**쿠키 vs 웹 스토리지**

-쿠키 : 한 개당 4kb만 저장가능, 20개로 제한됨, 보안에 취약함

-웹 스토리지 : 5mb 저장가능. 쿠키보다 안전하고 속도도 빠름.

-웹 스토리지 안에서 데이터는 키/값으로 저장됨

**localStorage와 sessionStorage**

-localStorage : 만료 날짜가 없는 데이터를 저장함. 데이터는 브라우저가 닫힐 때 삭제되지 않으며

1년 동안 사용가능하다. 도메인이 다르면 서로의 로컬 스토리지에 접근할 수 없다.

-sessionStorage : 각 세션(하나의 윈도우)마다 데이터가 별도로 저장됨. 해당 세션이 종료되면 데이터가

사라짐. 같은 도메인에 연결했어도 브라우저 윈도우가 다르면 서로 다른 영역을 사용함

새로 윈도우를 열어서 동일한 도메인에 접속해도 세션스토리지는 서로 다르다.

-사용법은 책 참고

**파일 API**

-웹 브라우저가 사용자 컴퓨터에 잇는 로컬 파일을 읽어올 수 있도록 해주는 API

-주로 사용자가 파일을 선택해서 원격서버로 전송하는 작업을 수행함 but 보안상의 문제로 현재는

사용자가 input 요소를 통하여 선택한 파일 또는 드래그 앤 드롭 방식을 이용해 사용자가

선택한 파일로 한정함.

**애플리케이션 캐시**

-애플리케이션이 사용하는 파일을 클라이언트의 캐시에 저장하면, 웹 애플리케이션의 오프라인 버전을

만들 수 있다. -> 인터넷에 연결하지 않고 실행될 수 있다.

**웹 소켓**

-웹 소켓은 하나의 소켓 위에서 동작하며 HTML5와 자바스크립트를 통하여 사용할 수 있다.

-웹 소켓을 사용하게 되면, HTTP의 구속에서 벗어나 TCP/IP가 제공하는 모든 기능 사용가능

-채팅과 같은 실시간 양방향 통신을 할 수 있다.

**서버 전송 이벤트**

-웹페이지가 서버로부터 자동적으로 데이터를 전달받을 수 있는 기법

-서버가 주기적으로 보내주는 주식 가격 업데이트, 뉴스 피드 등을 받을 때 사용함

-이전에는 업데이트가 있는지 웹 페이지가 서버에 항상 문의해야 했지만, 서버 전송이벤트를 사용하면

업데이트가 자동으로 진행된다.