

# 프론트엔드 개발자 취업 & 이직 팁

## 1 과제와 기술 면접 대비

## 만남을 위한 과정

채용공고 > 지원 > 서류 검토 > 코딩테스트 or 과제 > 면접 > 협상 > 채용

회사          나          회사          나                                  함께          함께          함께

코딩 테스트를 준비해야하는 경우

leetcode, programmers 등 유명한 사이트들이 많이 있다.

코딩 테스트에 자신이 없다면?

불행중 다행

프론트엔드 개발자를 뽑을 때

코딩 테스트를 보는 경우는 많지 않다.

하지만 기술 면접 등에서 질문을 받을 때를 대비해서

스스로 부족하다고 생각하면 보강해두는 것 추천

결국,

과제와 기술 면접 준비가 핵심

과제가 없는 경우,

좋을 수도 있지만,

오히려 기술 면접시에 이야기 거리가 별로 없어서,

원론적인 내용을 다루는데 시간을 많이 할애하게 될 수도 있다.

컴퓨터 공학의 **기초 지식**을 물어볼 수도..

과제가 있는 경우,

아마도 실무와 많이 연관이 있는 주제일 것  
과제를 통해 지금의 **내 실력을 파악**해볼 수도 있을 것

여러 회사의 과제를 경험해보면,  
**유사한 부분**을 검증하려고 한다고 느낄 수 있을 수도

과제는 **요구사항**을 주고,  
**제한된 시간**동안 구현하는 것이 일반적

과제를 할때,  
굳이 안써본 기술을 활용하는 것은 비추  
**익숙한 기술**로 요구사항을 구현하는 것을 추천

만약 사용할 기술을 전제한다면 해당 기술의 간단한 기능 위주로 활용



과제를 제시간에 하지 못한 경우,

과제를 하면서 궁금한 점이 생기는 경우,

물어볼 수 있는 기회를 충분히 활용하시길...

회사도 과제를 통해 나를 검증하고자 한다.

잘못된 커뮤니케이션으로 나를 온전히 검증하지 못하는 회사도 손해  
다.

실무형 과제는

결국 연습 / 포트폴리오를 만들면서 **경험했던 내용**들이 주를 이룰 것

API 요청 / 상태 관리 / 검색 기능 구현 / 컴포넌트 설계 등...

## 실무형 과제를 출제하는 이유?

- 실제로 실무를 할 수 있는 역량을 확인
- 기술 면접에서 지원자의 역량을 파악해볼 수 있는 기본 정보

=> 구현하지 못한 부분이 있다면, 기술 면접 이전에 꼭 **보강**해서 가자

요구사항을 구현하는 것이 최우선이다.

그 외의 어필은 어디까지는 + @ 일 뿐이다.

과제를 제출한 이후 기술 면접을 진행할때까지의 시간을 활용해도 된다.

과제를 하면서  
알게되었거나 부족했던 부분을  
스스로 채워가는 사람인가?

이부분에서 합격점을 받는다면 큰 가점이 될 수 있다.

## 기술 면접

프론트엔드 로드맵

(<https://roadmap.sh/frontend>)

리액트 로드맵

(<https://roadmap.sh/react>)

구글링으로

프론트엔드 **기술 면접 대비** 질문 정리

꽤나 많은 양질의 정보들이 나온다.(크로스 체크 필수)

이 질문들에 대한 대답을 준비하면 된다.

답을 알고 있는 것보다 더 중요한 것,

그 답을 외운 것이 아니라

**이해**하고 있는지?

이해를 못하고 있더라도 그 **맥락**은 아는지!



기술적인 문제를 만났을때,

어떻게 대처할 수 있는지?

- 이미 그 문제의 해답을 알고 있는 경우,
- 그 문제의 키포인트를 아는 경우,
- 그 문제의 맥락을 아는 경우,
- 해결책을 검색해낼 수 있는 경우,

전혀 알 수 없는 질문을 받았을때?

주어진 내용들을 활용해서  
합리적인 답을 유추해보자.

면접관의 **질문의 의도**를 파악하는 것이 중요하다.

질문을 한 의도는 여러가지 일 수 있음

- 집요한 꼬리질문으로 정말 해당 내용을 잘 아는게 맞는지 확인
- 낯선 문제에 대한 접근법 확인
- 앞에서 잘못된 답을 정정할 수 있는 기회 제공
- 당사자가 고민하고 있는 부분에 대한 의견 수렴

## 과제와 관련된 질문

코드 리뷰를 하는 경우도 있을 것이고,  
그렇지 않더라도 과제에 대한 질문이 있을 수 있다.

기술 면접을 진행하기 전에  
과제의 내용을 훑어보고 질문의 대상이 될 수 있는 코드를 **미리 파악**하  
고  
가상의 질문에 대한 답을 **미리 대비**해두자

과제와 관련된 질문을 시작하는 타이밍에

스스로 준비해왔던 아쉬웠던 부분에 대한 보충을  
먼저 이야기하거나 다루면 좋다.

면접관이 과제와 관련해서 질문할 거리를 미리 인지하고 있다는 사실  
만으로도

신뢰를 얻을 수 있다.

다만, 그 문제에 대한 나름의 해결책을 가져가는 것이 중요

대답을 하지 못하는 질문이 나오거나  
뭔가 잘못 대답을 했다는 생각이 들 때는

역으로 면접관에게 물어보자

부족한 부분을 배울 수 있고,  
운이 좋다면 성장가능성을 어필할 수도 있다.

과제 혹은 기술 면접  
탈락하더라도

피드백을 요청하자

실제로 만나서 피드백을 받을 수도 있고,  
그게 불가능하다면 이메일로라도 나의 부족한 부분을 확인하는 것  
그 자체로 가치있고, 다음 기회를 준비할 수 있다.

기술 면접의 과정을

녹음해보고 내가 대답하는 방식에 대해서  
회고해보자

녹음을 들어보면서 내가 대답하지 못했던 부분에 대해서  
보강하고 다음 기회가 온다면 잡아내자



## 블로그를 활용하자

기술 면접을 하면서 알게된 내용을 정리해두고,  
면접 준비를 하면서 알게된 부분들도 정리해두자

채용/이직 프로세스를 회고하면서 나의 부족을 기록하고 채워보자

## 정리

### 과제와 기술 면접 대비

실무형 과제

요구사항 구현에 집중

부족한 과제

기술 면접 기회를 얻는다면 꼭 보강

기술 면접 질문

이해를 베이스로 한 대비

꿀팁

피드백 요청 / 녹음분 회고