



YOUR TECHNICAL RECRUITING JOURNEY

KAT WANG

Software Engineer @Google
CSE 492J Instructor

Past Experience:

Legacy CSE 332 TA

Software Engineering Intern @
Google, Bloomberg, EA

Email: siyukw@gmail.com

LinkedIn: [linkedin.com/in/siyukw](https://www.linkedin.com/in/siyukw)



“What am I even looking for?”

Roles

- Software Engineering***
- Program/Product Manager
- Quality/Test Engineering
- Data Scientist
- Technical Consulting
- Business Analyst

Technology Areas

- Applications
- Web Development
- Enterprise Software
- Artificial Intelligence
- Machine Learning
- AR/VR
- Gaming
- Operating Systems
- Cloud
- Information Technology

“Okay but from where?”







\$100k – \$120k

Self-reported starting salary range



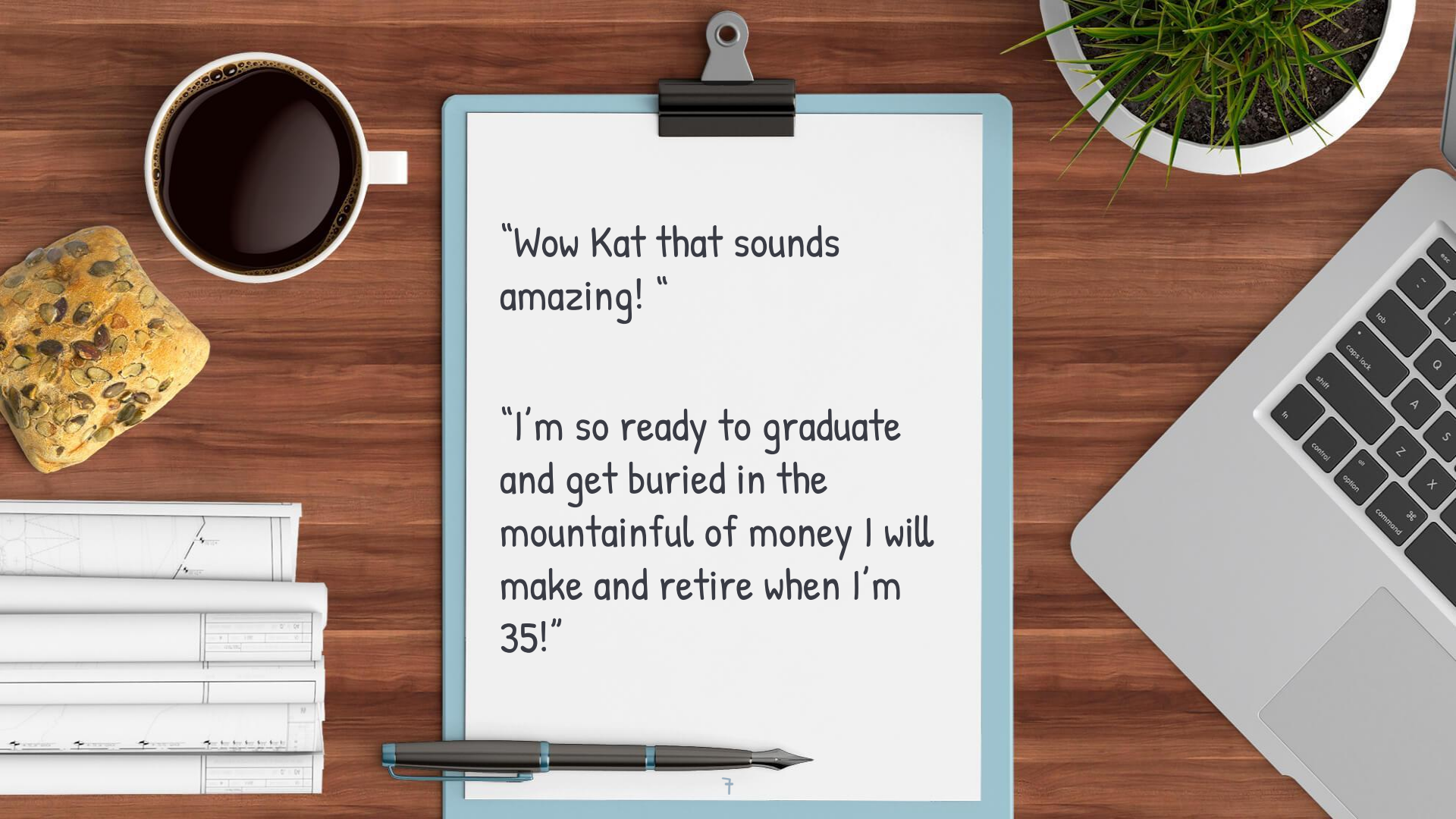
\$75k - \$120k

Equity packages from large, publicly traded companies

(4 year vesting schedule)

\$15k – \$75k

Signing bonus range from
competitive compensation packages



"Wow Kat that sounds amazing!"

