JavaScript basic

1. Write a JavaScript program to display the current day and time in the following format.
Sample Output : Today is : Friday.
Current time is : 4 PM : 50 : 22

2. Write a JavaScript program to print the contents of the current window.

3. Write a JavaScript program to get the current date.
Expected Output :
mm-dd-yyyy, mm/dd/yyyy or dd-mm-yyyy, dd/mm/yyyy

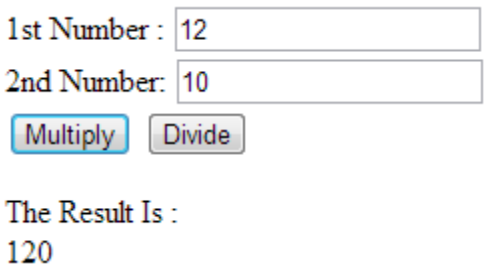
4. Write a JavaScript program to find the area of a triangle where lengths of the three of its sides are 5, 6, 7.

5. Write a JavaScript program to rotate the string 'w3resource' in right direction by periodically removing one letter from the end of the string and attaching it to the front.

6. Write a JavaScript program to determine whether a given year is a leap year in the Gregorian calendar.

7. Write a JavaScript program to find 1st January be a Sunday between 2014 and 2050.

8. Write a JavaScript program where the program takes a random integer between 1 to 10, the user is then prompted to input a guess number. If the user input matches with guess number, the program will display a message "Good Work" otherwise display a message "Not matched". **9.** Write a JavaScript program to calculate days left until next Christmas. **10.** Write a JavaScript program to calculate multiplication and division of two numbers (input from user). *Sample form* :



11. Write a JavaScript program to convert temperatures to and from celsius, fahrenheit. [Formula : c/5 = (f-32)/9 [where c = temperature in celsius and f = temperature in fahrenheit]
Expected Output :
60°C is 140 °F
45°F is 7.222222222222222°C

12. Write a JavaScript program to get the website URL (loading page).

JavaScript functions

---

1. Write a JavaScript function that reverse a number.
Example x = 32243;
Expected Output : 34223

2. Write a JavaScript function that checks whether a passed string is palindrome or not?
A palindrome is word, phrase, or sequence that reads the same backward as forward, e.g., madam or nurses run.

3. Write a JavaScript function that generates all combinations of a string.
Example string : 'dog'
Expected Output : d,do,dog,o,og,g

4. Write a JavaScript function that returns a passed string with letters in alphabetical order.
Example string : 'webmaster'
Expected Output : 'abeemrstw'
Assume punctuation and numbers symbols are not included in the passed string.

5. Write a JavaScript function that accepts a string as a parameter and converts the first letter of each word of the string in upper case.
Example string : 'the quick brown fox'
Expected Output : 'The Quick Brown Fox '

6. Write a JavaScript function that accepts a string as a parameter and find the longest word within the string.
Example string : 'Web Development Tutorial'
Expected Output : 'Development'

7. Write a JavaScript function that accepts a string as a parameter and counts the number of vowels within the string.
Note : As the letter 'y' can be regarded as both a vowel and a consonant, we do

not count 'y' as vowel here.
Example string : 'The quick brown fox'
Expected Output : 5

8. Write a JavaScript function that accepts a number as a parameter and check the number is prime or not.
Note : A prime number (or a prime) is a natural number greater than 1 that has no positive divisors other than 1 and itself.

9. Write a JavaScript function which accepts an argument and returns the type.
Note : There are six possible values that typeof returns: object, boolean, function, number, string, and undefined.

10. Write a JavaScript function which returns the n rows by n columns identity matrix.

11. Write a JavaScript function which will take an array of numbers stored and find the second lowest and second greatest numbers, respectively.
Sample array : [1,2,3,4,5]
Expected Output : 2,4
.
12. Write a JavaScript function which says whether a number is perfect.
According to Wikipedia : In number theory, a perfect number is a positive integer that is equal to the sum of its proper positive divisors, that is, the sum of its positive divisors excluding the number itself (also known as its aliquot sum). Equivalently, a perfect number is a number that is half the sum of all of its positive divisors (including itself).
Example : The first perfect number is 6, because 1, 2, and 3 are its proper positive divisors, and 1 + 2 + 3 = 6. Equivalently, the number 6 is equal to half the sum of all its positive divisors: (1 + 2 + 3 + 6) / 2 = 6. The next perfect number is 28 = 1 + 2 + 4 + 7 + 14. This is followed by the perfect numbers 496 and 8128.
.
13. Write a JavaScript function to compute the factors of a positive integer.
.
14. Write a JavaScript function to convert an amount to coins.
Sample function : amountTocoins(46, [25, 10, 5, 2, 1])
Here 46 is the amount. and 25, 10, 5, 2, 1 are coins.
Output : 25, 10, 10, 1
.
15. Write a JavaScript function to compute the value of b^n where n is the exponent and b is the bases. Accept b and n from the user and display the result.
.

16. Write a JavaScript function to extract unique characters from a string.
Example string : "thequickbrownfoxjumpsoverthelazydog"
Expected Output : "thequickbrownfxjmpsvlazydg"
.
17. Write a JavaScript function to get the number of occurrences of each letter in specified string.
.
18. Write a function for searching JavaScript arrays with a binary search.
Note : A binary search searches by splitting an array into smaller and smaller chunks until it finds the desired value.
.
19. Write a JavaScript function that returns array elements larger than a number.
.
20. Write a JavaScript function that generates a string id (specified length) of random characters.
Sample character list :
"ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789"
.
21. Write a JavaScript function to get all possible subset with a fixed length (for example 2) combinations in an array.
Sample array : [1, 2, 3] and subset length is 2
Expected output : [[2, 1], [3, 1], [3, 2], [3, 2, 1]]
.
22. Write a JavaScript function that accepts two arguments, a string and a letter and the function will count the number of occurrences of the specified letter within the string.
Sample arguments : 'w3resource.com', 'o'
Expected output : 2

23. Write a JavaScript function to find the first not repeated character.
Sample arguments : 'abacddbec'
Expected output : 'e'

24. Write a JavaScript function to apply Bubble Sort algorithm.
Note : According to wikipedia "Bubble sort, sometimes referred to as sinking sort, is a simple sorting algorithm that works by repeatedly stepping through the list to be sorted, comparing each pair of adjacent items and swapping them if they are in the wrong order".
Sample array : [12, 345, 4, 546, 122, 84, 98, 64, 9, 1, 3223, 455, 23, 234, 213]
Expected output : [3223, 546, 455, 345, 234, 213, 122, 98, 84, 64, 23, 12, 9, 4, 1]

25. Write a JavaScript function that accept a list of country names as input and returns the longest country name as output.
Sample function : Longest_Country_Name(["Australia", "Germany", "United States of America"])
Expected output : "United States of America"

26. Write a JavaScript function to find longest substring in a given a string without repeating characters.

27. Write a JavaScript function that returns the longest palindrome in a given string.
Note: According to Wikipedia "In computer science, the longest palindromic substring or longest symmetric factor problem is the problem of finding a maximum-length contiguous substring of a given string that is also a palindrome. For example, the longest palindromic substring of "bananas" is "anana". The longest palindromic substring is not guaranteed to be unique; for example, in the string "abracadabra", there is no palindromic substring with length greater than three, but there are two palindromic substrings with length three, namely, "aca" and "ada".
In some applications it may be necessary to return all maximal palindromic substrings (that is, all substrings that are themselves palindromes and cannot be extended to larger palindromic substrings) rather than returning only one substring or returning the maximum length of a palindromic substring.

28. Write a JavaScript program to pass a 'JavaScript function' as parameter.

29. Write a JavaScript function to get the function name.

JavaScript Recursion

Use recursion to solve the following exercises.

