

🇺🇸 SDE学习&面试路线规划

本文给大家介绍SDE面试如何准备，以及怎么利用有限的时间合理规划、快速达到面试要求。

很多同学去🇺🇸美国读硕转码，遇到一个非常常见的问题就是，美国那边intern或全职找工通常要提前很长时间投递简历，可能人刚到美国，很过公司就开放岗位了。而且投递的越早越有优势。这样对于转码选手特别不友好，还完全没来得及准备刷题+项目+OOD+System Design等内容，那怎么破？

SDE面试考察哪些内容？

SDE面试通常考察如下内容：

- 算法题
- 项目
- OOD (object oriented design)
- System Design
- BQ

零基础转码选手如何制定学习规划路线？

计算机的方向多，知识杂，那么怎么进行合理的规划以快速达到面试要求呢？

很多同学有如下疑问：

- 计算机的几门核心专业课（OS、Database、Network、系统结构等）要不要花大力气先学？
- 编程语言怎么选，Java VS Python VS C++？
- project应该怎么找？
- 应该学习哪些skills？
- 刷题感觉自己刷得很慢。刷题应该刷多少道？hard的是否需要刷？

从快速应对面试的角度来看，应该要重点针对面试考察的点来学习。

1. 首先是掌握一门编程语言，优先推荐Java。在🇺🇸这边现在Java使用得还是最多的，而且它是一门面向对象的语言，生态也成熟，用来写各种项目非常方便。当然Python也OK，不过入职后大概率还是要学一门其他面向对象语言的。

2. 学语言的时候可以同时学下数据结构和算法。注意学得时候要多动手练习，掌握原理，以及要会活学活用。然后就是去leetcode刷题了。刷题也有技巧，按类型来专项刷，先刷easy的，完整刷完一遍后再刷medium，hard的考察频率低可以适当少投入时间。刷题是一个长期的事情，需要坚持，比如每天花2-3小时刷题，坚持就会有效果，最终要刷300-400道左右比较好。
3. 除了刷题，另一个大头就是项目。项目直接决定了简历筛选通过的概率。简历上的项目要体现技术深度和亮点，最好用到流行的技术栈，效果要有明确的数据指标。另外是针对不同的公司jd，可以准备多份简历，和jd越匹配简历筛选通过率越高。然后几个项目之间最好有一些差异化和侧重点，推荐从如下几个方面去考虑：**MERN的全栈项目、Springboot&微服务项目、Infra的项目、Mobile项目**。
4. 项目和刷题会话费比较多的时间。OOD其实范围比较小，看几道真题，找找感觉总结下规律应该差不多了。
5. 至于核心的几门专业课。比如操作系统、计算机体系结构、计算机网络、数据库，重要肯定是重要的，因为这些是计算机程序运行的基础，掌握扎实对实际的工作肯定有帮助。这里面，数据库是必学的，因为做项目需要用到，其他几门如果暂时时间仓促，可以先不学，先按面试准备，只要项目没有涉及相关的知识那大概率也不会问，后面拿到offer有空余的时间再补上。
6. BQ。部分公司比如Amazon把BQ看的很重，最好找一些面经多看看，总结下高频问题和回答方式。

学习资料

刷题

刷题就是去leetcode刷，网上免费的教程也很多，没啥捷径，就是反复刷，加强理解活学活用。

项目

去哪里找项目

网上常见的找项目的平台有如下几个，当然有一些是付费的。

- udemy
- udacity
- YouTube
- github

<https://github.com/codecrafters-io/build-your-own-x>

这个仓库汇集了非常多的优质资源，带你从0实现各种技术框架，比如web server、爬虫、DB、OS等等，代码量不大，但是都是一个核心功能具备的完整项目，非常助于理解框架的设计，非常推荐。

2 Project based learning

<https://github.com/practical-tutorials/project-based-learning>

一系列优秀课程的集合，以项目实战的方式来带你设计一个自己的应用。涵盖了C/C++/Python/Java/R等多种语言。

3 Free project course

<https://github.com/resumejob/free-project-course>

整理了网络上免费的项目实战课，涵盖大厂高频考点，很多都有完整的视频和文档，包括Java、Python、C++的项目实战。

- freecodecamp
- CodeCademy
- Educative.io
- DataCamp

这里给一些比较典型的案例：

MERN项目：

- YouTube-Build a Fullstack Blog App using MERN (mongo, express, react, node):
<https://www.youtube.com/watch?v=xKs2lZZya7c>

Springboot项目：

- YouTube-Full Stack Development with Java Spring Boot, React, and MongoDB — Full Course
(电影评论网站)：<https://www.youtube.com/watch?v=5PdEmeopJVQ>
- Spring Boot Angular Project in 8 Hours 🔥🔥 – Build a Youtube Clone
🔥:<https://www.youtube.com/watch?v=DW1nQ4o3sCI>

infra的项目比如说实现一个KV存储引擎（参考leveldb），或实现一个数据库路由组件等等。

做项目过程中需要注意哪些点

需要注意的是，网络上的这些项目教程大多时长比较短就几个小时，只是教你一步一步把代码敲完，很多更加重要的内容没有体现：

1. 项目的完整开发流程
2. 技术选型是怎么做的
3. 技术方案的思考以及design上的trade-off，遇到问题和挑战怎么解决，这个是面试官非常关心的
4. 项目效果如何获取的，怎么做的性能测试
5. 团队合作方面。比如自己承担的是什么角色，如何分工，怎么一起获得项目结果

