**Q.1 Write a JavaScript to demonstrate properties and methods of String reference Type.**

*/\* PRACTICAL-1: Write a JavaScript to demonstrate properties and methods of String reference Type.*

*Name: Angat Shah*

*Enrollment No: 202203103510097*

*Branch: B.Tech Computer Science and Engineering \*/*

*// CODE:*

let string = "Hi, Tony Mikaelson!";

console.log("--> Length:", string.length);

console.log("--> Uppercase:", string.toUpperCase());

console.log("--> Lowercase:", string.toLowerCase());

console.log("--> Character at Index 7:", string.charAt(7));

console.log("--> Index of 'Tony':", string.indexOf("Tony"));

console.log("--> Substring from Index 7-15:", string.substring(7, 15));

let newString = string.replace("Mikaelson", "Stark");

console.log("--> After replacing 'Mikaelson' with 'Stark':", newString);

let words = string.split(",");

console.log("--> Split into words:", words);

let anotherString = " Welcome!";

let combinedString = string.concat(anotherString);

console.log("--> Combined string:", combinedString);

let spacedString = " Pepper ";

console.log("--> Original string:", spacedString);

console.log("--> Trimmed string:", spacedString.trim());

console.log("\n-\*-\*-\*-\*-\*-END OF PRACTICAL 1-\*-\*-\*-\*-\*-")

**Q.2 Write a JavaScript to demonstrate properties and methods of Array Collection Reference Type.**

*/\* PRACTICAL-2: Write a JavaScript to demonstrate properties and methods of Array Collection Reference Type.*

*Name: Angat Shah*

*Enrollment No: 202203103510097*

*Branch: B.Tech Computer Science and Engineering \*/*

*// CODE:*

let array = [1, 5, 7, 11, 30];

console.log("--> Length:", array.length);

console.log("--> First Element:", array[0]);

array.push(8);

console.log("--> Array after Pushing Element-8:", array);

array.unshift(0);

console.log("--> Array after Unshifting 0:", array);

let poppedElement = array.pop();

console.log("--> Popped Element:", poppedElement);

console.log("--> Array after Popping:", array);

let shiftedElement = array.shift();

console.log("--> Shifted Element:", shiftedElement);

console.log("--> Array after Shifting:", array);

console.log("--> Index of 5:", array.indexOf(5));

let slicedArray = array.slice(1, 3);

console.log("--> Sliced Array:", slicedArray);

let anotherArray = [-1, -2, -3];

let combinedArray = array.concat(anotherArray);

console.log("--> Combined array:", combinedArray);

let joinedString = array.join(", ");

console.log("--> Joined string:", joinedString);

let unsortedArray = [5, 2, 8, 1, 9];

console.log("--> Unsorted Array:", unsortedArray);

let sortedArray = unsortedArray.sort();

console.log("--> Sorted array:", sortedArray);

let reversedArray = array.reverse();

console.log("--> Reversed array:", reversedArray);

console.log("\n-\*-\*-\*-\*-\*-END OF PRACTICAL 2-\*-\*-\*-\*-\*-")

**Q.3 Write a JavaScript to demonstrate properties and methods of Date reference Type.**

*/\* PRACTICAL-3: Write a JavaScript to demonstrate properties and methods of Date reference Type.*

*Name: Angat Shah*

*Enrollment No: 202203103510097*

*Branch: B.Tech Computer Science and Engineering \*/*

*// CODE:*

let currentDate = new Date();

let year = currentDate.getFullYear();

let month = currentDate.getMonth();

let day = currentDate.getDate();

let hours = currentDate.getHours();

let minutes = currentDate.getMinutes();

let seconds = currentDate.getSeconds();

console.log(`Current Date --> ${day}/${month + 1}/${year}`);

console.log(`Current Time --> ${hours}:${minutes}:${seconds}`);

let futureDate = new Date();

futureDate.setDate(currentDate.getDate() + 11);

console.log(`Date after 11 days --> ${futureDate.getDate()}/${futureDate.getMonth() + 1}/${futureDate.getFullYear()}`);

console.log("\n-\*-\*-\*-\*-\*-END OF PRACTICAL 3-\*-\*-\*-\*-\*-")

**Q.4 Write a JavaScript to create a cart contains items of categories like groceries, apparels, accessories and gadgets. Each category offers discounts 10%, 20%, 5% and 50% discounts. Apply discount on the times present in the cart and generate a Final bill. [Hint: Use objects containing key-value pair]**