1. Write a JavaScript program to calculate the factorial of a number.
In mathematics, the factorial of a non-negative integer n, denoted by n!, is the product of all positive integers less than or equal to n. For example, 5! = 5 x 4 x 3 x 2 x 1 = 120

2. Write a JavaScript program to find the greatest common divisor (gcd) of two positive numbers.

.
3. Write a JavaScript program to get the integers in range (x, y).
Example : range(2, 9)

Expected Output : [3, 4, 5, 6, 7, 8]
.
4. Write a JavaScript program to compute the sum of an array of integers.
Example : var array = [1, 2, 3, 4, 5, 6]
Expected Output : 21
.
5. Write a JavaScript program to compute the exponent of a number.
Note : The exponent of a number says how many times the base number is used as a factor.
8^2 = 8 x 8 = 64. Here 8 is the base and 2 is the exponent.
.
6. Write a JavaScript program to get the first n Fibonacci numbers.
Note : The Fibonacci Sequence is the series of numbers: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, . . . Each subsequent number is the sum of the previous two.
.
7. Write a JavaScript program to check whether a number is even or not.
.
8. Write a JavaScript program for binary search.
Sample array : [0,1,2,3,4,5,6]
console.log(l.br_search(5)) will return '5'
.
9. Write a merge sort program in JavaScript.
Sample array : [34,7,23,32,5,62]
Sample output : [5, 7, 23, 32, 34, 62]

JavaScript conditional statements and loops

1. Write a JavaScript program that accept two integers and display the larger.

2. Write a JavaScript conditional statement to find the sign of product of three numbers. Display an alert box with the specified sign.
Sample numbers : 3, -7, 2
Output : The sign is -

3. Write a JavaScript conditional statement to sort three numbers. Display an alert box to show the result.
Sample numbers : 0, -1, 4
Output : 4, 0, -1

4. Write a JavaScript conditional statement to find the largest of five numbers. Display an alert box to show the result.

Sample numbers : -5, -2, -6, 0, -1
Output : 0

5. Write a JavaScript for loop that will iterate from 0 to 15. For each iteration, it will check if the current number is odd or even, and display a message to the screen.
Sample Output :
"0 is even"
"1 is odd"
"2 is even"

---

6. Write a JavaScript program which compute, the average marks of the following students Then, this average is used to determine the corresponding grade.

Student Name	**Marks**
David	80
Vinoth	77
Divya	88
Ishitha	95
Thomas	68

The grades are computed as follows :

Range	**Grade**
<60	F
<70	D
<80	C
<90	B
<100	A

7. Write a JavaScript program which iterates the integers from 1 to 100. But for multiples of three print "Fizz" instead of the number and for the multiples of five print "Buzz". For numbers which are multiples of both three and five print

"FizzBuzz".

8. According to Wikipedia a happy number is defined by the following process : "Starting with any positive integer, replace the number by the sum of the squares of its digits, and repeat the process until the number equals 1 (where it will stay), or it loops endlessly in a cycle which does not include 1. Those numbers for which this process ends in 1 are happy numbers, while those that do not end in 1 are unhappy numbers (or sad numbers)".
Write a JavaScript program to find and print the first 5 happy numbers.

9. Write a JavaScript program to find the armstrong numbers of 3 digits.
Note : An Armstrong number of three digits is an integer such that the sum of the cubes of its digits is equal to the number itself. For example, 371 is an Armstrong number since 3**3 + 7**3 + 1**3 = 371.

10. Write a JavaScript program to construct the following pattern, using a nested for loop.

```
*

* *

* * *

* * * *

* * * * *
```

11. Write a JavaScript program to compute the greatest common divisor (GCD) of two positive integers.

12. Write a JavaScript program to sum the multiples of 3 and 5 under 1000.

JavaScript array

1. Write a JavaScript function to check whether an `input` is an array or not.
Test Data :
console.log(is_array('w3resource'));
console.log(is_array([1, 2, 4, 0]));
false

true

2. Write a JavaScript function to clone an array.
Test Data :
console.log(array_Clone([1, 2, 4, 0]));
console.log(array_Clone([1, 2, [4, 0]]));
[1, 2, 4, 0]
[1, 2, [4, 0]]

3. Write a JavaScript function to get the first element of an array. Passing a parameter 'n' will return the first 'n' elements of the array.
Test Data :
console.log(first([7, 9, 0, -2]));
console.log(first([],3));
console.log(first([7, 9, 0, -2],3));
console.log(first([7, 9, 0, -2],6));
console.log(first([7, 9, 0, -2],-3));
Expected Output :
7
[]
[7, 9, 0]
[7, 9, 0, -2]
[]

4. Write a JavaScript function to get the last element of an array. Passing a parameter 'n' will return the last 'n' elements of the array.
Test Data :
console.log(last([7, 9, 0, -2]));
console.log(last([7, 9, 0, -2],3));
console.log(last([7, 9, 0, -2],6));
Expected Output :
-2
[9, 0, -2]
[7, 9, 0, -2]

5. Write a simple JavaScript program to join all elements of the following array into a string.
Sample array : myColor = ["Red", "Green", "White", "Black"];
Expected Output :
"Red,Green,White,Black"
"Red,Green,White,Black"
"Red+Green+White+Black"

6. Write a JavaScript program which accept a number as input and insert dashes (-) between each two even numbers. For example if you accept 025468 the output should be 0-254-6-8.

7. Write a JavaScript program to sort the items of an array.
Sample array : var arr1 = [3, 8, 7, 6, 5, -4, 3, 2, 1];
Sample Output : -4,-3,1,2,3,5,6,7,8

8. Write a JavaScript program to find the most frequent item of an array.
Sample array : var arr1=[3, 'a', 'a', 'a', 2, 3, 'a', 3, 'a', 2, 4, 9, 3];
Sample Output : a (5 times)

9. Write a JavaScript program which accept a string as input and swap the case of each character. For example if you input 'The Quick Brown Fox' the output should be 'tHE qUICK bROWN fOX'.

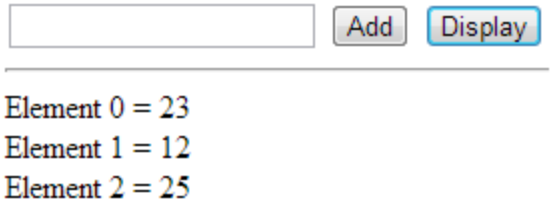
10. Write a JavaScript program which prints the elements of the following array.
Note : Use nested for loops.
Sample array : var a = [[1, 2, 1, 24], [8, 11, 9, 4], [7, 0, 7, 27], [7, 4, 28, 14], [3, 10, 26, 7]];
Sample Output :
"row 0"
" 1"
" 2"
" 1"
" 24"
"row 1"

------

11. Write a JavaScript program to find the sum of squares of a numeric vector.

12. Write a JavaScript program to compute the sum and product of an array of integers.

13. Write a JavaScript program to add items in an blank array and display the items.
Sample Screen :



14. Write a JavaScript program to remove duplicate items from an array (ignore case sensitivity).

15. We have the following arrays :
color = ["Blue ", "Green", "Red", "Orange", "Violet", "Indigo", "Yellow "];
o = ["th","st","nd","rd"]
Write a JavaScript program to display the colors in the following way :
"1st choice is Blue ."
"2nd choice is Green."
"3rd choice is Red."
- - - - - - - - - - - - -
Note : Use ordinal numbers to tell their position.

