

week.3

비전공자가 개발자가 되기까지

왜 개발자라는 직업을 택했는가?

- 다른 직무에 비해 자유도가 높은 편
 - 개발 스택을 정하고, 무엇을 개선할지, 때론 무엇을 만들지 고민할 수 있고...
- 내가 하는 만큼 나에게 쌓이는 직업
 - 공부한 것들 어디 안간다고 생각
- 아직도 자유로운 편

부트캠프 시절

- 무식하게 공부
- 시간 투자가 중요하다고 생각해서 4~5개월간 새벽 2~3시에 취침, 기상 오전 7시
- 스타트업에 지원했는지라 취준 기간은 1달로 짧았음

부트캠프 기간 동안

챙긴 것

- 함께 공부한 동료
- CS 기초 지식(http, db)
- 프로젝트

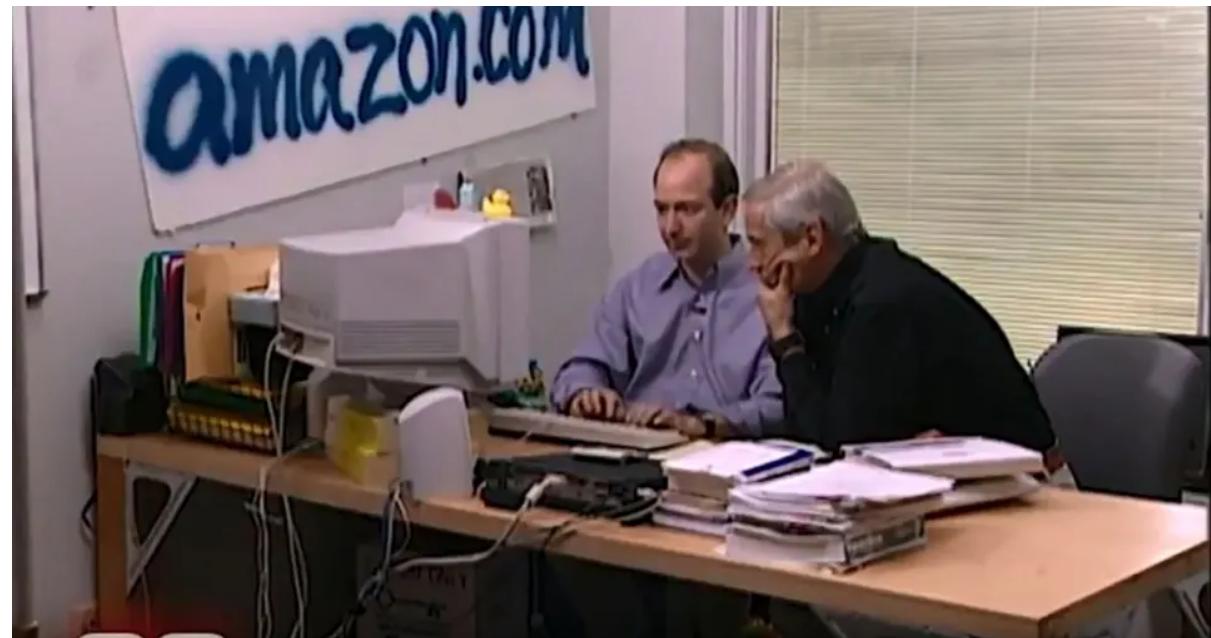
못 챙긴 것

- 알고리즘
- 약간의 건강

제 커리어를 간단히 말씀드리자면...

신생 스타트업(8개월) - 혼자 살아 남는 방법을 터득

- 3명일때 입사, 5명이 되고 회사가 망함
(투자를 못받음)
- git repo부터 시작해서 user site, admin site Front-end를 혼자 담당
 - 여기서 webpack, react 등 혼자 다하면서 이것저것 많이 알게 됨
 - 혼자 공부를 많이 했고, 혼자 공부하는 법을 스스로 좀 터득하게 됨
- 하지만 회사가 망하고 나서, 직업의 안정이 주는 소중함을 알게 됨



중소기업(2년) - FE가 큰 서비스에서 많은 것들을 개선 & 교육 & 리더

- 200여 명 규모의 탄탄한 중소기업
- 한창 성장기를 함께 경험
 - 입사 초반: 50여 명
 - 퇴사 시기: 200여 명
- Front-end application이 커서 많은 것을 배움(FE 개발자만 20여 명)
- 2년동안 인턴 6명의 온보딩을 담당, 한 파트(feature)의 리더로 일 함
 - 2시간 동안 React 강의함..
- 2년 후, 현 회사로 이직
 - 비즈니스 규모가 큰 회사에서 경험을 쌓아 보고 싶다.