*/\* PRACTICAL-4: Write a JavaScript to create a cart contains items of categories like groceries apparels, accessories and gadgets. Each category offersdiscounts 10%, 20%, 5% and 50% discounts Apply discount on the times present in the cart and generate a Uinal bill. [Hint: Use objects containing key-value pair]*

*Name: Angat Shah*

*Enrollment No: 202203103510097*

*Branch: B.Tech Computer Science and Engineering \*/*

*// CODE:*

const cart = [

{ item: 'Croissant', category: 'Groceries', price: 5 },

{ item: 'LV Cotton Shirt', category: 'Apparels', price: 1250 },

{ item: 'Apple Watch Ultra 2', category: 'Accessories', price: 799 },

{ item: 'Iphone 15 Pro Max', category: 'Gadgets', price: 1199 },

];

const categoryDiscounts = {

Groceries: 10,

Apparels: 20,

Accessories: 5,

Gadgets: 50,

};

let total = 0;

cart.forEach(item => {

const category = item.category;

const discountPercentage = categoryDiscounts[category] || 0;

const discountAmount = (discountPercentage / 100) \* item.price;

const discountedPrice = item.price - discountAmount;

total += discountedPrice;

console.log(`[${category}] \n${item.item} --> $${item.price.toFixed(2)} (${discountPercentage}% off) = $${discountedPrice.toFixed(2)}`);

});

console.log(`\nTotal Bill --> $${total.toFixed(2)}`);

console.log("\n-\*-\*-\*-\*-\*-END OF PRACTICAL 4-\*-\*-\*-\*-\*-")

**Q.5 Create a regular expression to find pattern from the given text as follows: 1) All the words starting with 'A'. 2) All the words starting with consonants. [Bonus Program: Upload a file using PHP and find above mentioned patterns using JavaScript RE only from the text present in the file. ]**

*/\* PRACTICAL-5: Create a regular expression to find pattern from the given text as follows: 1) All the words starting with 'A'. 2) All the words starting with consonants. [Bonus Program: Upload a file using PHP and find above mentioned patterns using JavaScript RE only from the text present in the file. ]*

*Name: Angat Shah*

*Enrollment No: 202203103510097*

*Branch: B.Tech Computer Science and Engineering \*/*

*// CODE:*

var text = "He is not here to compete, He is here to rule And He is Ruling";

var matches = text.match(/\bA\w\*/g);

console.log("Words Starting with 'A' --> ",matches);

var matches = text.match(/\b[b-df-hj-np-tv-zB-DF-HJ-NP-TV-Z][a-z\d.+-]\*/g);

console.log("Words Starting with Consonants --> ",matches);

console.log("\n-\*-\*-\*-\*-\*-END OF PRACTICAL 5-\*-\*-\*-\*-\*-")

**Q.6 Take input from user through prompt box and store it in an array. Print the Fibonacci series till the indexed number in the array. [Hint: Use Iterative Methods of array]**

*<!-- PRACTICAL-6: Take input from user through prompt box and store it in an array. Print the Fibonacci series till the indexed number in the array.[Hint: Use Iterative Methods of array]*

*Name: Angat Shah*

*Enrollment No: 202203103510097*

*Branch: B.Tech Computer Science and Engineering -->*

*<!-- CODE -->*

*<!DOCTYPE html>*

*<html lang="en">*

*<head>*

*<meta charset="UTF-8">*

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

*<title>PRACTICAL6</title>*

*</head>*

*<body>*

*<script>*

*const userInput = prompt('Enter the Index to Generate Fibonacci Series: ');*

*const n = parseInt(userInput);*

*if (isNaN(n) || n < 0) {*

*alert('Invalid input. Enter a non-negative number.');*

*} else {*

*const fibonacciArray = [0, 1];*

*for (let i = 2; i <= n; i++) {*

*fibonacciArray[i] = fibonacciArray[i - 1] + fibonacciArray[i - 2];*

*}*

*document.write(`Fibonacci series up to index ${n} --> `, fibonacciArray.slice(0, n + 1));*

