

Hello,

**KDT** 웹 개발자 양성 프로젝트

5기!

with





# 취업 관련 Tip



취벤



가즈아!!!!!!!!!!



대기업 vs

스타트업 + 중소기업

## 대기업

- 당장에 일할 사람 많아요
- 성장 시간에 투자 가능!
- 큰 규모의 Legacy 시스템 보유
- 나중에 진짜 잘할 사람이 필요

→ CS 지식을 많이 봅니다(컴과?)  
→ 코테가 빠셉니다  
→ 문제 해결 능력을 봅니다

기본이 탄탄해야 합니다!

VS

## 스타트업 + 중소

- 당장에 일할 사람 필요
- 업무로 성장할 기회 제공
- 다양한 개발 경험 가능
- 지금 당장 개발할 사람이 필요

→ 실무 위주의 지식을 봅니다  
→ 프로젝트, 포폴 질문 위주  
→ 인간성을 많이 봅니다

빠르게 적응이 가능한!





# CS 공부는 어찌?



# CS 공부 얼마나 해야하죠?

- 상대적으로 큰 기업을 원하면 CS랑 코테는 필수입니다!
- 다만 기업이 클 수록 면접을 보는 사람들이 짬이 찬 분들 → 자신들이 하는 분야가 아닌 이상 예전 학교에서 배웠던 CS에 대한 지식을 기반으로 질문 → 개발자에게 필수적인 CS 지식을 물어 봅니다!
- 그러니 모든 것 커버하겠다는 마인드 보다는 정말 중요한 것 부터 차근차근 시작하세요!



# 꿀팁 사이트 공유







# 참고할 만한 사이트 리스트(FE)

- <https://github.com/baeharam/Must-Know-About-Frontend#tada-%EC%86%8C%EA%B0%9C>
  - FE 개발자로 알아야할 CS 지식
- <https://github.com/alex/what-happens-when>
- <https://github.com/SantonyChoi/what-happens-when-KR>
  - 여러분이 주소창에 주소를 치고 엔터를 쳤을 때, 실제로서 생기는 일!
  - 매우 유명한 FE 기술 면접 내용입니다! 아래의 번역본으로 공부 하세요!



# 참고할 만한 사이트 리스트(CS-KOR)

- <https://github.com/gyoogle/tech-interview-for-developer>
  - CS 전반 / 기술 면접 Wiki
- [https://github.com/JaeYeopHan/Interview\\_Question\\_for\\_Beginner](https://github.com/JaeYeopHan/Interview_Question_for_Beginner)
  - CS 전반 / 기술 면접 대비



# 참고할 만한 사이트 리스트(EN)

- <https://www.frontendinterviewhandbook.com/introduction/CS>
  - 프론트 엔드
- <https://www.techinterviewhandbook.org/software-engineering-interview-guide/>
  - CS 전반



프로듀스 101 선공개  
소혜야 <Irony> 1조 무대 연습

데뷔가  
하고  
싶어?

배윤정  
댄스 트레이너  
소혜야 가수가 하고 싶어?

네카라쿠배당토 가고 싶어?



# 기술 블로그 눈팅하기!

- 요즘 규모가 있는 IT 회사면 대부분 기술 블로그를 운영합니다
- 해당 블로그에는 현재 이 회사가 관심을 가지는 분야가 어디인지, 그리고 가끔 다양한 꿀팁(!) 들이 나오게 됩니다!
- 가고 싶은 회사에 대해서 잘 알면 당연히 도움이 되겠죠!?
  - <https://d2.naver.com/home>
  - <https://tech.kakao.com/blog/>
  - <https://techblog.woowahan.com/>
  - 등등



자소서는요?

# 사실 이걸 너무나 케바케 입니다!



- 그래도 지켜야 할 건 있겠죠?





# 자소서 기본!

- 짧은 시간에 어필을 해야 합니다!
  - 결국 면접 보는 사람은 개발자 → 사이드 업무일 확률이 높아요!
- 첫번째 내용
  - 공고에 나온 자격조건 / 우대조건을 최 상단에 배치하여 전체 이력서를 요약
- 두번째 내용
  - 포폴 중심으로 포폴에 대한 개별 요약 내용 및 추가 설명 페이지
- 세번째 내용
  - 그 외에 어필하고 싶은 부분 작성하기



# 무조건 기술 스택의 양에 집중 하지 말 것!

- 무조건 양에 치중하면 안됩니다!
- 양으로 서류 통과해도 기술 면접에서 답변을 잘 못하면, 오히려 다른 잘 하는 기술 스택에 대한 신뢰도가 떨어져서 큰 마이너스가 됩니다!
- 정말 단순하게 'Hello, world' 찍어본 수준이라면 쓰지 않는 것을 추천 드립니다!



