Evening Assessment

21. Use for loop to print each item in an array.

<script>

const subjects = ["Math", "Science", "History", "English"];

for (let i = 0; i < subjects.length; i++) {

console.log(subjects[i]);

}

</script>

22. What does the array.length property return?

array.length property returns the total number of elements in the array.

23. Use forEach to iterate over an array and print all items.

<h2>See Console for Array Items</h2>

<script>

const animals = ["Tiger", "Elephant", "Giraffe"];

animals.forEach(animal => console.log(animal));

</script>

24. How do you check if a variable is an array?

<script>

const marks = [90, 85, 78];

const name = "Abhi";

console.log(Array.isArray(marks)); -> this returns true

console.log(Array.isArray(name)); ->this returns false

</script>

25–30: Arrow Functions

25. Write an arrow function to multiply two numbers.

<h2>Check Console for Answer</h2>

<script>

const multiply = (a, b) => a \* b;

console.log("Product is, multiply(5, 5));

</script>

Ans is 25.

26. Convert a traditional function to an arrow function.

Traditional function:

function greet(name) {

return "Hello " + name;

}

Arrow function:

const greet = name => "Hello " + name;

27. Write an arrow function that returns the square of a number.

<h2>Check Console for Square</h2>

<script>

const square = n => n \* n;

console.log("Square of 5 is:", square(5));

</script>

o/p is 25.

28. Create an arrow function that returns a greeting message.

<script>

const getGreeting = name => `Hello, ${name}!`;

console.log(getGreeting("Abhi"));

</script>

29. Use an arrow function inside map() to double each number in an array.

<script>

const numbers = [1, 2, 3, 4, 5];

const doubled = numbers.map(num => num \* 2);

console.log("Original nums:", numbers);

console.log("Doubled nums:", doubled);

</script>

30. What is the difference in this context between arrow and regular functions?

->Regular functions have their own this.They use this.

->Arrow functions(->) do not have their own this. They inherit this from their parent scope.

31–35: Event Handling

31. How do you add a click event to a button in JavaScript?

<button id="button">Submit</button>

<script>

document.getElementById("button").addEventListener("click", function() {

alert("Button clicked!");

});

</script>

32. Write JavaScript that changes the text of a paragraph when a button is clicked.

<p id="para">Original text</p>

<button onclick="changeText()">Change Text</button>

<script>

function changeText() {

document.getElementById("para").textContent = "Text is changed!";

}

</script>

33. How do you add a mouseover event to an element?

document.getElementById("myBox").addEventListener("mouseover", () => {

console.log("Mouse is over the element");

});

34. Write an event handler that logs the value of an input field when typing.

<input type="text" id="nameInput" placeholder="Enter text" />

<script>

const input = document.getElementById("nameInput");

input.addEventListener("input", () => {

console.log("Current value:", input.value);

});

</script>

35. What’s the purpose of preventDefault() in event handling?

preventDefault() -> it is used to stop the default behavior like form submission or link navigation.

36–40: Forms

36. How do you get the value of an input field using JavaScript?

<input type="text" id="userInput" placeholder="Enter something" />

<button onclick="showValue()">Show Value</button>

<script>

function showValue() {

const value = document.getElementById("userInput").value;

alert("Input value: " + value);

}

</script>

37. Write JavaScript to validate if a form field is not empty.

<input type="text" id="name" placeholder="Enter your name" />

<button onclick="validate()">Submit</button>

<script>

function validate() {

const name = document.getElementById("name").value;

if (name === "") {

alert("Name field cannot be empty!");

} else {

alert("Hello, " + name + "!");

}

}

</script>

38. How do you handle form submission using JavaScript?

document.addEventListener("DOMContentLoaded", function () {

const form = document.getElementById("myForm");

form.addEventListener("submit", function (event) {

event.preventDefault();

const name = document.getElementById("name").value;

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

if (name === "" || email === "") {

alert("Please fill in all fields.");

return;

}

console.log("Form Submitted!");

console.log("Name:", name);

console.log("Email:", email);

form.reset();

});

});

39. How do you stop a form from refreshing the page when submitted?

<form onsubmit="handleSubmit(event)">

<input type="text" placeholder="Enter data" >

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

</form>

<script>

function handleSubmit(event) {

event.preventDefault();

alert("Sorry,Page can’t be refreshed");

}

</script>

40. How can you reset a form using Javascript?

<form id="resetForm">

<input type="text" placeholder="Type your text here" >

<button type="button" onclick="resetForm()">Reset</button>

</form>

<script>

function resetForm() {

document.getElementById("resetForm").reset();

}

</script>