

***퍼블리셔/프론트엔드 면접 Q&A**

1.공통질문

왜 퍼블리셔/개발자가 되려고 하는가

퍼블리셔/개발자로서의 본인의 비전을 이야기 해달라

비전공자로서 갖고 있는 컴플렉스가 있는가

운영체제같은 컴퓨터공학(cs)에 대한 기초지식이 있는가

최근에 관심갖거나 공부 하고 싶은 개발 기술은 무엇인가

프로젝트 협업 과정을 경험한 적이 있는가

퍼블리셔/개발자가 되기 위해서 어떻게 공부하였는가

학습시 주로 이용하는 웹페이지나, 동영상 강좌 페이지는 어디인가

최근에 읽은 개발 관련 서적은 무엇인가

즐거 보는 개발 관련 유튜브가 있는가

회사 기술 스택에 맞추어 단기간 내에 언어와 프레임워크를 학습 하여야 할 때, 어떻게 공부하고 해결할 것인가

포트폴리오 제작시에 비인기 라이브러리를 사용한 경험이 있는가

이러한 비인기 라이브러리에 대한 정보를 어디서 얻는가 왜 활용하였는가

2.HTML

XML과 HTML의 차이점을 설명해보세요

XML은 웹 브라우저 외에도 포괄적으로 사용되는 형식으로 데이터 교환에 특화되어 있으며 이것 만으로 데이터를 가지고 있을 뿐 화면에 표시되지 않는다. HTML은 웹 브라우저에서 사용하는 문서 형식이며 XHTML은 HTML규격이 가진 문제를 극복하고 확장기능을 제공한다.

Http와 Https 통신 방식의 차이를 설명해보세요

결정적 차이는 보안이다. 1. http방식은 네트워크상에서 정보를 누군가가 마음대로 열람, 수정이 가

/ https는 누가 볼수없도록 막음. 2. http방식이 https방식보다 빠르다. 3. Http방식은 민감한정보를 다룰 때 항상 변조, 해킹 가능성을 생각해야한다. Https는 설치 및 인증서를 유지하는데 추가적인 비용이 발생. -> 따라서, 민감한 정보가 있는 페이지의 경우 Https 그럴필요가없으면 http로 만들면 된다.

window 객체란 무엇이며, DOM은 무엇인가요?

window 객체 : 실행 시 가장 상위에 존재하는 객체. 변수 또는 함수 선언 시 이 window 객체 안에 선언된다. 이 window 객체의 프로퍼티를 탐색하는 방법에는 여러가지가 있다.

window.location : 현재 브라우저의 주소 window.location.href : 창에 입력된 주소

window.navigator : 브라우저 자체에 대한 정보 window.document : 문서의 html/css등에 접근가능

DOM : Document of Model의 약자.

javascript에서 document로 접근 가능하다. 자식의 최상의 문서는 html

홈페이지가 사용자에게 보이는 순서에 대해서 설명해보세요.

브라우저의 기본 구조

브라우저의 주요 구성 요소는 다음과 같다.(1.1)

1. 사용자 인터페이스 - 주소 표시줄, 이전/다음 버튼, 북마크 메뉴 등. 요청한 페이지를 보여주는 창을 제외한 나머지 모든 부분이다.
2. 브라우저 엔진 - 사용자 인터페이스와 렌더링 엔진 사이의 동작을 제어.
3. 렌더링 엔진 - 요청한 콘텐츠를 표시. 예를 들어 HTML을 요청하면 HTML과 CSS를 파싱하여 화면에 표시함.
4. 통신 - HTTP 요청과 같은 네트워크 호출에 사용됨. 이것은 플랫폼 독립적인 인터페이스이고 각 플랫폼 하부에서 실행됨.
5. UI 백엔드 - 폼, 박스와 창 같은 기본적인 장치를 그림. 플랫폼에서 명시하지 않은 일반적인 인터페이스로서, OS 사용자 인터페이스 체계를 사용.
6. 자바스크립트 해석기 - 자바스크립트 코드를 해석하고 실행.
7. 자료 저장소 - 이 부분은 자료를 저장하는 계층이다. 쿠키를 저장하는 것과 같이 모든 종류의 자료를 하드 디스크에 저장할 필요가 있다. HTML5 명세에는 브라우저가 지원하는 '웹 데이터 베이스'가 정의되어 있다.

