BLOG WEB APPLICATON

Introduction	This website is a blog app in which a user can view some recently added blog posts on the main index page and user can create new posts and user can also perform update and delete operations on the post. The site is fully responsive for all screen types.
Index.html	On the main index.html file there are few sections, all sections contain jquery, bootstrap css/js and relevant css of each section:
Header	The header section contains a top nav and main a navbar.
Top nav	The top nav contains social links, search bar icon and a button for adding new posts. When search bar icon is clicked then an input is shown with focus to search different posts, currently search functionality is not available, only an input box. The button “Add new post” will redirect the user to a new page for adding new post.
Main navbar	The main navbar contains logo and menu icon which is visible on smaller screens and below that is the menu for navigation on site, the menu will be hidden and can be accessed using menu icon when the screen is smaller below 1000px. The menu is shown vertically and links are animated.
Header image	A beautiful responsive header image with text is added below the navbar to enhance the UI.
Recent posts	In recent posts section the 6 latest posts are shown with image, title and some content, below this a button is added which will redirect to page with all posts. recentPost.js script is used for adding logic and elements to UI of recent posts section, in this file first the array post is populated with dummy data and then stored inside the localStorage so it can be used everywhere in the app. This script uses a function recentPost() to get 6 latest posts from the localStorage and then renders the elements with content on screen.
Contact us	This section gives a form with name, email and message inputs so the user can contact the author, this form is also responsive for all screen size.
About	A simple about section to tell the user about the author, which is responsive.
Footer	In the end there is footer to show some information.

All posts page	In this page the user can view all post in beautiful and responsive UI. This page contains header same as mentioned before. The script allPosts.js is used to add logic to this page in which posts are retrieved from the localStorage and then map function is used in which few html elements are created and then post data is added inside these elements and then the posts are appended to the div inside the html page. Each post is clickable which will take the user to its respective details page using the id which is added in the url params.
Single Post Page	The post.html shows the details of a single post in responsive UI with image, title and content. It also contains two action buttons “Edit” and “Delete”. Post.js script is used to add logic which gets the post id from url params using URLSearchParams(window.location.search) method of javascript, this postId is used to find the post from the posts retrieved from localStorage and then the post elements are rendered on the screen.
Add Post	When the user clicks on the add post button on the top nav then it is redirected to the add_post.html page where responsive UI with a form is given to add image, title and content of the new post, when the user saves or adds the post then an alert is shown that post has been added to the localStrage and page is redirected to the newly added post. The script used to add logic is addPost.js which takes the value of form input using id tag and then adds the values inside an object which is the added to array using push() method and then the new array is added to localStorage(). The id is added to each post so CRUD operations can be performed, the id is generated using Math.random() method and then added to the new post object. When user adds the new post then user is given alert message that new post has been saved.
Edit Post	The edit_post.html page has a responsive UI that shows a form with prefilled data of the current post so it can be edited and saved, when user saves the data an alert message is given to let user know that post has been edited.

Delete Post	editPost.js script is used to add logic to this html page using the postId taken from the url params the relevant post is found from posts array using jquery grep() method and then input fields are filled with the post data so the user can update the post information, when user saves the post then user is given alert message that post have been saved and redirected to the edited post page. When the user lands on the post details page then at the bottom there are two buttons, one is delete button which user can click perform delete operation, the user is given alert message that the post has been deleted and redirected to the main page.