대기업(1년 다 되감) - 다시 막내급...

- 대기업 복지와 여유를 누리며 살아가는 중...
- 요즘 관심사는 `unit test` 와 `functional programming`

취업 준비는 어떻게 해야하나

제 경험에서만 나온 이야기기 때문에 적당히 들어주시면 감사하겠습니다.

오늘도 개발자가 **안 된다고** 말했다

개발 협업이 어려운
기획자&디자이너를 위한
필수 가이드북

김중철·김수지 저

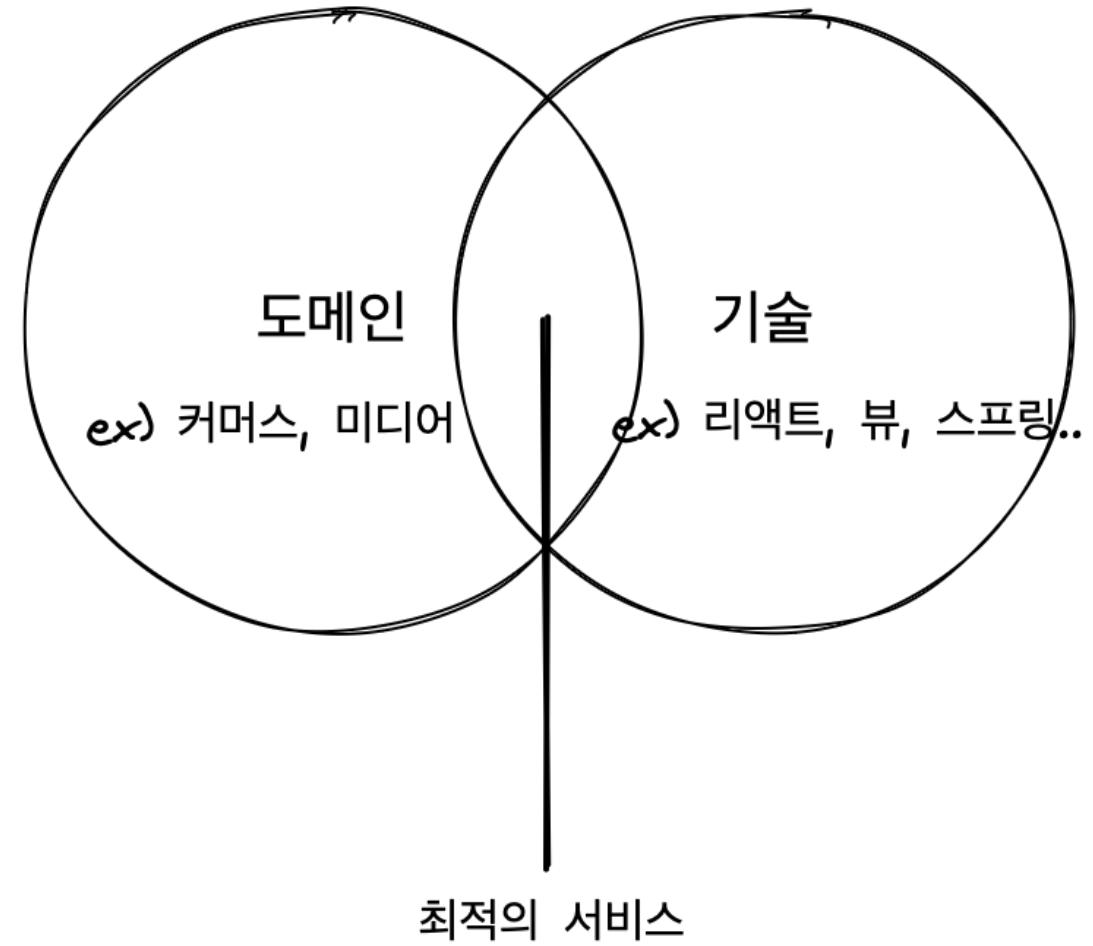
'개발자'는 회사를 다니는 이상
회사 입장에서 비즈니스적으로
필요한 것(요구사항)을 기한 내에 해결해줘
야하는 직무

- 해결: 새로운 기능, 버그 수정, 기능 유
지/보수 등



1. 가고 싶은 회사 정하기

- 도메인(서비스 분야)
 - ex) 커머스, 미디어(웹툰, ott)...
- 기술(stack)
 - ex) 리액트, 뷰, 스프링 등등..
- 연봉, 복지 등등..
- 남들이 뭐라고 해도, 자기 기준 좋은 회사가 좋은 회사임
- 커리어가 쌓일 수록 달라질 수 있음!



