## Front End Engineering-II

Project Report

Semester-IV (Batch-2022)

**FAQ page**

A red and white sign

Description automatically generated with low confidence

**Supervised By: Submitted By:**

Raveesh Samkaria Vanshika goyal

2210990946(G-31)

**Department of Computer Science and Engineer** Chitkara University Institute of Engineering & Technology, Chitkara University, Punjab

**Abstract**

**In this project, we embark on a journey to create a visually stunning and highly functional FAQ (Frequently Asked Questions) page using Tailwind CSS, a utility-first CSS framework renowned for its simplicity and efficiency. Tailwind CSS offers a novel approach to styling web interfaces by providing a comprehensive set of utility classes that enable developers to design intricate layouts and components without the need for traditional CSS. With its intuitive syntax and robust design system, Tailwind CSS empowers both novice and seasoned developers to streamline their workflow and build captivating user experiences.**

**Throughout this project, we delve into the essential principles of designing and developing a responsive FAQ page that seamlessly adapts to various screen sizes and devices. We begin by structuring the layout of our FAQ page, leveraging Tailwind CSS's grid system and responsive design utilities to create a visually appealing and easily navigable interface. Next, we explore techniques for styling individual FAQ sections and questions, employing Tailwind CSS's extensive collection of utility classes for typography, spacing, and colour to achieve a cohesive and aesthetically pleasing design.**

**One of the key features of our FAQ page is its interactive accordion behaviour, which allows users to expand and collapse individual questions to view their corresponding answers. By harnessing Tailwind CSS's utility classes for toggling visibility and applying transitions, we implement this functionality in a simple and elegant manner, enhancing the user experience and promoting engagement.**

**Throughout the project, we emphasize best practices for structuring HTML markup and organizing CSS classes to ensure code maintainability and scalability. Whether you're a beginner eager to learn the basics of Tailwind CSS or a seasoned developer seeking to optimize your workflow, this project provides valuable insights and practical techniques for creating polished and responsive web interfaces that captivate and delight users.**

**Table of Contents**

1. Introduction
2. Objectives
3. Significance

4. Problem Definition and Requirements

5.Proposed Design / Methodology

6.Code /File format

7.Results

## ****Introduction****

## **In today's digital landscape, user experience reigns supreme. As web developers, our goal is not just to convey information but to do so in a manner that is intuitive, engaging, and visually appealing. One often-overlooked yet crucial aspect of this user experience is the Frequently Asked Questions (FAQ) page. Frequently serving as a central hub for addressing common queries and concerns, a well-designed FAQ page can significantly enhance user satisfaction and streamline interaction with a website or application.**

## **Enter Tailwind CSS – a revolutionary utility-first CSS framework that has gained widespread acclaim for its simplicity, flexibility, and efficiency. Tailwind CSS offers developers a robust toolkit of utility classes that enable them to rapidly prototype and style interfaces without the need for writing custom CSS. With its intuitive syntax and comprehensive design system, Tailwind CSS empowers developers to focus on crafting exceptional user experiences rather than wrestling with CSS intricacies.**

## **In this project, we embark on a journey to leverage the power of Tailwind CSS in creating a modern and responsive FAQ page. Our objective is clear: to design a visually stunning and highly functional FAQ page that not only addresses users' common queries but also enhances the overall browsing experience. Throughout this project, we will explore the fundamental principles of building a responsive web interface, from structuring layout and styling elements to implementing interactive behaviours.**

## **By following along with this project, you will gain valuable insights into the following key aspects:**

## **1. Structuring Layout: We'll utilize Tailwind CSS's grid system and responsive design utilities to create a flexible and visually appealing layout that adapts seamlessly to various screen sizes and devices.**

## **2. Styling Elements: Leveraging Tailwind CSS's extensive collection of utility classes, we'll style individual FAQ sections and questions to achieve a cohesive and aesthetically pleasing design.**

## **3. Implementing Interactivity: We'll employ Tailwind CSS's utility classes to create an interactive accordion behaviour, allowing users to expand and collapse individual questions to view their corresponding answers.**

## **Whether you're a beginner eager to learn the basics of Tailwind CSS or a seasoned developer seeking to optimize your workflow, this project offers valuable insights and practical techniques for creating polished and responsive web interfaces that captivate and delight users. Let's dive in and embark on this exciting journey together.**