*}*

*</script>*

*</body>*

*</html>*

*<!-- -\*-\*-\*-\*-\*-END OF PRACTICAL 6-\*-\*-\*-\*-\*- -->*

**Q.7 Take input from user through prompt box and store it in an array. Calculate the Factorial of each number present in the array using iterative method of array and generate the resultant array of factorial values. [Bonus Program: Create the same program combining the iterative methods concept and recursion.]**

*<!-- PRACTICAL-7: Take input from user through prompt box and store it in an array. Calculate the Factorial of each number present in the array using iterative method of array and generate the resultant array of factorial values. [Bonus Program: Create the same program combining the iterative methods concept and recursion.]*

*Name: Angat Shah*

*Enrollment No: 202203103510097*

*Branch: B.Tech Computer Science and Engineering -->*

*<!-- CODE -->*

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>PRACTICAL7</title>

</head>

<body>

<div id="display"></div>

<script>

function calculateFactorialIterative(number) {

let factorial = 1;

for (let i = 1; i <= number; i++) {

factorial \*= i;

}

return factorial;

}

function calculateFactorialRecursive(number) {

if (number <= 1) {

return 1;

}

return number \* calculateFactorialRecursive(number - 1);

}

const userInput = prompt("Enter numbers separated by commas: ");

const inputArray = userInput.split(',').map(Number);

const factorialArrayIterative = inputArray.map(number => calculateFactorialIterative(number));

const factorialArrayRecursive = inputArray.map(number => calculateFactorialRecursive(number));

const display = document.getElementById("display");

display.innerHTML += `<p>Inputed Array --> [${inputArray.join(', ')}]</p>`;

display.innerHTML += `<p>Factorial Array By Iterative Method --> [${factorialArrayIterative.join(', ')}]</p>`;

display.innerHTML += `<p>Factorial Array By Recursive Method --> [${factorialArrayRecursive.join(', ')}]</p>`;

</script>

</body>

</html>

*<!-- -\*-\*-\*-\*-\*-END OF PRACTICAL 7-\*-\*-\*-\*-\*- -->*

**Q.8 a) Write a demonstrative JavaScript for i) Function methods, ii) Function internals**

**b) Write a JavaScript to implement function overloading using function internals.**

*/\* PRACTICAL-8: a) Write a demonstrative JavaScript for i) Function methods, ii) Function internals*

*b) Write a JavaScript to implement function overloading using function internals.*

*Name: Angat Shah*

*Enrollment No: 202203103510097*

*Branch: B.Tech Computer Science and Engineering \*/*

*// CODE:*

*// A(I). Function Methods*

function greet(name) {

console.log("--> Hello, " + name + "!");

}

greet("Tony Mikaelson");

*// A(II). Function Internals*

function add(a, b) {

return a + b;

}

console.log("# Function Internals \n" + add.toString());

console.log("--> Number of arguments expected by 'add' function: " + add.length);

*// B. Function Overloading*

function calculate() {

if (*arguments*.length === 2) {

return *arguments*[0] + *arguments*[1];

} else if (*arguments*.length === 3) {

return *arguments*[0] \* *arguments*[1] \* *arguments*[2];

} else {

throw new Error("--> Invalid Number of Arguments");

}

}

console.log("\n# Function Overloading")

console.log("--> Addition when Two Arguments are passed:" + calculate(2, 3));

console.log("--> Multiplication when Three Arguments are passed:" + calculate(2, 3, 4));

*//console.log(calculate(2, 3, 4, 5));*

console.log("\n-\*-\*-\*-\*-\*-END OF PRACTICAL 8-\*-\*-\*-\*-\*-")

**Q.9 Write a demonstrative JavaScript for i) Map, weka map and Set, ii) Function Closure and iii) Typed array with ArrayBuffers.**

*/\* PRACTICAL-9: Write a demonstrative JavaScript for i) Map, weka map and Set, ii) Function Closure and iii) Typed array with ArrayBuffers.*

*Name: Jenil Bhikadiya*

*Enrollment No: 202203103510005*

*Branch: B.Tech Computer Engineering \*/*

*// CODE:*

*// I. Map, WeakMap, and Set*

*// Map*

let map = new Map();

map.set('Tony', 'Mikaelson');

map.set('Pepper', 'Pots');

console.log("#Map");

console.log("--> The Value of 'Tony':"+ map.get('Tony'));

console.log("--> Map Size:" + map.size);