16. Find the leap years in a given range of years.

17. Write a JavaScript program to shuffle an array.

18. Write a JavaScript program to perform a binary search.
Note : A binary search or half-interval search algorithm finds the position of a specified input value within an array sorted by key value.
Sample array :
var items = [1, 2, 3, 4, 5, 7, 8, 9];
Expected Output :
console.log(binary_Search(items, 1)); //0
console.log(binary_Search(items, 5)); //4

19. There are two arrays with individual values, write a JavaScript program to compute the sum of each individual index value from the given arrays.
Sample array :
array1 = [1,0,2,3,4];
array2 = [3,5,6,7,8,13];
Expected Output :
[4, 5, 8, 10, 12, 13]

20. Write a JavaScript program to find duplicate values in a JavaScript array.

21. Write a JavaScript program to flatten a nested (any depth) array. If you pass shallow, the array will only be flattened a single level.
Sample Data :
console.log(flatten([1, [2], [3, [[4]]],[5,6]]));
[1, 2, 3, 4, 5, 6]
console.log(flatten([1, [2], [3, [[4]]],[5,6]], true));
[1, 2, 3, [[4]], 5, 6]

22. Write a JavaScript program to compute the union of two arrays.
Sample Data :
console.log(union([1, 2, 3], [100, 2, 1, 10]));
[1, 2, 3, 10, 100]

23. Write a JavaScript function to find the difference of two arrays.
Test Data :
console.log(difference([1, 2, 3], [100, 2, 1, 10]));
["3", "10", "100"]
console.log(difference([1, 2, 3, 4, 5], [1, [2], [3, [[4]]],[5,6]]));
["6"]
console.log(difference([1, 2, 3], [100, 2, 1, 10]));
["3", "10", "100"]

24. Write a JavaScript function to remove. 'null', '0', '""', 'false', 'undefined' and 'NaN' values from an array.
Sample array : [NaN, 0, 15, false, -22, '',undefined, 47, null]
Expected result : [15, -22, 47]

25. Write a JavaScript function to sort the following array of objects by title value.
Sample object :

```
var library = [

   { author: 'Bill Gates', title: 'The Road Ahead', libraryID: 1254},

   { author: 'Steve Jobs', title: 'Walter Isaacson', libraryID: 4264},

   { author: 'Suzanne Collins', title: 'Mockingjay: The Final Book of The Hunger Games
', libraryID: 3245}

   ];
```

Expected result :

```
[[object Object] {

  author: "Suzanne Collins",

  libraryID: 3245,

  title: "Mockingjay: The Final Book of The Hunger Games"
```

```
}, [object Object] {
  author: "Bill Gates",
  libraryID: 1254,
  title: "The Road Ahead"
}, [object Object] {
  author: "Steve Jobs",
  libraryID: 4264,
  title: "Walter Isaacson"
}]
```

26. Write a JavaScript program to find a pair of elements (indices of the two numbers) from an given array whose sum equals a specific target number.
Input: numbers= [10,20,10,40,50,60,70], target=50
Output: 3, 4

27. Write a JavaScript function to retrieve the value of a given property from all elements in an array.
Sample array : [NaN, 0, 15, false, -22, '',undefined, 47, null]
Expected result : [15, -22, 47]

28. Write a JavaScript function to find the longest common starting substring in a set of strings.
Sample array : console.log(longest_common_starting_substring(['go', 'google']));
Expected result : "go"

29. Write a JavaScript function to fill an array with values (numeric, string with one character) on supplied bounds.
Test Data :
console.log(num_string_range('a', "z", 2));
["a", "c", "e", "g", "i", "k", "m", "o", "q", "s", "u", "w", "y"]

30. Write a JavaScript function to merge two arrays and removes all duplicates elements.
Test data :
var array1 = [1, 2, 3];
var array2 = [2, 30, 1];
console.log(merge_array(array1, array2));
[3, 2, 30, 1]

31. Write a JavaScript function to remove a specific element from an array.

Test data :
console.log(remove_array_element([2, 5, 9, 6], 5));
[2, 9, 6]

32. Write a JavaScript function to find an array contains a specific element.
Test data :
console.log(remove_array_element([2, 5, 9, 6], 5));
[2, 9, 6]

33. Write a JavaScript script to empty an array keeping the original.
.
34. Write a JavaScript function to get nth largest element from an unsorted array.
Test Data :
console.log(nthlargest([43, 56, 23, 89, 88, 90, 99, 652], 4));
89

35. Write a JavaScript function to get a random item from an array.

36. Write a JavaScript function to create a specified number of elements with pre-filled numeric value array.
Test Data :
console.log(array_filled(6, 0));
[0, 0, 0, 0, 0, 0]
console.log(array_filled(4, 11));
[11, 11, 11, 11]

37. Write a JavaScript function to create a specified number of elements with pre-filled string value array.
Test Data :
console.log(array_filled(3, 'default value'));
["default value", "default value", "default value"]
console.log(array_filled(4, 'password'));
["password", "password", "password", "password"]

38. Write a JavaScript function to move an array element from one position to another.
Test Data :
console.log(move([10, 20, 30, 40, 50], 0, 2));
[20, 30, 10, 40, 50]
console.log(move([10, 20, 30, 40, 50], -1, -2));
[10, 20, 30, 50, 40]

39. Write a JavaScript function to filter false, null, 0 and blank values from an array.
Test Data :
console.log(filter_array_values([58, '', 'abcd', true, null, false, 0]));
[58, "abcd", true]

40. Write a JavaScript function to generate an array of specified length, filled with integer numbers, increase by one from starting position.
Test Data :
console.log(array_range(1, 4));
[1, 2, 3, 4]
console.log(array_range(-6, 4));
[-6, -5, -4, -3]

41. Write a JavaScript function to generate an array between two integers of 1 step length.
Test Data :
console.log(rangeBetwee(4, 7));
[4, 5, 6, 7]
console.log(rangeBetwee(-4, 7));
[-4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6, 7]

42. Write a JavaScript function to find the unique elements from two arrays.
Test Data :
console.log(difference([1, 2, 3], [100, 2, 1, 10]));
["1", "2", "3", "10", "100"]
console.log(difference([1, 2, 3, 4, 5], [1, [2], [3, [[4]]],[5,6]]));
["1", "2", "3", "4", "5", "6"]
console.log(difference([1, 2, 3], [100, 2, 1, 10]));
["1", "2", "3", "10", "100"]

JavaScript Date

JavaScript date object is used to create dates and times. Date objects are based on a time value that is the number of milliseconds since 1 January, 1970 UTC.

Constructor :

```
new Date();

new Date(value);

new Date(dateString);

new Date(year, month[, day[, hour[, minute[, second[, millisecond]]]]]);
```

1. Write a JavaScript function to check whether an `input` is a date object or not.
Test Data :
console.log(is_date("October 13, 2014 11:13:00"));
console.log(is_date(new Date(86400000)));
console.log(is_date(new Date(99,5,24,11,33,30,0)));
console.log(is_date([1, 2, 4, 0]));
Output :
false
true
true
false

2. Write a JavaScript function to get the current date.
Note : Pass a separator as an argument.
Test Data :
console.log(curday('/'));
console.log(curday('-'));
Output :
"11/13/2014"
"11-13-2014"

3. Write a JavaScript function to get the number of days in a month.
Test Data :
console.log(getDaysInMonth(1, 2012));
console.log(getDaysInMonth(2, 2012));
console.log(getDaysInMonth(9, 2012));
console.log(getDaysInMonth(12, 2012));

Output :
31
29
30
31

4. Write a JavaScript function to get the month name from a particular date.
Test Data :
console.log(month_name(new Date("10/11/2009")));
console.log(month_name(new Date("11/13/2014")));
Output :
"October"
"November"

5. Write a JavaScript function to compare dates (i.e. greater than, less than or equal to).
Test Data :
console.log(compare_dates(new Date('11/14/2013 00:00'), new Date('11/14/2013 00:00')));
console.log(compare_dates(new Date('11/14/2013 00:01'), new Date('11/14/2013 00:00')));
console.log(compare_dates(new Date('11/14/2013 00:00'), new Date('11/14/2013 00:01')));
Output :
"Date1 = Date2"
"Date1 > Date2"
"Date2 > Date1"

