St. Francis Institute of Technology

Mount Poinsur, S.V.P. Road, Borivali (W), Mumbai-103

**Class: TE IT A & B Academic Year: 2023-2024**

**Experiment – 6**

**Menu driven program using arrow functions in JavaScript**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**1. Aim:** To write a menu driven program in JavaScript to demonstrate the use of arrow

functions.

**2. Objective:** After performing the experiment, the students will be able to understand and

implement the basic concepts of JavaScript including use of arrow functions.

**3. Lab objective mapped:** Students will be able to use JavaScript to develop interactive web pages (PO3, PO5, PSO3, PSO4)

**4. Prerequisite:** JavaScript

**5. Requirements:** The following are the requirements **–**

* PC/Laptop
* Visual Studio Code
* Browser

**6. Pre-Experiment Theory:**

Arrow functions allow us to write shorter function syntax.

The handling of this is also different in arrow functions compared to regular functions.

In short, with arrow functions there are no binding of this.

In regular functions the this keyword represented the object that called the function, which could be the window, the document, a button or whatever.

With arrow functions the this keyword always represents the object that defined the arrow function.

With a regular function this represents the object that calls the function

With an arrow function this represents the owner of the function

**7. Laboratory Exercise:**

**A. Procedure**

Open Visual Studio Code

Select File, New to write to a new file

Save the file as .js

Write the code

To view the output, right-click on the file and select Open With option. Then choose any

web browser that is available or check output on console.

Check output

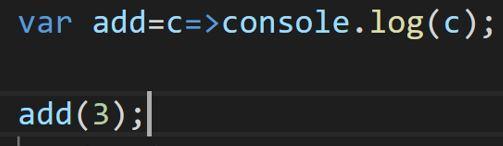
**B. Program Code**

Write a Menu driven program in JavaScript to carry out the following in order to demonstrate the use of arrow functions -

1. Write an arrow function to accept an array of subject marks of a student and calculate its average.
2. Write an arrow function to display Fibonacci series.
3. Write an arrow function to check whether the entered string is palindrome or not.

**8. Post Experimental Exercise-**

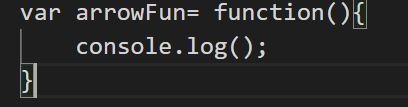
a) Find the output:





**Output:**

b)Write the given anonymous method using arrow function.



**Output:**

A black screen with red text

Description automatically generated

**9. Results/Observations/Program output:**

Present the program code and output

| **CODE:**  **<!DOCTYPE html>**  **<html>**  **<head>**  **<title>Arrow Function Examples</title>**  **</head>**  **<body>**  **<script>**  **// Arrow function to calculate average of an array of numbers**  **const calculateAverage = marks => {**  **const sum = marks.reduce((total, mark) => total + mark, 0);**  **return sum / marks.length;**  **};**  **// Arrow function to generate Fibonacci series**  **const generateFibonacciSeries = count => {**  **const series = [0, 1];**  **for (let i = 2; i < count; i++) {**  **series.push(series[i - 1] + series[i - 2]);**  **}**  **return series;**  **};**  **// Arrow function to check if a string is palindrome**  **const isPalindrome = str => {**  **const cleanStr = str.toLowerCase().replace(/[^a-zA-Z0-9]/g, '');**  **const reversedStr = cleanStr.split('').reverse().join('');**  **return cleanStr === reversedStr;**  **};**  **// Main menu function**  **const mainMenu = () => {**  **const choice = parseInt(prompt(`Select an option:**  **1. Calculate average of subject marks**  **2. Generate Fibonacci series**  **3. Check for palindrome**  **4. Exit`));**  **switch (choice) {**  **case 1:**  **const marks = prompt("Enter subject marks separated by commas:").split(',').map(Number);**  **const average = calculateAverage(marks);**  **alert(Average: ${average});**  **break;**  **case 2:**  **const count = parseInt(prompt("Enter the count of Fibonacci numbers to generate:"));**  **const fibonacciSeries = generateFibonacciSeries(count);**  **alert(Fibonacci Series: ${fibonacciSeries.join(', ')});**  **break;**  **case 3:**  **const inputString = prompt("Enter a string:");**  **const palindromeResult = isPalindrome(inputString) ? "is" : "is not";**  **alert(The entered string ${palindromeResult} a palindrome.);**  **break;**  **case 4:**  **alert("Exiting the program.");**  **break;**  **default:**  **alert("Invalid choice.");**  **break;**  **}**  **if (choice !== 4) {**  **mainMenu(); // Show menu again**  **}**  **};**  **// Start the program**  **mainMenu();**  **</script>**  **</body>**  **</html>** |
| --- |

| **Output:**  **A screenshot of a computer error  Description automatically generated**  A screenshot of a computer  Description automatically generated  A number on a white background  Description automatically generated  A screenshot of a computer  Description automatically generated      A white background with black dots  Description automatically generated  A blue and white line  Description automatically generated  A black and white text  Description automatically generated |
| --- |

**10. Conclusion:**

Write what was performed in the experiment

Write which all features of JavaScript you used to perform the experiment

**11.** **References:**

* HTML 5 Black Book (Covers CSS3, JavaScript, XML, XHTML, AJAX, PHP, jQuery) 2Ed., DT Editorial Services
* <https://www.w3schools.com/js/default.asp>
* <https://www.tutorialspoint.com/javascript/index.htm>
* [https://www.youtube.com/watch?v=W6NZfCO5SI](https://www.youtube.com/watch?v=W6NZfCO5SIk)