Blog.js

// TODO: Create a variable that selects the main element, and a variable that selects the back button element

const main = document.getElementById("main");

const back = document.getElementById("back");

// TODO: Create a function that builds an element and appends it to the DOM

function appendBlogPost(post){

let postDiv = document.createElement('div');

postDiv.className = 'card';

let titleElement= document.createElement('h2');

titleElement.textContent = post.title;

let contentElement= document.createElement('p');

contentElement.textContent = post.content;

let authorElement= document.createElement('p')

postDiv.appendChild(titleElement);

postDiv.appendChild(contentElement);

postDiv.appendChild(authorElement);

let blogMain = document.getElementById("main");

blogMain.appendChild(postDiv)

};

// TODO: Create a function that handles the case where there are no blog posts to display

function noBlogPosts(){

let blogMain = document.getElementById("main");

let messageElement = document.createElement("p");

messageElement.textContent = "No blog posts yet...";

blogMain.appendChild(messageElement)

}

// TODO: Create a function called `renderBlogList` that renders the list of blog posts if they exist. If not, call the no posts function.

function renderBlogList() {

let storedBlogPosts = localStorage.getItem('blogPosts');

storedBlogPosts = JSON.parse(storedBlogPosts);

if (storedBlogPosts && Array.isArray(storedBlogPosts) && storedBlogPosts.length >0) {

storedBlogPosts.forEach(post => {

amendDOMWithBlogPost(post);

});

} else {

noBlogPosts();

}

}

// TODO: Call the `renderBlogList` function

renderBlogList();

// TODO: Redirect to the home page using the `redirectPage` function found in logic.js when the back button is clicked

back.addEventListener("click", function (event) {

event.preventDefault();

redirectPage("https://web.us.edusercontent.com/7f2jn4brhnop8a4m8kl0johbhc/index.html");

});

Form.js

document.addEventListener("DOMContentLoaded", function () {

const form = document.getElementById("form");

const username = document.getElementById("username");

const title = document.getElementById("title");

const content = document.getElementById("content");

const errorElement = document.getElementById("error");

const submit = document.getElementById("submit")

form.addEventListener("submit", function (event) {

if (

username.value.trim() === "" ||

title.value.trim() === "" ||

content.value.trim() === ""

) {

event.preventDefault();

errorElement.textContent = "Please complete the form.";

} else {

event.preventDefault();

storeLocalStorage();

const redirectPage = function (url) {

redirectURL = url;

location.assign(url);

};

redirectPage("https://web.us.edusercontent.com/7f2jn4brhnop8a4m8kl0johbhc/blog.html");

};

});

});

Blog.css

/\*

Blog Styles

\*/

/\*

Tags

\*/

a {

text-decoration: none;

}

article {

flex: 1 1 50%;

border: 2px solid;

border-radius: 4px;

padding: 1rem;

position: relative;

}

article blockquote {

margin: 1rem;

margin-bottom: 2rem;

font-style: italic;

}

article h2 {

text-transform: capitalize;

border-bottom: thin solid;

}

article p {

position: absolute;

left: 1rem;

bottom: 0.5rem;

opacity: .75;

}

footer {

position: fixed;

inset: 0;

top: calc(100% - 12rem);

}

li {

margin: 1rem;

}

main {

flex-direction: column;

gap: 1em;

padding: 1rem;

height: calc(100vh - 18rem);

overflow-y: scroll;

justify-content: flex-start;

}

ul {

list-style: none;

}

/\*

Classes

\*/

.card {

flex: 1 1 50%;

border: 2px solid;

border-radius: 4px;

}

.dark a {

color: var(--light);

}

.light footer {

border-top: double 1rem;

}

.light a {

color: var(--dark);

}

/\*

IDs

\*/

#back {

padding: 0.5rem 1rem;

border-radius: 4px;

margin-right: 1rem;

border: groove black;

background-color: var(--dark-accent);

color: var(--light);

}

#back:active {

box-shadow: 1px 1px 1px 1px rgba(0, 0, 0, .5);

background-color: var(--dark-accent);

}

#back:hover {

border-style: inset;

background-color: var(--dark-accent-action);

}

#love {

text-align: center;

}

Logic.js

// TODO: Create logic to toggle the light/dark mode styles for the page and circle. The mode should be saved to local storage.

// DO: Create a function called `readLocalStorage` that reads from local storage and returns the data. If no data exists, return an empty array.

function readLocalStorage(){

let storedBlogPostsArray = localStorage.getItem('blogPostArray');

if (storedBlogPostsArray !== null){

return JSON.parse(storedBlogPostsArray);

} else {

return [];

}

}

// TODO: Create a function called `storeLocalStorage` that takes a given object and saves the new data to the existing blog data in local storage.

const blogPostKey = 'blogPosts';

let storedBlogPosts = localStorage.getItem(blogPostKey)

let blogPostArray = storedBlogPosts ? JSON.parse(storedBlogPosts) : [];

function storeLocalStorage() {

const blogPostKey = 'blogPosts';

let storedBlogPosts = localStorage.getItem(blogPostKey)

let blogPostArray = storedBlogPosts ? JSON.parse(storedBlogPosts) : [];

let newBlogPost = {

username: username.value,

title: title.value,

content: content.value.trim(),

};

blogPostArray.push(newBlogPost);

localStorage.setItem(blogPostKey, JSON.stringify(blogPostArray));