6. Write a JavaScript function to add specified minutes to a Date object.
Test Data :
console.log(add_minutes(new Date(2014,10,2), 30).toString());
Output :
"Sun Nov 02 2014 00:30:00 GMT+0530 (India Standard Time)"

7. Write a JavaScript function to test whether a date is a weekend.
Note : Use standard Saturday/Sunday definition of a weekend.
Test Data :
console.log(is_weekend('Nov 15, 2014'));
console.log(is_weekend('Nov 16, 2014'));
console.log(is_weekend('Nov 17, 2014'));
Output :
"weekend"

"weekend"
undefined

8. Write a JavaScript function to get difference between two dates in days.
Test Data :
console.log(date_diff_indays('04/02/2014', '11/04/2014'));
console.log(date_diff_indays('12/02/2014', '11/04/2014'));
Output :
216
-28

9. Write a JavaScript function to get the last day of a month.
Test Data :
console.log(lastday(2014,0));
console.log(lastday(2014,1));
console.log(lastday(2014,11));
Output :
31
28
31

10. Write a JavaScript function to calculate 'yesterday day'.
Test Data :
console.log(yesterday('Nov 15, 2014'));
console.log(yesterday('Nov 16, 2015'));
console.log(yesterday('Nov 17, 2016'));
Output :
"Fri Nov 14 2014 00:00:00 GMT+0530 (India Standard Time)"
"Sun Nov 15 2015 00:00:00 GMT+0530 (India Standard Time)"
"Wed Nov 16 2016 00:00:00 GMT+0530 (India Standard Time)"

11. Write a JavaScript function to get the maximum date from an array of dates.
Test Data :
console.log(max_date(['2015/02/01', '2015/02/02', '2015/01/03']));
Output :
"2015/02/02"

12. Write a JavaScript function to get the minimum date from an array of dates.
Test Data :
console.log(min_date(['2015/02/01', '2015/02/02', '2015/01/03']));
Output :
"2015/01/03"

13. Write a JavaScript function that will return the number of minutes in hours and minutes.
Test Data :
console.log(timeConvert(200));
Output :
"200 minutes = 3 hour(s) and 20 minute(s)."

14. Write a JavaScript function to get the amount of days of a year.
Test Data :
console.log(days_of_a_year(2015));
365
console.log(days_of_a_year(2016));
366

15. Write a JavaScript function to get the quarter (1 to 4) of the year.
Test Data :
console.log(quarter_of_the_year(new Date(2015, 1, 21)));
2
console.log(quarter_of_the_year(new Date(2015, 10, 18)));
4

16. Write a JavaScript function to count the number of days passed since beginning of the year.
Test Data :
console.log(days_passed(new Date(2015, 0, 15)));
15
console.log(days_passed(new Date(2015, 11, 14)));
348

17. Write a JavaScript function to convert a Unix timestamp to time.
Test Data :
console.log(days_passed(new Date(2015, 0, 15)));
15
console.log(days_passed(new Date(2015, 11, 14)));
348

18. Write a JavaScript program to calculate age.
Test Data :
console.log(calculate_age(new Date(1982, 11, 4)));
32
console.log(calculate_age(new Date(1962, 1, 1)));
53

19. Write a JavaScript function to get the day of the month, 2 digits with leading zeros.
Test Data :
d= new Date(2015, 10, 1);
console.log(day_of_the_month(d));
"01"

20. Write a JavaScript function to get a textual representation of a day (three letters, Mon through Sun).
Test Data :
dt = new Date(2015, 10, 1);
console.log(short_Days(dt));
"Sun"

21. Write a JavaScript function to get a full textual representation of the day of the week (Sunday through Saturday).
Test Data :
dt = new Date(2015, 10, 1);
console.log(long_Days(dt));
"Sunday"

22. Write a JavaScript function to get ISO-8601 numeric representation of the day of the week (1 (for Monday) to 7 (for Sunday)).
Test Data :
dt = new Date(2015, 10, 1);
console.log(ISO_numeric_date(dt));
7

23. Write a JavaScript function to get English ordinal suffix for the day of the month, 2 characters (st, nd, rd or th.).
Test Data :
dt = new Date(2015, 10, 1);
console.log(english_ordinal_suffix(dt));
"1st"

24. Write a JavaScript function to get ISO-8601 week number of year, weeks starting on Monday.
Example : 42 (the 42nd week in the year)
Test Data :
dt = new Date(2015, 10, 1);
console.log(ISO8601_week_no(dt));
44

25. Write a JavaScript function to get a full textual representation of a month, such as January or June.
Test Data :
dt = new Date(2015, 10, 1);
console.log(full_month(dt));
"November"

26. Write a JavaScript function to get a numeric representation of a month, with leading zeros (01 through 12).
Test Data :
dt = new Date(2015, 10, 1);
console.log(numeric_month(dt));
"11"

27. Write a JavaScript function to get a short textual representation of a month, three letters (Jan through Dec).
Test Data :
dt = new Date(2015, 10, 1);
console.log(short_months(dt));
"Nov"

28. Write a JavaScript function to get a full numeric representation of a year (4 digits).
Test Data :
dt = new Date(2015, 10, 1);
console.log(full_year(dt));
2015

29. Write a JavaScript function to get a two digit representation of a year.
Examples : 79 or 04
Test Data :
dt = new Date(1989, 10, 1);
console.log(sort_year(dt));
"89"

30. Write a JavaScript function to get lowercase Ante meridiem and Post meridiem.

31. Write a JavaScript function to get uppercase Ante meridiem and Post meridiem.

32. Write a JavaScript function to swatch Internet time (000 through 999).
Test Data :

dt = new Date(1989, 10, 1);
console.log(internet_time(dt));
812

33. Write a JavaScript function to get 12-hour format of an hour with leading zeros.
Test Data :
dt = new Date(1989, 10, 1);
console.log(hours_with_zeroes(dt));
"12"

34. Write a JavaScript function to get 24-hour format of an hour without leading zeros.
Test Data :
dt = new Date(1989, 10, 1);
console.log(hours_without_zeroes(dt));
0

35. Write a JavaScript function to get minutes with leading zeros (00 to 59).
Test Data :
dt = new Date(1989, 10, 1);
console.log(minutes_with_leading_zeros(dt));
"00"

36. Write a JavaScript function to get seconds with leading zeros (00 through 59).
Test Data :
dt = new Date(1989, 10, 1);
console.log(seconds_with_leading_zeros(dt));
"00"

37. Write a JavaScript function to get Timezone.
Test Data :
dt = new Date();
console.log(seconds_with_leading_zeros(dt));
"India Standard Time"

38. Write a JavaScript function to find whether or not the date is in daylights savings time.
Test Data :
dt = new Date();

console.log(daylights_savings(dt));
1

39. Write a JavaScript function to get difference to Greenwich time (GMT) in hours.
Test Data :
dt = new Date();
console.log(diff_to_GMT(dt));
"+05.500"

40. Write a JavaScript function to get timezone offset in seconds.
Note : The offset for timezones west of UTC is always negative, and for those east of UTC is always positive.
Test Data :
dt = new Date();
console.log(timezone_offset_in_seconds(dt));
19800

41. Write a JavaScript function to add specified years to a date.
Test Data :
dt = new Date(2014,10,2);
console.log(add_years(dt, 10).toString());
Output :
"Sat Nov 02 2024 00:00:00 GMT+0530 (India Standard Time)"

42. Write a JavaScript function to add specified weeks to a date.
Test Data :
dt = new Date(2014,10,2);
console.log(add_weeks(dt, 10).toString());
Output :
"Sun Jan 11 2015 00:00:00 GMT+0530 (India Standard Time)"

