

디자인~개발자 면접 질문 모음집입니다.

대체로 이런걸 물어보겠구나 느낌을 보시고 비슷한 질문이 있고 중복된 질문이 있을 수 있어 참고해 주시길 바랍니다.

[인성 및 경험 관련 질문]

- 자기소개
- 공백기에 대한 질문
- 살아가면서 가장 어려웠던 일과 해결 방법
- 자신을 뽐냄으로써 어떤 이득 줄 수 있는지
- 자신의 강점과 약점은 무엇인가?(약점에 대한 보완을 얼마나 성실하고 지속적으로 하고 있는지에 대한 내용이 중요함)
- (5인 미만 회사)퇴사 안 하고 오랜시간 같이 열심히 할 자신 있는가
- 우리 회사가 뭐하는 회사인지
- 국비 교육과정으로 무엇을 배웠는지
- (코드리뷰하는 회사)주니어로서 선배에게 어떤 조언이나 의견을 주로 낼까라 예상하는가
- 학원 공부 외에 얼마나 노력했는지
- 이력서 기반 질문(전직이유)
- 가장 좋았던/나빴던 프로젝트가 무엇이었으며 그 이유는 각각 무엇인지
- 성향이 느린편인지 빠른편인지
- 어떤 것을 배울 때 습득이 빠른 편인지
- 현업에서는 소통이 중요한데 자신의 성격이 어떤 것 같은지
- 프론트엔드/백엔드 개발이 성향에 맞는 것 같은지
- 회사에서 많이 이야기하는 성향인지(또는 내향/외향?)
- 깃허브에 올린 내용은 본인이 다 만든건지?
- 테스트 코드를 작성한 적이 있는지
- 본인이 이 회사에서 더 성장할 수 있을거라 생각하는지
- 회사에서 안좋은 일이 생기면, 스트레스 받는 상황이 되면 어떻게 푸는지
- 개발을 할 때 중요하다고 생각하는 3가지 말씀해주세요.
- 입사를 해서 와서 물어보면 알려줄수 있지만 누군가 나서서 알려주지 않을수도 있는데 괜찮은지?
- 새로운 지식을 익혀야 할 때 보통 어떤 방법을 사용하는지
- (프론트엔드에 지원했는데 포폴이 약할 경우) 퍼블리싱 포트폴리오로 보이는데 프론트엔드쪽을 어느정도 할 수 있는지
- 주력 언어는 무엇인지
- 기록을 남기기 위한 프로그램은 어떤걸 사용해봤나요?
- 가장 즐겁게 임했던 전공 활동은?
- 학부 강의 중 가장 기억에 남는 걸 설명해보기
- 이전 회사에서 개인적 성장에 도움이 된 점은 무엇이고 그 이유는?
- 주도적으로 뭔가를 이뤄본 경험은 어떤 것이 있는지?
- 입사한다면 이루고 싶은 것?
- 그간의 경험이 새 업무에서 어떻게 활용될 수 있을지?
- 면접관이라면 어떤 부분을 가장 중점적으로 보시겠습니까?
- 그 부분을 고려했을때 본인은 100점 만점에 몇점이라고 보시나요?

[공통 질문]

개발자로서의 역량과 가치관

해당 과정을 학습하면서 프론트/백엔드 개발자가 가져야할 가장 필수요건은 뭐라고 생각하나요?

본인이 생각하기에 좋은 개발자란 어떤 사람이라고 생각하는 지

플랫폼 회사가 무엇이라고 생각하나요?

프로그래밍 학습 경험

비전공자로서 어떻게 프로그래밍을 시작했나요?

어떤 언어로 시작했으며, 현재 주로 어떤 언어를 사용하고 있나요?

자기 학습 능력

프론트엔드/백엔드 개발에 어쩌다 관심을 가지게 되었나요?

어떤 학습 자료나 온라인 강의를 활용하여 프론트엔드/백엔드 개발을 스스로 학습한 경험이 있나요?

프로젝트 설계와 구현

큰 규모의 프로젝트를 시작할 때 어떤 계획과 설계과정을 거치나요?

복잡한 기능을 구현할 때, 구현 전에 어떤 고민을 하고 있는지 설명해보세요.

프로젝트 경험과 포트폴리오

어떠한 개인 또는 팀 프로젝트를 진행해봤나요? 어떤 프로젝트를 진행했고 사용한 기술 스택과 개발 과정에 대해 언급해주세요. 특히 자랑하고 싶은 부분이 있다면 무엇인가요?

포트폴리오 관련해 가장 중점적으로 고민해보고 주도적으로 만들어 본 프로젝트 한 개와 기능을 설명해주세요.

프로젝트 중 가장 도전적이었던 부분은 무엇이었나요?

협업 프로젝트에서의 역할과 기여에 대해 언급하세요.

프로젝트에서 발생한 어려움이나 돌파구를 어떻게 다뤘는지에 대해 이야기하세요.

팀 프로젝트에서 협업하는 방식에 대해 어떤 경험이 있나요?

프로젝트 일정이 지연되었을 때, 이를 어떻게 다뤘는지 설명하세요.

이 프로젝트의 폴더관리는 왜 이렇게 한건지 설명

문제해결 능력과 커뮤니케이션, 협업

어떤 복잡한 문제를 마주했을 때, 어떻게 접근하고 해결했는지 예시를 들어 설명해주세요.

개발 프로젝트에서 팀원들과 원활한 의사소통을 유지하는 데 어떤 노력을 기울였나요?

팀에서 의견이 충돌했을 때 어떻게 해결했는지에 대해 언급해주세요.

프로젝트에서 팀원과의 협업에서 발생한 어려움에 대해 이야기해보세요.

어떤 상황에서 어떤 디버깅 도구를 사용하나요?

