**Lab02- Functional Programming**

**Name: Ashwanthika Umasankar**

**Student Id: 1001854976**

**Date : 06/29/2023**

**Windows 10 OS**

Question 1 : Start with an array called inputtable. The array should have numbers between 1 and 10.

// Answer 1: Creating an array inputtable from numbers 1 to 10 and outputiing it to the console.

var inputtable = [1,2,3,4,5,6,7,8,9,10];

  console.log("Question 1: Inputtable array", inputtable)

  console.log("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_")

Question 2 : Use inputtable from step 1 to create the following:

Question 2a : Set of multiples of 5 between 1 and 51. Name it fiveTable, print the contents to the console

var fiveTable = [];

//Function to calculate multiples of five from inputtable array

function answerFive (val) {

  // Check if val is less than the length of inputtable array

  if (val < inputtable.length) {

    var multiple = inputtable[val] \* 5;

    //checking if it is in range 1 to 51

    if (multiple >= 1 && multiple <= 51) {

      fiveTable.push(multiple);

    }

    //calling function recursively

    answerFive(val + 1);

  }

}

answerFive (0);

console.log("Question 2a Multiples of 5", fiveTable);

console.log("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_")

2b: Set of multiples of 13 between 1 and 131. Name it thirteenTable, print the contents to the console

var thirteenTable = [];

// Function to calculate multiples of thirteen from inputtable array

function answerThirteen (val) {

  // Check if val is less than the length of inputtable array

  if (val < inputtable.length) {

    var multiple = inputtable[val] \* 13;

    //checking if it is in range 1 to 131

    if (multiple >= 1 && multiple <= 131) {

      thirteenTable.push(multiple);

    }

    //calling function recursively

    answerThirteen(val + 1);

  }

}

answerThirteen (0);

console.log(" Question 2b Multiples of 13", thirteenTable);

console.log("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_")

2c: Set of squares of the numbers in inputtable. Name it squaresTable, print the contents to the console

var squaresTable = [];

function square(i) {

  // Check if val is less than the length of inputtable array

  if (i >= inputtable.length) return;

  //if yes find the square

  squaresTable.push(inputtable[i] \* inputtable[i]);

  square(i + 1);

}

square(0);

console.log("Question 2c Squares", squaresTable);

console.log("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_")

//---------------------------------------------------------

Question 3: Get (and then print) the odd multiples of 5 between 1 and 100. 5, 15, ..

//function to compute and store odd multiples of 5

function oddMultiplesFive(FiveMultiple, output) {

  if (5 \* FiveMultiple > 100) {

    return output;

  }

  if (FiveMultiple % 2 === 1) {

    output.push(5 \* FiveMultiple);

  }

  return oddMultiplesFive(FiveMultiple + 1, output);

}

var outputres = oddMultiplesFive(1, []);

console.log("Question 3 OddMultiples of 5", outputres); //printing the result array wich has 5's odd multiples

console.log("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_")

Question 4: Get (and then print) the sum of even multiples of 7 between 1 and 100.

//------------------------------------------------

// Calculate and store the sum of even multiples of 7

let add = 0;

function evenSeven(multiple) {

  (multiple > 100)

    ? (console.log("Question 4: Sum of even multiples of 7:", add), null)

    : (

        (multiple % 7 === 0 && multiple % 2 === 0)

          ? (add += multiple)

          : null,

          evenSeven(multiple + 1)

      );

}

evenSeven(1);

console.log("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_")

//-----------------------------------------------------

Question 5: Use currying to rewrite the function below:

//function to compute cylinder volume

function cylinder\_volume(r) {

  return function(h) {

    var volume = 3.14 \* r \* r \* h;

    return volume;

  }

  }

  //using currying and reqriting the function

  // a. Using r = 5 and h = 10 to call your curried function.

  var volume = cylinder\_volume(5)

  var ret1= volume(10);

  console.log("Volume of 5A:", ret1);

  // b. Reusing the function from part ‘a’ but using h = 17

  var ret2 = volume(17);

  console.log("VVolume of 5B:", ret2);

  // c. Reusing the function from part ‘a’ but using h = 11

  var ret3 = volume(11);

  console.log("Volume of 5C:", ret3);

  console.log("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_")

//-----------------------------------------------------

Question 6: Use the following code to take advantage of closures to wrap content with HTML tags, specifically show an HTML table consisting of a table row that has at least one table cell/element. You can use the console to output your results. Note: you MUST call makeTag() at least once but may NOT call it more than 4 times.

//function to create HTML tags

makeTag = function(beginTag, endTag) {

  return function(textContent) {

    return beginTag + textContent + endTag;

  }

  }

  console.log("Question 6:");

  //creating tag functions for table,tr and td. not calling it >4 times

  var table = makeTag("<table>\n", "</table>\n");

  var tablerow = makeTag("<tr>\n", "</tr>\n");

  var tabledata = makeTag("<td>", "</td>\n");

  //print the table using tag functions

  console.log(table(

    tablerow(

      tabledata("first name") +tabledata("ashwanthika")

  ) +

  tablerow(

    tabledata("last name") +tabledata("umasankar")

  ) +

  tablerow(

    tabledata("age") +tabledata("24")

  )

  ));

//------------------------------------

console.log("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_")

Question 7: Do the ‘generic’ version of questions 3 and 4

function evenOddMultiples(odd\_even\_var, number, init\_val) {

  var evenArray = [];

  var oddArray = [];

  //recursive function to compute multiples

  function compute(init\_val) {

    if (number \* init\_val > 100) {

      return;

    }

    //checking if the val passed with function call is odd or even, if it is odd then we print only odd multiples else even multiples

    if (odd\_even\_var == 1) {

      if (init\_val % 2 == 1) {

        oddArray.push(number \* init\_val);

      }

    } else {

      if (init\_val % 2 == 0) {

        evenArray.push(number \* init\_val);

      }

    }

    compute(init\_val + 1);

  }

  compute(init\_val);

  console.log("Question 7:");

//if its odd then we print as such, if its even we find the sum and return

  if (odd\_even\_var == 1) {

    console.log("Odd multiples of the given number %d array:", number, oddArray);

  } else {

    console.log("Even multiples of the given number %d array:", number, evenArray);

    var sumevenArray = evenArray.reduce((sum, current) => sum + current, 0);

    console.log("Sum of even multiples:", sumevenArray);

  }

}

evenOddMultiples(1, 11, 1); // Odd multiples of 11 between 1 and 100

evenOddMultiples(0, 3, 1);  // Even multiples of 3 between 1 and 100

console.log("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_")