**Task List**

HTML

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0"/>

    <title>ToDo List</title>

    <link

     rel="stylesheet"

     href="https://fonts.googleapis.com/css?family=Poppins&display=swap"

     />

     <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-QWTKZyjpPEjISv5WaRU9OFeRpok6YctnYmDr5pNlyT2bRjXh0JMhjY6hW+ALEwIH" crossorigin="anonymous">

     <link rel="stylesheet" href="./style.css"/>

</head>

<body>

    <header>

        <h1>Daiwik's To Do List</h1>

    </header>

    <form id="abc">

        <input type="text" class="todo-input"/>

        <button class="todo-button" type="submit">

            <i class="fas fa-plus-square">+</i>

        </button>

        <div class="select">

            <select name="todos" class="filter-todo">

                <option value="all">ALL</option>

                <option value="completed">Completed</option>

                <option value="uncompleted">uncompleted</option>

            </select>

        </div>

    </form>

    <div class="todo-container">

        <ul class="todo-list"></ul>

    </div>

    <script src="./app.js"></script>

</body>

</html>

CSS

\* {

    margin: 0;

    padding: 0;

    box-sizing: border-box;

}

body {

    background-image: linear-gradient(120deg, #f6d365 0%, #fda085 100%);

    color: white;

    font-family: "poppins", sens-serif;

    min-height: 100vh;

}

header {

    font-size: 2rem;

}

header,

form {

    min-height: 20vh;

    display: flex;

    justify-content: center;

    align-items: center;

}

form input,

form button {

    padding: 0.5rem;

    font-size: 2rem;

    border: none;

    background: white;

}

form button {

    color: #ff6f47;

    background: #f7fffe;

    cursor: pointer;

    transition: all 0.3s ease;

}

form button:hover {

    background: #ff6f47;

    color: white;

}

.todo-container {

    display: flex;

    justify-content: center;

    align-items: center;

}

.todo-list {

    min-width: 30%;

    list-style: none;

}

.todo {

    margin: 0.5rem;

    background: white;

    font-size: 1.5rem;

    color: black;

    display: flex;

    justify-content: space-between;

    align-items: center;

    transition: all 1s ease;

}

.filter-todo {

    padding: 1rem;

}

.todo li {

    flex: 1;

}

.trash-btn,

.complete-btn {

    background: #ff6f47;

    color: white;

    border: none;

    padding: 1rem;

    cursor: pointer;

    font-size: 1rem;

}

.complete-btn {

 background: rgb(11, 212, 162);

}

.todo-item {

    padding: 0rem 0.5rem;

}

.fa-trash,

.fa-check {

    pointer-events: none;

}

.fall {

    transform: translateY(10rem) rotateZ(20deg);

    opacity: 0;

}

.completed {

    text-decoration: line-through;

    opacity: 0.5;

}

/\*CUSTOM SELECTOR\*/

select {

    -webkit-appearance: none;

    -moz-appearance: none;

    -ms-appearance: none;

    appearance: none;

    outline: 0;

    box-shadow: none;

    border: 0 !important;

    background-image: none;

}

/\*custom select\*/

.select {

    margin: 1rem;

    position:relative;

    overflow: hidden;

}

select{

    color: #ff6f47;

    font-family: "poppins", sans-serif;

    cursor: pointer;

    width: 12rem;

}

/\* Arrow\*/

.select::after{

    content: "\25BC";

    position: absolute;

    top: 0;

    right: 0;

    padding: 1rem;

    background: #ff6f47;

    cursor: pointer;

    pointer-events: none;

}

/\*Transition\*/

/\*.select:hover::after{

transform: scale(1.5);

}

JS

//getting variables

const todoInput = document.querySelector(".todo-input");

const todoButton = document.querySelector(".todo-button");

const todoList = document.querySelector(".todo-list");

const todoFilter = document.querySelector(".filter-todo");

document.addEventListener('DOMContentLoaded',function(e){

    getTodo();

