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 – 7**

**Menu driven program to demonstrate the use of classes and inheritance in JavaScript**

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

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

inheritance.

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

implement the basic concepts of JavaScript including use of classes and inheritance.

**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:**

JavaScript classes are templates for JavaScript objects. A JavaScript class is not an object. It is a template for JavaScript objects.

Use the keyword class to create a class. Always add a method named constructor()

class ClassName {

constructor() { ... }

}

When you have a class, you can use the class to create objects

**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 classes and inheritance -

1. (a) Create a class ‘accholder’ with following attribute as ‘accnum’, ‘name’, ‘age’, and ‘balance’.

(b) Print details of two employees using display function.

1. (a) Create a parent class “Calculator” with attributes length, width.

(b) Create one derived class named “rectArea” from “calculator” with attribute radius to calculate and display area of a rectangle. Use super keyword to call parent’s constructor.

(c) Create derived class “circleArea” from “rectArea” to calculate and display area of a circle.

1. (a) Create a class “CheckNum” with attributes num1 and num2.

(b) Create a derived class “CheckNeonSpy” from “CheckNum”. Define methods CheckNeon() and CheckSpy().

*A neon number is a number where the sum of the digits of the square of the number is equal to the number. For example, 9 is a neon number; its square is 9\*9=81 and the sum of these digits is 8+1=9.*

*A spy number is a number where the sum of its digits equals the product of its digits. For example, 1124 is a spy number, the sum of its digits is 1+1+2+4=8 and the product of its digits is 1\*1\*2\*4=8.*

**8. Post Experimental Exercise-**

Google Form Quiz Link –

TEB:

<https://docs.google.com/forms/d/e/1FAIpQLSec97otb3dyAC8NQYL3buZ401Tfz2cM6Vx62vZG55nwuu1S4g/viewform?usp=sf_link>

TEA: <https://docs.google.com/forms/d/e/1FAIpQLSejHqadZ3mvGdmPsz3uskGvIdCyq_NCTu7TMEnkaI-TtuJF2w/viewform?usp=sf_link>

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

Present the program code and output

| **CODE:  <!DOCTYPE html>**  **<html lang="en">**  **<head>**  **<meta charset="UTF-8">**  **<meta name="viewport" content="width=device-width, initial-scale=1.0">**  **<title>Document</title>**  **<script>**  **class Accholder {**  **constructor(accnum, name, age, balance) {**  **this.accnum = accnum;**  **this.name = name;**  **this.age = age;**  **this.balance = balance;**  **}**  **display() {**  **console.log(Account Number: ${this.accnum});**  **console.log(Name: ${this.name});**  **console.log(Age: ${this.age});**  **console.log(Balance: ${this.balance});**  **console.log();**  **}**  **}**  **class Calculator {**  **constructor(length, width) {**  **this.length = length;**  **this.width = width;**  **}**  **}**  **class RectArea extends Calculator {**  **constructor(length, width, radius) {**  **super(length, width);**  **this.radius = radius;**  **}**  **calculateRectangleArea() {**  **return this.length \* this.width;**  **}**  **}**  **class CircleArea extends RectArea {**  **constructor(length, width, radius) {**  **super(length, width, radius);**  **}**  **calculateCircleArea() {**  **return Math.PI \* Math.pow(this.radius, 2);**  **}**  **}**  **class CheckNum {**  **constructor(num1, num2) {**  **this.num1 = num1;**  **this.num2 = num2;**  **}**  **}**  **class CheckNeonSpy extends CheckNum {**  **constructor(num1, num2) {**  **super(num1, num2);**  **}**  **sumOfDigits(number) {**  **return number.toString().split('').reduce((sum, digit) => sum + parseInt(digit), 0);**  **}**  **productOfDigits(number) {**  **return number.toString().split('').reduce((product, digit) => product \* parseInt(digit), 1);**  **}**  **checkNeon() {**  **return this.sumOfDigits(this.num1 \* this.num1) === this.num1;**  **}**  **checkSpy() {**  **return this.sumOfDigits(this.num2) === this.productOfDigits(this.num2);**  **}**  **}**  **// Taking input from user**  **// Part (a)**  **const accnum1 = parseInt(prompt("Enter Account Number for emp1:"));**  **const name1 = prompt("Enter Name for emp1:");**  **const age1 = parseInt(prompt("Enter Age for emp1:"));**  **const balance1 = parseFloat(prompt("Enter Balance for emp1:"));**  **const accnum2 = parseInt(prompt("Enter Account Number for emp2:"));**  **const name2 = prompt("Enter Name for emp2:");**  **const age2 = parseInt(prompt("Enter Age for emp2:"));**  **const balance2 = parseFloat(prompt("Enter Balance for emp2:"));**  **const emp1 = new Accholder(accnum1, name1, age1, balance1);**  **const emp2 = new Accholder(accnum2, name2, age2, balance2);**  **emp1.display();**  **emp2.display();**  **// Part (b)**  **const length = parseFloat(prompt("Enter Length:"));**  **const width = parseFloat(prompt("Enter Width:"));**  **const radius = parseFloat(prompt("Enter Radius for circle:"));**  **const rect = new RectArea(length, width, radius);**  **console.log("Rectangle Area:", rect.calculateRectangleArea());**  **const circle = new CircleArea(length, width, radius);**  **console.log("Circle Area:", circle.calculateCircleArea());**  **// Part (c)**  **const num1 = parseInt(prompt("Enter a number to check Neon:"));**  **const num2 = parseInt(prompt("Enter a number to check Spy:"));**  **const checkNeonSpy = new CheckNeonSpy(num1, num2);**  **console.log("Is", num1, "a neon number?", checkNeonSpy.checkNeon());**  **console.log("Is", num2, "a spy number?", checkNeonSpy.checkSpy());**  **</script>**  **</head>**  **<body>**    **</body>**  **</html>** |
| --- |

| Output:                            **Output:** |
| --- |

**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=W6NZfCO5SIk>
* <https://www.youtube.com/watch?v=PkZNo7MFNFg>