

```
<!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://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.4.0/css/all.min.css">
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <div class="container">
    <header>
      <div class="header">
        <h2>Work ToDo</h2>
      </div>
      <div class="new-todo">
        <form id="todoform">
          <input type="text" name="newtodo" id="newtodo" placeholder="work todo...">
          <button type="submit">
            <i class="fa-solid fa-square-plus"></i>
          </button>
        </form>
      </div>
    </header>
    <div id="todos-list">
      <!-- <div class="todo" id="0">
        <i class="fa-regular fa-circle"></i>
        <i class="fa-regular fa-circle-check"></i>
        <p class="checked">sid wants milk</p>
        <i class="fa-solid fa-pen-to-square"></i>
        <i class="fa-solid fa-trash"></i>
      </div> -->
    </div>
    <div class="notification"></div>
  </div>

  <script src="script.js"></script>
</body>
</html>
```

```
style
@import
url('https://fonts.googleapis.com/css2?family=Ubuntu&display=swap');

:root {
--container-height: 600px;
--contaier-width: 400px;
--header-height: 150px;
}

* {
padding: 0;
margin: 0;
box-sizing: border-box;
font-family: 'Ubuntu', monospace;
}
html {
font-size: 20px;
}

body {
width: 100vw;
height: 100vh;
display: flex;
align-items: center;
justify-content: center;
background-image: url('./img/ooo.jpg');
}

.container {
background-color: rgb(236, 149, 36);
width: var(--contaier-width);
height: var(--container-height);

position: relative;
overflow: hidden;
}
header {
width: 100%;
height: var(--header-height);
background-image: url('./img/ccc.jpg');
background-size: cover;
border-radius: 5px 5px 0 0;
}
.header {
width: 100%;
height: 100px;
display: flex;
align-items: center;
justify-content: center;
color: #fff;
}
.new-todo {
width: 100%;
height: 50px;
padding: 0.25rem 0;
}
.new-todo form {
display: flex;
align-items: center;
}
.new-todo form input {
flex: 1;
height: 40px;
background-color: transparent;
border: none;
border-right: 0px;
outline: transparent;
padding-left: 0.5rem;
font-size: 1rem;
color: #ffffff;
}
```

### style

```
.new-todo form button {  
    width: 80%;  
    height: 60px;  
    font-size: 2rem;  
    background-color: transparent;  
    color: #fff;  
    border: none;  
    border-left: 0px;  
    cursor: pointer;  
}  
.new-todo form button:hover {  
    text-shadow: 1px 1px 20px rgba(240, 237, 237, 0.8);  
}  
  
#todos-list {  
    height: calc(var(--container-height) - var(--header-height));  
    background-color: #c4c4c4;  
    padding: 0.5rem;  
    border: 1px solid rgb(219, 219, 219);  
    overflow: scroll;  
    border-radius: 0 0 5px 5px;  
}  
  
#todos-list .todo {  
    display: flex;  
    align-items: center;  
    padding: 0.75rem 0.5rem;  
    border-radius: 5px;  
}  
#todos-list .todo:hover {  
    background-color: rgba(0, 0, 0, 0.1);  
}  
#todos-list .todo * {  
    cursor: pointer;  
    margin-right: 0.5rem;  
}  
#todos-list .todo i {  
    font-size: 0.9rem;  
}  
#todos-list .todo p {  
    flex: 1;  
    word-break: break-all;  
}  
.checked {  
    text-decoration: line-through;  
    color: grey;  
}  
#todos-list .todo .bi-pencil-square {  
    color: rgb(31, 177, 48);  
}  
#todos-list .todo .bi-trash {  
    color: rgb(236, 82, 82);  
    margin: 0;  
}  
  
.notification {  
    position: absolute;  
  
    width: calc(3 * var(--contaier-width) / 4);  
    height: 60px;  
  
    display: flex;  
    justify-content: center;  
    align-items: center;  
  
    opacity: 0.9;  
  
    border-radius: 8px;  
    background-color: rgb(233, 81, 81);  
    top: 10px;  
    right: calc(-3 * var(--contaier-width) / 4);
```