## ****Background****

**In today's fast-paced digital landscape, websites and applications serve as primary touchpoints for businesses and individuals alike. As such, providing users with an intuitive and seamless browsing experience is paramount to success. Frequently Asked Questions (FAQ) pages play a crucial role in this experience, serving as a central repository for addressing common queries and concerns users may have.**

**Traditionally, FAQ pages have been viewed as mundane and static elements of a website, often overlooked in favor of more visually striking features. However, with the growing emphasis on user-centric design and accessibility, the importance of designing an engaging and functional FAQ page cannot be overstated.**

**Enter Tailwind CSS – a revolutionary CSS framework that has transformed the way developers approach web design. Unlike traditional CSS frameworks, which rely on pre-defined components and complex styling rules, Tailwind CSS embraces a utility-first approach, providing developers with a vast array of utility classes that can be used to style elements directly in the markup.**

**The combination of Tailwind CSS's simplicity and flexibility makes it an ideal choice for designing responsive and visually appealing interfaces, including FAQ pages. By leveraging Tailwind CSS's utility classes, developers can rapidly prototype and iterate on designs, focusing on creating an exceptional user experience without being bogged down by the intricacies of CSS.**

**Against this backdrop, this project seeks to explore the potential of Tailwind CSS in designing a modern and responsive FAQ page. By harnessing the power of Tailwind CSS's utility classes, we aim to create a visually stunning and highly functional FAQ page that not only addresses users' common queries but also enhances the overall browsing experience.**

**Through this project, we endeavor to demonstrate the versatility and efficiency of Tailwind CSS in web design, providing developers with practical insights and techniques for creating polished and user-friendly interfaces. Whether you're a seasoned developer looking to streamline your workflow or a beginner eager to learn the basics of Tailwind CSS, this project offers valuable lessons and inspiration for designing exceptional FAQ pages that captivate and delight users.**

**Objectives**

**The objective of this project is to design and develop a modern, responsive, and visually appealing Frequently Asked Questions (FAQ) page using Tailwind CSS. Tailwind CSS is a utility-first CSS framework known for its simplicity, flexibility, and efficiency, making it an ideal tool for rapidly prototyping and styling web interfaces without the need for writing custom CSS.**

**Through this project, we aim to achieve the following objectives:**

**1. Create a Responsive Layout: Utilize Tailwind CSS's grid system and responsive design utilities to craft a layout that seamlessly adapts to various screen sizes and devices, ensuring an optimal viewing experience for users across desktop and mobile platforms.**

**2. Style FAQ Sections and Questions: Employ Tailwind CSS's extensive collection of utility classes for typography, spacing, and colour to style individual FAQ sections and questions, creating a cohesive and visually appealing design that aligns with the overall aesthetic of the website or application.**

**3.Implement Interactive Accordion Behaviour : Utilize Tailwind CSS's utility classes to implement an interactive accordion behaviour for collapsible answers, allowing users to expand and collapse individual questions to view their corresponding answers with ease and efficiency.**

**4. Enhance User Experience: By combining the power of Tailwind CSS with thoughtful design choices and intuitive interactive elements, aim to enhance the overall user experience of the FAQ page, making it easy for users to find relevant information and navigate the content seamlessly.**

**5. Promote Best Practices : Throughout the project, emphasize best practices for structuring HTML markup, organizing CSS classes, and optimizing code maintainability and scalability, empowering developers to apply these principles to their own projects effectively.**

**By achieving these objectives, this project aims to demonstrate the capabilities of Tailwind CSS in designing polished and user-friendly FAQ pages that not only address users' common queries but also contribute to a positive and engaging browsing experience. Whether you're a beginner learning the basics of Tailwind CSS or a seasoned developer seeking to optimize your workflow, this project provides valuable insights and practical techniques for creating exceptional web interfaces.**

**Significance**

**The significance of this project lies in its contribution to the enhancement of user experience and the optimization of web development workflows through the utilization of Tailwind CSS in designing a modern Frequently Asked Questions (FAQ) page. Several key points highlight the significance of this endeavour.**

**1. Improved User Engagement: A well-designed FAQ page serves as a valuable resource for users seeking answers to common queries and concerns. By leveraging Tailwind CSS to create a visually appealing and intuitive interface, this project aims to increase user engagement and satisfaction, fostering positive interactions with the website or application.**

**2. Responsive Design: In today's multi-device landscape, responsive design is essential for ensuring that content is accessible and optimized across various screen sizes and devices. By implementing a responsive layout using Tailwind CSS, this project demonstrates the importance of adapting to evolving user preferences and technological advancements.**