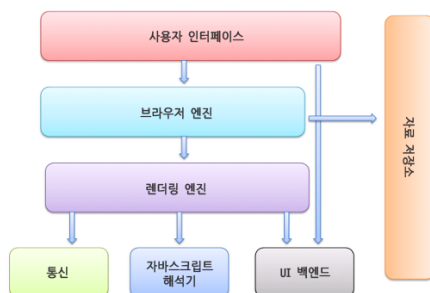


그림 1 브라우저의 주요 구성 요소

크롬은 대부분의 브라우저와 달리 각 탭마다 별도의 렌더링 엔진 인스턴스를 유지하는 것이 주목할만하다. 각 탭은 독립된 프로세스로 처리된다.

3.CSS

flex를 사용하는 이유가 무엇인가요?

flex는 레이아웃을 좀 더 편하게 잡기 위해서 만들어진 css 속성이다. flex를 사용하면 요소들의 크기나 위치를 쉽게 잡을 수 있다. 기존에 수평 구조를 만들 때 사용하는 속성이 float나 inline-block 등이 있었는데 여러가지 문제를 가지고 있었고 flex를 사용하면 이러한 속성의 한계를 보완할 수 있다. 물론 수평뿐만 아니라 수직도 가능하다.

flex는 컨테이너와 아이템 개념을 사용하여 요소의 크기가 불분명하거나 동적인 경우에도 요소를 효율적으로 정렬할 수 있게 해준다.

Box-Model 대하여 설명해보세요

CSS에 포함되는 모든 요소의 외장은 박스이며 주어진 태그의 특성에 따라, 블록 박스와 인라인 박스로 나뉜다. 블록 박스의 대표적인 태그는 div이며, 인라인 박스의 대표적인 태그는 span이다.

CSS box는 콘텐츠/패딩/테두리/여백 총 4가지 영역으로 구성되는데, 블록 박스는 해당 영역이 모두 적용되어 구성되며, 인라인 박스는 콘텐츠 영역만으로 구성된다.

또한 블록 박스는 줄바꿈이 일어나며, width와 height을 지정해줄 수 있고 padding/margin/border 영역이 적용된다. 인라인 박스는 세 특징들 모두 해당되지 않는다. padding은 border와 콘텐츠 영역 사이에 위치하며, margin은 박스 주변의 여백을 지정하며 border는 박스 여백과 패딩 사이에 그려진다.

그리고 몇 가지 복잡성을 추가하기 위한 도구인 대체 박스 모델이 있다. 표준 박스 모델에선 블록 박스의 width, height을 지정하면, 실제 박스의 넓이와 높이는 margin과 padding을 포함한 값이다. 그러나 대체 박스 모델에선 블록 박스의 크기는 지정된 width와 height 값이다. 또한 이를 활성화하기 위해선, 해당 태그에 box-sizing: border-box 속성을 추가해줘야 한다.

class와 id의 차이점을 설명해 주세요.

id와 class의 차이는 id는 유일한 요소에 적용할 때, 그리고 css는 복수의 요소에 적용할 때 사용한다는 점이다. 하나의 id는 한 문서에서 한 번만 사용이 가능하지만, 하나의 class는 여러 번 사용이 가능하다. 우선순위는 id가 class보다 높다.

display 속성에 어떤 것들이 있는지 설명해 주세요.

display 속성에는 block, inline, inline-block, none이 있다

block : 항상 새로운 라인에 요소가 시작되고 화면 크기의 전체 가로폭을 영역으로 차지한다.
width 속성 값을 부여해주면 그 너비만큼 영역을 차지한다.

inline : 새로운 라인에서 시작되지 않으며 다른 요소들과 같은 줄에 배치될 수 있고 content 너비만큼의 영역을 차지한다. 그리고 width, height, margin-top, margin-bottom 속성이 적용이 되지 않는다.

inline-block: block 레벨 요소와 inline 레벨 요소의 특징을 모두 가지고 있다. 한 줄에서 inline 레벨 요소들과 같이 배치될 수 있으며 width와 height 속성으로 영역의 크기를 지정할 수 있다.

none: 선택한 요소들을 화면에 나타나지 않게 한다. visibility: hidden과의 차이점은 영역이 남아있는지 여부가 다르다는 점이다.(display: none은 영역도 없음)

position에 대하여 설명해보세요.

