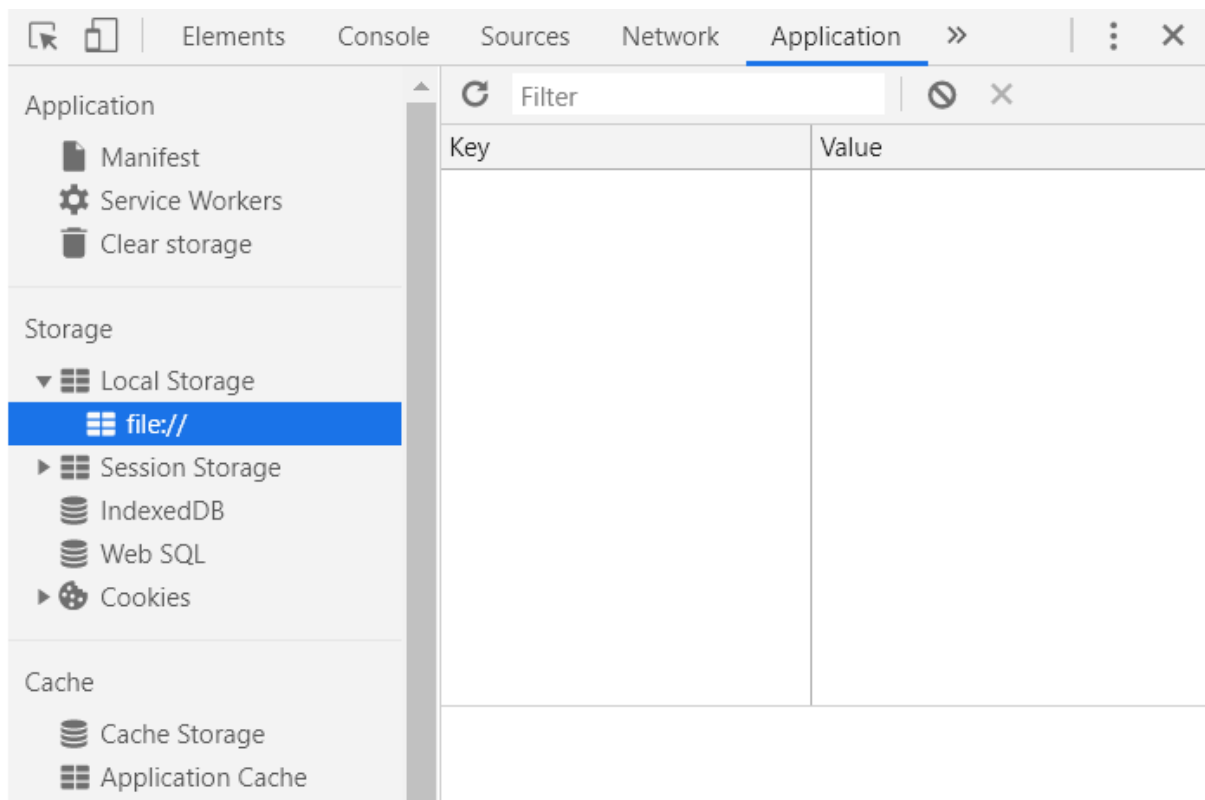




Web storage

🕒 생성일	@2023년 8월 2일 오전 9:58
☰ 태그	IndexedDB localStorage sessionStorage

Web storage란 데이터를 서버가 아닌 클라이언트에 저장 할 수 있게 해주는 기능이다. **Local Storage**, **Session Storage**, **Index DB** 등이 Web Storage에 속한다. Data를 클라이언트에 저장하기 때문에 이를 유지하고 활용하는데 유용하다.



Local Storage

`localStorage` 개체는 사용자의 웹 브라우저에 데이터를 저장하는 데 사용되는 웹 저장소 API입니다. 브라우저를 닫았다가 다시 열어도 데이터가 유지됩니다.

Description

쿠키와 달리 `localStorage` 는 만료되지 않으며 저장 한도가 더 큼니다(대부분의 최신 브라우저에서 약 5MB). 데이터는 키-값 쌍으로 저장되며 클라이언트 측에서만 사용할 수 있습니다. 즉, HTTP 요청에서 서버로 전송되지 않습니다.

특징

- `localStorage` 개체는 만료 시간 없이 데이터를 저장합니다. 즉, 브라우저를 닫아도 데이터가 삭제되지 않으며 수동으로 지우지 않는 한 지속적으로 사용할 수 있습니다.
- `localStorage` 데이터는 안전하지 않습니다. 암호나 신용카드 번호와 같은 민감한 정보를 저장하는 데 사용해서는 안 됩니다.
- `localStorage` 는 동기식이므로 많은 양의 데이터를 읽거나 매우 자주 액세스하는 경우 성능 문제가 발생할 수 있습니다.
- `localStorage` 에 저장된 데이터는 동일한 도메인에서만 접근할 수 있습니다. 즉, 도메인이 다르면 다른 `localStorage` 를 사용합니다.

Session Storage

`sessionStorage` 개체는 Web Storage API의 일부이며 단일 세션 내에서 데이터를 저장하는 데 사용됩니다. 데이터는 페이지 세션 기간 동안만 사용할 수 있으며 탭을 닫으면 삭제됩니다.