"I'm so ready to graduate and get buried in the mountainful of money I will make and retire when I'm 35!"





THE RECRUITING PROCESS

Fall is the PRIME recruiting season.

- > Most companies recruit during this time until roles are filled.
- > Different roles at different companies can have very different processes for different people.

- Figure out what you want
- Apply to a fuck ton of places
- Prepare for technical interviews

August

- Hit up ALL the career fairs
- Start handling phone screens, behavioral interviews, coding challenges

October

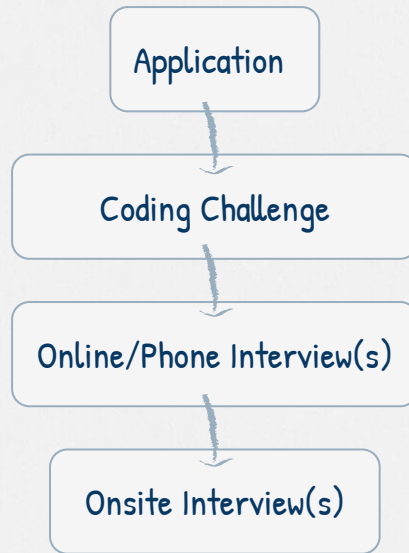
September

- Apply more aggressively
- Start emailing for referrals and hit up recruiters
- Practice on Leetcode or something

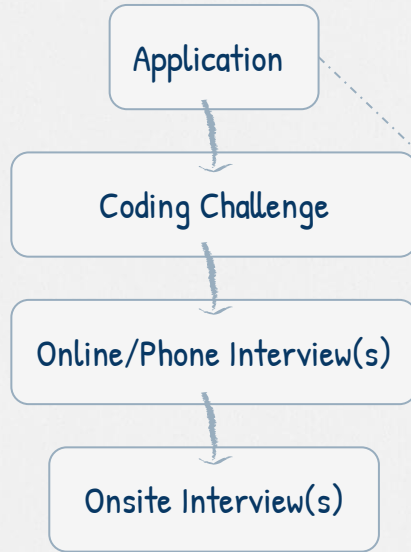
November

- CRUSH THOSE INTERVIEWS
- Don't give up

THE TYPICAL RECRUITING PROCESS

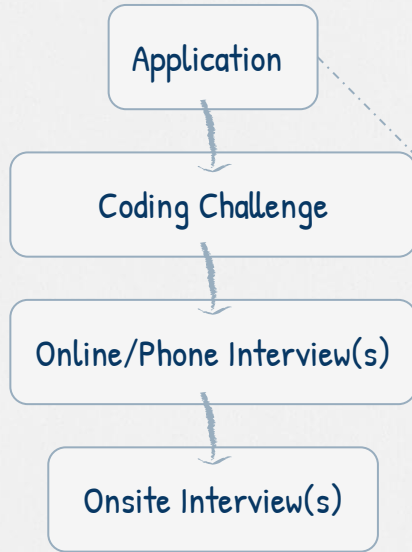


THE TYPICAL RECRUITING PROCESS



- Online (company websites, LinkedIn, Indeed, etc), Career Fairs
- Mostly ask for information on your resume
- Time consuming, tedious, repetitive, but definitely worth the effort!
- You won't hear back from most of them.
- Referrals are very helpful

