**Beginner Level:**

1. **To-Do List App**
   * Create a simple app where users can add, edit, and delete tasks.
   * Use localStorage to save the to-do list data.
2. **Calculator**
   * Build a basic calculator that performs simple arithmetic operations (addition, subtraction, multiplication, and division).
3. **Random Quote Generator**
   * Display a random quote each time a button is clicked. You can store the quotes in an array or fetch them from an API.
4. **Tip Calculator**
   * Create a tool to calculate tips based on the bill amount and selected tip percentage.
5. **Color Guessing Game**
   * A game where users must guess a randomly generated RGB color based on provided color values.
6. **Simple Form Validation**
   * Create a sign-up form that validates user input (e.g., checking for valid email, password length, etc.).
7. **Countdown Timer**
   * Create a countdown timer that counts down to a specific event or date, like New Year’s Eve.
8. **Weather App (API Integration)**
   * Use the OpenWeatherMap API to fetch and display current weather information for a city entered by the user.
9. **Palindrome Checker**
   * Build a small app that checks whether a given word or phrase is a palindrome.
10. **Unit Converter**
    * Create a simple tool to convert between units like kilometers to miles, Celsius to Fahrenheit, etc.

**Intermediate Level:**

1. **Quiz App**
   * Build a multiple-choice quiz app where users can answer questions, and the app keeps score. Store the questions in an external JSON file.
2. **Memory Game**
   * Create a card matching game where users flip cards to find matching pairs. Add a timer and scoring system.
3. **Digital Clock**
   * Build a live digital clock that updates every second and displays the current time.
4. **Movie Search App (API Integration)**
   * Use an API like OMDB to fetch and display movie details when a user searches for a movie by name.
5. **Currency Converter**
   * Create a currency converter app that uses an API like ExchangeRate-API to fetch real-time currency conversion rates.
6. **Drag and Drop Interface**
   * Implement a simple drag-and-drop interface where users can drag items into different boxes or containers.
7. **Image Slider**
   * Build an image slider or carousel that automatically changes images after a set time and allows users to manually navigate between them.
8. **Real-time Chat Application (Using WebSocket)**
   * Implement a basic real-time chat app using WebSockets, where users can join a room and send messages to others in real-time.
9. **Stopwatch**
   * Create a stopwatch app with start, stop, and reset functionality.
10. **Markdown Previewer**
    * Build a tool where users can write in markdown on one side of the screen, and see a preview of the formatted text on the other.
11. **Recipe App**
    * Create an app where users can search for recipes based on ingredients. Use an external API like Spoonacular.
12. **Shopping Cart (Frontend Only)**
    * Develop a basic shopping cart system where users can add/remove items, adjust quantities, and view the total price.
13. **Interactive Quiz Game**
    * A trivia quiz where questions are fetched from an external API (like Open Trivia DB), and users can answer them in real-time.
14. **Expense Tracker**
    * Build a personal finance app where users can add, view, and delete expense items. Include features like income tracking and expense categorization.
15. **Music Player**
    * Create a music player with play, pause, stop, and next/previous song functionality. You can use pre-uploaded audio files for simplicity.
16. **Real-time Search Filter**
    * Build a search interface where users can type keywords and instantly filter through a list of items (e.g., products, books, or movies).
17. **Form Wizard**
    * Develop a multi-step form where users fill out different sections and can navigate back and forth between steps.
18. **Personal Portfolio Website**
    * Create a personal portfolio website with different sections (About Me, Projects, Contact), including responsive design and JavaScript interactivity.
19. **GitHub User Search**
    * Use the GitHub API to build a user search app where you can input a GitHub username and display their profile information and repositories.
20. **Light/Dark Mode Toggle**
    * Create a website that allows users to toggle between light and dark themes. Store the theme preference using localStorage.

**Key Concepts to Focus On:**

* **DOM Manipulation:** Projects like the to-do list, calculator, and quiz app help with understanding how to manipulate the DOM using JavaScript.
* **API Integration:** Projects like the weather app, movie search app, and GitHub user search help with learning how to work with APIs and handle asynchronous data.
* **Event Handling:** Projects like the calculator, color guessing game, and stopwatch help with managing user interaction via events (clicks, key presses, etc.).
* **LocalStorage:** To save data locally on the browser (e.g., storing theme preferences or to-do lists).