关于项目类型

项目类型来说有如下几种：

- 真实的工业项目，有真实的用户，有impact，公司实习会更加有含金量
- 有影响力的git开源项目
- 独立开发者项目，用户量可能很小，但是能解决一个场景下的具体问题。比如写了一个插件记录leetcode submissions，或者制作了一个网页版本的小游戏。经历了完整的开发流程，能够正常部署运营。有用户数最好了，用户数就是背书。
- 学校课程小项目，或者跟着网上教程上做的项目。这种项目查重率极高，需要进行包装。另外是需要之其人知其所以然，能理解并且讲清楚。
- 另外就是自己在网上找的项目，跟着一些视频做的。需要注意不能是那种让面试官一眼就看出来是培训班的项目，尽量针对性的做一些优化和改造。

关于技术栈

全栈的比较合适，纯后端的话，匹配的岗位没那么多。

尽量多用一些前端、云服务AWS等、mobile app等技术。

- web全栈。比较容易出一个完整的东西，能够快速了解基本的从前端到后端的整个流程。
- java backend。可以有一些比较有深度的，比如分布式、并发编程相关，涉及多个技术组件比如redis、mysql等等。

关于后端的选择：

各种喜欢新型技术的企业/产品用 node, 大公司/稳定的产品用 Java/C++, go 好像很牛但是小众, python 好像比较常见于小 project.

最常见的组合就是精通一个 frontend, 然后了解 backend. 或者精通 backend, 了解其中一个 frontend。

OOD

网络上有一些免费的教程，比如设计电梯系统、停车场系统等等，看几个视频课总结下规律。

学习教程推荐：

(1) https://www.youtube.com/watch?v=odS_mmxDA&list=PLvylyKZVcfAI07QZ-zhebl5Zsf1EAbzID



【面向对象设计OOD】 - Elevator 设计电梯

YouTube · Edward Shi
Nov 16, 2021



【面向对象设计OOD】 - Parking Lot 设计停车场 (2)

YouTube · Edward Shi
Nov 16, 2021



【面向对象设计OOD】 - Amazon Locker 设计亚马逊快递柜

YouTube · Edward Shi
Nov 16, 2021



【面向对象设计OOD】 - Parking Lot 设计停车场 (4)

YouTube · Edward Shi
Nov 16, 2021

(2) 前面的部分章节是免费的，可以看。 <https://www.educative.io/courses/grokking-the-object-oriented-design-interview/RMIM3NgjAyR>

System Design

System Design一般intern不会考，NG的全职可能考简单的，社招考察比较多。

重点是要掌握一个完整的系统有哪些模块以及相关理论基础、数据流程是怎么样的、涉及哪些技术框架或组件、伸缩性等怎么考量。

推荐教程：

- system-design-primer: github star164K, 理论+实践涵盖范围都非常齐全。强烈推荐。:
 [GitHub – donnemartin/system-design-primer: Learn how to design large-scale systems. Prep for the system design interview. Includes Anki flashcards.](https://github.com/donnemartin/system-design-primer)
- <https://www.youtube.com/c/SystemDesignInterview/videos>
- <https://www.hiredintech.com/courses/system-design>
- <https://www.youtube.com/playlist?app=desktop&list=PLAd5bt5mn3V3TrrJFBpnu4PH9e8KZMvNA>

关于面试

Mock Interview

可以多和同学、学长学姐做mock interview, 一方面是练习英语口语, 另一方面是可以把项目的思路、细节捋清楚。

一些平台：

1. paramp (免费) : <https://www.pramp.com/#/sign-in>
2. interviewing.io
3. <https://www.interviewbit.com/peer-mock-interview/>
4. (付费) : <https://www.techmockinterview.com/>

关于Online Coding

对于Phone Interview, 平时可以多在Online Coding网站上练习, 比如: Codepad, GoogleDoc, Dollabedit 等。确保在真实面试过程中能够编译运行代码, 遇到问题快速Debug。

对于Onsite Interview, 大多公司比较喜欢用白板进行面试, 这就要求同学们平时刷题的时候不要过度依赖IDE的Grammar Check, 同时如果条件允许, 增加用白板联系写代码的经验, 在正确的基础上也要一定程度保证美观。

面试Coding部分时, 一定要首先跟面试官确定题目的隐含假设, 例如输入是否有序、输入/输出是否可以有重复元素等。了解题意之后, 尽量先与面试官交流自己的大致思路, 确保该思路是面试官想要的方向再继续进行。

PS: 如果需要SDE项目辅导、面试辅导、简历修改也可以联系我哦~~【V:
meta1101】

一亩田指南

一亩田指南