**3. Streamlined Development Process: Tailwind CSS's utility-first approach streamlines the development process by providing a comprehensive set of utility classes that can be used to style elements directly in the markup. This project showcases how Tailwind CSS enables developers to prototype, iterate, and deploy interfaces more efficiently, saving time and effort in the development cycle.**

**4. Accessibility and Inclusivity: Designing with accessibility in mind is crucial for ensuring that all users, regardless of ability, can access and interact with digital content effectively. Through thoughtful design choices and adherence to accessibility best practices, this project underscores the importance of creating inclusive experiences that cater to diverse user needs and preferences.**

**5. Educational Value : As a learning resource, this project offers valuable insights and practical techniques for developers at all skill levels. By providing step-by-step guidance on designing and developing a FAQ page with Tailwind CSS, this project empowers developers to expand their skill set, explore new technologies, and enhance their proficiency in web development.**

**In summary, this project's significance lies in its ability to contribute to the improvement of user experience, the advancement of web development practices, and the dissemination of knowledge within the developer community. By showcasing the potential of Tailwind CSS in designing modern and responsive interfaces, this project seeks to inspire and empower developers to create exceptional digital experiences that resonate with users worldwide.**

## ****Problem definition and requirements****

**Problem Definition**

**The problem at hand is the need to design and develop a modern and user-friendly Frequently Asked Questions (FAQ) page for a website or application. While FAQ pages are essential for providing users with answers to common queries and concerns, they are often overlooked in terms of design and functionality. The challenge lies in creating a FAQ page that not only effectively addresses users' questions but also enhances the overall browsing experience, promoting engagement and satisfaction.**

**Key aspects of the problem include:**

**1. User Engagement: Traditional FAQ pages may lack visual appeal and interactivity, resulting in low user engagement and limited effectiveness in guiding users to relevant information. There is a need to design a FAQ page that captures users' attention and encourages them to explore the content further.**

**2. Responsive Design : With the proliferation of mobile devices and varying screen sizes, ensuring that the FAQ page is accessible and functional across all devices is crucial. The design must be responsive and adaptable, providing a seamless experience regardless of the user's device.**

**3. Ease of Navigation: Users should be able to navigate the FAQ page intuitively, finding answers to their questions quickly and efficiently. The layout and organization of the content should facilitate easy navigation and promote discoverability.**

**4. Visual Presentation : Aesthetic appeal plays a significant role in user perception and engagement. The design of the FAQ page should be visually pleasing, with clear typography, appropriate use of colour, and well-defined sections to enhance readability and comprehension.**

**5. Accessibility: It's essential to ensure that the FAQ page is accessible to all users, including those with disabilities. Accessibility considerations should be integrated into the design process to ensure that the page can be navigated and understood by everyone, regardless of their abilities.**

**Addressing these challenges requires a holistic approach that combines thoughtful design, effective use of technology, and adherence to best practices in web development. By defining and addressing these key aspects, we can create a FAQ page that not only meets users' needs but also contributes to a positive and engaging browsing experience.**

## ****Requirements****

Requirements for the Future of Web Design:

1. User Interface Design:

- The FAQ page should have a visually appealing and modern design that aligns with the overall aesthetic of the website or application.

- Clear typography, appropriate use of colour, and consistent styling should be applied to enhance readability and comprehension.

- The layout should be responsive and adaptable, ensuring optimal viewing and usability across various devices and screen sizes.

2. Content Organization:

- FAQ sections and questions should be logically organized and easy to navigate.

- Each FAQ section should have a clear heading, and questions within each section should be presented in a structured manner.

- A collapsible accordion-style layout can be implemented to allow users to expand and collapse individual questions to view their corresponding answers.

3. Interactivity and Functionality:

- Interactive features such as accordion behaviour for collapsible answers should be implemented to enhance user experience and promote engagement.

- Users should be able to navigate the FAQ page seamlessly, with smooth transitions and intuitive interactions.

4. Accessibility:

- The FAQ page should adhere to accessibility standards and best practices to ensure that it is usable by all users, including those with disabilities.

- Proper semantic HTML markup should be used, and ARIA attributes can be applied where necessary to enhance accessibility.

- The page should be navigable and understandable via keyboard navigation and screen readers.

5. Performance and Optimization:

- The page should be optimized for performance, with fast loading times and minimal resource usage.

- CSS and JavaScript files should be minified and concatenated to reduce file size and improve load times.