43. Write a JavaScript function to add specified months to a date.
Test Data :
dt = new Date(2014,10,2);
console.log(add_months(dt, 10).toString());
Output :
"Wed Sep 02 2015 00:00:00 GMT+0530 (India Standard Time)"

44. Write a JavaScript function to get time differences in minutes between two dates.
Test Data :
dt1 = new Date("October 13, 2014 11:11:00");

dt2 = new Date("October 13, 2014 11:13:00");
console.log(diff_minutes(dt1, dt2));
2

45. Write a JavaScript function to get time differences in hours between two dates.
Test Data :
dt1 = new Date("October 13, 2014 08:11:00");
dt2 = new Date("October 13, 2014 11:13:00");
console.log(diff_hours(dt1, dt2));
3

46. Write a JavaScript function to get time differences in days between two dates.
Test Data :
dt1 = new Date("October 13, 2014 08:11:00");
dt2 = new Date("October 19, 2014 11:13:00");
console.log(diff_days(dt1, dt2));
6

47. Write a JavaScript function to get time differences in weeks between two dates.
Test Data :
dt1 = new Date("June 13, 2014 08:11:00");
dt2 = new Date("October 19, 2014 11:13:00");
console.log(diff_weeks(dt1, dt2));
18

48. Write a JavaScript function to get time differences in months between two dates.
Test Data :
dt1 = new Date("June 13, 2014 08:11:00");
dt2 = new Date("October 19, 2014 11:13:00");
console.log(diff_months(dt1, dt2));
5

49. Write a JavaScript function to get time differences in years between two dates.
Test Data :
dt1 = new Date("June 13, 2014 08:11:00");
dt2 = new Date("October 19, 2017 11:13:00");

console.log(diff_years(dt1, dt2));
3

50. Write a JavaScript function to get the week start date.

51. Write a JavaScript function to get the week end date.

52. Write a JavaScript function to get the month start date.

53. Write a JavaScript function to get the month end date.

JavaScript String

1. Write a JavaScript function to check whether an `input` is a string or not.
Test Data :
console.log(is_string('w3resource'));
true
console.log(is_string([1, 2, 4, 0]));
false

2. Write a JavaScript function to check whether a string is blank or not.
Test Data :
console.log(is_Blank(''));
console.log(is_Blank('abc'));
true
false

3. Write a JavaScript function to split a string and convert it into an array of words.
Test Data :
console.log(string_to_array("Robin Singh"));
["Robin", "Singh"]

4. Write a JavaScript function to remove specified number of characters from a string.
Test Data :
console.log(truncate_string("Robin Singh",4));

"Robi"
.
5. Write a JavaScript function to convert a string in abbreviated form.
Test Data :
console.log(abbrev_name("Robin Singh"));
"Robin S."
.
6. Write a JavaScript function to hide email addresses to protect from unauthorized user.
Test Data :
console.log(protect_email("robin_singh@example.com"));
"robin...@example.com"
.
7. Write a JavaScript function to parameterize a string.
Test Data :
console.log(string_parameterize("Robin Singh from USA."));
"robin-singh-from-usa"
.
8. Write a JavaScript function to capitalize the first letter of a string.
Test Data :
console.log(capitalize('js string exercises'));
"Js string exercises"
.
9. Write a JavaScript function to capitalize the first letter of each word in a string.
Test Data :
console.log(capitalize_Words('js string exercises'));
"Js String Exercises"
.
10. Write a JavaScript function that takes a string which has lower and upper case letters as a parameter and converts upper case letters to lower case, and lower case letters to upper case.
Test Data :
console.log(swapcase('AaBbc'));
"aAbBC"
.
11. Write a JavaScript function to convert a string into camel case.
Test Data :
console.log(camelize("JavaScript Exercises"));
console.log(camelize("JavaScript exercises"));
console.log(camelize("JavaScriptExercises"));
"JavaScriptExercises"
"JavaScriptExercises"

"JavaScriptExercises"
.
12. Write a JavaScript function to uncamelize a string.
Test Data :
console.log(uncamelize('helloWorld'));
console.log(uncamelize('helloWorld','-'));
console.log(uncamelize('helloWorld','_'));
"hello world"
"hello-world"
"hello_world"
.
13. Write a JavaScript function to concatenates a given string n times (default is 1).
Test Data :
console.log(repeat('Ha!'));
console.log(repeat('Ha!',2));
console.log(repeat('Ha!',3));
"Ha!"
"Ha!Ha!"
"Ha!Ha!Ha!"
.
14. Write a JavaScript function to insert a string within a string at a particular position (default is 1).
Test Data :
console.log(insert('We are doing some exercises.'));
console.log(insert('We are doing some exercises.','JavaScript '));
console.log(insert('We are doing some exercises.','JavaScript ',18));
"We are doing some exercises."
"JavaScript We are doing some exercises."
"We are doing some JavaScript exercises."
.
15. Write a JavaScript function to humanized number (Formats a number to a human-readable string.) with the correct suffix such as 1st, 2nd, 3rd or 4th.
Test Data :
console.log(humanize_format());
console.log(humanize_format(1));
console.log(humanize_format(8));
console.log(humanize_format(301));
console.log(humanize_format(402));
"1st"
"8th"
"301st"

"402nd"
.
16. Write a JavaScript function to truncates a string if it is longer than the specified number of characters. Truncated strings will end with a translatable ellipsis sequence ("…") (by default) or specified characters.
Test Data :
console.log(text_truncate('We are doing JS string exercises.'))
console.log(text_truncate('We are doing JS string exercises.',19))
console.log(text_truncate('We are doing JS string exercises.',15,'!!'))
"We are doing JS string exercises."
"We are doing JS ..."
"We are doing !!"
.
17. Write a JavaScript function to chop a string into chunks of a given length.
Test Data :
console.log(string_chop('w3resource'));
console.log(string_chop('w3resource',2));
console.log(string_chop('w3resource',3));
["w3resource"]
["w3", "re", "so", "ur", "ce"]
["w3r", "eso", "urc", "e"]
.
18. Write a JavaScript function to count the occurrence of a substring in a string.
Test Data :
console.log(count("The quick brown fox jumps over the lazy dog", 'the'));
Output :
2
console.log(count("The quick brown fox jumps over the lazy dog", 'fox',false));
Output :
1
.
19. Write a JavaScript function to escape a HTML string.
Test Data :
console.log(escape_HTML('<a href="javascript-string-exercise-17.php" target="_blank">'));
Output :
"<a href="javascript-string-exercise-17.php" target="_blank">"
.
20. Write a JavaScript function that can pad (left, right) a string to get to a determined length.
Test Data :
console.log(formatted_string('0000',123,'l'));

console.log(formatted_string('00000000',123,''));
Output :
"0123"
"12300000"
.
21. Write a JavaScript function to repeat a string a specified times.
Test Data :
console.log(repeat_string('a', 4));
console.log(repeat_string('a'));
Output :
"aaaa"
"Error in string or count."
.
22. Write a JavaScript function to get a part of a string after a specified character.
Test Data :
console.log(subStrAfterChars('w3resource: JavaScript Exercises', ':','a'));
console.log(subStrAfterChars('w3resource: JavaScript Exercises', 'E','b'));
Output :
"w3resource"
"xercises"
.
23. Write a JavaScript function to strip leading and trailing spaces from a string.
Test Data :
console.log(strip('w3resource '));
console.log(strip(' w3resource'));
console.log(strip(' w3resource '));
Output :
"w3resource"
"w3resource"
"w3resource"
.
24. Write a JavaScript function to truncate a string to a certain number of words.
Test Data :
console.log(truncate('The quick brown fox jumps over the lazy dog', 4));
Output :
"The quick brown fox"
.
25. Write a JavaScript function to alphabetize a given string.
Alphabetize string : An individual string can be alphabetized. This rearranges the letters so they are sorted A to Z.
Test Data :
console.log(alphabetize_string('United States'));