*// WeakMap*

let weakMap = new WeakMap();

let obj1 = {};

let obj2 = {};

weakMap.set(obj1, 'Gucci');

weakMap.set(obj2, 'Dior');

console.log("\n#WeakMap");

console.log("--> 'obj1' in WeakMap:" +weakMap.get(obj1));

console.log("--> WeakMap has 'obj2':" + weakMap.has(obj2));

*// Set*

let set = new Set();

set.add('A');

set.add('B');

set.add('C');

console.log("\n#Set");

console.log("--> Size of Set:"+set.size);

console.log("--> Set has 'A':" +set.has('A'));

*// II. Function Closure*

function outerFunction() {

let outerVariable = '--> This is the Outer Variable';

function innerFunction() {

console.log(outerVariable);

}

return innerFunction;

}

let closure = outerFunction();

console.log("\n#Function Closure");

closure();

*// III. Typed array with ArrayBuffers*

let buffer = new ArrayBuffer(16);

let view = new Int32Array(buffer);

view[0] = 30;

view[1] = 11;

view[2] = -47;

console.log("\n#Typed Array with ArrayBuffers");

console.log(view);

console.log("\n-\*-\*-\*-\*-\*-END OF PRACTICAL 9-\*-\*-\*-\*-\*-")

**Q.10 a) Write a JavaScript to list the properties of an object using iterator and generator.**

**b) Write a JavaScript to delete the name property and change the class property value from the following object. Also print the object before and after the said operations. [Hint: Use data properties of a JavaScript object.]**

**c) Write a JavaScript to update the rollno property value based on the class value. Perform this using accessor property of a JavaScript object. [Hint: for class: VI, the rollno is 12. If the class: V, the rollno should be 13.]**

**Sample object:**

**let student = { name : "David Rayy", class : "VI", rollno : 12 };**

*/\* PRACTICAL-10: a) Write a JavaScript to list the properties of an object using iterator and generator.*

*b) Write a JavaScript to delete the name property and change the class property value from the following object. Also print the object before and after the said operations. [Hint: Use data properties of a JavaScript object.]*

*c) Write a JavaScript to update the rollno property value based on the class value. Perform this using accessor property of a JavaScript object. [Hint: for class: VI, the rollno is 12. If the class: V, the rollno should be 13.]*

*Sample object:*

*let student = { name : "David Rayy", class : "VI", rollno : 12 };*

*Name: Angat Shah*

*Enrollment No: 202203103510097*

*Branch: B.Tech Computer Science and Engineering \*/*

*// CODE:*

*// I. Iterator & Generator*

let obj = {

Key1: '1',

Key2: '2',

Key3: '3'

};

console.log("--> Listing Properties Using Iterator:");

let iterator = Object.keys(obj)[Symbol.iterator]();

for (let key of iterator) {

console.log(key);

}

console.log("\n--> Listing Properties Using Generator:");

function\* generateKeys(object) {

for (let key in object) {

yield key;

}

}

let generator = generateKeys(obj);

let next = generator.next();

while (!next.done) {

console.log(next.value);

next = generator.next();

}

*// II. Delete Name Property*

let student = { name: "David Rayy", sclass: "VI", rollno: 12 };

console.log("\n--> Before Deletion:");

console.log(student);

delete student.name;

student.sclass = "VII";

console.log("--> After Deletion:");

console.log(student);

*// III. Accessor Property*

let student1 = {

\_name: "David Rayy",

\_sclass: "VI",

get rollno() {

return *this*.\_sclass === "VI" ? 12 : 13;

}

};

console.log("\n--> Before Update:");

console.log(student1);

console.log("Roll No:", student1.rollno);

student.\_sclass = "V";

console.log("--> After Update:");

console.log(student1);

console.log("Roll No:", student1.rollno);

console.log("\n-\*-\*-\*-\*-\*-END OF PRACTICAL 10-\*-\*-\*-\*-\*-")

**Q.11 a) Write a JavaScript to get the volume of a Cylinder with four decimal places using object classes. [Hint: Volume of a cylinder : V = πr2h, where r is the radius and h is the height of the cylinder.]**

**b) Write a Bubble Sort algorithm in JavaScript using OOP concept.**

**[Bonus: Create a php page which take names of students. List the name using the algorithm developed in this practical.]**

