Introduction to JavaScript

* JS is a OOP language
* JS is used to make interative web pages for mobile and PC, mobile apps (iOS/Android), Full-fledged desktop apps,
* JS is case sensitive
* Objects have properties and behaviours
* HTML and CSS make websites static while JS make them lively (Dynamic and interactive)
* JS instructs the browser how to dynamically change the content and presentation of the website based on user interactions and user inputs
* JS is a programming language - Variables, functions, loops, conditions
* We add JS code to HTML within <script> tag

Document Object Model (DOM)

* DOM is the bridge between HTML and JS
* Object oriented representation of a document
* It structures a document as a tree
  + Root: <html> document
  + Main branches: <body> and <head>
  + <head> has 2 branches
    - Sub branches: <meta> and <title>
    - <meta> has one attribute: charset = “UTF-8”
  + <body> has 2 branches
    - Sub branches: <a> and <p>
    - <p> has id attribute
    - <a> has href attribute
* To access a property - objectName.propertyName - Ex: car.color
* To access behaviour - objectName.functionName() - Ex: car.accelerate()

Variables

* var is used to declare variables
* Ex: var x = “Programming Hub”;
* var x = "Programming Hub";

document.write(x);

document.write(x);

* var username;

username = “Alice”;

document.getElementById(“pid”).textContent = “Hello” + username + “!”;

var greeting;

greeting = “Good Morning!”;

document.getElementById(“pid2”).textContent = greeting;

document.write()

* This is used to print text in JS
* Ex: document.write(“Hello World”);

Adding to HTML document: Alert

* Adds a dialog box that pops up in the screen
* <script> alert(‘Hello world!’); </script>

Boolean

* true and false
* Ex: var x = false;

Functions

* var username;

username = “Alice”;

display(“pid”, “Hello” + username + “!”);

var greeting;

greeting = “Good Morning!”;

display(“pid2”, greeting);

* Here function is: display(id,text)
* Below code shows how to define a function
* function display(id,text) {

document.getElementById(id).textContent = text;

}

Events

* Occurance (action) that happen in the system
* happens when user/browser manipulate a web pages
* <html>

<head>

    <h1> JavaScript Events </h1>

</head>

<body>

    <script language="JavaScipt" type="text/JavaScript">

        function clickevent() {

            document.write("Hello");

        }

    </script>

    <form>

        <input type = "button" onclick="clickevent()" value = "Submit">

    </form>

</body>

</html>

* <html>

<head>

    <h1> JavaScript Events </h1>

</head>

<body>

    <script language="JavaScipt" type="text/JavaScript">

        function clickevent() {

            document.write("Hello");

        }

    </script>

    <form>

        <input type = "text" id="input" onkeydown = "clickevent()">

    </form>

</body>

</html>

Loops

* for - loop through a block of code
  + for (let i = 0; i < 5; i++) {

console.log(“Walking each one step”);

}

* for/in - loop through properties of object
  + for (i in A) {

//code

}

* for/of - loop through values of inerrable object (Ex: variable)
  + for (i of A) {

//code

}

* while
  + while (i < 5) {

//code

}

* do/while

Conditions

* if (condition) {

//code

}

Handling forms (input fields, checkboxes, radio buttons, validations)

* <script type = “text/javascript”>

function validateform() {

let x = document.forums[“myForm”][“fname”].value;

if (x == “”) {

alert(“Name must be filled out”);

return false;

}

}

</script>

<form name = “Application” action = “/action\_page.php” onsubmit = “return

validateForm()” method = “post”>

Name: <input type = “text” name = “fname”>

<input type = “submit” value = “Submit”>

</form>

JS error handling

* try {

//code

}

catch(err) {

//code

}

* ‘finally’ is optional
* <script>

function myFunction() {

const message = document.getElementById(“a1”)

message.innerHTML = “”;

let x = document.getElementById(“sample”).value;

try {

if(x == “”) throw “empty”;

if(isNaN(x)) throw “not a number”;

x = Number(x);

if (x < 5) throw “too low”;

if (x > 10) throw “too high”;

}

catch (err) {

message.innerHTML = “Input is ” + err;

}

finally {

document.getElementById(“demo”).value = “”;

}

}

</script>

* Reference error
* Syntax error
* Type error
* Range error

TypeScript influence

* Superset of JavaScript with design-time support for type safety and tooling
* Doesn’t run in the browser
* TypeScript code is compiled to JS, which then runs in the browser

Angular

* AngularJS is a JS framework
* Extends HTML attributes with directives and binds data to HTML with expressions
* Free and open source framework used for frontend web application development

React

* JS library to build UI
* Used to build single page applications easily

React Native

* Cross platform mobile application development framework (Can run in iOS and Android)
* Similar to react, but uses native component instead of web components

ReactJS

* Free and open source frontend JS library

VueJS

* Progressive JS framework
* Vue.js lets you extend HTML with HTML attributes called directives

AJAX

* AJAX - Asynchronous JavaScript And XML
* Not a programming language
* AJAX uses combination of below 2
  + Browser built-in XMLHttpRequest object
  + JS and HTML DOM

REST API

* REST is a protocol
* REST API provide flexible and lightweight way to integrate applications
* Used in connecting components in microservices architechtures
* Use AJAX to send RESTful requests
* XMLHttpRequest object is used to exchange data with the server
* xhttp.open(“GET”, “ajax\_info.txt”, true);

xhttp.send();

Client-server architecture

* Modern applications run on client server architecture
* Architecture seperates application software to 2 categories
  + Servers
  + Clients
* It’s effective when client and server applications are hosted and executed on different machines connected via a network

JavaScript web application hosting

* Web hosting is required to make the website available for others
* Frontend: HTML, CSS, JS
* Backend: JS running on a Node.js server
* Solutions to host JS web application
  + Firebase hosting
  + Typical web hosting service