CSS position 속성은 문서 상에 요소를 배치하는 방법을 지정한다. static, relative, absolute, fixed, sticky 속성을 값으로 갖는다.

static은 position 속성의 기본 값이며, 요소를 문서 흐름에 맞게 배치한다.

relative는 요소가 차지하는 공간은 static과 같으나, top/right/left/bottom 값을 적용시킬 수 있다.

absolute는 일반적인 문서 흐름에서 제거되고 static이 아닌 부모 블록의 박스 안에 속하여 위치가 지정된다. 만약 전부 static 값을 갖으면, 최상위 블록에 의존한다.

fixed은 요소가 페이지의 평소 위치에서 제거되고 뷰포트를 기준으로 지정된 위치에 배치되며 스크롤 할 때 이동하지 않는다.

sticky는 지정된 임계값을 넘을 때까지 relative 위치로 처리되며, 특정 지점에서 fixed 위치로 처리된다.

4.JS

자바스크립트를 왜 객체 지향 프로그래밍 언어라고 부르는지 아시나요?

데이터 처리 과정 시, 객체를 생성하여 하나의 독립된 과정으로 운영될 수 있도록 함으로서 코드 재사용성을 증가시키기 때문에 자바스크립트를 객체 지향 프로그래밍 언어라고 부름.

객체 지향 프로그래밍 언어의 특징으로는 추상화(*같은 기능의 함수에 같은 이름을 부여)와 다형성(*상속으로 받은 함수를 그대로 사용하지 않고 새로 만들어 사용), 캡슐화(*보호가 필요한 객체

내용을 해당 객체 내용의 함수로만 접근 가능하게 함), 상속화 등이 있음

Promise 객체는 무엇이며, 언제 왜 사용하나요?

순차적이지 않은 비동기 함수의 실행 순서를 제어. callback 대신 Promise 함수를 사용했을 때의 장점! callback hell에서 벗어날 수 있음 .then() 을 이용하여 가독성 좋은 연속적인 비동기 코드를 작성가능 / Promise.all()을 통해 병렬 비동기 코드를 작성 가능. callback 함수를 제 때 실행하지 못하거나 변수 전달 실패 등이 없음

Promise와 Callback의 차이점은 무엇이며 각각의 장단점에 대해 설명해보세요

Callback을 간단하게 설명하자면 자바스크립트의 비동기 통신의 단점을 보완하기위해 나온 개념입니다. 쉽게 설명하자면 서버가 요청받은 내용을 처리하고 있을때 특정 요청받은 내용의 차례가 올 경우 그 내용을 콜백을 통해 불러서 처리합니다.

동기 함수와 비동기 함수의 차이점을 설명해보세요

동기함수 : 블로킹, 명령문이 순서대로 진행된다.

비동기함수 : 요청과 응답의 결합이 비동기적이라 꼭 순차적으로 응답하지 않아도 되며, 의도적으로 응답시간을 지연시킬 수 있음

자바스크립트에서 프로토타입이란?

자바스크립트는 프로토타입을 기반으로 상속을 구현하여 불필요한 중복을 제거(중복 제거 방법은 기존의 코드를 재사용하는것!!)

즉, 생성자 함수가 생성할 모든 인스턴스가 공통적으로 사용할 프로퍼티나 메소드를 프로토타입에 미리 구현해 놓음으로써 또 구현하는것이 아니라 상위(부모) 객체인 프로토타입의 자산을 공유하여 사용할 수 있다.

__proto__ 접근자 프로퍼티로 자신의 프로토타입, 즉 Prototype 내부슬롯에 접근 할 수 있음.

프로토타입체인이란? 객체의 프로퍼티에 접근하려고 할때 객체에 접근하려는 프로퍼티가 없으면, __proto__ 접근자 프로퍼티가 가리키는 링크를 따라 자신의 부모역할을 하는 프로토타입의 프로퍼티를 순차적으로 검색한다. 프로토타입체인의 최상위 객체는 Object.prototype이다. 이 객체의 프로퍼티와 메소드는 모든 객체에게 상속된다.

prototype 프로퍼티 는 생성자함수가 생성할 인스턴스의 프로토타입을 가르킨다.

자바스크립트에서 .this 객체가 하는 역할은 무엇인가요?

함수 스코프 내에서 자동으로 설정되는 특별한 식별자. 함수를 호출하는 방법에 의해 결정된다.

함수 호출 시 new ⇒ 완전히 새로운 객체. apply(동적호출) 또는 call(정적호출)을 통해 호출 ⇒ 인수로 전달된 객체 / 함수를 메서드로 호출 ⇒ 함수가 프로퍼티인 객체

함수형 프로그래밍에 대해 설명해보세요

프로그래밍 패러다임에는 명령형 프로그래밍과 선언형 프로그래밍이 있다. 함수형 프로그래밍은 후자인 명령형 프로그래밍에 속한다. 이 함수형 프로그래밍은 기본적으로 기존 명령형 프로그래밍은 어떻게 구현하겠다 에 집중돼 있었다면 함수형 프로그래밍은 무엇을 구현하겠다에 초점이 맞추어져 보다 함수를 수학적으로 사용하여 설계가 가능하다.