*/\* PRACTICAL-11: a) Write a JavaScript to get the volume of a Cylinder with four decimal places using object classes. [Hint: Volume of a cylinder : V = πr2h, where r is the radius and h is the height of the cylinder.]*

*b) Write a Bubble Sort algorithm in JavaScript using OOP concept.*

*[Bonus: Create a php page which take names of students. List the name using the algorithm developed in this practical.]*

*Name: Angat Shah*

*Enrollment No: 202203103510097*

*Branch: B.Tech Computer Science and Engineering \*/*

*// CODE:*

*// I. Volume of Cylinder*

class Cylinder {

constructor(radius, height) {

*this*.radius = radius;

*this*.height = height;

}

getVolume() {

let volume = Math.PI \* Math.pow(*this*.radius, 2) \* *this*.height;

return volume.toFixed(4);

}

}

let cylinder = new Cylinder(11,30);

console.log("--> Volume of Cylinder:", cylinder.getVolume());

*// II. Bubble Sort*

class BubbleSort {

static sort(arr) {

let n = arr.length;

for (let i=0; i<n-1; i++) {

for (let j=0; j<n-i-1; j++) {

if (arr[j] > arr[j+1]) {

let temp = arr[j];

arr[j] = arr[j+1];

arr[j+1] = temp;

}

}

}

return arr;

}

}

let unsortedArray = [79, 8, 1, 11, 4, 5, 30, 4, 7];

console.log("--> Unsorted Array:", unsortedArray);

let sortedArray = BubbleSort.sort(unsortedArray);

console.log("--> Sorted Array:", sortedArray);

console.log("\n-\*-\*-\*-\*-\*-END OF PRACTICAL 11-\*-\*-\*-\*-\*-")

**Q.12 Write a JavaScript to create a tip calculator that help determine how much to tip at restaurants or whenever the need arises.**

*<!-- PRACTICAL-12: Write a JavaScript to create a tip calculator that help determine how much to tip at restaurants or whenever the need arises.*

*Name: Angat Shah*

*Enrollment No: 202203103510097*

*Branch: B.Tech Computer Science and Engineering -->*

*<!-- CODE -->*

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>PRACTICAL12</title>

<style>

body {

font-family: 'Trebuchet MS', 'Lucida Sans Unicode', 'Lucida Grande', 'Lucida Sans', Arial, sans-serif;

}

.calculator {

max-width: 300px;

margin: 50px auto;

padding: 20px;

border: 1px solid #ccc;

border-radius: 5px;

background-color: #f9f9f9;

}

input[type="number"] {

width: 100%;

padding: 8px;

margin: 5px 0;

box-sizing: border-box;

}

input[type="button"] {

width: 100%;

padding: 10px;

margin-top: 10px;

background-color: #5E4DE0;

color: white;

border: none;

border-radius: 20px;

cursor: pointer;

}

</style>

</head>

<body>

<div class="calculator">

<h2>TIP CALCULATOR</h2>

<label for="billAmount">BILL AMOUNT ($):</label><br>

<input type="number" id="billAmount" placeholder="ENTER BILL AMOUNT"><br><br>

<label for="tipPercentage">TIP PERCENTAGE (%):</label>

<input type="number" id="tipPercentage" placeholder="ENTER TIP PERCENTAGE"><br><br>

<input type="button" value="Calculate Tip" onclick="calculateTip()"><br><br>

<div id="tipAmount"></div>

<div id="totalAmount"></div>

</div>

<script>

function calculateTip() {

var billAmount = parseFloat(document.getElementById('billAmount').value);

var tipPercentage = parseFloat(document.getElementById('tipPercentage').value);

var tipAmount = (billAmount \* (tipPercentage / 100)).toFixed(2);

var totalAmount = (billAmount + parseFloat(tipAmount)).toFixed(2);

document.getElementById('tipAmount').innerHTML = "TIP AMOUNT: $" + tipAmount;

document.getElementById('totalAmount').innerHTML = "TOTAL AMOUNT: $" + totalAmount;

}

</script>

</body>

</html>

**Q.13 Write a JavaScript to create election map. The code should have two candidates running for president and display the number of votes they received from each state, both in a table on the bottom right and when you hover mouse over a state. [Hint: BOM, DOM, Events] [Bonus: Created map should change the colour and pop the name of the state.]**

