

질문	답변
검색창에 <b>www.google.com</b> 을 검색한다면 어떤 일이 일어날까요?	
3-way handshake에 대해 설명해주세요.	<p><b>3-way handshake</b>는 <b>TCP/IP</b> 프로토콜로 통신하기전, 정확한 정보 전송을 위해 상대방의 컴퓨터와 세션을 수립하기위해 (연결을 하는) 과정입니다. (<b>TCP 초기화</b>)</p> <p>클라이언트가 서버에 '접속을 요청하는' <b>SYN</b>패킷을 보내게 되면 서버에서는 '요청을 수락하는' <b>ACK</b> 패킷을 포함하여 <b>SYN + ACK</b> 패킷을 다시 클라이언트에게 발송합니다.</p> <p>클라이언트는 다시 이를 수신한 후 서버에게 <b>ACK</b>를 발송하면 클라이언트-서버 간의 연결이 이루어지게 되고 데이터를 주고받을 수 있게됩니다.</p>
4-way handshake에 대해 설명해주세요.	<p><b>3-way handshake</b>로 연결을 수립했다면, <b>4-way handshake</b>로 연결을 종료합니다.</p> <p>해당 과정은</p> <ol style="list-style-type: none"> <li>1. 클라이언트에서 연결 종료를 하겠다는 <b>FIN</b> 세그먼트를 보냅니다.</li> <li>2. 서버에서는 <b>FIN</b> 세그먼트를 받았는 응답에 대한 <b>ACK</b>를 클라이언트에 보냅니다. 이때는 서버 내의 <b>process</b>에게 끝내겠다는 신호를 보내지만 아직 <b>process</b>가 <b>close</b>되지 않을 수 있습니다.</li> <li>3. 서버의 <b>process</b>에서 <b>passive close</b>를 신호를 받으면 서버에서는 다시 클라이언트에게 <b>FIN</b> 세그먼트를 보냅니다.</li> <li>4. 클라이언트가 서버로 <b>ACK</b>를 보내게 되고 서버가 이를 받게 되면 연결이 종료됩니다.</li> </ol>

reflow, repaint에 대해 설명해주세요.	<p>reflow와 repaint는 주로 브라우저 크기 변경이나 웹 인터렉션에 의해서 일어나게 되는데,</p> <p>reflow는 브라우저 렌더링 과정에서 재배치가 이루어 지는 것을 말하고 이는 비용이 비싸다.</p> <p>repaint는 페인트 과정이 다시 발생하게 되는것을 말하고 상대적으로 비용이 싸지만 둘 다 최소화 하는것이 좋다.</p> <p>이들을 줄이기 위한 방법으로는 인라인스타일 지양, 애니메이션이 있는 노드를 fixed나 absolute로 이용하고 애니메이션의 경우 transform로 스타일하는방법, CSS trigger와 같은 사이트에서 확인 후 적절한 속성을 사용</p>
layout과정과 paint과정 중 어떤 것이 더 성능적으로 유리할까요?	<p>paint과정이 성능적으로 더 유리하다고 생각합니다.</p> <p>layout과정은 렌더트리를 실제로 어떻게 배치시킬지 계산하기 때문에 성능적으로 더 좋지 않고 이를 줄이기 위해 노력해야합니다.</p>
http와 https의 차이점에 대해 설명해주세요.	<p>HTTP는 Hyper Text Transfer Protocol의 약자로 서버-클라이언트 모델을 따르면서 request/response 구조로 웹상에서 정보를 주고 받을 수 있는 프로토콜입니다.</p> <p>HTTP의 경우 텍스트로 정보를 송/수신 하기때문에 중간에 인터셉트를 할 경우 정보가 노출될 수 있어, 이를 보완하기위해 HTTP에 암호화를 추가한 프로토콜이 HTTPS입니다.</p>
웹 브라우저나 node.js 환경은 멀티 스레드 환경입니다. 자바스크립트는 메인 스레드가 싱글 스레드인데 해당 환경에서 어떻게 동작되는지 설명해주세요.	
싱글스레드 기반인데 어떻게 멀티스레드같이 보일까?	
이벤트 루프의 역할에 대해 설명해주세요.	
태스크 큐를 3가지로 분류한다면 어떤 큐들이 있는지와 각 큐들의 예시를 말해주세요.	
3가지 태스크큐의 우선순위에 대해 설명해주세요.	
setTimeout과 requestAnimationFrame에 대해 비교해주세요.	
setTimeout을 사용하여 특정 로직을 1분 후에 실행한다고 가정했을 때 이때 정확히 1분 후에 로직이 동작한다고 신뢰할 수 있을까요?	
Critical Rendering Path (CRP)라는 용어에 대해 들어보셨나요?	

