

Storage

저장하기

```
localStorage.setItem('key', 'value');
```

가져오기

```
localStorage.getItem('key');
```

삭제하기

```
localStorage.removeItem('key');
```

전체 삭제하기

```
localStorage.clear();
```

코드 분석

▼ 첫 번째 페이지

```
<!DOCTYPE html>  
<html lang="en">
```

```

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
  <h1>Hello World</h1>
  <input type="text" id="text_id">
  <input type="button" value="전송" onclick="send()">

  <script>
    function send() {
      const value = document.getElementById('text_id').value;
      localStorage.setItem('text', value);
      window.location.href = 'second.html';
    }
  </script>
</body>
</html>

```

▼ 🍀 두 번째 페이지

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body onload="load()">
  <h1 id="page_title"></h1>
  <input type="button" value="다음페이지" onclick="move()">
  <script>

```

```

        function load() {
            console.log(localStorage.getItem('text'));
            document.getElementById('page_title').textContent = localStorage.getItem('text');
        }

        //다음페이지
        function move() {
            location.href = './third.html';
        }
    </script>
</body>
</html>

```

▼ 🍀 세 번째 페이지

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body onload="load()">
    <h1 id="page_title">세번째 페이지</h1>
    <h1 id="location"></h1>
    <script>
        function load() {
            console.log(localStorage.getItem('text'));
            document.getElementById('location').textContent = localStorage.getItem('text');
        }
    </script>
</body>
</html>

```



Cookie와 Location의 차이

1. 저장 용량

대상	용량
쿠키	4KB
localStorage	5MB

2. 서버 통신

대상	설명
쿠키	쿠키는 서버와 클라이언트 간의 요청과 응답에 포함된다. 서버가 클라이언트의 상태를 지속적으로 파악할 때 사용. 네트워크의 트래픽을 증가할 수 있다.
localStorage	순수하게 클라이언트측에서만 데이터를 저장하고 관리

3. 유효기간

대상	설명
쿠키	만료 날짜/시간을 설정할 수 있다.
localStorage	만료 날짜가 없다. 사용자가 브라우저의 캐시를 지워야만 가능하다.

4. 접근성

대상	설명
쿠키	http와 https 모두 사용 가능하다. 설정에 따라 서브도메인간에 공유가 가능
localStorage	같은 도메인 및 프로토콜에서만 접근이 가능. 서브도메인 간 공유가 기본적으로 제한

5. 보안

대상	설명
쿠키	보안 취약성이 존재, HttpOnly를 사용하여 JS의 접근을 막는다. Secure 속성을 사용하여 Https에서만 사용하도록 제한
localStorage	XSS 공격에 취약하다. 보안에 대한 기능을 제공하지 않는다.

? 언제 사용하는가?

쿠키: 세션관리, 사용자 인증, 사용자 선호 설정 등 서버가 관리해야 할 필요가 있는 정보를 저장.

localStorage: 비교적 큰 양의 데이터를 저장, 서버와 무관하게 클라이언트 측에서만 사용하는 데이터를 저장할 때 유용하다.