Description

`sessionStorage` 는 `localStorage` 와 비슷하지만 수명과 범위가 다릅니다. 브라우저에서 열리는 각 새 탭 또는 창에는 자체 독립 세션이 있으므로 별도의 `sessionStorage` 데이터가 있습니다. `localStorage` 와 마찬가지로 `sessionStorage` 는 데이터를 키-값 쌍으로 저장하며 쿠키에 비해 저장 제한이 더 큼니다(대부분의 최신 브라우저에서 약 5MB). 그러나 `localStorage` 와 달리 `sessionStorage` 데이터는 영구적이지 않습니다.

특징

- `sessionStorage` 는 탭에 따라 분리됩니다. 동일한 사이트에 속하더라도 다른 탭이나 창과 데이터를 공유하지 않습니다.

- `localStorage` 와 마찬가지로 `sessionStorage` 데이터는 안전하지 않습니다. 민감한 정보를 저장하는 데 사용해서는 안 됩니다.
- `sessionStorage` 는 동기식이므로 매우 자주 액세스하거나 많은 양의 데이터가 저장되는 경우 잠재적으로 성능 문제가 발생할 수 있습니다.
- `localStorage` 와 마찬가지로 `sessionStorage` 역시 동일한 도메인에서만 접근할 수 있습니다.

IndexedDB

IndexedDB는 파일/BLOB를 포함하여 상당한 양의 구조화된 데이터를 클라이언트 측에 저장하기 위한 저수준 API입니다. 이는 사용자의 브라우저에 데이터를 유지하여 애플리케이션이 온라인과 오프라인 모두에서 작동할 수 있도록 하는 방법입니다.

Description

`IndexedDB` 는 SQL 기반 RDBMS와 같은 트랜잭션 데이터베이스 시스템입니다. 그러나 SQL 기반 RDBMS와 달리 모든 `IndexedDB` 트랜잭션은 자동 커밋됩니다. `IndexedDB` 는 객체 지향입니다. IndexedDB에서는 열 데이터의 행이 아닌 개체를 저장합니다. 이 데이터의 고성능 검색을 허용하는 인덱스를 사용합니다.

특징

- `IndexedDB` API는 강력하지만 간단한 경우(예: 간단한 키-값 저장)에는 너무 복잡해 보일 수 있습니다.
- API는 비동기식입니다.
- 복잡한 쿼리는 API의 단순성으로 인해 구성하기 어려울 수 있습니다.
- `localStorage` 및 `sessionStorage` 와 마찬가지로 `IndexedDB` 데이터는 안전하지 않으며 사용자가 언제든지 지울 수 있습니다.

비교정리

데이터 보관

- `localStorage` `IndexedDB` : 데이터는 브라우저를 닫아도 유지됩니다.
- `sessionStorage` : 데이터는 브라우저 세션이 유지되는 동안에만 유지됩니다. 브라우저를 닫으면 데이터가 삭제됩니다.

데이터 저장 위치

- `localStorage` `sessionStorage` : 브라우저 내부에 저장됩니다.
- `IndexedDB` : 브라우저 외부의 별도의 데이터베이스에 저장됩니다.

데이터 저장 방식

- `localStorage` `sessionStorage` : 데이터는 키-값 쌍으로 저장됩니다.
- `IndexedDB` : NoSQL 데이터베이스로서, 객체를 저장하고 복잡한 쿼리를 수행할 수 있습니다.

localStorage 사용예시

`localStorage` 는 일반적으로 브라우저 세션 및 다시 로드에서 중요하지 않은 데이터를 유지하는 데 사용됩니다.

사례 예시: 사용자 기본 설정

사용자가 UI 테마를 사용자 정의할 수 있는 애플리케이션을 고려하십시오. `localStorage` 를 사용하면 브라우저에 직접 기본 설정을 저장할 수 있습니다. 다음에 사용자가 방문하면 `localStorage` 에서 테마를 로드하여 일관되고 개인화된 경험을 만들 수 있습니다.

sessionStorage 사용예시

`sessionStorage` 는 일반적으로 단일 브라우저 세션의 여러 페이지에서 유지하려는 임시 데이터를 저장하는 데 사용됩니다.

사례 예시: 다단계 양식의 데이터

설문조사와 같은 다단계 양식이 있는 경우 `sessionStorage` 에 사용자의 답변을 저장할 수 있습니다. 사용자가 페이지를 다시 로드하거나 양식에서 앞뒤로 탐색하는 경우 이전 답변이 손실되지 않습니다.

인덱스 DB 사용예시

`IndexedDB` 는 더 많은 양의 구조화된 데이터를 저장해야 할 때 사용됩니다. 또한 데이터를 검색하거나 쿼리해야 할 때 사용됩니다.

사례 예시: 고객 기록 저장 및 쿼리

비즈니스 애플리케이션의 경우 `IndexedDB` 에 고객 레코드를 저장할 수 있습니다. 이를 통해 데이터를 쿼리하고 복잡한 검색을 수행하며 오프라인에서도 작업할 수 있습니다.