 的数据类型，你知多少？
- 8 为什么说函数是 JavaScript 的一等公民？
- 9 为什么代码没有按照编写顺序执行？



深入讲解前端核心知识点的底层原理以及相关实战经验

希望借此帮助你建立起系统的知识框架

提升综合技术能力

10 怎么复用你的代码？

11 为什么说 JavaScript 不适合大型项目？

12 浏览器如何执行 JavaScript 代码？

13 区分浏览器中的进程与线程

加餐2 手写 Promise、async/await

14 HTTP 协议和它的“补丁”们

15 如何让浏览器更快地加载网络资源？

16 浏览器同源策略与跨域方案详解

17 前后端如何有效沟通？

## 模块二：前端实际应用场景解析

18 你是怎么理解“组件”这个概念的？

深入讲解前端核心知识点的底层原理以及相关实战经验

希望借此帮助你建立起系统的知识框架

提升综合技术能力

## 模块二：前端实际应用场景解析

18 你是怎么理解“组件”这个概念的？

19 把路由放在前端意味着什么？

20 详解组件通信之状态管理

21 你的代码到底是怎么编译的？

22 如何合理搭建前端项目？

23 谈性能优化到底在谈什么？

24 你的代码是怎么成为黑客工具的？

## 模块三：综合能力提升

25 Node.js == 全栈？

26 常用的数据结构了解多少？

深入讲解前端核心知识点的底层原理以及相关实战经验

希望借此帮助你建立起系统的知识框架

提升综合技术能力

## 模块三：综合能力提升

25 Node.js == 全栈？

26 常用的数据结构了解多少？

27 到底该不该了解算法？

28 你都了解过哪些编程方式？

29 框架到底用了哪些设计模式？

30 前端热点技术之 Serverless

31 微前端与功能的可重用性

## 彩蛋

32 如何获得心仪的 Offer？

33 谈谈前端工程师的职业规划（如何跳槽、如何晋升）

- 服务端有 Node.js
- 客户端有 Electron
- 移动端小程序以 JavaScript 作为主语言
- PWA 更有颠覆原生 App 之志

**前端工程师再也不只是“写页面”的了！**



## 如何提升竞争力？

欢迎你在留言区与我、或与同行的朋友  
分享你的成长和经验心得



Next: 第01讲 《那些看不见的 HTML 标签》

# 拉勾教育

— 互联网人实战大学 —



关注拉勾「教育公众号」  
获取更多课程信息