Composite 과정은 어떤 일을 처리하는 단계일까요?	
<b>display: none</b> , <b>visibility: hidden</b> 은 모두 요소가 보이지 않는 속성인데 어떤 것을 사용하는 것이 성능 측면에서 좋을지와 어떤 이유 때문에 그렇게 생각하시는지 설명해주세요.	
<b>==</b> 와 <b>===</b> 는 모두 같음을 나타내는 연산자입니다. 두가지 연산자의 차이점을 얘기해주세요.	
<b>0 === -0</b> 의 결과를 설명해주세요.	
<b>1/0 === 1/-0</b> 결과에 대해 설명해주세요.	
<b>falsy value</b> 에 대해 설명하고 <b>falsy value</b> 에 해당되는 예시를 보여주세요.	
<b>0.1 + 0.2 ≠ 0.3</b> 이 현상이 왜 그런지 설명해주세요.	
자바스크립트의 타입에 대해 설명해주세요.	
<b>null</b> 과 <b>undefined</b> 의 특징과 차이를 설명해주세요.	
자바스크립트의 <b>BigInt</b> 에 대해 설명해주세요.	
<b>Symbol</b> 객체에 대해 설명해주세요.	
얕은 복사와 깊은 복사를 설명하고 깊은 복사를 할 수 있는 방법에 대해 얘기해주세요.	
불변성이란 무엇이고 코드를 짤 때 불변성이 중요한 이유에 대해서 설명해주세요.	
<b>var</b> 와 <b>let</b> 의 차이점에 대해 설명해주세요.	
<b>const</b> 선언을 지향해야하는 이유에 관해 설명해주세요.	
TDZ에 대해 설명해주세요.	
<b>this</b> 의 역할에 대해 설명해주세요.	
<b>this</b> 를 바인딩하는 방법이 있을까요? 만약 바인딩하는 방법이 있다면 어떤 것들이 있을까요?	
<b>this</b> 암시적 바인딩과 명시적 바인딩에 대해 설명해주세요.	
일반 함수와 화살표 함수의 <b>this</b> 의 차이에 대해 설명해주세요.	
호이스팅은 왜 일어나는 것이고 실행 컨텍스트와는 무슨 관계가 있는지 설명해주세요.	
콜 스택이 무엇인지 설명해주세요.	
스코프 개념에 대해 설명해주시고 스코프 체인에 관해서 설명해주세요.	
일급 객체와 일급 함수에 대해 설명하고 자바스크립트가 일급 객체이기 때문에 가능한 것들에 대해 설명해주세요.	
순수함수를 설명하고 순수함수의 이점을 아는대로 열거해주세요.	