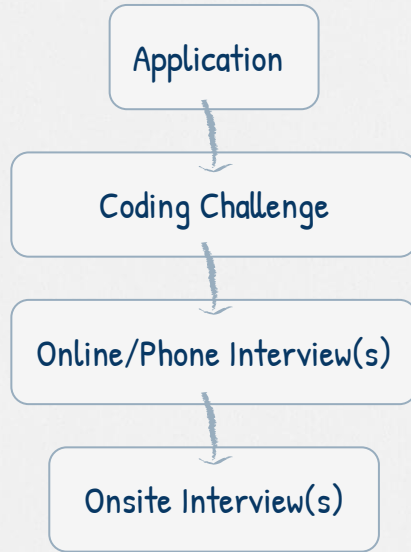
THE TYPICAL RECRUITING PROCESS



**MAKE SURE YOUR RESUME IS
UPDATED AND READY TO GO**

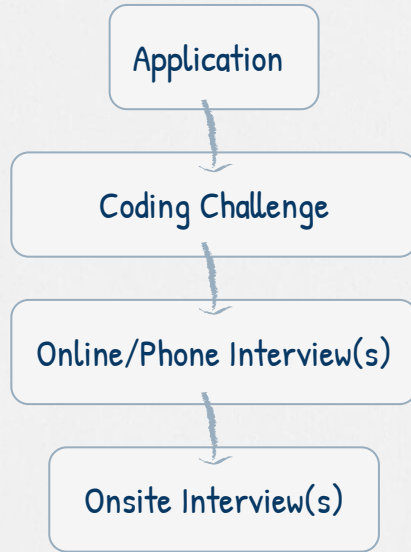
- One page
- Highlight the strongest area you have to present
- Make yourself a LinkedIn
- <http://bit.ly/cseresumeguide>

THE TYPICAL RECRUITING PROCESS



- Sent out by companies via email (so check your inbox often!)
- Typically expire in a week or two
- 60min - 90min time limit to solve 1 - 4 leetcode style questions on websites like Hackerrank
- Sometimes come after a recruiter call

THE TYPICAL RECRUITING PROCESS



- Video or audio call with an engineer/engineers (**MAKE SURE YOU HAVE CELL SERVICE + WIFI**)
- Usually 1 or 2 60 minute long interviews
- Coding questions on a shared screen
- Typical technical interview structure (We will talk more about this later!)

THE TYPICAL RECRUITING PROCESS



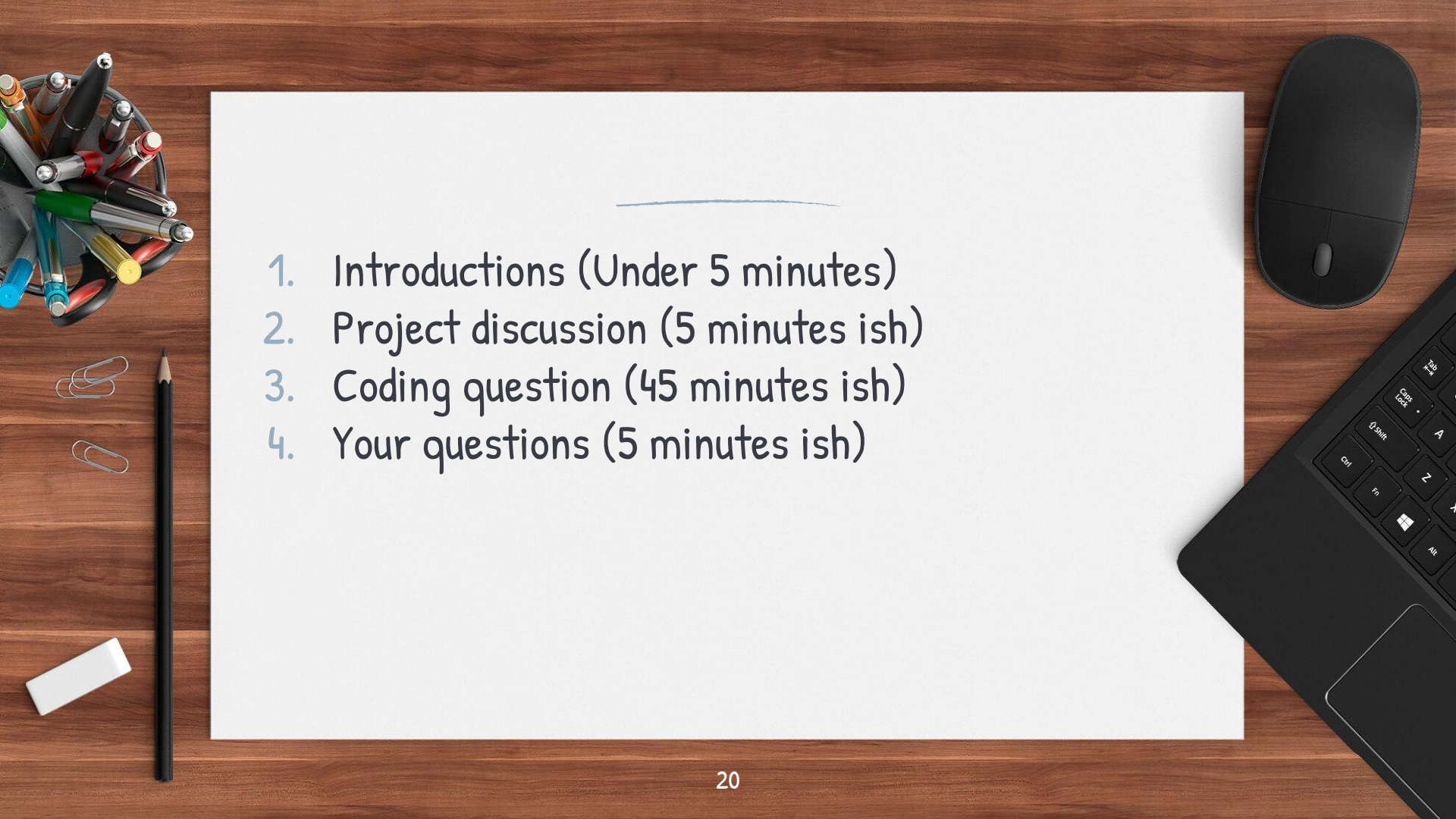
- Free lunch and (usually) free trip!
- Anywhere between 2 hours and a whole day, sometimes include a “lunch interview”
- Face to face with engineers
- Coding questions on a whiteboard or a laptop
- Typical technical interview structure