프로젝트에서 발생한 에러를 어떻게 디버깅하고 해결했는지에 대한 경험을 공유해보세요.

코드에서 버그를 찾는 방법과 해결하는 과정에 대해 설명하세요.

팀 내에서 기술적인 아이디어를 효과적으로 전달하는 방법에 대해 이야기해보세요.

팀 프로젝트나 협업 경험 중에서 가장 도전적이었던 상황에 대해 이야기해보세요.

코드 리뷰와 피드백

자신이 작성한 코드를 다른 사람이 리뷰할 때 어떤 피드백을 받았나요? 그 피드백을 어떻게 활용했나요?(코드 리뷰에서 받은 피드백을 어떻게 다뤘는지 설명해주세요.)

코드 리뷰에서 가장 중요하게 생각하는 점은 무엇인가요?

포트폴리오에서 사용한 코드에 대해 왜 사용했는지와 코드 리뷰 해 주세요.

기술적인 도전과 극복

프로그래밍 공부 중 어려움을 겪었던 경험이 있다면 어떤 것이었나요? 어떻게 극복했나요?

새로운 기술이나 개념을 처음 접할 때 겪는 어려움에 대해서 어떻게 대처하는지 설명해보세요.

비전공자로서 프론트엔드/백엔드 개발을 시작하면서 가장 큰 기술적 도전은 무엇이었나요? 어떻게 어려움을 이겨냈나요?

기술적인 어려움에 부딪혔을 때, 자신을 격려하거나 도와준 자원이 있었나요?

개발자 커뮤니티 및 네트워킹

어떻게 개발자 커뮤니티에 참여하게 되었나요? 어떤 활동을 했나요?

개발자 커뮤니티에서 얻은 경험이나 도움이 되었던 에피소드가 있다면 공유해주세요.

개발 도구

코드 에디터로 어떤 것을 사용하고, 사용하는 이유는 무엇인가요?

자주 사용하는 개발 도구 및 확장 프로그램이 있나요?

코드 버전 관리

Git을 사용하는 이유는 무엇이며, 기본적인 Git 명령어를 나열해보세요.

최근 기술적 도전 혹은 프로젝트, 적응력과 학습 능력

최근에 도전한 기술적인 과제나 프로젝트에 대해 이야기해보세요.

새로운 기술이나 도구를 학습할 때 어떤 방식을 사용하나요?

그 도전에서 얻은 교훈이나 성취감에 대해 언급해주세요.

비전공자로 시작했지만, 다른 개발 분야나 기술에 대한 호기심이 있나요?

지금까지 프론트엔드 외에 다른 기술 스택에 대한 공부나 실험을 해본 경험이 있나요?

어떤 방법으로 최신 기술 동향을 계속해서 따라갈 수 있을까요?

개발자로서 성장 방향

앞으로 어떤 기술이나 분야에 대해 더 깊이 공부하고 싶으신가요?

자신이 개발자로서 어떤 방향으로 성장하고자 하는지에 대해 이야기해보세요.

비전공자의 강점

비전공자로서의 강점이 무엇이라고 생각하나요? 어떤 면에서 다른 개발자들과 차별화되어 있을까요?

직무 이해와 열정

프론트엔드/백엔드 개발자로서의 역할에 대한 정확한 이해가 있나요? 어떤 부분에서 가장 열정적이십니까?

프론트엔드와 백엔드 간의 협업 방식에 대한 생각을 말해보세요.

이 회사에서 일하고 싶은 이유는 무엇인가요?

이 회사에서 프론트엔드/백엔드 개발자로 일하고 싶어하는 이유와, 여러 지원자 중에서 자신을 강조하고 싶은 강점은 무엇인가요?

질문 및 호기심

면접관에게 질문을 하고, 회사에 대한 궁금증을 표현하세요.

디자인 면접 질문

비전공자라면 전공자가 아닌데 왜 이 일이 하고싶은지?

디자인 관점과 가치관

가장 좋아하는 브랜드와 그 이유는?

브랜드 디자인이란 뭐라고 생각하는지?

좋은 디자인과 나쁜 디자인은 무엇인가?

본인이 생각하는 좋은 디자인이란 무엇인가

디자인 할때 제일 중요하게 생각하는 것

클라이언트가 선호하는 디자인이 뭐라고 생각하는가

가장 많이 참고하는 사이트는?

회사가 원하는 디자인은 어떤 방향이라고 생각하나요

디자인 할 때 제일 자주 사용하는 색상은 무엇인지? 이유는 뭔지?

제일 좋아하는 폰트는 무엇인지

디자인 작업물에 대한 설명

본인 포트폴리오에서 가장 마음에 드는 작업물 설명해보세요

영어로 포트폴리오에 있는 프로젝트중 하나를 발표해 보세요

업무 방식이나 극복 방법

다수의 동료가 안맞는 의견을 제시하면 어떻게 하실건가요?

프로젝트 중 직면하신 과제에 대해 하나 말씀해 줄수있나요? 어떻게 극복하셨나요?

Front-end 개발자 면접 질문 [비전공자 신입~전공자 신입]

사이트 추천:

[현직 개발자가 정리해주는 프론트엔드 기술 면접 필수 질문, 답변 20선 \(feat. 전 카카오 엔터프라이즈 개발자\) | zero-base](#)

HTML과 CSS

HTML5와 CSS3에서 어떤 새로운 기능이 추가되었는지 설명해보세요.

시맨틱 HTML이 무엇이며 왜 중요한가요?

HTML 문서가 브라우저에서 어떻게 파싱되는지에 대한 간단한 설명을 해보세요.

CSS의 박스 모델에 대해 설명하세요.

HTML 태그 중 어떤 것들이 페이지의 성능에 영향을 미칠 수 있나요?

CSS에서 margin과 padding의 차이는 무엇인가요?

CSS의 display 속성의 각 특징들을 설명하세요.