Output :
"SUadeeinsttt"
.
26. Write a JavaScript function to remove the first occurrence of a given 'search string' from a string.
Test Data :
console.log(remove_first_occurrence("The quick brown fox jumps over the lazy dog", 'the'));
Output :
"The quick brown fox jumps over lazy dog"
.
27. Write a JavaScript function to convert ASCII to Hexadecimal format.
Test Data :
console.log(ascii_to_hexa('12'));
console.log(ascii_to_hexa('100'));
Output :
"3132"
"313030"
.
28. Write a JavaScript function to convert Hexadecimal to ASCII format.
Test Data :
console.log(hex_to_ascii('3132'));
console.log(hex_to_ascii('313030'));
Output :
"12"
"100"
.
29. Write a JavaScript function to find a word within a string.
Test Data :
console.log(search_word('The quick brown fox', 'fox'));
console.log(search_word('aa, bb, cc, dd, aa', 'aa'));
Output :
"'fox' was found 1 times."
"'aa' was found 2 times."
.
30. Write a JavaScript function check if a string ends with specified suffix.
Test Data :
console.log(string_endsWith('JS PHP PYTHON','PYTHON'));
true
console.log(string_endsWith('JS PHP PYTHON',''));
false
.

31. Write a JavaScript function to escapes special characters (&, <, >, ', ") for use in HTML.
Test Data :
console.log(escape_html('PHP & MySQL'));
"PHP & MySQL"
console.log(escape_html('3 > 2'));
"3 > 2"
.
32. Write a JavaScript function to remove non-printable ASCII chars.
Test Data :
console.log(remove_non_ascii('äÄçÇéÉêPHP-MySQLöÖÐþúÚ'));
"PHP-MySQL"
.
33. Write a JavaScript function to remove non-word characters.
Test Data :
console.log(remove_non_word('PHP ~!@#$%^&*()+`-={}[]|\\:";\'/?><., MySQL'));
"PHP - MySQL"
.
34. Write a JavaScript function to convert a string to title case.
Test Data :
console.log(sentenceCase('PHP exercises. python exercises.'));
"Php Exercises. Python Exercises."
.
35. Write a JavaScript function to remove HTML/XML tags from string.
Test Data :
console.log(strip_html_tags(' PHP Exercises '));
"PHP Exercises"
.
36. Write a JavaScript function to create a Zerofilled value with optional +, - sign.
Test Data :
console.log(zeroFill(120, 5, '-'));
"+00120"
console.log(zeroFill(29, 4));
"0029"
.
37. Write a JavaScript function to test case insensitive (except special Unicode characters) string comparison.
Test Data :
console.log(compare_strings('abcd', 'AbcD'));
true
console.log(compare_strings('ABCD', 'Abce'));
false

.
38. Write a JavaScript function to create a case-insensitive search.
Test Data :
console.log(case_insensitive_search('JavaScript Exercises', 'exercises'));
"Matched"
console.log(case_insensitive_search('JavaScript Exercises', 'Exercises'));
"Matched"
console.log(case_insensitive_search('JavaScript Exercises', 'Exercisess'));
"Not Matched"

JavaScript Validation with regular

1. Write a JavaScript program to test the first character of a string is uppercase or not.

2. Write a JavaScript program to check a credit card number.

3. Write a pattern that matches e-mail addresses.
The personal information part contains the following ASCII characters.

- Uppercase (A-Z) and lowercase (a-z) English letters.
- Digits (0-9).
- Characters ! # $ % & ' * + - / = ? ^ _ ` { | } ~
- Character . (period, dot or fullstop) provided that it is not the first or last character and it will not come one after the other.

4. Write a JavaScript program to search a date within a string.

5. Write a JavaScript program that work as a trim function (string) using regular expression.

6. Write a JavaScript program to count number of words in string.
Note :
- Remove white-space from start and end position.
- Convert 2 or more spaces to 1.
- Exclude newline with a start spacing.

7. Write a JavaScript function to check whether a given value is IP value or not.

8. Write a JavaScript function to count the number of vowels in a given string.
Test Data :
console.log(alphabetize_string('United States'));
Output :
"SUadeeinsttt"

9. Write a JavaScript function to check whether a given value is an valid url or not.

10. Write a JavaScript function to check whether a given value is alpha numeric or not.

11. Write a JavaScript function to check whether a given value is time string or not.

12. Write a JavaScript function to check whether a given value is US zip code or not.

13. Write a JavaScript function to check whether a given value is UK Post Code or not.

14. Write a JavaScript function to check whether a given value is Canada Post Code or not.

15. Write a JavaScript function to check whether a given value is a social security number or not.

16. Write a JavaScript function to check whether a given value is hexadecimal value or not.

17. Write a JavaScript function to check whether a given value is hexcolor value or not.

18. Write a JavaScript function to check whether a given value represents a domain or not.

19. Write a JavaScript function to check whether a given value is html or not.

20. Write a JavaScript function to check a given value contains alpha, dash and underscore.

21. Write a JavaScript function to print an integer with commas as thousands separators.
Test Data :
console.log(thousands_separators(1000));
"1,000"
console.log(thousands_separators(10000.23));
"10,000.23"
console.log(thousands_separators(100000));
"100,000"

JavaScript DOM

1. Here is a sample html file with a submit button. Now modify the style of the paragraph text through javascript code.
Sample HTML file :

```
<!DOCTYPE html>
<html><br><head>
<meta charset=utf-8 />
<title>JS DOM paragraph style</title>
</head>
<body>
<p id ='text'>JavaScript Exercises - w3resource</p>
<div>
<button id="jsstyle"
onclick="js_style()">Style</button>
</div>
</body>
</html>
```

Clicking on the button the font, font size, and color of the paragraph text will be changed.

2. Write a JavaScript function to get the values of First and Last name of the following form.
Sample HTML file :
view plaincopy to clipboardprint?

```
<!DOCTYPE html>
<html><head>
<meta charset=utf-8 />
<title>Return first and last name from a form - w3resource</title>
```

```
</head><body>
<form id="form1" onsubmit="getFormvalue()">
First name: <input type="text" name="fname" value="David"><br>
Last name: <input type="text" name="lname" value="Beckham"><br>
<input type="submit" value="Submit">
</form>
</body>
</html>
```

3. Write a JavaScript program to set the background color of a paragraph.

4. Here is a sample html file with a submit button. Write a JavaScript function to get the value of the href, hreflang, rel, target, and type attributes of the specified link.

view plaincopy to clipboardprint?

```
<!DOCTYPE html>
<html><head>
<meta charset=utf-8 />
</head>
<body>
<p><a id="w3r" type="text/html" hreflang="en-
us" rel="nofollow" target="_self" href="http://www.w3resource.com/">w3reso
urce</a></p>
<button onclick="getAttributes()">Click here to get  attributes value</but
ton>
</body></html>
```

5. Write a JavaScript function to add rows to a table.
Sample HTML file :

view plaincopy to clipboardprint?

```
<!DOCTYPE html>
<html><head><br><meta charset=utf-8 />
<title>Insert row in a table - w3resource</title>
</head><body>
<table id="sampleTable" border="1">
<tr><td>Row1 cell1</td>
<td>Row1 cell2</td></tr>
<tr><td>Row2 cell1</td>
<td>Row2 cell2</td></tr>
</table><br>
<input type="button" onclick="insert_Row()" value="Insert row">
</body></html>
```

6. Write a JavaScript function that accept row, column, (to identify a particular cell) and a string to update the content of that cell.
Sample HTML file :


```
<!DOCTYPE html>
<html><head><br><meta charset=utf-8 />
<title>Change the content of a cell</title>
</head><body>
<table id="myTable" border="1">
<tr><td>Row1 cell1</td>
<td>Row1 cell2</td></tr>
<tr><td>Row2 cell1</td>
<td>Row2 cell2</td></tr>
<tr><td>Row3 cell1</td>
<td>Row3 cell2</td></tr>
</table><form>
<input type="button" onclick="changeContent()" value="Change content">
</form></body></html>
```