- Images should be optimized for the web to minimize bandwidth usage without sacrificing quality.

6. Maintainability and Scalability:

- The codebase should be well-organized and maintainable, following best practices in code structuring and documentation.

- CSS should be modularized and organized using a scalable architecture to facilitate future updates and additions.

- The FAQ page should be designed with scalability in mind, allowing for easy expansion and addition of new sections or questions in the future.

By meeting these requirements, the FAQ page will effectively address users' common queries and concerns while providing a seamless and enjoyable browsing experience.

**Hardware requirements**

For a project focused on designing and developing a FAQ page using Tailwind CSS, the hardware requirements are relatively minimal. Here's a basic outline of the hardware needed:

1. Computer : Any modern computer capable of running web development tools and browsers should suffice. This includes desktops, laptops, or even tablets with appropriate development environments installed.

2. Operating System: The choice of operating system depends on the developer's preference. Tailwind CSS development is platform-agnostic, meaning it can be used on Windows, macOS, or Linux systems.

3. Processor and RAM: While Tailwind CSS itself does not have high hardware requirements, a computer with a reasonably fast processor (such as an Intel Core i3 or AMD Ryzen 3) and sufficient RAM (8GB or more recommended) will ensure smooth performance when running development tools and browsers simultaneously.

4. Storage: Adequate storage space is necessary for storing project files, development tools, and other related resources. A minimum of 256GB of storage is recommended, but more significant storage capacity may be necessary depending on the size and complexity of the project and any additional software or files required.

5. Internet Connection: An internet connection is essential for accessing documentation, resources, and libraries related to Tailwind CSS and other web development tools. A stable and reasonably fast internet connection will facilitate seamless development and testing processes.

Overall, the hardware requirements for developing a FAQ page using Tailwind CSS are modest and should be readily accessible to most developers, regardless of their hardware setup. The focus should primarily be on having a reliable computer system with enough processing power, memory, and storage to support web development tasks effectively.

**Proposed design and methodology**

**Proposed Designs for the faq page:**

**Design Proposal for FAQ Page:**

**1. Header Section:**

**- The header section will contain the website or application logo on the left and navigation links on the right for seamless user navigation.**

**- A prominent title such as "Frequently Asked Questions" will be displayed to clearly indicate the purpose of the page.**

**2. FAQ Sections:**

**- Each FAQ section will be visually distinct, with a bold heading to identify the category of questions.**

**- Questions within each section will be presented in a clean and organized manner, with sufficient spacing between each question to enhance readability.**

**- A "" icon next to each question will indicate its expandable/collapsible nature, allowing users to reveal or hide the answer as needed.**

**3. Answer Display:**

**- Answers to the questions will be displayed in an accordion-style layout, ensuring that only one answer is visible at a time to minimize clutter.**

**- When a user clicks on a question, the corresponding answer will smoothly expand below it, providing a seamless and intuitive user experience.**

**- A subtle animation or transition effect can be applied to enhance the interaction and make the accordion behaviour more engaging.**

**4. Visual Elements:**

**- A minimalist design approach will be adopted, with a focus on clean lines, ample white space, and subtle colour accents to maintain visual interest.**

**- Use of a cohesive colour palette that complements the website or application's branding will help create a harmonious visual identity.**

**- Clear typography with readable font sizes and styles will ensure that content is easy to understand and accessible to all users.**

**5. Navigation and Accessibility:**

**- A sticky navigation bar at the top of the page will enable users to quickly jump to different sections of the FAQ, enhancing accessibility and ease of navigation.**

**- Proper semantic HTML markup and ARIA attributes will be employed to ensure that the FAQ page is accessible to users of all abilities, including those using assistive technologies.**

**6. Mobile Responsiveness:**

**- The design will be fully responsive, ensuring optimal display and usability across a wide range of devices, including smartphones and tablets.**

**- For smaller screens, the FAQ sections and questions will stack vertically, and the accordion behaviour will remain intact to provide a seamless mobile experience.**

**7. Call to Action (CTA):**

**- A subtle CTA button or link at the bottom of the FAQ page can encourage users to take further action, such as contacting support or exploring related resources.**

**- The CTA should be strategically placed and visually distinct without detracting from the primary purpose of the FAQ page.**

**By incorporating these design elements and principles, the proposed FAQ page will not only effectively address users' common queries and concerns but also deliver a visually appealing and user-friendly experience that enhances engagement and satisfaction.**

## ****Methodology****

**Methodology for the faq page:**

1. Research and Content Gathering:

- Conduct research to identify common questions and concerns that users may have about the website or application.

