Filename::main.jss

\* {

margin: 0;

box-sizing: border-box;

font-family: "Fira Sans", sans-serif;

}

:root {

--dark: #374151;

--darker: #1F2937;

--darkest: #111827;

--grey: #6B7280;

--pink: #e2187d;

--purple: #491ab8;

--light: #EEE;

}

body {

display: flex;

flex-direction: column;

min-height: 100vh;

color: #FFF;

background-color: #a2b6a0;

background-image: linear-gradient(to right, #8b8f8a, #1d98cc);

}

header {

padding: 2rem 1rem;

max-width: 800px;

width: 100%;

margin: 0 auto;

text-align: center;

}

header h1 {

font-size: 2.5rem;

font-weight: 550;

color: rgb(208, 24, 100);

margin-bottom: 0.6rem;

margin-top: 3rem;

}

header h1 span {

color: rgb(0, 0, 0);

}

#new-task-form {

display: flex;

flex-wrap: wrap;

justify-content: center;

gap: 1rem;

margin-top: 1rem;

}

input, button {

appearance: none;

border: none;

outline: none;

background: none;

font-family: inherit;

}

#new-task-input {

flex: 1 1 60%;

min-width: 250px;

background-color: #f8f8f8;

padding: 1rem;

border-radius: 1rem;

color: var(--light);

font-size: 1.25rem;

}

#new-task-input::placeholder {

color: var(--grey);

}

#new-task-submit {

color: var(--pink);

font-size: 1.5rem;

font-weight: 700;

background-image: linear-gradient(to right, var(--pink), var(--purple));

-webkit-background-clip: text;

-webkit-text-fill-color: transparent;

cursor: pointer;

transition: opacity 0.4s ease;

}

#new-task-submit:hover {

opacity: 0.8;

}

#new-task-submit:active {

opacity: 0.6;

}

main {

flex: 1;

max-width: 800px;

width: 100%;

margin: 0 auto;

}

.task-list {

padding: 1rem;

}

.task-list h2 {

font-size: 1.5rem;

font-weight: 400;

color: var(--darker);

margin-bottom: 1rem;

}

#tasks .task {

display: flex;

justify-content: space-between;

align-items: center;

background-color: var(--darkest);

padding: 1rem;

border-radius: 1rem;

margin-bottom: 1rem;

flex-wrap: wrap;

}

.task .content {

flex: 1 1 60%;

}

.task .content .text {

color: var(--light);

font-size: 1.125rem;

width: 100%;

display: block;

transition: color 0.4s;

background: transparent;

border: none;

}

.task .content .text:focus {

outline: none;

}

.task .content .text:not(:read-only) {

color: var(--pink);

}

.task .actions {

display: flex;

flex-wrap: wrap;

gap: 0.5rem;

}

.task .actions button {

cursor: pointer;

font-size: 1.125rem;

font-weight: 700;

text-transform: uppercase;

transition: 0.4s;

background: transparent;

}

.task .actions button:hover {

opacity: 0.8;

}

.task .actions button:active {

opacity: 0.6;

}

.task .actions .edit {

background-image: linear-gradient(to right, var(--pink), var(--purple));

-webkit-background-clip: text;

-webkit-text-fill-color: transparent;

}

.task .actions .delete {

color: crimson;

}

@media (max-width: 998px) {

html {

font-size: 85%;

}

}

cursor: pointer;

font-size: 1.125rem;

font-weight: 700;

text-transform: uppercase;

transition: 0.4s;

background: transparent;

}

.task .actions button:hover {

opacity: 0.8;

}

.task .actions button:active {

opacity: 0.6;

}

.task .actions .edit {

background-image: linear-gradient(to right, var(--pink), var(--purple));

-webkit-background-clip: text;

-webkit-text-fill-color: transparent;

}

.task .actions .delete {

color: crimson;

}

@media (max-width: 998px) {

html {

font-size: 85%;

}

}

@media (max-width: 768px) {

html {

filename::main.js

window.addEventListener('load', () => {

const form = document.querySelector("#new-task-form");

const input = document.querySelector("#new-task-input");

const list\_el = document.querySelector("#tasks");

form.addEventListener('submit', (e) => {

e.preventDefault();

const task = input.value;

const task\_el = document.createElement('div');

task\_el.classList.add('task');

const task\_content\_el = document.createElement('div');

task\_content\_el.classList.add('content');

task\_el.appendChild(task\_content\_el);

const task\_input\_el = document.createElement('input');

task\_input\_el.classList.add('text');

task\_input\_el.type = 'text';

task\_input\_el.value = task;

task\_input\_el.setAttribute('readonly', 'readonly');

task\_content\_el.appendChild(task\_input\_el);

const task\_actions\_el = document.createElement('div');

task\_actions\_el.classList.add('actions');

const task\_edit\_el = document.createElement('button');

task\_edit\_el.classList.add('edit');

task\_edit\_el.innerText = 'Edit';

const task\_delete\_el = document.createElement('button');

task\_delete\_el.classList.add('delete');

task\_delete\_el.innerText = 'Delete';

task\_actions\_el.appendChild(task\_edit\_el);

task\_actions\_el.appendChild(task\_delete\_el);

task\_el.appendChild(task\_actions\_el);

list\_el.appendChild(task\_el);

input.value = '';

task\_edit\_el.addEventListener('click', (e) => {

if (task\_edit\_el.innerText.toLowerCase() == "edit") {

task\_edit\_el.innerText = "Save";

task\_input\_el.removeAttribute("readonly");

task\_input\_el.focus();

} else {

task\_edit\_el.innerText = "Edit";

task\_input\_el.setAttribute("readonly", "readonly");

}

});

task\_delete\_el.addEventListener('click', (e) => {

list\_el.removeChild(task\_el);

});

});

});

Filename::todolist.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

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

<title>Task List - CHATHURYA'S</title>

<link rel="icon" href="p.jpg" type="image/png">

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

</head>

<body>

<header>

<h1><span>Aruna's</span> task list</h1>

<form id="new-task-form">

<input

type="text"

name="new-task-input"

id="new-task-input"

placeholder="What do you have planned?" />

<input

type="submit"

id="new-task-submit"

value="Add task" />

</form>

</header>

<main>

<section class="task-list">

<h2>Tasks</h2>

<div id="tasks">

<!-- <div class="task">

<div class="content">

<input

type="text"

class="text"

value="A new task"

readonly>

</div>

<div class="actions">

<button class="edit">Edit</button>

<button class="delete">Delete</button>

</div>

</div> -->

</div>

</section>

</main>

<script src="main.js"></script>

</body>

</html>