**1. blogLoader.js: Dynamic loading and displaying of blog content**

* Loads blog data from blogData.json.
* Dynamically generates blog titles, dates, categories, and content.
* Updates blog content easily by editing the JSON file.

**2. dynamicGallery.js: Creates a dynamic image gallery**

* Loads image data from gallery.json.
* Dynamically generates a responsive image gallery with fade-in animation.
* Uses IntersectionObserver for lazy loading, optimizing page performance.

**3. fetchWeather.js: Fetches and displays real-time weather data**

* Uses OpenWeatherMap API to retrieve weather data based on the user-entered city name.
* Displays temperature, weather conditions, humidity, and wind speed.
* Includes error handling to ensure smooth user experience even if API fails.

**4. formValidation.js: Validates user inputs in forms**

* Ensures users provide valid data before submitting forms.
* Validates fields like email format and required fields with instant feedback.
* Enhances user experience by preventing invalid submissions.

**5. imageCarousel.js: Implements image carousel functionality**

* Loads image carousel data from imageCarousel.json.
* Dynamically creates carousel images with fade-in and fade-out animations.
* Automatically loops through images and uses IntersectionObserver for optimized loading.

**6. main.js: Main script for initializing core functionalities**

* Integrates modules like scroll.js, blogLoader.js, and imageCarousel.js.
* Loads all necessary JSON data (e.g., blogs, gallery, carousel) using Promise.all.
* Initializes features such as scroll button, weather data, and form validation.

**7. scroll.js: Controls back-to-top button**

* Dynamically displays a back-to-top button when scrolling down.
* Implements smooth scrolling to the top when clicked.
* Modular design makes it reusable across multiple pages.

**8. gallery.json: Stores data for the image gallery**

* Contains information about gallery images, such as paths and descriptions.
* Used by dynamicGallery.js to dynamically generate the gallery.

**9. blogData.json: Stores data for blog content**

* Holds information about blog posts, including titles, dates, categories, and content.
* Used by blogLoader.js to populate blog sections dynamically.

**10. imageCarousel.json: Stores data for the image carousel**

* Contains information about carousel images, such as paths and descriptions.
* Used by imageCarousel.js to generate carousel content dynamically.