- Collaborate with stakeholders, customer support teams, and end users to gather relevant content and insights for the FAQ page.

2. Information Architecture:

- Define the structure and organization of the FAQ page, including the categorization of questions into logical sections.

- Create a hierarchical outline of the FAQ content to ensure that it is easy to navigate and understand.

3. Wireframing and Prototyping:

- Develop wireframes or low-fidelity prototypes to visualize the layout and functionality of the FAQ page.

- Iterate on the design based on feedback from stakeholders and usability testing to refine the user experience.

4. Design Implementation:

- Translate the finalized design into code using HTML, CSS, and JavaScript.

- Utilize the Tailwind CSS framework to apply styling and layout components efficiently, leveraging utility classes for rapid development.

5. Accordion Behaviour:

- Implement the accordion behaviour using JavaScript or a JavaScript framework such as Vue.js or React.

- Write code to toggle the visibility of answer content when a question is clicked, ensuring smooth transitions and accessibility.

6. Responsive Design:

- Ensure that the FAQ page is fully responsive, adapting seamlessly to different screen sizes and devices.

- Test the responsiveness of the page across various breakpoints and make adjustments as needed to optimize the user experience.

7. Accessibility Considerations:

- Incorporate accessibility best practices into the design and development process to ensure that the FAQ page is usable by all users, including those with disabilities.

- Use semantic HTML markup, ARIA attributes, and keyboard navigation support to enhance accessibility.

8. Testing and Quality Assurance:

- Conduct thorough testing of the FAQ page across different browsers, devices, and assistive technologies to identify and address any compatibility issues.

- Perform usability testing with real users to gather feedback on the clarity, effectiveness, and usability of the FAQ page.

9. Deployment and Iteration:

- Deploy the FAQ page to a staging environment for final review and approval.

- Monitor user feedback and analytics data to identify opportunities for further optimization and iteration, such as adding new questions or refining the design based on user interactions.

10. Documentation and Maintenance:

- Document the implementation details, including code structure, dependencies, and design decisions, for future reference.

- Establish a maintenance plan to ensure that the FAQ page remains up-to-date and continues to meet the evolving needs of users and stakeholders.

By following this methodology, you can systematically design, develop, and deploy a high-quality FAQ page that effectively addresses users' common questions and enhances the overall user experience.

**Algorithms used**

The algorithm used for implementing the accordion behaviour on the FAQ page can vary depending on the developer's preferences and the specific requirements of the project.

This algorithm follows these steps:

1. Selects all question elements on the FAQ page.

2. Adds a click event listener to each question element.

3. Toggles the 'active' class on the clicked question element to visually indicate its active state (e.g., by changing its background colour).

4. Selects the answer element corresponding to the clicked question using the `nextElementSibling` property.

5. Toggles the visibility of the answer element by adjusting its `maxHeight` property. If the answer is currently hidden, it sets the `maxHeight` to its scroll height, making it visible. If the answer is already visible, it resets the `maxHeight` to `null`, hiding it.

This algorithm provides a simple and effective way to implement accordion behaviour on the FAQ page using vanilla JavaScript. However, frameworks like Vue.js or React can also be used to achieve similar functionality with additional features and optimizations.

**Code**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>FAQ Page</title>

    <link href="https://cdn.jsdelivr.net/npm/tailwindcss@2.2.19/dist/tailwind.min.css" rel="stylesheet">

</head>

<body class="font-sans bg-center bg-cover" style="background-image: url('https://th.bing.com/th?id=OIP.q-K1Dljdyo5-lERKBT0YwAHaEc&w=322&h=193&c=8&rs=1&qlt=90&o=6&dpr=1.3&pid=3.1&rm=2');">

<div class="container mx-auto ">

    <div id="faq-section" class="  max-w-4xl mt-2 px-2s py-20 rounded shadow-3g">

        <h1 class="text-6xl font-bold px-4 font-serif text-darkblue-800 mb-10">Frequently Asked Questions</h1>

        <div class="search-container flex  mb-10 ml-6">

            <input type="text" placeholder="Search..." id="search-box" class="flex-1 px-4 py-2 rounded border border-gray-400">

        </div>

        <div class="space-y-4">

            <div class="faq-item bg-gray-200 p-4 rounded cursor-pointer" id="faq1">

                <h2 class="toggle flex justify-between items-center text-xl font-bold text-gray-700">

                    What is JavaScript?