2. 그 회사가 무엇을 원하는지/하는지 살펴보기

- 네이버 테크 직군 신입 채용
- [크리마 (크리마팩토리, 크리마랩)] Junior Software Engineer (신입가능) 채용 | 원티드
- 토스채용

서류전형

- 사람들마다 갖고 있는 것들이 다르니 가장 설명하기가 어려움
- 오탈자/가독성 등 기본은 지키자
- 서류전형 통과 후에는 면접에서 활용됨
- 처음에 서류 작성이 힘들다면 주변에 개발자 친구들 혹은 커리어 코치들의 도움을 받는 것도 괜찮음
 - 피드백/코칭을 받으면서 제일 중요한 것은 자기가 뭘 말하고 싶은지는 명확해야함

알고리즘

- 이론 공부 후, 혼자 많이 풀어보는 수 밖에...
- 수능 수학 풀듯이 풀고, 오답노트, 풀고, 오답노트의 반복
- 강의, 스터디 혹은 주변에 잘하는 친구가 있다면 풀이법 공유
- (저한테는) 제일 재미 없고, 성장이 더딘 영역...

과제전형

- 신입한테도 할지 모르겠지만...
- 평소에 코딩 많이 해보고, 구조 많이 짜봐야 바로 잘 할 수 있음
- 갑자기 잘하긴 힘듬... 정답이 없어서 겁 먹을 필요는 없음

1차 면접(실무 면접, 기술 면접)

- (학부 생일 경우) DB, OS, Network 등 컴퓨터 공학 지식을 물어봄
 - 그래도 취업을 떠나서 틈틈이 기초 지식은 공부하는 게 좋습니다!
- 비전공인걸 감안하고 서류를 통과시킨 거라면 프로젝트, Front-end 기초 지식을 물어봄
- 공부를 해서 들어가야하는 면접
- 가장 중요한 것은 내가 이력서에 작성한 기술들에 대해서는 이야기할 수 있어야함

저는 React, scss(sass), webpack, javascript를
활용하여 SPA 서비스인 맛집 지도를 개발한 경험이 있습니다.

저는 React, scss(sass), webpack, javascript를 활용하여 SPA 서비스인 맛집 지도를 개발한
경험이 있습니다.

- React는 어떤 장점이 있나요?
 - React 단점은 없나요? / state와 props는 무엇인가요? / 다른 프레임워크는 사용해봤나요?
/ virtual DOM은 뭔가요? / 기존 방식들은 무엇이 문제였을까요?
- scss는 무엇이죠?
 - css랑 뭐가 다르나요? / trade-off도 있을텐데요?
- webpack은 왜 필요하죠?
 - 직접 설정해본 경험은 있나요? / babel은 아시나요?

1차 면접은 꼬리 물기 질문이 많이 들어옴

- 완벽히 모든 것을 방어할 순 없지만 그래도 공부를 많이 해서 대답을 잘하길 바람
- 1 depth 대답을 못하면, 혹은 정확한 대답을 못하면 모른다고 생각을 함
- 차라리 진짜 모르면 '모릅니다.'라고 이야기를 하는 게 맞음
 - 신입이고, 모를 수도 있지...
 - 면접관이 진짜 궁금해서(몰라서) 물어보는 경우도 있음

1차 면접은 '답이 있는 질문'과 '없는 질문'으로 나눠져 있음

- (어느 정도) 답이 있는 질문
 - 프론트엔드 면접 질문 치면 자료 많이 나옴
 - state와 props는 무엇인가요?
 - react hook은 무엇인가요? 언제 사용하나요?
 - 신입 기준에서 기술적인 질문들은 답이 어느 정도 있다고 봄
 - 경력직은 또 다름

답이 있는 질문에 대해 대답을 잘하기 위해서
공부를 열심히 함.

그리고 저는 제가 사용하는 용어가 입에 붙을 수 있게
샤워할때마다 말하는 연습을 함.