THE TYPICAL RECRUITING PROCESS





STRUCTURE OF A TYPICAL
TECHNICAL INTERVIEW

- 
-
1. Introductions (Under 5 minutes)
 2. Project discussion (5 minutes ish)
 3. Coding question (45 minutes ish)
 4. Your questions (5 minutes ish)

1. INTRODUCTIONS

- > "Hello."
- > "How are you?"
- > "Who are you?"
- > "Why do you want to work here?"

Be nice, be polite, be confident.



Don't do this ^^^^^



Do this ^^^^^



2. PROJECT DISCUSSION

Have 1 or 2 project(s) that you feel confident talking about in front of people who know what's up.

> "Are school/internship projects okay?"

2. PROJECT DISCUSSION

"I spent this summer working at an advertising network, specifically trying to drive engagement on our video ads by A/B testing new ad content and formats. I worked primarily in the backend and used Python and R for data analysis. I produced a 8% improvement in click-through rates across the board over six weeks of testing."

How long did you work on this project?

How big was the team working on this, what was your role specifically?

Why did you choose that technology stack?

What was the biggest bug you encountered and how did you fix it?

If you redid the project what would you do differently?

3. CODING QUESTION

- > This is the bulk of technical interviews.
- > Don't freak out, I promise you will be fine.

3. CODING QUESTION

General Tips:

- > Treat the interview like a standardized test
- > Practice coding without an IDE/Compiler/Computer
- > Practice coding and talking aloud at the same time
- > Essential Practice Resources:
 - Cracking the Coding Interview (book)
 - LeetCode (practice site)
 - Hackerrank (practice site)
 - Data Structures and Algorithms (course)

Picking a language:

- > Strongly recommended: Pick something OOP
- > Syntax typically doesn't matter
- > Review helpful APIs
 - String -> Int
 - String manipulation
 - Popular data structures
 - Searching and Sorting algorithms
- > Be able to talk about why you picked that language

3. CODING QUESTION

Question Patterns:

- > **String or Array manipulation** - Great for tech screens, shorter, sometimes mathy
- > **Linked Lists** - Often used in whiteboard interviews because they *expect* you to draw pictures
- > **Trees** - Often used when building up directories or searching for something ie phone trees
- > **Hash Tables** - If you are organizing data for lookups... chances are the answer is a hash table

3. CODING QUESTION

Structure Your Thoughts:

- > **Talk** - clarify the question
- > **Example** - talk through sample input and expected output
- > **Brute Force** - what is the simplest way to solve this?
- > **Optimize** - can you save run time or memory?
- > **Walk Through** - clarify your algorithm
- > **Implement** - write the code!
- > **Test** - list test cases, does your code address these?

4. YOUR QUESTIONS

Yeah I was also surprised that this part actually matters.