})

todoButton.addEventListener("click", addTodo);

todoList.addEventListener("click", deletecheck);

todoFilter.addEventListener("click", filterTodo);

function addTodo(e) {

    //creating li element

    const todoDiv = document.createElement("div")

    todoDiv.className = "todo";

    const todos = document.createElement("li");

    todos.className = "todo-item";

    todos.appendChild(document.createTextNode(todoInput.value));

    todoDiv.appendChild(todos);

    todoInput.value = "";

    //create complete-button

    const completeBtn = document.createElement('button');

    completeBtn.className = "complete-btn";

    completeBtn.innerHTML = '<i class="fas fa-check"></i>';

    todoDiv.appendChild(completeBtn);

    //creating trash-button

    const trashBtn = document.createElement('button');

    trashBtn.className = "trash-btn";

    trashBtn.innerHTML = '<i class="fas fa-trash"></i>';

    todoDiv.appendChild(trashBtn);

    todoList.appendChild(todoDiv);

    todoInput.value = "";

    // //create complete-buttonateElement("button");

    // trashBtn.className = "trashbtn";

    // trashBtn.innerHTML = '<i class="fas fa-trash"></i>';

    // todoDiv.appendChild(trashBtn);

    // todoList.appendChild(todoDiv);

     e.preventDefault();

}

function deletecheck(e) {

    //Delete functionlities

    const items = e.target;

    if (items.className === "trash-btn") {

        const todo = items.parentElement;

        todo.classList.add("fall");

        removeTodo(todo);

        todo.addEventListener("transitionend", function () {

            todo.remove();

        });

}

//completed functionalits

if(items.className === "complete-btn") {

    const todo = items.parentElement;

    console.log(todo);

    todo.classList.toggle("completed");

    console.log(todo);

}

    e.preventDefault();

}

function filterTodo(e){

 const todos = todoList.childNodes;

 console.log(todos);

 todos.forEach(function(todo){

    switch(e.target.value) {

        case "all":

            todo.style.display = "flex";

            break;

        case "completed":

            if (todo.classList.contains("completed")) {

                todo.style.display = "flex";

            } else {

                todo.style.display = "none";

            }

            break;

        case "uncompleted":

            if (!todo.classList.contains("completed")) {

                todo.style.display = "flex";

            } else {

                todo.style.display = "none";

            }

            break;

          default:

            console.log(e.target.value);

    }

 })

    e.preventDefault();

}

//TO SET TODO IN LOCAL STORAGE

function setTodos(todo){

    let todos;

    if(localStorage.getItem('todos')){

        todos= [];

    }else{

        todos = JSON.parse(localStorage.getItem("todos"));

    }

    todos.push(todo);

    //setting to loacal storage

    localStorage.getItem('todos', JSON.stringify(todos));

}

function getTodo() {

    let todos;

    if (localStorage.getItem("todos") === null) {

        todos = [];

    }else {

        todos.JSON.parse(localStorage.getItem("todos"));

    }

    todos.forEach(function(todo){

        const completeBtn = document.createElement("button");

        completeBtn.className = "complete-btn";

        completeBtn.HTML = '<i class="fas fa-check"></i>';

        todoDiv.appendChild(completeBtn);

        //creating trash-button

        const trashBtn = document.createElement("div");

        todoDiv.className="todos";

        const todos = document.createElement('li');

        todos.className = "todo-item";

        setTodos(todoInput.value);

        todos.appendChild(document.createTextNode(todoInput.value));

        todoDiv.appendChild(todos);

        todoInput.value = "";

        //create complete-buttonateElement("button");

        trashBtn.className = "trashbtn";

        trashBtn.innerHTML = '<i class="fas fa-trash"></i>';

        todoDiv.appendChild(trashBtn);

        todoList.appendChild(todoDiv);

    });

}

function removeTodo(todo){

    let todos;

    if (localStorage.getItem("todos") === null){

        todos = [];

    }else {

        todos = JSON.parse(localStorage.getItem("todos"));

    }

   const todoIndex = todo.children[0].innerText;

   todos.splice(indexOf(todoIndex), 1);

   localStorage.setItem("todos", JSON,stringify(todos));

}