                    <svg xmlns="http://www.w3.org/2000/svg" class="h-6 w-6 arrow rotated" fill="none" viewBox="0 0 24 24" stroke="currentColor"><path stroke-linecap="round" stroke-linejoin="round" stroke-width="2" d="M19 9l-7 7-7-7"></path></svg>

                </h2>

                <p class="hidden">JavaScript is the most powerful and versatile programming language used in the web</p>

            </div>

            <div class="faq-item bg-gray-200 p-4 rounded cursor-pointer" id="faq2">

                <h2 class="toggle flex justify-between items-center text-xl font-bold text-gray-700">

                    Why to learn JavaScript?

                    <svg xmlns="http://www.w3.org/2000/svg" class="h-6 w-6 arrow rotated" fill="none" viewBox="0 0 24 24" stroke="currentColor"><path stroke-linecap="round" stroke-linejoin="round" stroke-width="2" d="M19 9l-7 7-7-7"></path></svg>

                </h2>

                <p class="hidden">JavaScript is an essential programming language for web developers to learn as it grows rapidly. </p>

            </div>

            <div class="faq-item bg-gray-200 p-4 rounded cursor-pointer" id="faq3">

                <h2 class="toggle flex justify-between items-center text-xl font-bold text-gray-700">

                    Who developed JavaScript?

                    <svg xmlns="http://www.w3.org/2000/svg" class="h-6 w-6 arrow" fill="none" viewBox="0 0 24 24" stroke="currentColor"><path stroke-linecap="round" stroke-linejoin="round" stroke-width="2" d="M19 9l-7 7-7-7"></path></svg>

                </h2>

                <p class="hidden">Brendan Eich first developed JavaScript</p>

            </div>

            <div class="faq-item bg-gray-200 p-4 rounded cursor-pointer" id="faq4">

                <h2 class="toggle flex justify-between items-center text-xl font-bold text-gray-700">

                    How does it work?

                    <svg xmlns="http://www.w3.org/2000/svg" class="h-6 w-6 arrow" fill="none" viewBox="0 0 24 24" stroke="currentColor"><path stroke-linecap="round" stroke-linejoin="round" stroke-width="2" d="M19 9l-7 7-7-7"></path></svg>

                </h2>

                <p class="hidden">It works seamlessly alongside HTML and CSS .</p>

            </div>

            <div class="faq-item bg-gray-200 p-4 rounded cursor-pointer" id="faq5">

                <h2 class="toggle flex justify-between items-center text-xl font-bold text-gray-700">

                    What can JavaScript Do?

                    <svg xmlns="http://www.w3.org/2000/svg" class="h-6 w-6 arrow" fill="none" viewBox="0 0 24 24" stroke="currentColor"><path stroke-linecap="round" stroke-linejoin="round" stroke-width="2" d="M19 9l-7 7-7-7"></path></svg>

                </h2>

                <p class="hidden">JavaScript Can Change HTML Content, HTML Attribute Values and HTML Styles (CSS)</p>

            </div>

            <div class="faq-item bg-gray-200 p-4 rounded cursor-pointer" id="faq6">

                <h2 class="toggle flex justify-between items-center text-xl font-bold text-gray-700">

                    Where can I download JavaScript?

                    <svg xmlns="http://www.w3.org/2000/svg" class="h-6 w-6 arrow" fill="none" viewBox="0 0 24 24" stroke="currentColor"><path stroke-linecap="round" stroke-linejoin="round" stroke-width="2" d="M19 9l-7 7-7-7"></path></svg>

                </h2>

                <p class="hidden">JavaScript is already running in your browser on your computer, on your tablet, and on your smart-phone.</p>

            </div>

        </div>

        </div>

        <p class="not-found hidden text-4xl text-white mt-10">No results found.</p>

    </div>

</div>

<script>

    const toggleElements = document.querySelectorAll('.toggle');

    toggleElements.forEach((toggleElement) => {

        toggleElement.addEventListener('click', () => {

            const answer = toggleElement.nextElementSibling;

            const arrow = toggleElement.querySelector('.arrow');

            if (answer.classList.contains('hidden')) {

                answer.classList.remove('hidden');

                arrow.classList.add('rotated');

            } else {

                answer.classList.add('hidden');

                arrow.classList.remove('rotated');

            }

        });