style

```
color: #fff;  
transition: 300ms right ease-in-out;  
}  
.notif-enter {  
    right: 10px;  
}
```

```

//element select cheskundam dewdssssss
const form = document.getElementById('todoform');
const todoInput = document.getElementById('newtodo');
const todosListElements = document.getElementById('todos-list');
const notificationElement = document.querySelector('.notification');

//variables here
let todos = JSON.parse(localStorage.getItem('todos')) || [];
let EditTodoId = -1;

//first render
rendertodos();

//form submission boltheyy dewdsss
form.addEventListener('submit', function(event){
    event.preventDefault();

    saveTodo();
    rendertodos();
    localStorage.setItem('todos', JSON.stringify(todos))
});

//saving Todo functions
function saveTodo() {
    const todovalue = todoInput.value;

    //what if its empty
    const isEmpty = todovalue === "";

    //duplicate elements
    const isDuplicate = todos.some((todo) => todo.value.toUpperCase() === todovalue.toUpperCase());

    if(isEmpty){
        shownotification("your input is empty"); // alert place lo shownotification attribute use
    } else if(isDuplicate){
        shownotification('you entered the same values dewdss');
    } else{
        if(EditTodoId >= 0 ){
            todos = todos.map((todo , index) => ({
                ...todo,
                value : index === EditTodoId ? todovalue : todo.value,
            }));
            EditTodoId = -1;
        } else {
            todos.push({
                value : todovalue,
                checked : false,
                color:'#' + Math.floor(Math.random()*16777215).toString(16),
            });
        }
        todoInput.value = "";
    }
}

```

```

//render todo
function rendertodos(){
    if(todos.length === 0){
        todosListElements.innerHTML = '<center> ADD TASK..... </center>';
        return
    }
    //clearing rerender elements
    todosListElements.innerHTML = "";

    //rendering todo
    todos.forEach((todo,index) => {
        todosListElements.innerHTML += `
            <div class="todo" id=${index}>
                <i
                    class="fa-regular ${todo.checked? 'fa-circle-check' : 'fa-circle' }"
                    style = "color : ${todo.color}"
                    data-action="check"
                ></i>
                <p class="${todo.checked? 'checked' : ''}" data-action="check">${todo.value}</p>
                <i class="fa-solid fa-pen-to-square" data-action="edit"></i>
                <i class="fa-solid fa-trash" data-action="delete"></i>
            </div>
        `;
    });
}

//all in one todo listerner
todosListElements.addEventListener('click',(event)=>{
    const target = event.target;
    const parentElement = target.parentNode;

    if(parentElement.className !== 'todo') return;

    //todo id
    const todo = parentElement;
    const todoId = Number(todo.id);

    //targeting action
    const action = target.dataset.action;

    action === "check" && checkTodo(todoId);
    action === "edit" && editTodo(todoId);
    action === "delete" && deleteTodo(todoId);

});
//checking todo
function checkTodo(todoId){
    todos = todos.map((todo,index) =>
        ({
            ...todo,
            checked : index === todoId ? !todo.checked : todo.checked,
        }));
}

rendertodos();

```

```
localStorage.setItem('todos',JSON.stringify(todos))
}

//edit the todo
function editTodo(todoId){
    todoInput.value = todos[todoId].value;
    EditTodoId = todoId;
}

//deleting todo
function deleteTodo(todoId){
    todos = todos.filter((todo,index) => index !== todoId);
    EditTodoId = -1;

//then it should rerender
    renderTodos();
    localStorage.setItem('todos',JSON.stringify(todos))
}

//enable notifications
function showNotification(msg){
    //change the message
    notificationElement.innerHTML = msg;

    //presence of notification
    notificationElement.classList.add('notif-enter');

    //end of notification
    setTimeout(()=>{
        notificationElement.classList.remove('notif-enter')
    },2000)
}
```