

2.1 SSR의 장점

- 더 안전한 웹 애플리케이션
 - 쿠키 관리, 주요 API, 데이터 검증 등과 같은 작업을 서버에서 처리
- 더 뛰어난 웹 사이트 호환성
 - 클라이언트 환경이 자바스크립트를 사용하지 못하거나 오래된 브라우저라도 웹 페이지 제공
- 더 뛰어난 SEO

2.2 CSR의 장점

- 네이티브 애플리케이션처럼 느껴지는 웹 애플리케이션
- 쉬운 페이지 전환
- 지연된 로딩과 성능
- 서버 부하 감소

React.useEffect 훅

컴포넌트가 마운트 된 후 해당 기능을 실행(브라우저에서 실행)

Process.browser 변수

```
function IndexPage() {  
  const side = process.browser ? 'client' : 'server';  
  
  return <div>You're currently on the {side}-side.</div>;  
}  
  
export default IndexPage;
```

하지만 process.browser 에 대한 Vercel 팀의 지원은 곧 중단될 예정이므로 대신 좀 더 정확한 의미를 갖는 typeof window를 사용할 수 있다.

```
function IndexPage() {  
  const side = typeof window === "undefined" ? 'server' : 'client';  
  
  return <div>You're currently on the {side}-side.</div>;  
}  
  
export default IndexPage;
```

`typeof window`를 서버에서 실행하면 그 값은 문자열 "undefined"가 되며, 그렇지 않으면 클라이언트에서 실행하는 것임을 알 수 있다.

SSG (동적 컴포넌트 로딩)의 장점

일부 또는 전체 페이지를 빌드 시점에 미리 렌더링

- 쉬운 확장
- 뛰어난 성능
- 더 안전한 API 요청

ISR(증분 정적 재생성)

- Next.js가 어느 정도의 주기로 정적 페이지를 다시 렌더링하고 해당 내용을 업데이트 할지 정할 수 있다.
-