- > Show your interest
- > Learn more about the company

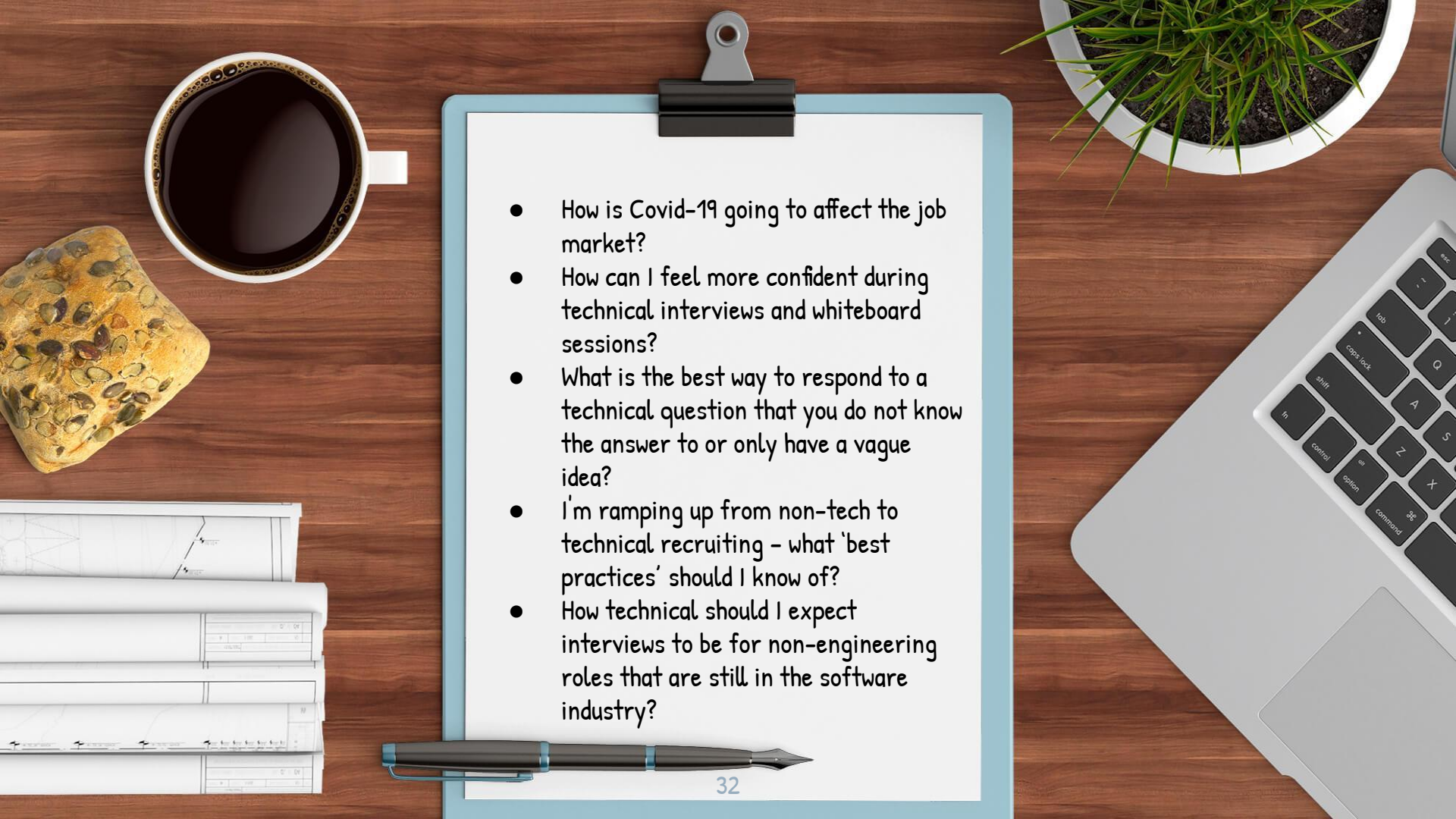
TODOs

- > Update your resume
- > Update your LinkedIn
- > Ask for referrals
- > Practice interviews
- > Start applying to places





Q & A

- 
- How is Covid-19 going to affect the job market?
 - How can I feel more confident during technical interviews and whiteboard sessions?
 - What is the best way to respond to a technical question that you do not know the answer to or only have a vague idea?
 - I'm ramping up from non-tech to technical recruiting - what 'best practices' should I know of?
 - How technical should I expect interviews to be for non-engineering roles that are still in the software industry?