    });

    document.getElementById('search-box').addEventListener('input', function() {

        const searchTerm = this.value.trim().toLowerCase();

        const faqItems = document.querySelectorAll('.faq-item');

        const notFoundMessage = document.querySelector('.not-found');

        let found = false;

        faqItems.forEach((faqItem) => {

            const question = faqItem.querySelector('h2').textContent.trim().toLowerCase();

            const answer = faqItem.querySelector('p').textContent.trim().toLowerCase();

            if (question.includes(searchTerm) || answer.includes(searchTerm)) {

                faqItem.classList.remove('hidden');

                found = true;

            } else {

                faqItem.classList.add('hidden');

            }

        });

        if (found) {

            notFoundMessage.classList.add('hidden');

        } else {

            notFoundMessage.classList.remove('hidden');

        }

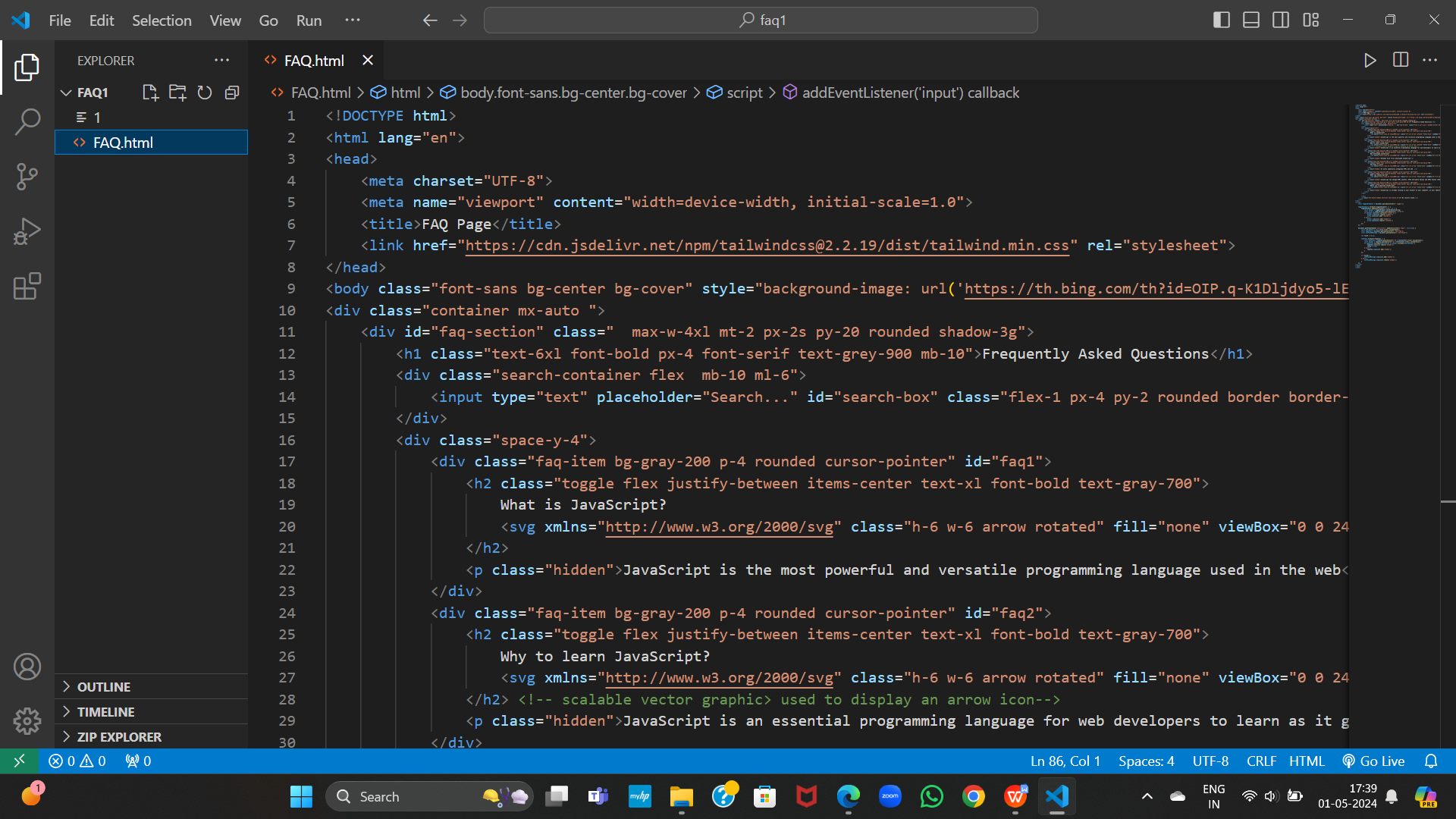
    });