7. Write a JavaScript function that creates a table, accept row, column numbers from the user, and input row-column number as content (e.g. Row-0 Column-0) of a cell.
Sample HTML file :


```
<!DOCTYPE html>
<html>
<head>
<meta charset=utf-8 />
<title>Change the content of a cell</title>
<style type="text/css">
body {margin: 30px;}
</style>
</head><body>
<table id="myTable" border="1">
</table><form>
<input type="button" onclick="createTable()" value="Create the table">
</form></body></html>
```

8. Write a JavaScript program to remove items from a dropdown list.
Sample HTML file :

```
<!DOCTYPE html>
<html><head>
<meta charset=utf-8 />
<title>Remove items from a dropdown list</title>
</head><body><form>
<select id="colorSelect">
<option>Red</option>
<option>Green</option>
<option>White</option>
<option>Black</option>
</select>
<input type="button" onclick="removecolor()" value="Select and Remove"><br
></form></body></html>
```

9. Write a JavaScript program to count and display the items of a dropdown list, in an alert window.
Sample HTML file :

view plaincopy to clipboardprint?

```
<!DOCTYPE html>
<html><head>
<meta charset=utf-8 />
<style type="text/css">
body {margin: 30px;}
</style>
<title>Count and display items of a dropdown list - w3resource</title>
</head><body><form>
Select your favorite Color :
<select id="mySelect">
<option>Red</option>
<option>Green</option>
<option>Blue</option>
<option>White</option>
</select>
<input type="button" onclick="getOptions()" value="Count and Output all it
ems">
</form></body></html>
```

10. Write a JavaScript program to calculate the volume of a sphere.
Sample Output of the form :

Input radius value and get the volume of a sphere.

Radius

Volume

0.0000

Calculate

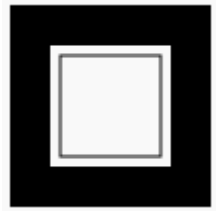
11. Write a JavaScript program to display a random image (clicking on a button) from the following list.
Sample Image information :
"http://farm4.staticflickr.com/3691/11268502654_f28f05966c_m.jpg", width: "240", height: "160"
"http://farm1.staticflickr.com/33/45336904_1aef569b30_n.jpg", width: "320", height: "195"
"http://farm6.staticflickr.com/5211/5384592886_80a512e2c9.jpg", width: "500", height: "343"

12. Write a JavaScript program to highlight the bold words of the following paragraph, on mouse over a certain link.
Sample link and text :
[On mouse over here bold words of the following paragraph will be highlighted]
We have just started **this** section for the users (**beginner** to intermediate) who **want** to work with **various** JavaScript **problems** and write scripts online to **test** their JavaScript **skill**.

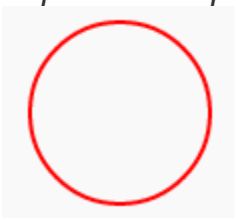
13. Write a JavaScript program to get the width and height of the window (any time the window is resized).

JavaScript drawing

1. Write a JavaScript program to draw the following rectangular shape.
Expected Output :



2. Write a JavaScript program to draw a circle.
Expected Output :

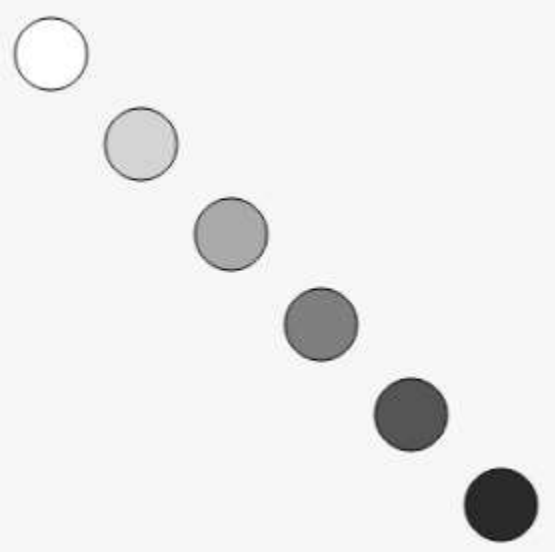


3. Write a JavaScript program to draw two intersecting rectangles, one of which has alpha transparency.
Expected Output :

4. Write a JavaScript program to draw the following right-angled triangle.
Expected Output :

5. Write a JavaScript program to draw the following diagram [use moveto() function].
Expected Output :

6. Write a JavaScript program to draw the following diagram [diagonal, white to black circles].
Expected Output :



JavaScript Object

1. Write a JavaScript program to list the properties of a JavaScript object.
Sample object :
var student = {
name : "David Rayy",
sclass : "VI",
rollno : 12 };
Sample Output : name,sclass,rollno

2. Write a JavaScript program to delete the rollno property from the following object. Also print the object before or after deleting the property.
Sample object :
var student = {

name : "David Rayy",
sclass : "VI",
rollno : 12 };

3. Write a JavaScript program to get the length of an JavaScript object.
Sample object :
var student = {
name : "David Rayy",
sclass : "VI",
rollno : 12 };

4. Write a JavaScript program to display the reading status (i.e. display book name, author name and reading status) of the following books.

```
var library = [

   {

       author: 'Bill Gates',

       title: 'The Road Ahead',

       readingStatus: true

   },

   {

       author: 'Steve Jobs',

       title: 'Walter Isaacson',

       readingStatus: true

   },

   {

       author: 'Suzanne Collins',

       title:  'Mockingjay: The Final Book of The Hunger Games',

       readingStatus: false

   }];
```

5. Write a JavaScript program to get the volume of a Cylinder with four decimal places using object classes.
Volume of a cylinder : $V = \pi r^2 h$
where r is the radius and h is the height of the cylinder.

6. Write a Bubble Sort algorithm in JavaScript.
Note : Bubble sort is a simple sorting algorithm that works by repeatedly stepping

through the list to be sorted,
Sample Data : [6,4,0, 3,-2,1]
Expected Output : [-2, 0, 1, 3, 4, 6]

7. Write a JavaScript program which returns a subset of a string.
Sample Data : dog
Expected Output : ["d", "do", "dog", "o", "og", "g"]

8. Write a JavaScript program to create a Clock.
Note : The output will come every second.
Expected Console Output :
"14:37:42"
"14:37:43"
"14:37:44"
"14:37:45"
"14:37:46"
"14:37:47"

9. Write a JavaScript program to calculate the area and perimeter of a circle.
Note : Create two methods to calculate the area and perimeter. The radius of the circle will be supplied by the user.