*Name: Angat Shah*

*Enrollment No: 202203103510097*

*Branch: B.Tech Computer Science and Engineering -->*

*<!-- CODE -->*

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>PRACTICAL-13 (ELECTION MAP)</title>

<style>

#map {

width: 600px;

height: 400px;

position: relative;

border: 1px solid #ccc;

}

#map svg {

width: 100%;

height: 100%;

}

#info {

position: absolute;

bottom: 10px;

right: 10px;

background-color: white;

padding: 10px;

border: 1px solid #ccc;

}

</style>

</head>

<body>

<div id="map"></div>

<div id="info">

<h3>Election Results</h3>

<table id="results">

<tr>

<th>State</th>

<th>Candidate 1</th>

<th>Candidate 2</th>

</tr>

</table>

</div>

<script src="election-map.js">

const electionData = {

"California": { candidate1: 3000000, candidate2: 2500000 },

"Texas": { candidate1: 2800000, candidate2: 3200000 },

"New York": { candidate1: 2000000, candidate2: 1800000 },

};

const mapContainer = document.getElementById('map');

const infoDiv = document.getElementById('info');

const resultsTable = document.getElementById('results');

const electionMap = document.createElementNS("http://www.w3.org/2000/svg", "svg");

electionMap.setAttribute('width', '100%');

electionMap.setAttribute('height', '100%');

mapContainer.appendChild(electionMap);

electionMap.addEventListener('mouseover', (event) => {

if (event.target.tagName === 'path') {

const stateName = event.target.getAttribute('data-state');

const stateData = electionData[stateName];

if (stateData) {

const html = `<h3>${stateName}</h3>

<p>Candidate 1: ${stateData.candidate1}</p>

<p>Candidate 2: ${stateData.candidate2}</p>`;

infoDiv.innerHTML = html;

}

}

});

for (const stateName in electionData) {

const statePath = document.createElementNS("http://www.w3.org/2000/svg", "path");

statePath.setAttribute('d', getStatePath(stateName));

statePath.setAttribute('data-state', stateName);

statePath.setAttribute('fill', '#ccc');

statePath.setAttribute('stroke', '#333');

electionMap.appendChild(statePath);

}

function getStatePath(stateName) {

const paths = {

"California": "M 10,30 L 40,30 L 40,70 L 10,70 Z",

"Texas": "M 50,50 L 80,50 L 80,90 L 50,90 Z",

"New York": "M 90,10 L 120,10 L 120,50 L 90,50 Z",

};

return paths[stateName];

}

for (const stateName in electionData) {

const stateData = electionData[stateName];

const row = resultsTable.insertRow();

row.insertCell(0).textContent = stateName;

row.insertCell(1).textContent = stateData.candidate1;

row.insertCell(2).textContent = stateData.candidate2;

}

</script>

</body>

</html>

**Q.14 Write a JavaScript to perform form validation using regular expression.**

*<!-- PRACTICAL-14: Write a JavaScript to perform form validation using regular expression.*

*Name: Angat Shah*

*Enrollment No: 202203103510097*

*Branch: B.Tech Computer Science and Engineering -->*

*<!-- CODE -->*

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>PRACTICAL14</title>

<style>

.error {

color: red;

}

body {

font-family: 'Trebuchet MS', 'Lucida Sans Unicode', 'Lucida Grande', 'Lucida Sans', Arial, sans-serif;

}

#myForm {

max-width: 300px;

margin: 50px auto;

padding: 20px;

border: 1px solid #ccc;

border-radius: 5px;

background-color: #f9f9f9;

}

input[type="text"] {

width: 100%;

padding: 8px;

margin: 5px 0;

box-sizing: border-box;

}

input[type="password"] {

width: 100%;

padding: 8px;

box-sizing: border-box;

margin: 5px 0;

}

button {

width: 100%;

padding: 10px;

margin-top: 10px;

background-color: #5E4DE0;

color: white;

border: none;

border-radius: 20px;

cursor: pointer;

}

</style>

</head>

<body>

<form id="myForm">

<h2>VALIDATION</h2>

<label for="email">Email:</label><br>

<input type="text" id="email" name="email" placeholder="ENTER YOUR EMAIL-ID"><br>

<span id="emailError" class="error"></span><br><br>

<label for="password">Password:</label><br>

<input type="password" id="password" name="password" placeholder="ENTER YOUR PASSWORD"><br>