# 고민의 흔적을 보여 줄 것!

- 내가 그 기술을 왜 선택하고 썼는지는 반드시 이해하고 어필 하기!
- 물론 이러한 과정 없이, 포폴에 올라가는 것들이 있겠지만 면접 전에는 반드시 해당 내용에 대한 이해를 하고 있어야만 합니다!
- 프로젝트를 하심에 있어서도 원하는 걸 만들기 위해서는 이 기술 스택을 사용하는게 좋겠다! 라는 생각을 하고 접근 하시면 자소서 쓰기 좋아 집니다!



<https://www.youtube.com/watch?v=Yc56NpYW1DM&t=798s>

<https://wonny.space/writing/work/engineer-resume>



# Toy Project



# 고민의 흔적을 보여 줄 것!

- 팀 프로젝트에서는 아무래도 자신의 고민을 100% 보여주고 반영하기 어렵습니다!
- 스스로 가장 자신 있게 어필 할 수 있는 분야를 잡아서 토이 프로젝트를 하나 하면 정말 큰 도움이 됩니다!
- 다만, 좀 어려운 부분이 되겠죠?
  - 성능 최적화 / 브라우저 별 웹 접근성 맞추기 / 웹 페이지 테스트 / 오픈 소스 참여 등등



코테!



# 코테는 성실성의 증거이다!

- 보통 코테는 똑똑하면(X) **성실하면**(O) 이기는 싸움입니다!
- 물론 본인이 빌 게이츠나, 워즈니악, 리누스 토르발즈, 존 카멕 같은 천재라면 예외죠 ㅎㅎㅎ 😊





# 코테는 성실성의 증명이다!

- 코테는 1등을 뽑는 해커톤 또는 올림피아드가 아닙니다! → 특정 수준 이상의 지원자를 뽑기 위한 hurdles이죠!
- 따라서, 틀린 문제에 대한 오답 노트 준비 + 맞춘 문제는 모범 답안 정리를 하면서 성실히 수행하면 코테를 통과할 레벨은 이룰 수 있습니다!
- 사실 답안 이해가 안가도 특정 패턴에 대한 답안을 외워도 되기 때문에, 너무 걱정 마시고 성실하게 준비 하시면 됩니다!



# 기술 면접!



# 주요 팁!

- 모든 질문은 두괄식 대답 + 핵심 키워드 언급하기!
  - ex. 해당 문제 풀이를 위해서는 Greedy 알고리즘으로 접근 하면 될 것 같습니다  
→ 그 뒤에 부연 설명
  - 여러분이 상대하는 사람이 개발자임을 잊지 마세요 😊
- 너무 외운 티를 내기 보다는 자신 만의 언어로 자신감 있게 개념을 요약 정리 할 것!(외운 티가 나면 마이너스!)



# 주요 팁!

- 한국어 보다는 영어 이름 쓰기
  - 외래 키 (X) / Foreign Key (O)
  - 면접관들은 아무래도 예전에 영어로 된 단어로 배웠을 확률이 높습니다 + 사대 주의 😊
- 공부할 분야를 나누고 해당 분야의 핵심 키워드를 정리 후 암기 → 그리고 그 핵심 키워드를 토대로 설명하는 연습을 하시면 됩니다!



주제	키워드	설명
TABLE 정규화	정보 무손실, 중복 제거, 분리 원칙, 1NF, 2NF, 3NF	이상 현상을 발생 시키는 속성간의 중복성, 중복성을 제거하 고 데이터의 무결성을 보장하기 위해 Table 을 나누는 과정



# 자기 소개서, 포폴 내용 숙지는 기본

- 자소서랑 포폴 내용 숙지 안하고 가면? 바로 탈락입니다!
- 자기 소개서의 내용이랑 포폴에서 사용한 기술 스택에 대한 이해는 반드시 하고 가야 합니다!



# 면접 질문을 모르면!?

- 일단 아는 부분 까지는 답변을 하세요! 면접관 자체가 다 알고 있을 거라는 기대가 없을 겁니다!
- 대신, 모르는 것을 아는척 하는 것은 절대 피하셔야 합니다!
- 회사에서 신입으로 키울 예정인데, 모르는 부분을 인정하지 않는다면 성장 가능성이 적다고 판단하기 때문에 반드시 피하셔야 합니다!



# 인간성!?





# 개발자 인간성이 꼭 필요한가요?

- 여러분도 다양한 인간 관계를 경험해 보셨겠지만, 회사도 결국 사람이 모여 있는 곳입니다!
- 그렇기 때문에 좋은 인간성을 어필하는 부분은 상당히 중요합니다!
- 그리고 규모가 작은 회사 일 수록, 흔히 말하는 못된 사람을 뽑아서 개발 팀의 분위기가 저해되고 잘하는 사람이 퇴사하는 것을 극도로 꺼리기 때문에 이러한 부분은 매우 중요합니다!



수고하셨습니다!