

질문	답변
SSR이 무엇인가요?	SSR은 서버 측에서 페이지를 렌더링 하는 기술입니다. 즉, 사용자가 페이지를 요청할 때, 서버에서 이미 완성된 페이지를 제공합니다.
CSR은 무엇인가요?	웹에서 페이지를 동적으로 생성하고 렌더링 하기 위해 클라이언트에서 자바스크립트를 사용하는 기술입니다. 클라이언트가 웹 페이지를 요청하면 서버는 초기 HTML 템플릿과 JavaScript 파일을 제공하고, 이후에 웹 브라우저는 JavaScript를 실행하여 페이지 내용을 동적으로 업데이트합니다.
CSR의 장점과 단점은 무엇인가요?	초기 로딩 속도는 느리지만 이후에는 서버에 다시 요청할 필요없이 클라이언트 내에서 작업이 이루어 지므로 빠릅니다. 하지만 SEO 측면에서는 불리합니다.  초기 요청에는 빈 페이지나 로딩 화면이 나타날 수 있습니다. 예를 들어 네트워크 상황이 좋지 않다면 CSR을 이용할 경우 사용자들은 글을 보기 전에 상당 시간 하얀 화면을 봐야 할 수도 있다.
SSR의 장점은 뭐가 있나요?	SSR을 통해 모든 데이터가 매핑된 서비스 페이지를 클라이언트에게 바로 보여줄 수 있습니다. 초기 로딩속도가 빠르고 더불어 SEO 또한 쉽게 구성할 수 있다.
SSR의 단점은 무엇이 있을 수 있나요?	뷰 변경시에는 서버에 계속 요청을 해야 하므로 서버 부하가 커질 수 있습니다. 또 개발이나 배포 복잡성이 상승할 수 있습니다. 간단한 데이터 수정에도 서버를 거쳐야한다.
SSR을 사용함으로써 왜 SEO 이점을 얻을 수 있나요?	검색엔진들이 이미 서버쪽에서 만들어진 html을 크롤링 할 수 있기 때문에 검색에 유리해진다. CSR에서는 기본적으로 브라우저가 서버에서 받아오는 최초의 HTML은 고작 <div id="app"></div> 혹은 <div id="root"></div> 한 줄 인데, 이렇게 될 경우 구글 검색엔진이 사이트의 내용을 파악할 수 없다는 것입니다. 물론 meta 태그를 이용한다면 어느정도 극복할 수 있으나 역시 정교한 색인(indexing)에는 한계가 존재합니다.

어떤 상황에서 SSR을 사용해야 하는지 설명해주세요	<p><b>정적 콘텐츠 로드</b>          서버 PC는 대개 연산처리능력이 좋고 네트워크 속도가 빠른 컴퓨터이다. 때문에 예상 처리 요청 건수를 처리하는 데에 우리가 없고, 결과적으로 <b>Server</b>에서 콘텐츠를 요청하고 받아오는 속도가 비교적 빠르다.          반면에 <b>Client System</b>은 제한된 컴퓨팅 성능을 가졌기 때문에 동적 콘텐츠를 가져오고 렌더링하는데 시간이 더 걸릴 수 있다. 콘텐츠를 렌더링하는데 소요되는 전체 시간이 더 오래걸리기 때문에 웹사이트가 <b>반복적인 동적 콘텐츠 렌더링</b>을 포함하는 경우 <b>CSR</b>보다 <b>SSR</b>이 더 나은 선택이다.</p> <p><b>검색엔진 최적화</b>          비즈니스 쇼핑몰의 경우 검색 엔진에 노출되고 유입이 되는 것이 유리하고          블로그 또한 경우 검색 엔진에 많이 <b>노출</b>되면서 다른 웹사이트에 비해 사용자와 상호작용이 많지 않기 때문에  <b>SSR</b>이 합리적인 선택이 될 수 있습니다.</p>
CSR과의 차이점은 무엇인가요?	<p><b>CSR</b>은 브라우저에서 <b>JavaScript</b>로 페이지를 렌더링하는 반면, <b>SSR</b>은 서버에서 <b>HTML</b> 페이지를 렌더링하여 클라이언트에 전달합니다.  <b>CSR</b>은 초기 로딩 속도가 느리지만 동적으로 페이지를 업데이트할 수 있고, <b>SSR</b>은 초기 로딩 속도가 빠르지만 서버 부하가 있을 수 있습니다.</p>
어떻게 서버 측에서 렌더링하는지 간략하게 설명해주세요	<ol style="list-style-type: none"> <li>1. 사용자가 웹사이트에 방문</li> <li>2. 브라우저에서 서버에 콘텐츠 <b>요청</b></li> <li>3. 서버는 페이지에 필요한 <b>모든 데이터 및 css까지 적용</b>해서 렌더링 준비를 마친 <b>HTML</b>과 자바스크립트 코드를 브라우저에 <b>반환</b>한다.</li> <li>4. 브라우저는 전달받은 페이지를 띄우고 자바스크립트 코드를 다운로드하고 <b>HTML</b>에 자바스크립트 <b>로직을 연결</b>한다.</li> </ol>
SSR을 사용하면 어떻게 초기 페이지 로딩 시간을 개선할 수 있을까요?	초기 페이지 로딩 시간을 개선하기 위해 코드 스플리팅, <b>CDN</b> 캐싱, 서버 사이드 캐싱, 이미지 최적화 및 비동기 데이터 로딩과 같은 최적화 기술을 사용할 수 있습니다.
코드 스플리팅(Code Splitting)이 SSR과 어떤 관련이 있나요?	코드 스플리팅은 번들 크기를 줄이는 기술로, <b>SSR</b> 과 함께 사용하면 초기 로딩 속도를 개선할 수 있습니다. 페이지에 필요한 <b>JavaScript</b> 코드만 클라이언트로 전송하여 효율적인 로딩을 가능하게 합니다.
SEO	<p><b>SEO</b>는 "검색 엔진 최적화"의 약어로,          웹 사이트나 웹 페이지를 검색 엔진 결과 페이지(<b>SERP</b>)에서 높은 순위로 표시되도록 최적화하는 프로세스를 가리킵니다.          검색 엔진은 사용자가 정보를 찾을 때 웹에서 수많은 페이지를 검색하여 가장 관련성이 높은 결과를 표시하는데, 이때 <b>SEO</b>는 웹 사이트나 페이지가 검색 결과에서 상위에 나타나도록 도와줍니다.</p>
	그런데 브라우저가 발전하는 과정에서 Javascript 라는 언어 또한 점점 성숙해졌고, Javascript로 할 수 있는 일들이 점점 많아지게 되었습니다. 그러면서 DOM을 점점 더 정교하게 다뤄야했고, jQuery 같은 라이브러리로는 한계가 있었습니다. 그래서 구글에서 Angular를 만들었고, 이어서 페이스북에서는 Component 기반으로 개발할 수 있는 React를 만들었습니다. 다시 Angular와 React의 장점만 수용하여 만든 Vue도 등장했죠.