JSON은 무엇인가요?

자바스크립트 객체를 문자열로 표현할 수 있는 객체표기문법을 따른 웹문서. 서버 ⇒ 브라우저로 정보를 보낼 때 JSON 파일(stringify)로 보내고 브라우저는 이를 자바스크립트로(parse) 파싱함

forEach와 Map함수의 차이점을 설명해보세요

둘 다 배열의 요소를 반복하지만 forEach는 값을 반환하지 않고 map은 새로운 배열을 반환하며 원본 배열을 변경하지 않는다.

고차함수를 사용해본 적이 있으신가요? 있다면 고차함수를 설명해보세요.

다른 함수를 매개변수로 사용하여 데이터를 처리하거나 함수를 반환하는 함수 (ex : forEach, filter)

자바스크립트의 원시 타입(Primitive Data Type)은 몇가지이며, 전부 말해보세요.

Boolean String Number Null Undefined

var 와 let, const의 가장 큰 차이점은 무엇인가?

let은 변수에 재할당이 가능하지만 const는 불가능하다.

호이스팅(Hoisting)의 개념을 설명해보세요

함수 안에 있는 선언들을 모두 끌어올려서 해당 함수 유효 범위의 최상단에 선언하는 것을 말한다. 호이스팅이란? 자바스크립트 함수는 실행되기 전에 함수 안에 필요한 변수값들을 모두 모아서 유효 범위의 최상단에 선언한다.

자바스크립트 Parser가 함수 실행 전 해당 함수를 한 번 훑는다.

함수 안에 존재하는 변수/함수선언에 대한 정보를 기억하고 있다가 실행시킨다.

유효 범위: 함수 블록 {} 안에서 유효

즉, 함수 내에서 아래쪽에 존재하는 내용 중 필요한 값들을 끌어올리는 것이다.

실제로 코드가 끌어올려지는 건 아니며, 자바스크립트 Parser 내부적으로 끌어올려서 처리하는 것이다.

실제 메모리에서는 변화가 없다.

호이스팅의 대상

var 변수 선언과 함수선언문에서만 호이스팅이 일어난다.

var 변수/함수의 선언만 위로 끌어 올려지며, 할당은 끌어 올려지지 않는다.

let/const 변수 선언과 함수표현식에서는 호이스팅이 발생하지 않는다.

자바스크립트는 싱글 스레드 언어인가요? 멀티 스레드 언어인가요?

싱글스레드 언어입니다.

그런데, 왜 멀티 스레드 언어처럼 사용할 수 있는건가요?

5.TS

타입스크립트를 사용해본 경험이 있는가, 타입스크립트에 대한 본인의 생각과 도입시의 장점을

말해보세요.

Typescript는 동적타입언어인 Javascript의 약점을 보완하기 위해서 타입을 지정해주는 것이다.

타입이 필요한 이유는 결론은 메모리를 절약하기 위해서이다. 메모리에 저장된 것을 읽어들이고, 값을 메모리에 저장할때, 값이저장되어있는 것을 참조할때의 크기들을 알아야 하기 때문이다.

또한, 에러를 잡기가 쉬워지고, 다른 동료와 협업 할때 코드의 예측도 가능해지고, 코드에디터의 도움을 더 받을 수 있다. 리액트의 경우 (브라우저는 javascript밖에 모르기때문에) tsx파일을 javascript로 변환하는 트랜스파일링이 필요하다. 이때 변환하는 과정에서 에러를 잡을 수 가있다. 런타임에 오류를 잡는 것보다 훨~~~좋다!

라이브러리와 프레임워크의 차이점에 대해서 설명해보세요

라이브러리와 프레임워크의 차이는 자유도의 차이 인것 같다. 프레임워크는 짜여진 패턴이나 틀 기반에서 내가 코딩을 하는 것이고, 라이브러리는 내가 가져다 사용해서 자유롭게 사용하는 방식이다.

6.React

CRA란 무엇인가요?

CRA(create-react-app)은 React의 공식 CLI(명령줄 인터페이스)로서, SPA 개발 초기 세팅을 편리하게 할 수 있도록 한다. 애플리케이션을 개발할 때 create-react-app을 사용하면 개발자가 빌드 구성을 직접 하지 않아도 되기 때문에 코드에만 집중할 수 있어 효율적으로 개발을 할 수 있다.