JavaScript

변수와 상수의 차이는 무엇인가요?

클로저(Closure)가 무엇이며 어떻게 동작하는지 설명하세요.

- 함수가 선언될 때 자동으로 생성되는 렉시컬 환경에 대한 설명입니다. 이러한 렉시컬 환경은 스코프 체인(scope chain)을 형성하게 되는데, 스코프 체인은 함수가 선언될 때의 모든 변수와 함수를 포함하는 렉시컬 스코프(lexical scope)를 형성합니다. 외부 함수가 실행 되고 반환된 후에도 외부 함수의 범위 내의 함수에 체이닝을 할 수 있는 함수 입니다. 정보를 은닉하기 위해서 주로 사용 합니다.

→매우 중요한 개념이므로 꼭 이해하고 면접에 임하세요!

이벤트 위임(Event Delegation)에 대해 설명하세요.

let, const, var의 차이는 무엇인가요?

JavaScript에서 동기적(Synchronous)과 비동기적(Asynchronous) 코드의 차이를 설명하세요.

자바스크립트 ES5와 ES6의 차이점을 설명하세요.

ES6문법을 지원하지 않는 인터넷 브라우저 등에서 ES6문법을 사용하기 위해 어떤 방법을 사용해야 하는지

브라우저 렌더링

브라우저 렌더링 원리에 대해서 설명해보세요.

- 브라우저 렌더링은 HTML, CSS, JavaScript 등의 웹 페이지 자원을 브라우저가 화면에 그리는 과정을 말합니다. 브라우저 렌더링 원리와 순서는 크게 다음과 같은 단계로 구성됩니다.

먼저 DOM을 생성합니다.(DOM설명 바로가기) 브라우저는 HTML 문서를 파싱하여 DOM 트리를 생성합니다.

이때, HTML 태그를 노드로 변환하고, 노드간의 계층 관계를 형성합니다.

두 번째로 CSSOM을 생성합니다. 브라우저는 CSS 파일을 파싱하여 CSSOM 트리를 생성합니다. 이때, CSS 속성을 노드로 변환하고, 노드간의 계층 관계를 형성합니다.

세 번째로 DOM트리와 CSSOM을 결합하여 렌더 트리를 생성 합니다. 이때, 실제 화면에 표시될 요소만을 선택하여 렌더 트리를 형성합니다.

이제, 브라우저는 렌더 트리를 이용하여 각 요소의 크기와 위치를 계산하는 과정인 레이아웃을 거쳐 화면에 요소를 그리는 페인팅 과정을 거치게 됩니다. 이때, 요소의 배경, 테두리, 글자 등을 그리게 됩니다.

→전체적인 과정을 머릿 속에 그려놓고, 꼭 위와 같이 설명하지 않아도 흐름을 대략적으로 설명하실 수 있다면 대답하셔도 충분할 것이라 생각 됩니다.

브라우저 렌더링 엔진이 HTML, CSS, Javascript를 해석하는 순서에 대해 설명해보세요.

브라우저에서 HTML, CSS, JavaScript이 어떻게 처리되어 사용자에게 웹페이지가 보여지는지에 대한 기본적인 과정을 설명해보세요.

주소창에 naver.com을 입력하면 생기는 일을 설명하세요.(DNS 조회, 서버와의 통신 등이 어떻게 이루어지는 지에 대해 설명해보세요.)

HTTP와 HTTPS의 차이점은?

HTTP와 HTTPS의 프로토콜 차이와 보안 측면에서의 이점에 대해 설명해보세요.

HTTPS를 사용했을 때의 장점은?

HTTPS를 사용함으로써 얻을 수 있는 보안적인 이점에 대해 설명해보세요.

프론트엔드 프레임워크와 라이브러리

프레임워크와 라이브러리의 차이는 무엇인가요?

React, Vue.js, Angular와 같은 프론트엔드 프레임워크에 대한 이해와 경험을 어필하세요. 세 가지 프레임워크의 특징과 각각의 장단점에 대해 설명해보세요.

React, Vue.js, Angular중에서 어떤 라이브러리나 프레임워크를 주로 사용하나요? 그 이유는 무엇인가요?

선택한 라이브러리나 프레임워크의 장단점에 대해 어떻게 평가하나요?

React의 상태관리 방법에 대해 설명하세요.

React의 라이프사이클에 대해 설명해주세요.

React를 이용해 화면을 만들 때의 로직을 설명하세요.

100페이지를 React로 작업할 시 하루 8시간 기준 몇 일 이내로 작업할 수 있는지?

Vue를 배워야 한다면 지금 React를 쓰는 수준으로 몇 일 이내로 터득할 수 있는지?

프론트엔드 트렌드와 관심 분야

최근에 주목한 프론트엔드 개발 트렌드가 있나요?

최근에 주목한 기술이나 개발 트렌드가 있나요? 그것에 대해 어떤 생각이나 계획이 있나요?

앞으로 어떤 분야에 대해 더 깊이 공부하고 싶은지 말씀해보세요.

프론트엔드 테스트

유닛 테스트와 통합 테스트의 차이는 무엇인가요?

테스트 주도 개발(TDD)에 대해 어떻게 생각하나요?

배포와 자동화

CI/CD(Continuous Integration/Continuous Deployment)가 무엇이며, 어떻게 구현할 수 있나요?

배포 프로세스에서 발생할 수 있는 문제에 대비하는 방법에 대해 이야기해보세요.

반응형 디자인과 크로스 브라우징

반응형 웹 디자인이 무엇이며 왜 중요한가요?

레이지 로딩(Lazy Loading)이 무엇이며, 어떻게 구현할 수 있나요?

크로스 브라우징 이슈를 다루는 방법에 대해 설명하세요.

