**Name:** Md Ali Hasan Jidan

**Student ID:** B01011849

**Assigned Tasks:** Home and About page- HTML, CSS, JavaScript

**Explanation of HTML code**

My HTML file actually creates the “bones” of the entire webpage—the structure and basic content (text, images, links). The very first thing is <!DOCTYPE html> it tells the browser, “This document is written in the HTML5 standard.” Right after that, <html lang="en"> tells the browser that the language of the page is English.

The <head> tag contains some information about the webpage that is not directly visible on the screen, but is very useful for browsers and search engines. For example, <meta charset="UTF-8"> this ensures that all Unicode characters Bangla, English, Arabic are displayed correctly. <meta name="viewport" content="width=device-width, initial scale=1.0"> tells the page to be responsive on multiple size screens. <title>Coffee</title> determines the text that is displayed in the browser tab. In this <head>, I also linked a few external files like swipers CSS (used to create the slider), Font Awesome (used to display the icons), and my own css/style.css (the main style of my design). I also added a favicon (a small icon that appears in the tab). All of these helps make the page stylish and responsive.

The actual visible part of the page comes in the <body> tag. At the very top is <header class="header"> this is the top bar or header section of the website. Here, first, there is <div id="menu-btn" class="fas fa-bars"></div> this is a “hamburger” icon on small screens, pressing this button opens/closes the menu (this is done by JavaScript). Then, there is <a href="#" class="logo">coffee <i class="fas fa-mug-hot"></i></a> this is my website’s brand logo. Here, there is a coffee mug icon next to the “coffee” text. Then <nav class="navbar"> here are various links home and about these take you to sections within the same page (#home, #about). Menu, review these take you to another page called habibur.html. Book and the “book a table” button above take you to komal.html and My Booking to another page called myBooking.html.

Then comes the <! --HOME--> section. <section class="home" id="home">—this id="home"—the reason the href="#home" link in the top menu scrolls here. Inside is a <div class="row"> here on the left side, in the content div, there is the title “fresh coffee in the morning” and the “buy one now” button clicking on this button takes you to the habibur.html page. On the right side, in the. image div, there is a large image this is the main coffee cup image (class="main-home-image-1"). Its alt text is nicely described so that even prisoners can use my website.

Below the home section is a <div class="image-slider"> here are three small images (thumbnails). They look small, but the work is big when you click on any of these images, JavaScript changes the src of the main-home-image as a result, the large image changes immediately according to the thumbnail. That is, a very lightweight gallery effect has been created here, that too without any heavy plugins.

Next comes the <! -- ABOUT --> section. <section class="about" id="about"> here too, because of the id="about" it goes here when you click on the about link in the menu. The title of this section is <h1 class="heading">about us <span>why choose us</span></h1> here the large text on the outside “about us” and the small text on the inside span “why choose us” the two are displayed in very different styles with CSS. Below is an image on the left side of row an image of a barista at work in a cafe. On the right side, in content, there is an h3 title, a large paragraph where your coffee specialties (single-origin beans, direct-trade, SCA-certified baristas, etc.) are written these give the user an idea of the brand’s value. There is a button called “read more” (currently href="#" meaning it doesn’t take you anywhere, later I will link to the ‘read more’ page). Below that, there is a icons-container here are three icon blocks (quality coffee, our branches, free delivery) that visually show your service points. At the end of this section, I have placed a button in a small content div called “go habibur section” that takes you to habibur.html.

In short, this HTML file creates the structure and main content of the entire webpage. At the beginning, `<!DOCTYPE html>` and `<html lang="en">` are used to inform the browser about the HTML5 standard and the language of the page. The `<head>` section contains the meta tags, page title, Google Fonts, Swiper, Font Awesome, links to your own CSS files and favicon, which makes the page responsive, stylish and displayable correctly.

The `<body>` section first contains the header, which contains the hamburger menu button, logo and navigation links. The home section (id="home") contains the title and buttons on the left and the main coffee cup image on the right. Below is the thumbnail image slider, where clicking changes to the larger image.

Then the about section (id="about") contains the stylish heading, the barista image on the left, the content on the right, and the service icon. Overall, this HTML file represents the basic structure and information of the website through page structure, text, images, and links.

Here is the HTML Code:

**Explanation of the CSS code**

In my CSS styles, I first imported the “Poppins” font from Google Fonts, it gives the text a modern and clean look. Then I created some custom variables in (:root --main-color, --border-radius, --border, etc.) Variables are meant to set the color/size/border in one place and use this name wherever you need it. This keeps the code clean and makes it easy to change.

\*{margin:0; padding:0; box-sizing: border-box; …} These are global “resets”; I removed the browser default margins/padding and sized it my way. Text-transform: capitalize; capitalizes the first letter of each word this works for English text however, it is not always necessary to capitalize all words, so if necessary, set this setting on a separate element. Transition: all .2s linear makes it look smoothly animated when there is a slight change in style (such as hovering over a button).