User interface (UI) and user experience (UX) design

* 4 golden rules of UI/UX design
  + Place users in control of the interface
  + Make it comfortable to interact with a website
  + Reduce cognitive load (make it easier to understand)
  + Make UI consistent

Chrome developer tools for debugging UI

* Edit HTML, CSS, JS and preview changes in real time
* Debug JS code with breakpoint
* Perform website performance testing and generate reports

Designing responsive web pages for different devices

* Web pages working according to user’s device configurations and screen size
* <html>

<head>

    <style>

        .responsive {

            width: 100%;

            height: auto;

        }

    </style>

</head>

<body>

    <h2>Responsive Webpage</h2>

    <img class="responsive" src="Sample.jpg">

</body>

</html>

Final project - Webpage for “Prime Computers”

* HTML file
* <!DOCTYPE html>

<html>

<head>

    <title>Prime Computers Official Website</title>

    <h2><center>Prime Computers</center></h2>

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

</head>

<body>

    <div id="container">

        <h6>Dark Mode</h6>

        <label class="switch">

            <input type="checkbox" onclick="changeMode()">

            <span class="slider"></span>

        </label>

    </div>

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

    <img class="responsive1" src="main.jpg">

    <br>

    <h4>Categories</h4>

    <ul>

        <li>Laptops</li>

        <li>Monitors</li>

        <li>Keyboards</li>

        <li>Mouse</li>

        <li>Speakers</li>

        <li>Headsets</li>

        <li>Graphics Cards</li>

        <li>RAM</li>

        <li>Processors</li>

        <li>Power Supply</li>

        <li>Motherboards</li>

        <li>Storage Drives</li>

    </ul>

    <br>

    <h4 class="h4">Special Offers of the Week</h4><br>

    <table>

        <tr>

            <th>Item</th>

            <th>Preview</th>

            <th>Description</th>

            <th>Price</th>

        </tr>

        <tr>

            <td>MSI Katana GF66 Laptop</td>

            <td><img class="responsive2" src="msi\_katana.png"></td>

            <td><h5>Intel Core i7 CPU, 16GB DDR4 RAM, 512GB SSD, Nvidia GTX 1650 4GB, 15.6" 144Hz Screen</h5></td>

            <td><b>369,000LKR</b></td>

        </tr>

        <tr>

            <td>Nvidia RTX 3060Ti 8GB Graphics Card</td>

            <td><img class="responsive2" src="rtx\_3060ti.png"></td>

            <td>Minimum Power Req: 650W, Base Clock: 1410MHz, Real-Time Ray Tracing</td>

            <td><b>255,000LKR</b></td>

        </tr>

        <tr>

            <td>Asus ROG Swift PG259QNR 360Hz Monitor</td>

            <td><img class="responsive2" src="asus\_monitor.png"></td>

            <td>24.5" IPS Display, 1920x1080 True Resolution, 360Hz, 1ms Response Time</td>

            <td><b>235,000LKR</b></td>

        </tr>

        <tr>

            <td>Intel Core i7 12700K Processor</td>

            <td><img class="responsive2" src="core\_i7\_cpu.png"></td>

            <td>5GHz Turbo Frequency, 12 Cores, DirectX 12 Support, Max Memory Channels: 2</td>

            <td><b>165,000LKR</b></td>

        </tr>

    </table>

    <br>

    <h4>Follow Us on Socials,</h4>

    <a href="https://www.linkedin.com/in/dilshvn/">Facebook</a>

    <a href="https://www.linkedin.com/in/dilshvn/">Twitter</a>

    <a href="https://www.linkedin.com/in/dilshvn/">Instagram</a>

    <a href="https://www.linkedin.com/in/dilshvn/">YouTube</a>

    <a href="https://www.linkedin.com/in/dilshvn/">LinkedIn</a>

</body>

</html>

* CSS file
* \* {

    padding: 0px;

    margin: 20px;

}

.responsive1 {

    width: 100%;

    height: auto;

}

.responsive2 {

    width: 50%;

    height: auto;

}

.h4 {

    animation-duration: 12s;

    animation-name: slidein;

    animation-iteration-count: infinite;

}

@keyframes slidein {

    0% {

        margin-left: 0%;

    }

    50% {

        margin-left: 300px;

    }

    100% {

        margin-left: 0%;

    }

}

@import url('https://fonts.googleapis.com/css2?family=Roboto:wght@400&display=swap');

body {

    margin: 0;

    padding: 0;

    font-family: 'Roboto', sans-serif;

    background: #f7f0e9;

    transition: .6s;

}

h1 {

    font-size: 5em;

    color: #3d3935;

}

#container {

    width: 98%;

    height: 3vh;

    display: flex;

    flex-direction: row;

    justify-content: left;

    align-items: right;

}

.switch {

    position: relative;

    display: inline-block;

    width: 10px;

    height: 34px;

}

.switch input {

    display: none;

}

.slider {

    position: absolute;

    cursor: pointer;

    top: 0;

    left: 0;

    right: 0;

    bottom: 0;

    background: #ccc;

    border-radius: 34px;

    transition: .4s;

}

.slider:before {

    content: "";

    position: absolute;

    height: 20px;

    width: 20px;

    left: 0px;

    bottom: 0px;

    background: rgb(5, 160, 114);

    border-radius: 50%;

    transition: .4s;

}

input:checked+.slider {

    background: #b892ff;

}

input:checked+.slider:before {

    transform: translateX(20px);

}

.dark-mode {

    background: #000000;

    color: white;

}

.dark-mode h1 {

    color: #f7f0e9;

}

* JS file
* function changeMode() {

    var element = document.body;

    element.classList.toggle("dark-mode");

}