<span id="passwordError" class="error"></span><br><br>

<button type="submit">Submit</button>

</form>

<script>

document.getElementById("myForm").addEventListener("submit", function(event) {

event.preventDefault();

var email = document.getElementById("email").value;

var password = document.getElementById("password").value;

var emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;

var passwordRegex = /^(?=.\*\d)(?=.\*[a-z])(?=.\*[A-Z]).{8,}$/;

if (!emailRegex.test(email)) {

document.getElementById("emailError").textContent = "Invalid email address";

return;

} else {

document.getElementById("emailError").textContent = "";

}

if (!passwordRegex.test(password)) {

document.getElementById("passwordError").textContent = "Password must be at least 8 characters long and contain at least one digit, one lowercase letter, and one uppercase letter";

return;

} else {

document.getElementById("passwordError").textContent = "";

}

alert("FORM SUBMIT SUCCESSFULLY!");

});

</script>

</body>

</html>

**Q.15 Write a demonstrative JavaScript for JavaScript APIs.**

*<!-- PRACTICAL-15: Write a demonstrative JavaScript for JavaScript APIs.*

*Name: Angat Shah*

*Enrollment No: 202203103510097*

*Branch: B.Tech Computer Science and Engineering -->*

*<!-- CODE -->*

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>PRACTICAL15</title>

</head>

<body><center>

<div>

<button onclick="getLocation()">GET LOCATION</button>

<p id="location"></p>

<input type="text" id="textInput" placeholder="Enter some text">

<button onclick="saveText()">SAVE TEXT</button>

<button onclick="loadText()">LOAD TEXT</button>

<p id="savedText"></p>

</div>

<script>

function getLocation() {

if (navigator.geolocation) {

navigator.geolocation.getCurrentPosition(showPosition);

} else {

document.getElementById("location").innerHTML = "Geolocation is not supported by this browser.";

}

}

function showPosition(position) {

var latitude = position.coords.latitude;

var longitude = position.coords.longitude;

document.getElementById("location").innerHTML = "Latitude: " + latitude + "<br>Longitude: " + longitude;

}

function saveText() {

var text = document.getElementById("textInput").value;

localStorage.setItem("savedText", text);

document.getElementById("textInput").value = "";

document.getElementById("savedText").innerHTML = "Text saved to Local Storage.";

}

function loadText() {

var text = localStorage.getItem("savedText");

if (text) {

document.getElementById("savedText").innerHTML = "Loaded Text: " + text;

} else {

document.getElementById("savedText").innerHTML = "No text saved in Local Storage.";

}

}

</script></center>

</body>

</html>

**Q.16 Write a React code to build a simple search filter functionality to display a filtered list based on the search query entered by the user.**

*Name: Angat Shah*

*Enrollment No: 202203103510097*

*Branch: B.Tech Computer Science and Engineering \*/*

*// CODE:*

import React, { useState } from 'react';

import './styles.css';

const SearchFilter = ({ data }) => {

const [searchQuery, setSearchQuery] = useState('');

const handleChange = (e) => {

setSearchQuery(e.target.value);

};

const filteredData = data.filter(item =>

item.toLowerCase().includes(searchQuery.toLowerCase())

);

return (

<div className="search-box">

<input

type="text"

placeholder="Search..."

value={searchQuery}

onChange={handleChange}

/>

<ul>

{filteredData.map((item, index) => (

<li key={index}>{item}</li>

))}

</ul>

</div>

);

};

const App = () => {

const data = ["Apple", "Banana", "Orange", "Grapes", "Pineapple"];

return (

<div>

<h1>PRACTICAL-16 (Search Filter)</h1>

<SearchFilter data={data} />

</div>

);

};

export default App;

**Q.17 Write a react code to create a simple counter which increments or decrements count dynamically on-screen as the user clicks on the button.**

*Name: Angat Shah*

*Enrollment No: 202203103510097*

*Branch: B.Tech Computer Science andEngineering \*/*

*// CODE:*

import React, { useState } from 'react';

import './counter.css';

const Counter = () => {

const [count, setCount] = useState(0);

const increment = () => {

setCount(prevCount => prevCount + 1);

};

const decrement = () => {

setCount(prevCount => prevCount - 1);