도구 및 개발 환경

npm과 yarn의 차이는 무엇인가요?(node로 프로젝트를 진행할 때 많이 쓰이는 패키지 관리 툴)

웹팩(Webpack)이 무엇이며 어떻게 동작하는지 설명하세요.

웹 애니메이션

CSS와 JavaScript를 사용하여 웹 애니메이션을 만드는 방법에 대해 설명해보세요.

GPU 가속이 무엇이며, 웹 애니메이션에 어떻게 도움이 되나요?

웹 보안

XSS(Cross-Site Scripting)와 CSRF(Cross-Site Request Forgery)에 대해 설명하세요.

Content Security Policy(CSP)가 무엇이며, 어떻게 사용되나요?

웹 성능 최적화

웹 페이지 성능을 최적화/개선하기 위한 방법에 대해 설명하세요.

브라우저 렌더링 프로세스에 대해 설명하세요.

웹 접근성

웹 접근성이 왜 중요한가요? 어떻게 웹 사이트를 접근성을 고려하여 개발할 수 있나요?

ARIA(Accessible Rich Internet Applications) 속성이 무엇이며 어떤 상황에서 사용하나요?

DOM이 무엇이며 어떻게 작동하는지 설명하세요. / BOM(브라우저 객체 모델)이 무엇인가요?

[단어 먼저 알고 가기]

파싱: 프론트엔드 개발에서 '파싱'은 일반적으로 데이터를 해석하고 원하는 형식으로 추출하는 과정을 나타냅니다. 주로 데이터를 가져와서 이를 웹 페이지에 표시하기 위해 필요한 형식으로 가공하는 단계를 말합니다.

트리: DOM 트리에서의 '트리(Tree)'는 계층적인 구조를 가진 데이터를 나타내는 자료 구조입니다. 이는 계층적인 부모-자식 관계를 갖는 구조를 나타내어 트리 형태로 표현됩니다. DOM 트리는 문서 객체 모델(Document Object Model)의 표현으로, HTML 또는 XML 문서의 구조를 트리 구조로 표현한 것입니다.

여기서 간단히 설명하면,

루트(Root): 트리의 맨 위에 있는 노드로, 전체 구조의 시작점을 나타냅니다. HTML 문서에서는 <html>태그가 루트 노드가 됩니다.

부모(Parent)와 자식(Child): 각 노드는 다른 노드를 가리킬 수 있습니다. 부모 노드는 그 자식들을 가리키며, 자식 노드는 부모를 가리킵니다.

형제(Sibling): 같은 부모를 가진 노드들은 서로 형제 관계입니다.

DOM 트리는 HTML 문서의 요소, 속성, 텍스트 등을 노드로 표현하고, 이들 간의 부모-자식 관계를 트리 형태로 나타냅니다. 이 트리 구조를 통해 웹 페이지의 구조를 표현하고, JavaScript를 사용하여 동적으로 수정하거나 조작할 수 있습니다.

노드: '노드(Node)'는 트리 구조에서 각각의 개별적인 요소를 나타내는 단위를 의미합니다. DOM(문서 객체 모델)에서 사용되는 노드는 HTML 또는 XML 문서의 구조를 표현하는 데 사용됩니다. 다양한 유형의 노드가 있으며, 주요한 네 가지 유형은 문서 노드, 요소 노드, 속성 노드, 텍스트 노드입니다.

DOM(Document Object Model)은 웹 페이지의 구조화된 표현을 제공하고, 프로그래밍 언어가 이 구조에 접근하고 조작할 수 있도록 도와주는 인터페이스입니다. 간단히 말하면, DOM은 HTML 및 XML 문서의 객체 지향 표현이며, 이를 통해 문서의 구조, 스타일, 내용 등에 대한 동적인 변경이 가능합니다.

DOM은 트리 구조로 표현되며, 각 요소(element)는 노드(node)로 표현됩니다.

DOM의 주요 요소:

1. **문서 노드(Document Node):** 문서 전체를 나타내는 노드로, DOM 트리의 최상위에 위치합니다.
2. **요소 노드(Element Node):** HTML 요소를 나타내는 노드로, 문서의 구조를 형성합니다.
3. **속성 노드(Attribute Node):** HTML 요소의 속성을 나타내는 노드로, 요소 노드의 자식 노드로 존재합니다.
4. **텍스트 노드(Text Node):** HTML 요소의 텍스트 내용을 나타내는 노드로, 요소 노드의 자식 노드로 존재합니다.

DOM은 자바스크립트를 통해 다룰 수 있으며, 자바스크립트를 사용하여 DOM을 조작하면 웹 페이지의 동적인 변화를 구현할 수 있습니다. DOM 조작은 문서의 내용을 변경하거나, 새로운 요소를 추가하거나, 기존 요소를 제거하는 등의 작업을 포함합니다.

예를 들어, 다음은 간단한 HTML 문서와 그에 대응하는 DOM 구조입니다:


```

<!DOCTYPE html>
<html>
<head>
  <title>DOM 예제</title>
</head>
<body>
  <h1 id="title">안녕하세요! </h1>
  <p>이것은 간단한 예제입니다.</p>
</body>
</html>

```

위 HTML 문서에 대한 간단한 DOM 트리는 다음과 같습니다:

```

Document
|-- html
|   |-- head
|   |   |-- title
|   |       |-- "DOM 예제"
|   |-- body
|       |-- h1#title
|           |-- "안녕하세요! "
|       |-- p
|           |-- "이것은 간단한 예제입니다."

```

하나 더!

아래와 같은 HTML 코드가 있다고 가정해보면

```

<html>
  <head>
    <title>DOM Tree</title>
  </head>
  <body>
    <div>
      <p>Hello, DOM! </p>
    </div>
  </body>
</html>

```

이를 DOM 트리로는 나타내면 다음과 같습니다.

```

- html
  - head
    - title
      - #text: "DOM Tree"
  - body
    - div
      - p
        - #text: "Hello, DOM! "

```

이제 BOM(브라우저 객체 모델)에 대해 알아보시다. BOM은 브라우저와 상호 작용할 수 있게 해주는 객체의 집합을 나타냅니다. DOM이 문서를 다루는 데에 반해, BOM은 브라우저 창과 관련된 객체들을 제어합니다.

