PSG COLLEGE OF TECHNOLOGY

Department of Computer Applications

23MX15 Web Technologies JavaScript Worksheet 3

**If writing it as HTML code, please write an internal script. Do not zip your files**.

Use the prompt and process.stdout.write methods for solving this worksheet. Refer inpuoutpuJS.txt file shared in Google Classroom.

1. Write a menu driven JavaScript program to manipulate JS  array as stack:

  case 1: push elements on to stack ”myStack” - use JS push array method ( allow only numbers (both integer and float) to be pushed.  But JS array allows any data type in  arrays.  )

     case 2:  pop elements out of stack “myStack”- use JS pop array method  ( check whether empty or not . If not empty pop element outside)

      case 3:  find the presence of an element and its index ( should use only find() and findIndex() methods on array stack)

      case 4:  print number of elements in the stack ( use stack array length property).

      case 5:  count the total integers and total floats inside the stack

Case 6: fix the number of decimal places as ‘n’’ for all the floating point numbers inside the stack. - use the toFixed(n) number method.

      default: Use the reduce() array method to reduce the array values to a sum value. Print the sum.

2. Write a menu driven JavaScript program to manipulate JS  array as Queue:

        case 1: Insert elements at the end of queue 'myQueue' using JS push array method. (allow both int and strings)

        case 2:  Remove element from the front of Queue using JS shift() array method.

        case 3:  print the total number of elements in the Queue

        case 4:  sort the elements in Queue array and store it in another array. (use sort array method).

        case 5:  remove only the 'n' number of elements from 'ith ' position of the queue array. use splice array method and add two new items into it. Print the removed elements and the new array.

Example: if myQueue=[123,”Hi”,3567,8.9,”23MX”,987];

Var removed= myQueue.splice(2,2,”Hello”,”Welcome”);

should make the myQueue=[123,”Hi”,”Hello”,”Welcome”,”23MX”,987];

        case 6:  Divide the Queue into two different Queues  first, second - use JS slice array method.

        case 7:  Join all the elements of Queue as a single string - use join array method.

Default: Use the map() method on myQueue to convert all the elements to string data type.