CSR과 SSR의 차이점을 설명해보세요

CSR의 과정 : 서버가 브라우저에게 응답을 보냄 -> 브라우저는 JS를 다운 받음 -> 브라우저는 리액트를 실행 -> 페이지가 보여지고 상호작용 함

SSR의 과정 : 서버가 브라우저에게 HTML 응답 랜더링하기 위한 준비가 되었다고 보냄 -> 브라우저가 페이지랜더링, 페이지가 보여지고 브라우저는 JS 다운받음 -> 브라우저 리액트 실행 -> 페이지 상호작용 가능

CSR은 마지막 단계 전까지 화면에 보여지지않고 로딩중 / SSR은 미리 페이지가 보여진다.

즉, CSR은 초기로딩속도가 느리긴하지만, 화면전환에 있어서 클라이언트에서 이루어져서 빠른 전환이 가능

SSR은 초기로딩속도가 빨라서 사용자가 느끼기엔 좋지만, 동작은 하지않음. 그리고 화면전환에 있어서 서버에 요청해야하므로 서버에 부담을 줄 수 있음.

어떻게 더 좋다보다 서비스마다 사용자의 요구마다 다름.

리액트의 상태관리에 대해 알고 있는가? Redux를 사용해봤다면, 그것에 대한 설명

리액트에서 전역의 상태를 관리하기 위해서 사용하는 방법이다. 컴포넌트간의 상태들을 한군데다가 모아놓고 공유해서 사용하는 방식. 리액트의 상태관리는 Context API, Redux, MobX 등의 상태 관리가 있으며, Context API보다 Redux를 사용하는 이유는 대규모 개발에서 유지보수성이나 작업 효율을 높이기에는 Redux를 사용하는것이 좋기 때문에 많은 사람들이 Redux를 사용한다. 리액트 16.3이후 버전에서는 그래도 Context API가 개선되어 사용하기 좋아졌다.

Redux는 사실 다른 곳에서도 많이 쓰이는 기술이었지만, react-redux라는 것이 있어서 react에서 사용하기 좋아졌다.

만드는 순서는 액션 타입을 정하고, 액션 생성 함수를 만들고, 이 액션들을 사용하는 리듀서 함수 (초기상태 포함)를 만들고, index.js에서 스토어를 만들어 provider로 스토어를 props로 전달해준다.

가상돔 (virtual DOM)은 무엇인가요?

Virtual DOM은 실제 DOM 변화를 최소화 시켜주는 역할을 합니다.

먼저 브라우저는 HTML 파일을 스크린에 보여주기 위해 DOM 노드 트리 생성, 렌더트리 생성, 레이아웃, 페인팅 과정을 거칩니다. DOM 노드는 HTML의 각 엘리먼트와 연관되어 있기 때문에 HTML 파일에 20개의 변화가 생기면 DOM 노드가 변경되고 그 이후의 과정역시 20회 다시 이루어 집니다. 작은 변화에도 매우 복잡한 과정들이 다시 실행되기 때문에 DOM 변화가 잦을 경우 성능이 저하됩니다.

Virtual DOM은 뷰에 변화가 있다면, 그 변화가 실제 DOM에 적용되기 전에 Virtual DOM에 적용 시키고 최종 결과만 실제 DOM에 전달합니다. 따라서 20개의 변화가 있다면 Virtual DOM은 변화된 부분만 가려내어 실제 DOM에 전달하고 실제 DOM은 그 변화를 1회로 인식하여 단 한번의 렌더링 과정만 거치게 됩니다.

7.기타

1. React와 Vue의 차이점은?
2. React의 LifeCycle을 아는대로 답변해주세요.
3. Redux와 useContext의 차이점은?
4. React의 상태변화 감지에 대해 아는대로 답변해주세요.
5. React에서 컴포넌트란 무엇인가요
6. React에서 상태관리 라이브러리를 1개만 선택하라고 한다면 무엇을 선택할건가요? 그 이유는?
7. Dom과 Virtual Dom의 차이점은 무엇인가요.
8. 렌더링 동작 방식에 대해 말씀해주세요.
9. React 이외 프레임워크 혹은 라이브러리 사용 경험이 있나요
10. SPA란?
11. ES6 문법에 대해서 아는대로 답변해주세요.
12. 함수형 컴포넌트와 Class형 컴포넌트 방식의 차이점에 대해서 답변해주세요.
13. Rest API에 대해서 말씀해주세요.
14. SCSS, CSS의 차이점에 대해서 답변해주세요.
15. null과 undefined의 차이점에 대해서 말씀해주세요.