BOM의 주요 객체들:

1. **window 객체**:브라우저 창을 나타내며, 전역 객체로서 브라우저 창과 관련된 다양한 정보 및 메소드를 제공합니다.
2. **navigator 객체**:브라우저의 정보를 제공하는 객체로, 사용 중인 브라우저에 대한 정보를 얻을 수 있습니다.
3. **screen 객체**:사용자의 화면에 대한 정보를 제공하는 객체로, 화면의 크기 및 해상도 등을 확인할 수 있습니다.
4. **location 객체**:현재 문서의 URL 정보를 다루는 객체로, 페이지의 주소를 변경하거나 읽어올 수 있습니다.
5. **history 객체**:브라우저의 히스토리를 다루는 객체로, 페이지 이동 기록을 조작할 수 있습니다.

BOM을 이용하면 브라우저 창을 조작하고, 페이지를 새로고침하거나 이동하며, 브라우저의 크기를 조절하고 다양한 클라이언트 측 작업을 수행할 수 있습니다.

DOM (Document Object Model):

요소 조작 및 이벤트 처리:DOM을 이해하면 HTML 문서의 요소들을 자바스크립트를 통해 동적으로 조작하거나, 이벤트를 처리하는 방법을 익힐 수 있습니다.

동적 콘텐츠 업데이트:웹 페이지의 콘텐츠를 동적으로 업데이트하고 변경할 수 있는 능력은 프론트엔드 개발자로서 필수적입니다. DOM을 이해하면 페이지의 일부를 변경하거나 새로운 콘텐츠를 추가하는 등의 작업을 수행할 수 있습니다.

이벤트 위임 및 처리:DOM을 통해 이벤트를 효과적으로 처리하고, 이벤트 위임을 활용하여 성능을 향상시킬 수 있습니다.

동적 UI 개발:DOM을 다룰 수 있는 능력은 동적인 사용자 인터페이스(UI)를 개발하는 데 필수적입니다.

BOM (Browser Object Model):

브라우저 창과의 상호작용:BOM을 이해하면 브라우저 창을 조작하고 페이지를 새로고침하거나 이동하는 등의 작업을 할 수 있습니다.

브라우저 정보 활용:브라우저의 정보를 읽고 활용할 수 있는 능력은 웹 애플리케이션을 브라우저에 맞게 최적화하는 데 도움이 됩니다.

히스토리 관리:브라우저의 히스토리를 조작하여 사용자 경험을 향상시키는데 도움이 됩니다.

클라이언트 측 작업:BOM을 통해 브라우저의 다양한 기능을 활용하여 클라이언트 측에서 수행하는 작업을 구현할 수 있습니다.

비전공자 신입이 이러한 개념들을 학습하면서 프론트엔드 개발에 필요한 기본적인 도구와 능력을 키울 수 있으며, 이를 통해 웹 애플리케이션을 보다 동적이고 효과적으로 개발할 수 있습니다.

[좀 더 다양한 질문!]

- 데이터를 받아오는데 어떤 방법들을 사용할 수 있는가, 또한 그것들에 대해 설명해보세요.
- OOP의 개념에 대해 설명하세요.
- webRTC, Peer To Peer, web socket은 어떤 개념인지?
- Python과 GO 언어의 차이점을 설명하세요.
- 호이스팅에 대해 설명하세요. 변수와 함수가 어떻게 호이스팅되는지에 대한 기본적인 개념을 설명해보세요.
- this의 용법을 아는대로 설명하세요 자바스크립트의 this는 몇가지로 추론될 수 있는지?
- 브라우저 저장소에 대한 차이점을 설명하세요.
- 이벤트 버블링에 대해 말씀해주세요.
- 타입스크립트 사용해본 적 있는지? 있다면 어떤지?(장점과 사용 경험) 타입스크립트를 쓰는 이유는 무엇인가? 타입스크립트를 적용하면 어떤 어려가 있었는지?
- CSR과 SSR의 개념과 차이점은 무엇인지 설명해보세요.
- Prototype이란? Prototype Chaining은?(JavaScript의 프로토타입 개념과 프로토타입 체이닝에 대해 설명해보세요.)
- DOM과 가상 DOM이란?(DOM과 가상 DOM의 차이와 가상 DOM이 사용되는 이유에 대해 설명해보세요.)
- AJAX에 대해 자세히 설명하세요.(개념과 장단점)
- 이벤트 루프란?
- 화살표 함수와 일반함수의 차이점(화살표 함수와 일반 함수의 문법적인 차이와 동작 방식)
- "=="와"==="의 차이(동등 연산자와 일치 연산자의 차이에 대해 설명)
 - "=="는 동등 연산자로, 두 값을 비교할 때 형변환(type coercion)을 수행합니다. 즉, 비교하는 값의 데이터 타입이 다르더라도 자동으로 형변환을 수행한 후에 비교합니다. 이러한 형변환은 때로 예측하지 못한 결과를 초래할 수 있으므로, "=="를 사용할 때는 조심해야 합니다. "==="는 일치 연산자로, 두 값이 데이터 타입과 값이 모두 같은지 비교합니다. 따라서 "==="를 사용하면 형변환 없이 정확한 값을 비교할 수 있습니다. 이러한 일치 연산자를 사용하는 것이 더 안전하고 예측 가능한 결과를 얻을 수 있습니다.
- 헛갈리지 않게 핵심만 파악해보세요. 동등! 일치!
- 게시판 구축할 수 있는지(간단한 웹 게시판을 구축하는 과정과 필요한 기술에 대해 설명)
- RESTful API가 무엇인지 설명하세요.(개념과 특징)
 - RESTful API는 HTTP 프로토콜을 기반으로하는 웹 서비스 아키텍처입니다. 자원, 메소드, 메시지 등을 정의하여 클라이언트-서버 간의 통신을 가능하게 합니다. 또한, RESTful API는 표준 HTTP 메소드(GET, POST, PUT, DELETE)를 사용하여 서버와 통신합니다
- 이 질문을 대답하면서, HTTP 메소드에 대한 사용 방법에 대해서도 꼬리물기 질문이 들어올 수 있으니 준비하시는 것이 중요합니다.
- 그렇다면 HTTP메소드에 대한 설명도 해주실 수 있나요?
 - GET: 서버에서 리소스(데이터)를 요청하는 메소드입니다. 요청한 데이터를 가져와 응답합니다.
 - POST: 서버에 데이터를 전송하는 메소드입니다. 데이터를 전송하여 서버에서 처리하고, 처리 결과를 응답합니다.
 - PUT: 서버에 데이터를 업데이트하는 메소드입니다. 요청한 데이터를 서버에 저장하고, 처리 결과를 응답합니다.
 - DELETE: 서버에서 데이터를 삭제하는 메소드입니다. 요청한 데이터를 서버에서 삭제하고, 처리 결과를 응답합니다.
- 간략하게 기능 별 포인트를 함축적으로 이야기하는 것이 중요합니다.
- PUT, PATCH의 차이점이 무엇인지 설명하세요.(HTTP의 PUT과 PATCH 메서드의 차이)
- Promise에 대해 설명하세요.
- 실행 컨텍스트에 대해 아는대로 설명

