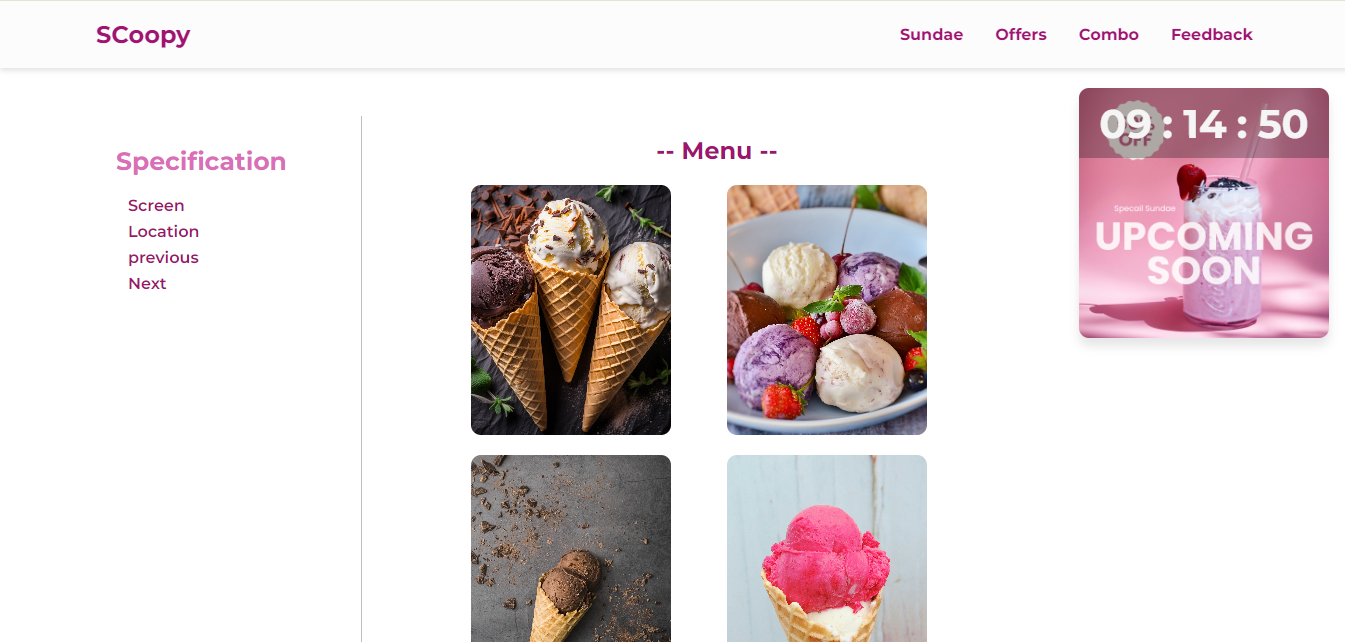
**— SCoopy —**

Note : CSS is pre-coded and doesn't need to be explained.

Scoopy is an ice cream webpage where we have a clear understanding of the browser object model concepts like screen, location, history, timing event, and their properties.



Index.html file

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<!-- use font-awesome -->

<!-- Pixabay -->

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.5.2/css/all.min.css">

<link rel="stylesheet" href="style.css">

<title>Ice Web Page</title>

</head>

<body onload="ad()">

<nav>

<h1>SCoopy</h1>

<ul class="navigation">

<li><a href="" class="nav-link">Sundae</a></li>

<li><a href="Offer.html" class="nav-link">Offers</a></li>

<li><a href="" class="nav-link">Combo</a></li>

<li><a href="" class="nav-link">Feedback</a></li>

</ul>

</nav>

<!-- offer section -->

<div class="count-container">

<div class="countdown">

<span id="hrs">00 </span>

<span>:</span>

<span id="mins">00</span>

<span>:</span>

<span id="sec">00</span>

</div>

</div>

<!-- ads -->

<div class="ads-position">

<div class="ads">

<img src="images/Offer.png" alt="">

<i class="fa-regular fa-circle-xmark" id="close" onclick="closeAd()"></i>

</div>

</div>

<!-- ----Products------ -->

<section class="home">

<div class="left">

<h3>Specification</h3>

<p id="ice1" onclick="size()">Screen</p>

<p id="ice2" onclick="host()">Location</p>

<p id="ice3" onclick="goBack()">previous</p>

<p id="ice4" onclick="goforward()">Next</p>

</div>

<div class="right">

<h1>-- Menu --</h1>

<div class="grid">

<div class="grid-box" id="ice1"><img src="images/sundae1.jpg" alt=""></div>

<div class="grid-box" id="ice1"><img src="images/sundae5.jpg" alt=""></div>

<div class="grid-box" id="ice2"><img src="images/sundae3.jpg" alt=""></div>

<div class="grid-box" id="ice2"><img src="images/sundae9.jpg" alt=""></div>

</div>

</div>

</section>

<script src="script.js"></script>

</body>

</html>

Window Screen:

Property: screen.width, screen.Height, screen.availwidth, screen.availHeight, screen.availcolorDepth, screen.pixelDepth.

**//The window.screen object can be written without the window prefix.**

**function size() {**

**document.getElementById("ice1").innerHTML = screen.width + " x " + screen.height ;**

**}**

**// screen.width - returns the width of the visitor's screen in pixels.**

**// screen.height - returns the height of the visitor's screen in pixels.**

**// screen.availWidth - returns the width of the visitor's screen, in pixels, minus interface features like the Windows Taskbar.**

**// screen.availHeight - returns the height of the visitor's screen, in pixels, minus interface features like the Windows Taskbar.**

**// screen.colorDepth - returns the number of bits used to display one color.**

**// screen.pixelDepth - returns the pixel depth of the screen.**

Window Location :

Property: window.location.href, window.location.hostname, window.location.pathname, window.location.protocol, window.location.assign().

function host() {

document.getElementById("ice2").innerHTML = window.location.protocol;

}

// window.location.href - returns the href (URL) of the current page

// window.location.hostname - returns the domain name of the web host

// window.location.pathname - returns the path and filename of the current page

// window.location.protocol - returns the web protocol used (http: or https:)

// window.location.assign(naturalicecreams.in) - loads a new document

Window History:

Property: window.history.back(), window.history.forward()

function goBack() {

document.getElementById("ice2").innerHTML = window.history.back();

}

// window.history.back() - loads the previous URL in the history list.

function goforward() {

document.getElementById("ice2").innerHTML = window.history.forward();

}

// window.history.forward() - loads the next URL in the history list

Timing event: The window object allows execution of code at specified time intervals.

Property: setTimeout ,setInterval

Settimeout-

function ad(){

setTimeout(() => {

document.querySelector(".ads-position").style.display = "block";

}, 6000);

}

setInterval-

function time(){

const futureDate = new Date('2024-06-06T00:00:00');

const timeDiff = futureDate - Date.now();

// The padStart() method pads a string with another string (multiple times) until it reaches a given length.

const hours = String(Math.floor(timeDiff / 3600000)).padStart(2, '0');

const minutes = String(Math.floor((timeDiff % 3600000) / 60000)).padStart(2, '0');

const seconds = String(Math.floor((timeDiff % 60000) / 1000)).padStart(2, '0');

document.getElementById("hrs").innerText=hours;

document.getElementById("mins").innerText=minutes;

document.getElementById("sec").innerText=seconds;

}

setInterval(time,1000);

-