};

return (

<div className="counter-container">

<h2 className="counter-title">PRACTICAL-17 (COUNTER)</h2>

<div className="counter-box">

<button onClick={decrement} className="counter-button">-</button>

<span className="counter-value">{count}</span>

<button onClick={increment} className="counter-button">+</button>

</div>

</div>

);

};

export default Counter;

**Q.18 Write a react code to create an accordion that toggles text content on click of the accordion header.**

*Name: Angat Shah*

*Enrollment No: 202203103510097*

*Branch: B.Tech Computer Science andEngineering \*/*

*// CODE:*

import React, { useState } from 'react';

import './Accordion.css';

function AccordionItem({ title, content }) {

const [isOpen, setIsOpen] = useState(false);

const toggleAccordion = () => {

setIsOpen(!isOpen);

};

return (

<div className="accordion-item">

<div className="accordion-header" onClick={toggleAccordion}>

<div>{title}</div>

<div>{isOpen ? '-' : '+'}</div>

</div>

{isOpen && <div className="accordion-content">{content}</div>}

</div>

);

}

function App() {

return (

<div className="App">

<h1>PRACTICAL-18 (ACCORDION)</h1>

<AccordionItem

title="Section 1"

content="Content for Section 1 Lorem ipsum dolor sit amet, consectetur adipiscing elit."

/>

<AccordionItem

title="Section 2"

content="Content for Section 2 Sed do eiusmod tempor incididunt ut labore et dolore magna aliqua."

/>

<AccordionItem

title="Section 3"

content="Content for Section 3 Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat."

/>

</div>

);

}

export default App;

**Q.19 Write a react code for simple login form where the user login by entering their username and password. The form inputs are validated to check if correct information is entered and the error messages are if the validation fails. The login form is hidden and the “Welcome, ${name}” message is shown when the user login is successful.**

*Name: Angat Shah*

*Enrollment No: 202203103510097*

*Branch: B.Tech Computer Science andEngineering \*/*

*// CODE:*

import React, { useState } from 'react';

import './Form.css';

const App = () => {

const [username, setUsername] = useState('');

const [password, setPassword] = useState('');

const [error, setError] = useState('');

const [isLoggedIn, setIsLoggedIn] = useState(false);

const handleSubmit = (e) => {

e.preventDefault();

if (username === 'user' && password === 'password') {

setIsLoggedIn(true);

} else {

setError('Incorrect username or password'); }

};

const handleLogout = () => {

setIsLoggedIn(false);

setUsername('');

setPassword('');

setError(''); };

return (

<div className="container">

{!isLoggedIn ? (

<form onSubmit={handleSubmit} className="login-form">

<h2>PRACTICAL-19 (LOGIN FORM)</h2>

<input

type="text"

placeholder="Username"

value={username}

onChange={(e) => setUsername(e.target.value)} />

<input

type="password"

placeholder="Password"

value={password}

onChange={(e) => setPassword(e.target.value)} />

{error && <p className="error-message">{error}</p>}

<button type="submit">Login</button>

</form>

) : (

<div className="welcome">

<h2>Welcome, {username}!</h2>

<button onClick={handleLogout}>Logout</button>

</div>

)}

</div>

);

};

export default App;

**Q.20 Write a react code to display a checklist with multiple options that can select and the selected options are dynamically displayed on the screen.**

*Name: Angat Shah*

*Enrollment No: 202203103510097*

*Branch: B.Tech Computer Science andEngineering \*/*

*// CODE:*

import React, { useState } from 'react';

import './Option.css';

function App() {

const options = ["Option 1", "Option 2", "Option 3", "Option 4"];

const [selectedOptions, setSelectedOptions] = useState([]);

const handleCheckboxChange = (option) => {

if (selectedOptions.includes(option)) {

setSelectedOptions(selectedOptions.filter(item => item !== option));

} else {

setSelectedOptions([...selectedOptions, option]);

}

};

return (

<div className="App">

<h1>PRACTICAL-20 (CHECKLIST)</h1>

<div className="checklist-container">

{options.map((option, index) => (

<label key={index} className="checkbox-label">

<input

type="checkbox"

checked={selectedOptions.includes(option)}

onChange={() => handleCheckboxChange(option)}

/>

{option}

</label>

))}

</div>

<div className="selected-options">

<h2>Selected Options:</h2>

<ul>

{selectedOptions.map((option, index) => (

<li key={index}>{option}</li>

))}

</ul>

</div>

</div>

);

}

export default App;