}

// ! Use the following function whenever you need to redirect to a different page

let redirectURL = '';

const redirectPage = function (url) {

redirectURL = url;

location.assign(url);

};

blog.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

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

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

<title>My First Blog</title>

<link rel="stylesheet" href="./assets/css/styles.css" />

<link rel="stylesheet" href="./assets/css/form.css" />

</head>

<body>

<body>

<header>

<h1>Blog Posts</h1>

</header>

<main id = "main" name = "main">

<section>

<ul>

<li></li>

</ul>

</section>

<button id="back" name = "back">Back</button>

</main>

<footer></footer>

<script defer src="./assets/js/blog.js" type="text/javascript"></script>

<script defer src="./assets/js/logic.js" type="text/javascript"></script>

</body>

</body>

</html>

Index.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

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

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

<title>My First Blog</title>

<link rel="stylesheet" href="./assets/css/styles.css" />

<link rel="stylesheet" href="./assets/css/form.css" />

</head>

<body>

<h1>My First Blog</h1>

<label class="switch">

<input type="checkbox">

<span></span>

</label>

<section>

<form id="form" name = "form">

<label>Username</label>

<input type= "text" id= "username" name="username"></input>

<label>Title</label>

<input type= "text" id= "title" name="title"></input>

<label>Content</label>

<textarea type= "text" id= "content" name="content"></textarea>

<p id="error" name="error"></p>

<button type="submit">Submit</button>

</form>

</section>

<script defer src="./assets/js/logic.js" type="text/javascript"></script>

<script defer src="./assets/js/form.js" type="text/javascript"></script>

</body>

</html>

Form.css

/\*

Form Styles

\*/

/\*

Tags

\*/

aside {

background: linear-gradient(

var(--light-accent),

var(--circle-color),

var(--dark-accent)

);

clip-path: circle(20rem at center);

display: flex;

justify-content: center;

align-items: center;

}

aside,

section {

flex: 1 1 50%;

}

aside > h2 {

font-size: 4rem;

color: white;

text-shadow: 1px 1px 1px var(--dark-accent);

}

form {

display: flex;

flex-flow: column wrap;

position: relative;

}

form > \*:not(label) {

margin: 1rem 0;

padding: 0.5rem;

}

h2 {

text-align: center;

font-size: 2rem;

margin-bottom: 2rem;

}

input[type=submit] {

padding: 1rem;

cursor: pointer;

font-weight: bold;

font-size: 1rem;

background-color: var(--dark);

color: var(--light);

}

input[type=submit]:active {

box-shadow: 1px 1px 1px 1px rgba(0, 0, 0, .5);

background-color: var(--dark-accent);

}

input[type=submit]:hover {

border-style: inset;

background-color: var(--dark-accent-action);

}

input,

textarea {

border-radius: 4px;

border-width: 1px;

font-size: 1rem;

background-color: var(--light-accent);

}

section {

padding: 2rem;

display: flex;

align-items: stretch;

flex-flow: column wrap;

justify-content: center;

border-left: double 1rem;

}

/\*

Classes

\*/

.dark section {

border-left: solid .5rem var(--disabled);

}

/\*

IDs

\*/

#error {

color: var(--error);

opacity: .75;

position: absolute;

bottom: -2rem;

}

Syles.css

/\*

Resets

\*/

\*,

\*:before,

\*:after {

margin: 0;

padding: 0;

box-sizing: border-box;

}

html {

height: 100%;

}

body {

min-height: 100%;

line-height: 1;

font-family: sans-serif;

}

​h1,

h2,

h3,

h4,

h5,

h6 {

font-size: 100%;

}

​input,

select,

option,

optgroup,

textarea,

button,

pre,

code {

font-size: 100%;

font-family: inherit;

}

​ol,

ul {

list-style: none;

}

/\*

General Styles

\*/

/\*

Universal

\*/

:root {

--light: #fff;

--light-accent: #f9f9fb;

--dark: #000;

--dark-accent: #222;

--dark-accent-action: #111;

--action: #563d7c;

--action-accent: #8570a5;

--disabled: #aaa;

--error: #ff0022;

--circle-color: #ffb100;

}

\* {

transition: background-color ease-in .5s;

}

/\*

Tags

\*/

header {

display: flex;

flex-flow: row wrap;

justify-content: space-between;

}

header,

footer {

padding: 2rem;

}

main {

height: calc(100vh - 6rem);

display: flex;

justify-content: center;

align-items: stretch;

border-top: double 1rem;

}

/\*

Classes

\*/

.dark header,

.dark footer {

background-color: var(--dark);

color: var(--light);

}

.dark main {

background-color: var(--dark-accent);

color: var(--light);

border-top: solid .5rem var(--disabled);

}

.light header,

.light footer {

background-color: var(--light);

color: var(--dark);

}

.light main {

background-color: var(--light-accent);

color: var(--dark);

}

/\*

IDs

\*/

#toggle {

font-size: 1.52em;

margin: 0;

padding: 0;

background: transparent;

border: none;

cursor: pointer;

text-shadow: 0px 0px .5rem var(--light-accent);

}