- 마이크로 태스크, 매크로 태스크의 개념
- call, apply 함수의 차이점
- 옵셔널 체이닝: 옵셔널 체이닝이 무엇이고, 어떤 상황에서 사용하는지에 대해 설명해보세요.
- defer, async 구분(<script>태그의 defer와 async속성의 차이에 대해 설명)
- 새로운 프로젝트를 시작한다면 어떤 구조를 가져갈 것인지(새로운 프로젝트에서 사용할 프로젝트 구조나 아키텍처에 대해 어떻게 결정할 것인지)
- 성능개선 경험(이전 프로젝트에서 어떻게 성능을 개선했는지, 어떤 도구나 기법을 사용했는지에 대해)
- 크로스 브라우징을 고려한 개발 경험이 있는지(이전 프로젝트에서 어떻게 크로스 브라우징을 고려했는지, 어떤 도구나 방법을 사용했는지에 대해)

***프론트엔드 면접 후기**

웹 퍼블리싱, 디자인 회사는 HTML, CSS로 단지 화면만 만들어내는 곳도 있었고
 프론트엔드 관련은 AJAX, axios, promise, callback등의 방식으로 API 데이터를 불러와 콘솔에 찍거나 데이터를 가공해(map, filter, forEach 등) 화면에 출력시키거나 단지 JSON배열을 선언해 가공하는 곳도 있었음.
 이처럼 내가 아직 많이 부족함을 깨닫고 리액트보다 자바스크립트의 기본 이론을 공부하되 주로 네트워크 통신, 배열 데이터 다루기 등을 중점적으로 더 공부해나갈 계획이다.

(출처: [신입 프론트엔드 면접 질문 & 후기 \(22.0. : 네이버블로그 \(naver.com\)\)](#))

Back-end 개발자 면접 질문 [비전공자 신입~전공자 신입]

보안과 성능최적화

웹 애플리케이션에서 어떤 보안 문제에 주의를 기울여야 할까요?

성능 최적화를 위한 백엔드 개발 방법에 대해 이야기해보세요.

보안에 대한 고려사항이 무엇인가요?

- 개발 중 보안에 주의를 기울이는 방법과 주요 보안 원칙에 대한 이해를 언급합니다.

인증(Authentication)과 권한 부여(Authorization)의 차이는 무엇인가요?

- 사용자 인증과 권한 관리에 대한 기본적인 이해를 설명합니다.

JAVA

Java에서 메모리 관리는 어떻게 이루어질까요?

- 가비지 컬렉션과 메모리 할당에 대한 기본 개념을 설명합니다.

오버로딩과 오버라이딩의 차이점을 설명해주세요.

- 메서드 오버로딩과 메서드 오버라이딩의 개념과 차이점을 설명합니다.

Java에서 인터페이스와 추상 클래스의 차이는 무엇인가요?

- 인터페이스와 추상 클래스의 특징과 사용 시점에 대한 이해를 제시합니다.

Java의 컬렉션 프레임워크에 대해 설명하세요.

- 주요 컬렉션 인터페이스와 클래스, 각각의 특징과 사용법에 대한 기본적인 이해를 언급합니다.

Java에서 예외 처리 방법에 대해 설명해주세요.

- try-catch 블록, throw, throws 등을 이용한 예외 처리에 대한 설명을 준비합니다.

DataBase

SQL과 NoSQL 데이터베이스의 차이는 무엇인가요?

- 각각의 특징과 사용 시나리오에 대한 이해를 언급합니다.

SQL의 기본적인 쿼리 문법에 대해 설명해주세요.

- SELECT, INSERT, UPDATE, DELETE 등의 SQL 쿼리 문법을 이해하고 있는지 확인합니다.

데이터베이스 정규화에 대해 설명하세요.

- 제 1, 2, 3 정규화에 대한 개념과 목적을 설명합니다.

데이터베이스와 SQL

- 관계형 데이터베이스와 NoSQL 데이터베이스의 차이에 대해 설명해주세요.

- SQL 쿼리에서 JOIN의 역할에 대해 설명하세요.