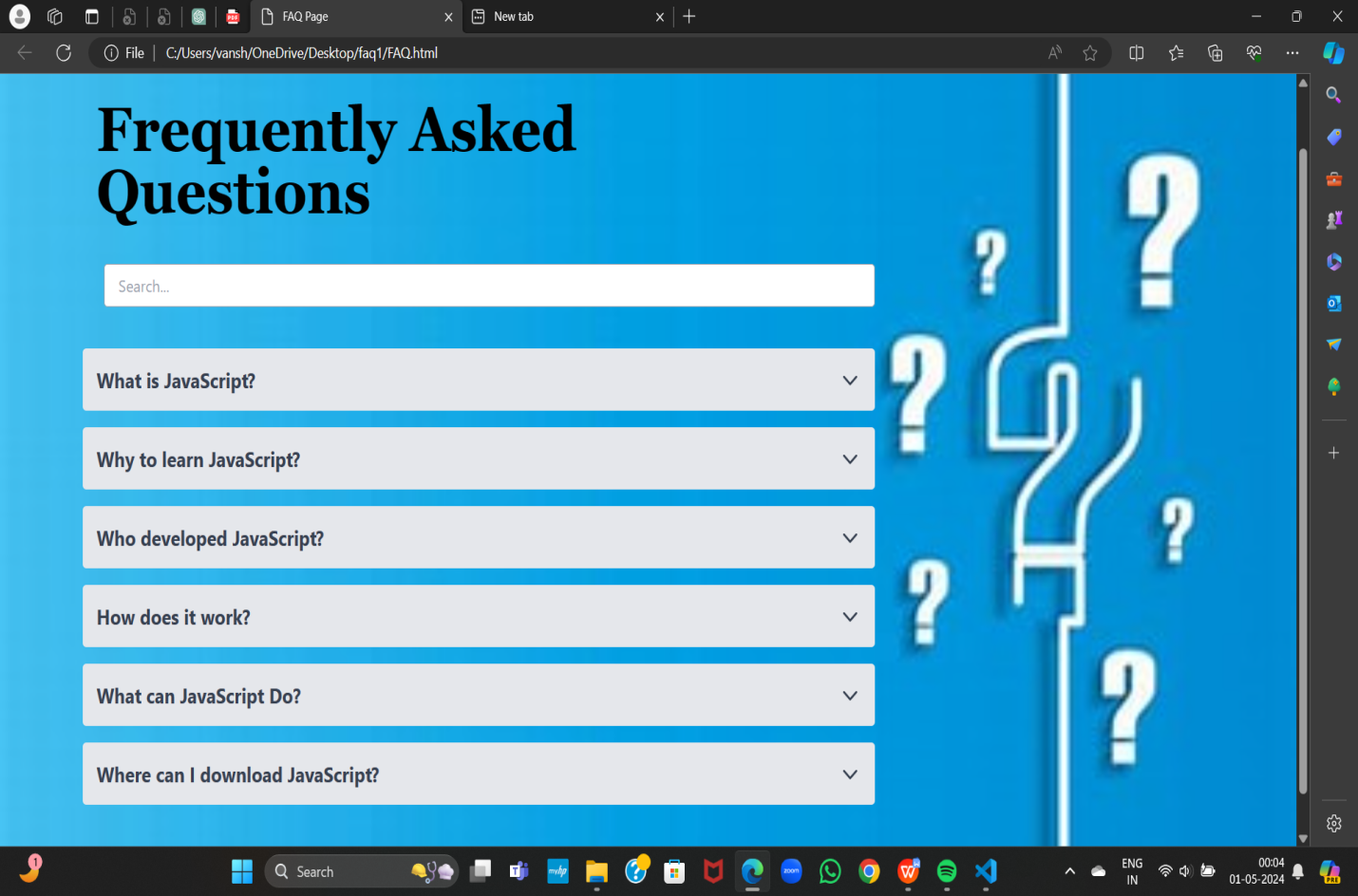
</script>

</body>

</html>

**File formate**

**Results**

****

**References:**

1. **<https://www.w3schools.com/js/>**
2. **<https://www.geeksforgeeks.org/javascript/>**
3. **<https://www.youtube.com/watch?v=PkZNo7MFNFg>**

**Thank you**