- 답이 없는 질문

- 혹시 ~~~한 이견차이가 발생하면 어떤 식으로 해결했나요?
- 코드리뷰 자주 하셨어요? 기분이 어떠셨어요?
- 혹시 일하다가 가장 스트레스 받을 때는 언제에요? 어떻게 해소하셨어요?
- 우리 서비스 뭐가 좋았어요? 왜 여기서 일하고 싶나요?
- 어떤 사람이 가장 잘맞고, 어떤 사람이랑 좀 일하기 싫었나요?
- 프로젝트하다가 힘든 건 없었나요?

답이 없는 질문에 대한 대답을 잘하기 위해
저는 질문에 대한 준비를 하지 않음.

평소에 자기 기준과 생각 정리를 잘해둠.
질문에 대해 느낀대로 말함.

1차 면접은 결국,

우리 팀에 들어와서 적응 잘할 수 있는

앞으로 몇 년 뒤 잘 성장할 수 있는, 성장 가능성 있는 사람을 '현재 기준'으로 판단

2차 면접(인성 면접, 임원 면접, 대표 면접)

- 캐주얼한 질문들이 많이 들어옴
 - 평소에 우리 회사 이미지 어떤가요?
 - 우리 회사 서비스 많이 활용하나요?
 - 쇼핑은 많이 하나요?
- 적당히 솔직하게 대답하는 게 중요한 면접
- 정말 가고 싶은 회사라면 회사 조사 많이 해가면 플러스 점수가 될 수 밖에 없음
- 오너, 임원 입장에서 우리 회사에 큰 관심을 주는 건 매우 기쁜 일
- 약간의 패기(?), 무리수는 안됨

2차 면접은 결국,

일을 잘할 거라고 실무진(1차 면접)에서 통과시켰으니

정말 우리 회사랑 잘 맞을까? 우리 회사에 적응을 잘할 수 있을까?

흔히들 말하는 컬쳐핏을 봄

(하지만 그렇다고 ~~인이 될 필요 절대 없음)



Add cover

Add comment

Interview Review

sk 엔카

마켓컬리

이마트

무신사 신사업

라포랩스

ns 홈쇼핑

야놀자

위메프

데일리 호텔

인터파크

네이버 파이낸셜(네이버 페이)

카카오 엔터테인먼트

카카오 스타일

롯데e커머스(롯데ON)

카카오 면접

면접 후,

- 면접은 잘본 면접이든 아닌 면접이든 무조건 복기
 - 1차 면접에서 대답 못한 질문들 다시 찾아 보기
 - 면접에서 반응이 좋았던 답, 안좋았던 답에 대해 복기

기업 유형에 따라 조금씩 다른 부분

스타트업

- '당장 1인분은 했으면 좋겠다...' 라는 약간의 바램은 있어 보임
- 성향상 조금 더 능동적이고, 스스로 학습을 잘했으면 좋겠음
- 대신 이것저것 해볼 수 있는 기회는 많음
- 사수가 없을 수도 있음(이건 득실 판단이 불가함)
 - 사수가 내가 기대한 사수가 아니라면?

대기업

- 입사 후, 몇 년 뒤를 바라보고 신입사원 채용(포텐을 본다)
- 아무래도 위에 많은 선배 개발자와 절차가 있어 업무 자유도가 권한은 좀 적을 수 있음
 - 심지어 일을 작게 작게 나눠서 함
 - ex) 검색 팀(검색 FE, BE), 추천 팀(추천 FE, BE), 상품 팀, 인프라... 등등..
- 같은 일을 하는 사람이 많음
 - 같은 소리를 낼 수 있는 사람이 많음
- 어쨌거나 안정적임

중소기업

- 대기업 + 스타트업
- 스타트업보다 안정적인 건 장점
- 복지, 처우가 애매할 가능성이 높음

기업 평판(후기) 알아보기

- 잡플래닛
- 크레딧잡
- 블라인드
 - 규모가 작은 중소기업, 스타트업은 얼추 맞음
 - 대기업은 진리의 부바부(부서 by 부서)
 - 삼성전자 그렇다던데? 응, 우리 회사 임직원 10만 명
 - 전체적인 분위기만 보면 되지, 디테일을 따질 필요 없음
 - 퇴사율이 높으면 눈 여겨 봐야함
 - 스타트업은 어쩔 수 없음

좋은 회사란, 결국 내가 좋다고 생각하는 회사

그럼에도... 저는

1. 면접 과정과 분위기

- i. 함께 일할 가능성이 높은 사람들인데...
- ii. 초면부터...