10. Write a JavaScript program to sort an array of JavaScript objects.
Sample Object :

```
var library = [
   {
       title:  'The Road Ahead',
       author: 'Bill Gates',
       libraryID: 1254
   },
   {
       title: 'Walter Isaacson',
       author: 'Steve Jobs',
       libraryID: 4264
   },
   {
       title: 'Mockingjay: The Final Book of The Hunger Games',
       author: 'Suzanne Collins',
```

```
       libraryID: 3245
   }];
```

Expected Output :

```
[[object Object] {
  author: "Walter Isaacson",
  libraryID: 4264,
  title: "Steve Jobs"
}, [object Object] {
  author: "Suzanne Collins",
  libraryID: 3245,
  title: "Mockingjay: The Final Book of The Hunger Games"
}, [object Object] {
  author: "The Road Ahead",
  libraryID: 1254,
  title: "Bill Gates"
}]
```

11. Write a JavaScript function to print all the methods in an JavaScript object.
Test Data :
console.log(all_properties(Array));
["length", "name", "arguments", "caller", "prototype", "isArray", "observe", "unobserve"]

12. Write a JavaScript function to parse an URL.
Sample Object :

```
console.log(parse_URL('https://github.com/pubnub/python/search?utf8=%E2%9C%93&q=python'));
```

Expected Output :

```
[object Object] {
  file: "search",
  hash: "",
  host: "github.com",
  params: [object Object] {
    q: "python",
    utf8: "%E2%9C%93"
  },
  path: "/pubnub/python/search",
```

```
  port: "",
  protocol: "https",
  query: "?utf8=%E2%9C%93&q=python",
  relative: "/pubnub/python/search?utf8=%E2%9C%93&q=python",
  segments: ["pubnub", "python", "search"],
  source: "https://github.com/pubnub/python/search?utf8=%E2%9C%93&q=python"
}
```

JavaScript validation

1. Write a JavaScript function to validate whether a given value type is boolean or not.

2. Write a JavaScript function to validate whether a given value type is error or not.

3. Write a JavaScript function to validate whether a given value type is NaN or not.

4. Write a JavaScript function to validate whether a given value type is null or not.

5. Write a JavaScript function to validate whether a given value is number or not.

6. Write a JavaScript function to validate whether a given value is object or not.

7. Write a JavaScript function to validate whether a given value type is pure json object or not.

8. Write a JavaScript function to validate whether a given value is RegExp or not.

9. Write a JavaScript function to validate whether a given value type is char or not.

10. Write a JavaScript function to check whether given value types are same or not.

Imp Questions:

The Movie Database

It's like IMDB, but much much smaller!

- Create an object to store the following information about your favorite movie: `title` (a string), `duration` (a number), and `stars` (an array of strings).
- Create a function to print out the movie information like so: "Puff the Magic Dragon lasts for 30 minutes. Stars: Puff, Jackie, Living Sneezes."

The Reading List

Keep track of which books you read and which books you want to read!

- Create an array of objects, where each object describes a book and has properties for the `title` (a string), `author` (a string), and `alreadyRead` (a boolean indicating if you read it yet).
- Iterate through the array of books. For each book, log the book title and book author like so: "The Hobbit by J.R.R. Tolkien".
- Now use an if/else statement to change the output depending on whether you read it yet or not. If you read it, log a string like 'You already read "The Hobbit" by J.R.R. Tolkien', and if not, log a string like 'You still need to read "The Lord of the Rings" by J.R.R. Tolkien.'

The Recipe Card

Never forget another recipe!

- Create an object to hold information on your favorite recipe. It should have properties for `title` (a string), `servings` (a number), and `ingredients` (an array of strings).
- On separate lines (one `console.log` statement for each), log the recipe information so it looks like:
- Mole
- Serves: 2
- Ingredients:
- cinnamon
- cumin
- cocoa

The Fortune Teller

Why pay a fortune teller when you can just program your fortune yourself?

- Write a function named `tellFortune` that:
  - takes 4 arguments: number of children, partner's name, geographic location, job title.
  - outputs your fortune to the screen like so: "You will be a X in Y, and married to Z with N kids."
- Call that function 3 times with 3 different values for the arguments.

The Age Calculator

Forgot how old you are? Calculate it!

- Write a function named `calculateAge` that:
  - takes 2 arguments: birth year, current year.
  - calculates the 2 possible ages based on those years.
  - outputs the result to the screen like so: "You are either NN or NN"
- Call the function three times with different sets of values.
- **Bonus**: Figure out how to get the current year in JavaScript instead of passing it in.

The Lifetime Supply Calculator

Ever wonder how much a "lifetime supply" of your favorite snack is? Wonder no more!

- Write a function named `calculateSupply` that:
  - takes 2 arguments: age, amount per day.
  - calculates the amount consumed for rest of the life (based on a constant max age).
  - outputs the result to the screen like so: "You will need NN to last you until the ripe old age of X"
- Call that function three times, passing in different values each time.
- **Bonus**: Accept floating point values for amount per day, and round the result to a round number.

The Geometrizer

Create 2 functions that calculate properties of a circle, using the definitions here.

Create a function called `calcCircumfrence`:

- Pass the radius to the function.
- Calculate the circumference based on the radius, and output "The circumference is NN".

Create a function called `calcArea`:

- Pass the radius to the function.
- Calculate the area based on the radius, and output "The area is NN".

The Temperature Converter

It's hot out! Let's make a converter based on the steps here.

Create a function called `celsiusToFahrenheit`:

- Store a celsius temperature into a variable.
- Convert it to fahrenheit and output "NN°C is NN°F".

Create a function called `fahrenheitToCelsius`:

- Now store a fahrenheit temperature into a variable.
- Convert it to celsius and output "NN°F is NN°C.

Other questions

1) Create an HTML page. Add JavaScript that includes 2 prompts for integer values from the user. Add JavaScript to add these 2 values together and display the result in an alert box.
2) Write a JavaScript function that returns a passed string with letters in alphabetical order.
3) Write a JavaScript function to extract unique characters from a string “thequickbrownfoxjumpsoverthelazydog”.
4) Create an HTML page. Add a JavaScript function to convert local time and date to UTC and display the result. Add a button to your page that calls this function. Add JavaScript to create a timer that displays the current date and time after a specified amount of time delay. Use a button (or other event) to call the timer function
5) Create a basic page in html that consists of one image. Using the onClick event handler, when the user clicks on the image, change it to a unique image

6) Reconfigure the previous question to execute using a function. Specifically, when the user clicks on the button, the onClick event handler calls a function. The function then executes the statement that changes the background color to blue.

7) Create an HTML page that includes 2 paragraphs of text. Add JavaScript to insert a new paragraph in between the other 2. Add JavaScript to use an event to change the style of the new paragraph using either the style property or the className property. Add JavaScript to use an event to change the style of the new paragraph back to the original.

8) Create a slide show of 5 images.

9) Use previous question images and add two image button (back and forward). Back button allows the user to move backward through the slide show one slide at a time. When the user reaches the end (or beginning when clicking on the back button) of the slide show, the slide show should not wrap around to the beginning (or end) and forward button should show next image. When at the beginning of the slide show, only the forward image button should be visible. Conversely, when at the end, only the back image button should be visible.

10) Create a basic page. Using the setTimeout() method, create an animation on the page. Allow the user to stop the animation by placing the cursor on any marble. Allow the user to restart the animation by removing the cursor from that marble.

11) Create a basic page in html. Then, display the following items in the page using only one or more document.write() statements:

1. Information about the web browser that the user is viewing this page with.
2. The height and width of the user's monitor, i.e. the resolution
3. The date that the page was created or last modified.

12) Create a basic page in html that displays 2 images. When the user places a cursor over any image, replace the image with a different image. When the user removes the cursor from the image and return it to its original state.

13) Create a signup form with validation before submit

14) Create a basic page in html that displays text and an image. Track how many times a user has visited your page by storing this information in a cookie. Display this information to the user, e.g. You have visited x number of times! The current visit should be included in this number. Display the

date and time of the user's last visit, e.g. "You lasted visted on..." If this is the user's first visit, display e.g. "You have never visited before" instead.

15) Create a "Mad-libs" game using JavaScript.

Create a blank page.
Using a prompt box, prompt the user to supply his or her name.
Then, using 5 additional prompt boxes, prompt the user to supply 5 words.
Save each word in a separate variable.
Then, using document.write() statements, use the information stored in variables to display a "Mad-libs" type of story, i.e. create a few paragraphs of information in story format.
Also, prompt the user for a color (i.e. one of the 16 named colors or a hex value).
Store this color in a variable.
In the "mad libs" story, highlight, using the color supplied by the user, each of the words that you previously collected. Do this by surrounding the words with tags and using an inline style.