인덱스(Index)에 대해 설명해주세요.

- 데이터베이스 인덱스의 역할과 성능 향상에 대한 이해를 언급합니다.

트랜잭션(Transaction)이란 무엇인가요?

- 트랜잭션의 개념과 ACID 특성에 대한 이해를 설명합니다.

JDBC(Java Database Connectivity)에 대해 아는 대로 설명하세요.

- Java에서 데이터베이스와의 연결을 위해 사용되는 JDBC에 대한 이해를 언급합니다.

[좀 더 다양한 질문!]

- DB는 어떤걸 사용해봤나요?
- 객체지향에 대해 설명해주세요.
- 자바의 장점과 단점을 설명해주세요
- 자바 말고 다른 언어 할줄 아시나요?(관련 경험이 없다면 학습 계획에 대해 물어봅니다.)
- HTTP 상태 코드에 대해 아는 대로 말해보세요.
- 코드 작성시 어떤 것을 가장 우선시 하는지(코드 작성시 주의할 점에 대해 물어봅니다. 가독성, 유지보수성 등이 포함될 수 있습니다.)
- 프레임워크 중 MVC모델은 무엇인가?
- XML의 효용성은 무엇인가요?
- 클래스 변수, 인스턴스 변수, 지역변수가 무엇인가?
- List와 배열의 차이점은? 차이가 있다면 왜 나느지요?
- HttpServletRequest가 클래스인데 이것을 쓰기 위해선 어디서 객체 생성을 해주나
- JPA의 영속성 컨텍스트가 해주는 역할은 무엇인가?
- AOP와 DI를 설명해보기

- AOP란?

Aspect oriented programming = 관점 지향 프로그래밍

프로그래밍 할 때 특정한 관심사를 가진 코드 부분을 별도의 모듈로 분리함으로써 기존 객체지향의 강력한 기능(상속, 위임)만으로는 처치가 곤란했던 중복을 할 수 있게 됨

- DI란?

Spring을 적용하여 applicationContext.xml에서 설정만 해주면 외부 설정파일(xml)에서 연관관계에 있는 객체를 주입해주기 때문에 의존객체를 찾기 위한 코드가 필요하지 않게 됨
즉, 외부설정에서 객체를 찾아서 쓰기 때문에 코드 내에 의존관계를 맺는 코드 생성이 불필요

[난이도 UP!]

- 프로세스와 스레드의 차이점에 대해 설명하시오.(멀티태스킹, 멀티쓰레딩에 대한 이해를 확인)
- 멀티 프로세스와 멀티 스레드의 개념에 대해 설명해주세요.(자원 공유, 성능 등에 대한 이해를 확인)
- 동시성과 병렬성은 무엇인가요?
- 교착상태(Deadlock)이란 무엇인가요?(다중 프로세스/스레드 환경에서 발생할 수 있는 문제에 대한 이해를 확인)
- 콘보이 현상(convoy Effect)란 무엇인가요?(운영체제에서 발생하는 현상 중 하나로, 성능 이슈와 관련)
- 임계구역은 무엇인가요?(프로세스 간에 공유되는 자원에 대한 접근을 동기화하기 위한 개념을 확인)
- 세마포어란 무엇인가요?(동기화 기법 중 하나로, 병렬 프로그래밍에서 사용되는 개념을 확인)
- 가상 기억장치란 무엇인가요?(가상 메모리와 관련된 개념)
- 페이징 기법과 알고리즘에 대해 설명해주세요.(운영체제에서 메모리 관리에 사용되는 기법과 알고리즘에 대한 이해를 확인)

[더 UP! UP!]

- 문맥 교환(Context Switching)이란 무엇인가요?
- GET 요청과 POST 요청의 차이점은 무엇인가요?
- WS(Web Server)와 WAS(Web Application Server)의 차이점에 대해 설명해주세요.
- 스프링 프레임워크에 대해 설명해주세요.
- Spring과 Spring Boot의 차이점은 무엇일까요?
- 서블릿에 대해 설명해주세요.
- 제어의 역전(Inversion of Control = IoC)이란 무엇인가요?
- 대용량 트래픽이 발생하면 어떻게 대응할 수 있을까요?

- 데이터베이스 정규화에 대해 설명해주세요.
- 데이터베이스 인덱스란 무엇인가요?
- 자바 데이터 타입 중 기본형과 참조형의 차이에 대해 설명해주세요.
- ORM이 무엇인지 설명해주세요.
- 생성자(Constructor)에 대해 설명해주세요.
- String, StringBuffer, StringBuilder의 차이를 설명해주세요.
- new String()과 리터럴("")의 차이에 대해 설명해주세요.
- Synchronized(동기화)에 대해 아는데로 말씀해주세요.
- AJAX의 어떤 부분이 백엔드 기술이고 어떤 부분이 프론트엔드 기술인지 설명해 주시겠어요?
- 사이트 접속시 화면이 뜨지 않을 때 어떤 부분을 체크해야 하는지?
- Java8에서 처음 소개된 Stream을 설명하고 어떤 장점이 있는지 설명해주세요.
- RDB와 NoSQL의 차이에 대해 설명해 보세요.
- DB에서 인덱스를 잘 사용하면 어떤 장점이 있을까요?
- 병렬 프로그래밍을 할 때 어떤 부분을 유의해야 하나요? 세마포어와 뮤텍스에 대해 알고 있나요? 생길 수 있는 문제는?
- 가장 기억에 남는 백엔드 프로젝트의 구조를 그려보세요.
- LRU(Least Recently Used) 캐시에 대해 설명하시오.
- 상호배제(Mutex 또는 Mutual Exclusion)는 무엇인가요?
- 선점형vs비선점형 스케줄링의 차이는 무엇인가요?
- OSI7 레이어 계층에 대해 말해주세요.