Html {font-size: 62.5%} This is a common trick: 62.5% of the browser's default 16px is ≈ 10px. As a result, 1rem ≈ 10px can be assumed. So, 1.7rem = 17px is easy to calculate in my head. With scroll-behavior: smooth, the page scrolls down smoothly when you click on the menu. Scroll-padding-top: 7rem, Since the header is fixed (always visible at the top), some “padding” has been left on top so that the title does not get hidden under the header when scrolling in-page.

.heading , This is the large heading style. Here, a hollow outline title has been created on the outside with color: transparent and -webkit-text-stroke. The span inside it has been brought to the very center (position: absolute; transform: translate (-50%, -50%)) As a result, it looks like a double-layer: large outline text, small filled text inside. This makes the heading look stylish.

The .btn style has your own custom border-radius (different curves on each side) and a dashed border on hover this “organic” button style gives the page a unique feel. .btn: hover changes both the border-style and border-radius, creating a soft, eye-catching animation that tells the user “this is clickable.”

.header fixed meaning it will stay on top even if you scroll. The box-shadow adds a light shadow under the header making the visual stand out. .navbar a has a fixed font-size and color. #menu-btn {display: none} Hiding the hamburger icon on desktop; usually it is displayed: inline-block in media queries for smaller screens and .navbar is hidden and shown with the active class first. Practically, a rule like .navbar.active {display: block } will be needed to work on mobile.

.home section has min-height: 100vh meaning the hero section will span the entire viewport height; background has a large image, background-size: cover that is, it will crop/fit properly across the screen. .home .row { display: flex } where content and images sit side by side in two columns, gap: 1.5rem—there is a gap between the columns.

The most interesting part is .home .row .image img{ animation: float 4s linear infinite } where the large image of the coffee cup slowly floats up and down on its own. In the @keyframes below, float at 0% and 100% means the normal position, and at 50%-translateY(0) means a little up in the middle. A cycle completes in 4 seconds and continues indefinitely. This animation is your “floating cup” effect.

.home .image-slider Here are three thumbnails. .image-slider img:hover { transform: translateY(-2rem) } When the mouse is hovered over, the images “jump” up a little, which encourages the user to click.

.about section .about .row .image img { animation: aboutImage 4s linear infinite }—Here the image is scaled up a little (from scale(.9) to scale(.8) to .9) and the border-radius is also changed resulting in a soft pulsing effect. On the content side, there are three icon cards for title, para, and .icons-container; these use your previous --border and --border-radius variables, so the style is consistent across the site.

Overall, this CSS code controls the colors, fonts, layout, and animations of the entire website. Google Fonts’ “Poppins” font, custom variables, global reset, smooth scroll, stylish headings, buttons with hover effects, fixed headers, responsive menus, hero images, flex layouts, floating coffee cups, and pulsing image effects all of which make the site modern, user-friendly, and attractive.

Here is the CSS code:

**Explanation of the JavaScript**

My first JavaScript task is to open/close the menu on a small screen. let menu = document.querySelector('#menu-btn'), This extracts the #menu-btn element (the hamburger icon). let navbar = document.querySelector('.navbar');—This extracts the navigation menu. Then menu.onclick = () => { ... } This means writing what happens when the icon is clicked. Inside menu.classList.toggle('fa-times'); Font Awesome's fa-times class is toggled; as a result, the “three dots” (hamburger) icon changes to a “cross/close” icon.

window.onscroll = () => { menu.classList.remove('fa-times') navbar.classList.remove('active'); } This line is very user-friendly the menu auto-closes as soon as the user starts scrolling, and the icon returns to its previous state. The big dropdown menu doesn't get in the way when scrolling.

The next part is the thumbnail-to-hero-image switching in the home section. document.querySelectorAll('.image-slider img') This extracts all the images in the .image-slider (a list). forEach(images => { images.onclick = () => { ... } }) The click event is set for each image. Inside, var src = images.getAttribute('src'); The source (image file) of the clicked image is taken. document.querySelector('.main-home-image').src = src; Then it is placed in the src of the large image. As a result, the large image changes immediately. No plugins were needed just a few lines of JavaScript did the job.

Finally, var swiper = new Swiper(".review-slider", { ... })—This is the code to run the Swiper slider.

This JavaScript code basically adds various interactive and user-friendly features to the webpage. First, it brings out the hamburger icon (#menu-btn) and the navigation menu (.navbar) on small screens. Then, clicking on the icon changes the hamburger icon to a cross icon by toggling the `fa-times` class and opens or closes the menu. The menu is automatically closed on the scroll event and the icon returns to its previous state, so the menu is not in the way while scrolling.

In the next part, the click event is set on all the thumbnail images in `.image-slider`. When a thumbnail is clicked, its `src` is set to the larger hero image, so the larger image changes immediately no additional plugins are required to do this, just a few lines of JavaScript are used to solve this.

Finally, an interactive slider is implemented for `.review-slider` using the Swiper library, which displays reviews or testimonials in an attractive way. Overall, this code creates a vibrant and smooth user experience on the webpage, including menu navigation, image switching, and sliders.

In short, JavaScript's job is to bring the page to life. Clicking will open/close the menu, scrolling will auto-collapse the menu, clicking on the thumbnail will change the larger image all of this is handled by JavaScript.