즉시 실행 함수가 무엇인지 얘기해주시고 해당 함수를 사용하는 이유에 대해 설명해주세요.	
이벤트 버블링과 캡처링에 대해 설명해주세요.	
이벤트 <b>delegation</b> (위임)에 대해 설명해주시고 이것은 버블링과 캡처링 중 어떤 것을 사용한 패턴인지 설명해주세요.	
이벤트 위임을 사용하는 예시를 설명해주세요.	
콜백함수를 사용하다보면 콜백헬 또는 콜백 지옥이라는 것을 경험하게 됩니다. 이를 해결하는 방법은 어떤 것들이 있을까요?	
<b>Promise</b> 는 3가지 상태를 가지고 있습니다. 각각의 상태에 대해 설명해주세요.	
API 서버 통신을 하다보면 <b>CORS (cross origin resource sharing, same origin policy)</b> 관련 이슈를 볼 수 있습니다. <b>CORS</b> 에 대해 설명해주시고 해결할 수 있는 방법으로는 어떤 것들이 있는지 설명해주세요.	
<b>XSS (Cross-Site Scripting)</b> 공격에 대해 설명해주시고 해당 공격을 방지할 수 있는 방법에 대해 설명해주세요.	
명령형 프로그래밍과 선언적 프로그래밍의 차이를 설명해주세요.	
<b>HTTP</b> 의 상태코드에는 어떤 것들이 있을까요?	
<b>HTTP</b> 의 각 메소드는 어떤 의도를 전달할까요?	
<b>DOCTYPE</b> 의 역할에 대해 설명해주세요. 만약 <b>html</b> 파일에 <b>DOCTYPE</b> 을 선언하지 않는다면 어떤 일이 벌어질까요?	
시멘틱 태그에 대해 설명해주시고 시멘틱 태그를 사용한 예시에 대해 말씀해주세요.	
<b>html</b> 파일은 보통 <b>&lt;head&gt;</b> 와 <b>&lt;body&gt;</b> 태그로 이루어집니다. 이때 각 태그들의 역할을 설명해주세요.	
<b>&lt;head&gt;</b> 태그 안에는 어떤 태그들이 들어갈까요?	
<b>meta</b> 태그의 역할과 해당 태그가 왜 필요한지 설명해주세요.	
웹 접근성에 대해 설명해주시고 지켜야하는 이유를 설명해주세요.	
웹 접근성을 지킬 수 있는 예시를 설명해주세요.	
<b>&lt;strong&gt;</b> 태그와 <b>&lt;b&gt;</b> 태그의 차이, <b>&lt;em&gt;</b> 태그와 <b>&lt;i&gt;</b> 태그 사이의 공통점은 뭐가 있을까요?	
<b>script</b> 태그에 포함되는 <b>defer, async</b> 에 대해 들어보셨나요? 속성이 들어있을 때 어떤 일이 일어나는지 설명해주세요.	
뷰 포트에 대해 설명해주세요.	

반응형 웹과 적응형 웹에 대해 설명해주세요.	
WAI-ARIA (Web Accessibility Initiative – Accessible Rich Internet Applications) 또는 ARIA 에 관련해서 설명해주세요	
IR기법에 대해 설명해주세요.	
이미지 스프라이트에 대해 설명해주세요.	
아래 코드들의 출력값을 작성해주세요.	
<pre> console.log("start");  setTimeout(() =&gt; {   console.log("macrotask 1"); }, 0);  Promise.resolve()   .then(() =&gt; {     console.log("microtask 1");   })   .then(() =&gt; {     console.log("microtask 2");   });  requestAnimationFrame(() =&gt; {   console.log("requestAnimationFrame callback"); });  console.log("end"); </pre>	
<pre> // 일반함수와 화살표함수에서의 this const obj = {   name: "John",   sayName() {     console.log(this.name); // ??   },   sayNameArrow: () =&gt; {     console.log(this.name); // ??   } }  obj.sayName(); obj.sayNameArrow(); </pre>	

<pre>// 엄격모드에서의 this  "use strict"; function myFunction() {   return this; }  console.log(myFunction());  var count = 0; function addCount() {   this.count = 100;   count++; } addCount(); console.log(count); // ??</pre>	
<pre>// 이벤트 핸들러에서의 this var btn = document.querySelector("#btn"); btn.addEventListener("click", function () {   console.log(this); });</pre>	

// 비동기 함수에서의 this

```
class Counter {  
  constructor() {  
    this.count = 0;  
  }  
}
```

```
  start() {  
    console.log(this); // ??
```

```
    setInterval(function() {  
      console.log(this) // ??  
      console.log(this.count++) // ??  
    }, 1000);
```

```
    setInterval(() => {  
      console.log(this) // ??  
      console.log(this.count++) // ??  
    }, 1000);  
  }  
}
```

```
const counter = new Counter();
```