***백엔드 면접 후기**

웹개발 - (주)메디오피아테크

2명의 담당자분들과 면접 진행, 내 포트폴리오와 이력서를 잘 읽으셨다는 느낌이 들어서 설명할때 재밌었습니다. 용어 위주

유지보수, 네트워크 - (주)아티언스 (1~3년차 경력면접)

1 : 다수 방식으로 면접이 진행 되었습니다. 면접 질문은 업무 지식, 인성, 사회 경험 등에 대하여 질의 응답이 이루어졌습니다.

AI인공지능 - (주)인피닉

서류 전형 이후 다대다 면접 진행, 이력서 및 경력 사항에 대한 질문, 면접 종료 후 희망 연봉 및 직전 연봉 기재, 다음날 유선으로 합격 통보

풀스택 - (주)디엔에이링크 (1~3년차 경력면접)

팀장님과 1:1 직무 및 인성 면접 진행하였습니다. 향후 맡을 업무에 관해 설명해주신 후 이력서 바탕으로 질문해주셨습니다.

임베디드 - (주)이로움

현재 사용할 수 있는 기술 스택들과 회사에서 어떤 것들을 개발했는지 알고 있는가? 포트폴리오에 제시한 프로젝트에 대한 설명

면접

자취방에서 가까운 곳에 평이 좋았던 기업에 파이썬 백엔드 개발자를 구인하는 것을 보고 지원했다. 최근에 스프링 공부를 많이 하고 있었는데 파이썬 백엔드 개발자로 급 노선 변경하여 지원한 것은 너무나도 뜬금 없

있을 수 있지만 나름 본인 기준에서는 좋은 회사라고 판단한 부분들이 많아서 선택하게 됐다.

서류 제출 후 지원자에게 맞춰서 1주일 이내 일정으로 면접 일정이 조율됐다.

프론트엔드 기술 면접은 경험해본 적이 있지만 백엔드 기술 면접은 첫 경험이었기에 네트워크, 데이터베이스 관련해서 내용만 정리해서 면접을 보러갔다.

면접은 지원자 한명 / 팀장 두분 팀원 한분으로 총 세분의 면접관분으로 구성됐다.

(앞으로 A팀장님 B팀장님 C팀원님으로 정의한다.)

A 팀장님은 편안한 분위기로 긴장을 풀어주려고 본인 소개를 먼저 해주셨다.

그에 맞춰서 본인이 간단한 본인 소개를 진행했고, 기술 면접으로 바로 진행됐다.

기술 면접은 진행했던 프로젝트 관련 질문과 코딩 테스트를 진행했다.

[프로젝트 관련 질문]

본인의 경우 로그인, 회원가입을 구현한 것을 질문받았고, node.js 환경에서 어떻게 진행했는지 얘기했다. 그런데 B팀장님 면접관라이브러리를 통해서 만든 것은 사실상 프론트엔드 개발자가 진행한 것과 다름이 없다는 듯한 얘기를 해주셨을 때앗..싶었다.

그동안 A를 풀어야 했는데 B를 보고 고민해왔던 느낌이었다.

너무 정체성 없이 프론트 담당, 백엔드 담당, 퍼블리싱, 기획 등에 투입됐던 지난 일들이 본인에게 얼마나 독이 됐는지를 느꼈고, 백엔드 개발자에게 중요한 것은 화면에 보여지는 것이 아닌 실제 로직이 어떻게 나왔는지 스스로 라이브러리처럼 만들고 생각하고, 설명할 수 있는지가 포인트였다고 생각됐다.

백엔드 개발자의 프로젝트 퀄리티는 결국 로직을 제대로 만들고 구현했는지가 핵심이 된 것이었다.

그동안 프로젝트를 너무 구역구역 진행했던 본인에게는 매우 중요한 사항이 무엇인지를 느끼게 해줬던 부분이다.

백엔드 핵심은 화면이 보이는 것 보다. 로직이 돌아가는 기능이 의도와 목적에 맞게 정확하게 잘 동작했는지가 중요한 부분임을 느끼게 됐다.

[코딩 테스트]

A팀장님과 C팀원님이 평가를 주도해주셨다.

문제의 핵심이 무엇인지 어떻게 풀어내는지 얼마나 최적화할 수 있는지 등을 다루는 일이었다.

점근 표기법으로 봤을 때. 복잡도를 계산하여 적절한 알고리즘을풀어내는 것이 문제의 핵심이다.

정말 어려운 알고리즘의 문제가 아니었다. 그런데 제대로 풀이하지 못했던 부분이 많이 아쉬움이 남았다. CS 관련 지식으로 최대한 준비했는데 막상 제대로 준비한 것은 보여주지 못한 것이 아쉬움이 컸다. 단순 구현에 치중된 코딩 테스트를 준비했는데 복잡도를 계산하고 최적화를 하는 단계까지 고려할 수 있게 친절하게 시각을 넓혀 주신 것 같아서 개인적으로 A팀장님께는 감사한 마음이 우선 들었다.

스스로 구현을 위한 알고리즘 역량이 부족함을 알 수 있었고, 개선할 이유를 찾아서 좋았다.

개선하거나 성장할 수 있는 방법을 찾았을 때 너무나도 기분이 좋지만 또 한편으로는 씁쓸한 마음이 존재하는 것도 있는 것 같습니다.

백엔드 기술 면접에서 기분 좋은 경험을 할 수 있었고, 앞으로는 어떻게 면접을 준비해야 하는 것인지 배울 수 있었습니다.

좋은 면접 기회와 저의 기술 블로그를 검토해 주시고, 긍정적인 평가와 조언을 남겨주셨던 A팀장님에게는 많은 부분을 배울 수 있었습니다. 감사드립니다.

출처: [일상] 신입 백엔드 개발자 면접 후기 — 성조의 기술 블로그 (tistory.com)