

PORTFOLIO

프론트엔드 개발자 편상건 포트폴리오

이름	편 상 건
생년월일	1996. 02. 19
이메일	hubintheroot@gmail.com
전화번호	010-8724-9661
주소	서울시 영등포구

PORTFOLIO

CONTENTS

01. Vanilla JS로 TODO List 만들기 JavaScript 사용 경험 및 숙련도 향상을 위하여 진행

Project 1.

Vanilla JS로 TODO List 만들기

About project

JavaScript의 경험과 숙련도 향상을 위해 진행한 프로젝트입니다.

Todo List의 보존을 위해 LocalStorage를 활용했습니다.

사용 편의성을 고민하여 할 일 목록을 얼마나 완료했는지, 직관적으로 볼 수 있도록 완료율을 백분율로 나타냈습니다.



Introduce project

작업 기간	2023. 06 ~ 2023. 06 (2일)
인력 구성(기여도)	FE 1명 (FE 기여도 100%)
프로젝트 목적	JavaScript의 경험과 숙련도 향상을 위해 진행된 프로젝트입니다.
프로젝트 내용	할 일 목록을 추가, 제거, 완료로 처리하여 일정관리를 지원하는 웹 서비스
주요 업무 및 상세 역할	1) 서비스 기획부터 프론트엔드 기능 구현까지 전담하여 업무 수행 2) LocalStorage 처리 및 저장된 data 처리 기능 구현 3) 완료한 Todo list를 백분율로 표시하는 기능 구현
사용언어 및 개발 환경	HTML5, CSS3, JavaScript, VScode, live server
참고 자료	-

Main work ① 할 일 목록 작성



• 기능 소개

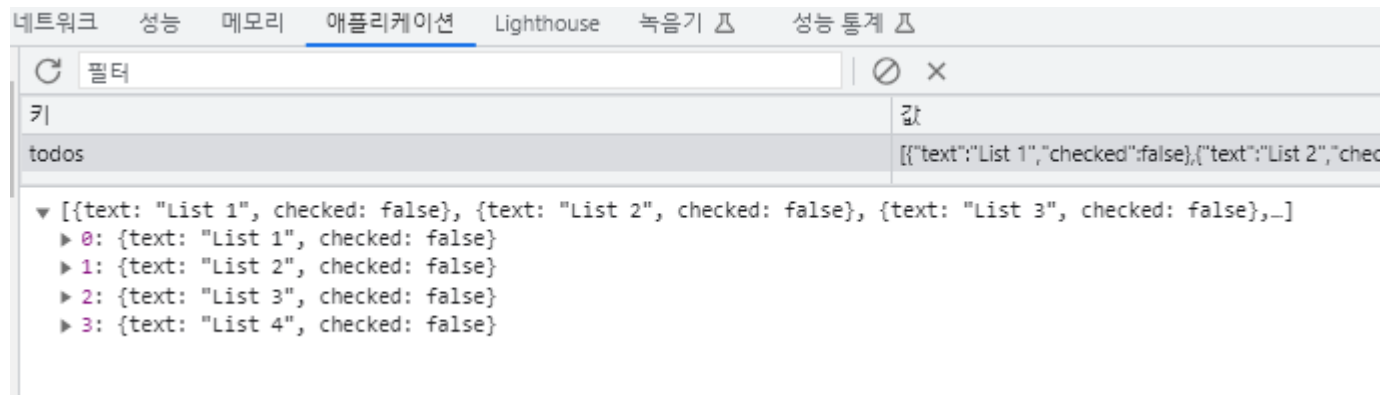
- 할 일 목록 작성 / Enter Key 또는 추가하기 버튼 클릭 시 할 일 목록 추가