2. 기술 스택과 개발 문화

- i. 개발자에겐 최고의 복지 중 하나

3. 연봉과 복지

- i. 절대 간과해서는 안되는 요소

4. 채용 공고를 꼼꼼히 작성하여 올리는 회사

- i. 초면부터...

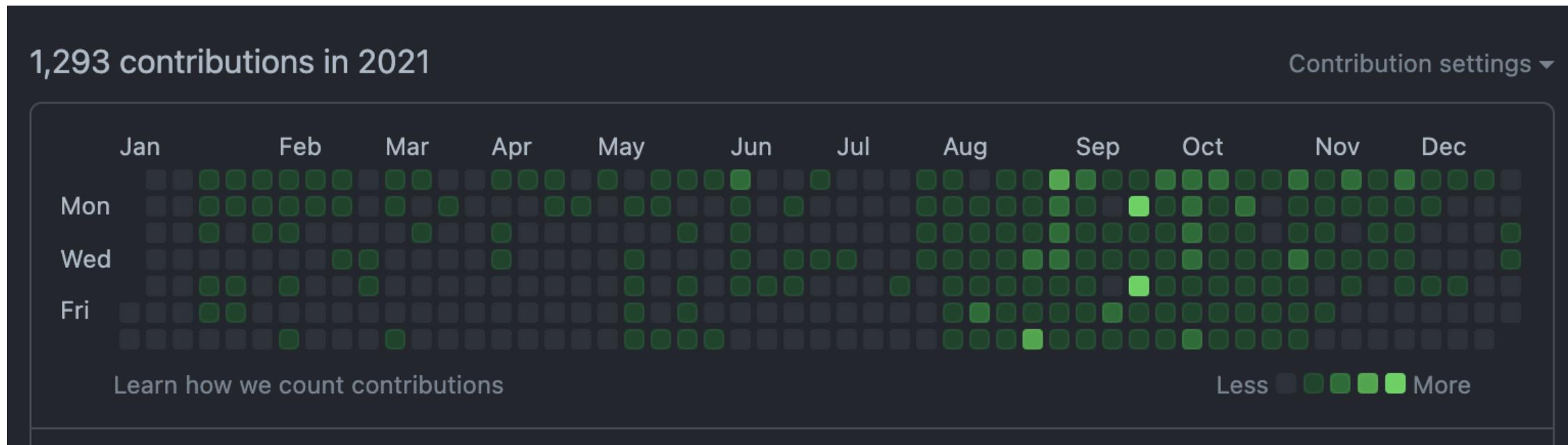
내가 해왔던 것

1. 자기 기준 명확히 하기

- 기준은 시기마다 달라질 수 있음
 - ex) 연봉 3000만원 이하면 지원 안한다/안간다.
 - ex) CTO만 있으면 일단 입사한다.
 - ex) 개발 문화 모르겠고, 회사 네임밸류 좋으면 일단 입사한다.
 - 등등..

2. 꾸준히, 스스로 성장하기

- 새로 알게 된 것 기록하기
 - 대신 기록을 위한 기록은 하지 않기
 - 알고리즘 꾸준히 풀기 (~~요즘은 안함...~~)



3. 면접 있으면 무조건 가기!

- 경험만큼 중요한 건 없음
- 면접에 많이 들어갈수록 면접에 능숙해진 자신을 발견할 수 있음
 - 말도 잘하게 됨
 - 당황도 안함
 - 어디서 들어본 질문임



4. 채용 공고 면밀히 보기

- 지원하고자 하는 회사의 첫 이미지
- 좋아하는 키워드와 싫어하는 키워드가 있는지 살펴보기
- 회사의 근황은 잘 기재되어 있는지

5. 건강한 몸, 건강한 멘탈 관리

- 건승하기!
- 나만 손해는 아니다! 훌륭한 인재를 알아보지 못하다니!!!
 - 채용 합/불은 굉장히 복합적인 이유
 - 기술적으로 A가 뛰어나도, 함께 일하고 싶은 사람은 B였다.
- 면접 복기

임원면접일정안내
자세한 사항은
이메일 참고 바랍니다.
(일정 조정 불가)

(오늘) 오전 9:30

[Web발신]
좋은소식을 전해드리지 못하게되었습니다. 앞으로도 건승하시길 기원합니다.

Q&A

짧은 시간에 말로 하려고 하니 내용이 많아 보이지만
여러분들이 채용 절차를 밟다보면 체득할 수 있는 (당연한) 내용입니다.
모두 화이팅입니다!