• 작업 내용

- 입력된 텍스트 처리
 - > 정규식 사용으로 공백 예외처리
- Enter Key 또는 추가하기 버튼 클릭 이벤트
 - > addEventListener 사용
 - > e.key로 Enter Key 여부 체크

```
function addTodo(){
  if (!input_todo.value || input_todo.value.replace(/\s/g, '') === ''){ // 정규식으로 공백만 존재할 경우를
    제외
    input_todo.value=''
    return 0
  }
  todos.push({'text': input_todo.value, 'checked': false});
  saveData();
  createTodoList();
  refreshProgressBar();
  input_todo.value = '';
}
const input_todo = document.querySelector('.input-todo');
const add_todo_button = document.querySelector('.btn-add-todo');
input_todo.addEventListener('keydown', e => { // 엔터키를 누를 때 이벤트 발생
  if (e.key === 'Enter') {
    addTodo();
  }
});
add_todo_button.addEventListener('click', e => { // 버튼을 클릭할 때 이벤트 발생
  addTodo();
});
```

Main work ② 로컬 스토리지 사용



• 기능 소개

- 할 일 목록의 보존을 위해 로컬 스토리지 저장 및 사용

• 작업 내용

- 로컬 스토리지에서 할 일 목록 읽어오기
 - > 할 일 목록을 배열로 처리
 - > JSON의 stringify, parse 사용

```
function saveData(){ // object 형태로 배열에 추가하고
    storage.setItem('todos', JSON.stringify(todos));
}
function loadStorage(){ // 읽어올 todo가 있다면 배열로 반환, 아니면 빈 배열로 반환
    return storage.length !== 0 ? JSON.parse(storage.getItem('todos')):[]
}
const storage = window.localStorage;
let todos = loadStorage() // todo를 담은 배열
```

Main work ③ 할 일 목록 표시



• 기능 소개

- 할 일 목록 작성 / Enter Key 또는 추가하기 버튼 클릭 시
할 일 목록 추가

• 작업 내용

- 배열에서 가져온 할 일 목록을 html의 li 태그로 생성
- 할 일 목록과 버튼에 이벤트 추가

```
function createTodoList(){
  todo_list_container.innerHTML = ''; // elements를 지워 중복된 todo list를 방지
  todos.forEach((todo, index) => {
    const li = document.createElement('li');
    const span = document.createElement('span');
    const button = document.createElement('button');

    if (todo.checked) {
      span.classList.add('checked');
    }
    li.classList.add('flex', 'horizontal-between', 'vertical-center', 'li-spacing');
    span.classList.add('hover-pointer', 'test-class');
    button.className = 'del-btn';

    span.textContent=todo.text;
    button.textContent = 'x';

    span.addEventListener('click', () => {
      if (span.getAttribute('class').split(' ').length === 2){
        todo.checked = true;
        span.classList.add('checked');
        saveData();
      }else{
        todo.checked = false;
        span.classList.remove('checked');
        saveData();
      }
    });
    refreshProgressBar();
  });
  button.addEventListener('click', () => { // todo 삭제
    todos.splice(index, 1);
    saveData();
    createTodoList();
    refreshProgressBar();
  });

  li.appendChild(span);
  li.appendChild(button);
  todo_list_container.appendChild(li);
}
```

Main work ④ 완료율을 백분율로 표시 및 완료한 목록 실선처리



```
function refreshProgressBar(){
  const total = todos.length;
  let progress = 0;
  todos.forEach( todo => todo.checked ? progress++ : 0);
  let percentage = Math.round((progress / total) * 100);
  percentage = percentage? percentage+"%":"";
  todo_percentage.innerHTML = percentage;
  progress_bar.style.width = percentage;
}
```

```
span.addEventListener('click', () => {
  if (span.getAttribute('class')
    .split(' ').length === 2){
    todo.checked = true;
    span.classList.add('checked');
    saveData();
  }else{
    todo.checked = false;
    span.classList.remove('checked');
    saveData();
  }
  refreshProgressBar();
});
```

- 기능 소개
 - 완료율을 백분율로 표시 및 완료한 목록 실선 처리
- 작업 내용
 - 완료율을 백분율로 표시
 - > 반올림을 위해 Math의 round함수를 사용
 - > 부드러운 게이지 움직임을 위해 css의 translate를 사용
 - 완료한 목록 실선처리
 - > checked의 Boolean 값을 기준으로 css 처리

End of Document
