JS Assignment Loops

Level =1

//Use for loop & while-loop to iterate from 0 to 100 and print only even numbers

// loop

// console.log("even the number using loop :");

// for(let i=0;i<=100 ;i++){

//     if(i % 2 == 0){

//         console.log(i)

//     }

// }

//while-loop

// console.log("Even numbers using while loop:");

//  let i=0;

//  while(i<= 100) {

//     if(i % 2 == 0){

//         console.log(i)

//     }

//     i++;

// }

// o/p==>

//     Even numbers using while loop: Loops.js:17:9

// 0 Loops.js:21:17

// 2 Loops.js:21:17

// 4 Loops.js:21:17

// 6 Loops.js:21:17

// 8 Loops.js:21:17

// 10 Loops.js:21:17

// 12 Loops.js:21:17

// 14 Loops.js:21:17

// 16 Loops.js:21:17

// 18 Loops.js:21:17

// 20 Loops.js:21:17

// 22 Loops.js:21:17

// 24 Loops.js:21:17

// 26 Loops.js:21:17

// 28 Loops.js:21:17

// // 30 Loops.js:21:17

// // 32 Loops.js:21:17

// // 34 Loops.js:21:17

// // 36 Loops.js:21:17

// // 38 Loops.js:21:17

// // 40 Loops.js:21:17

// // 42 Loops.js:21:17

// // 44 Loops.js:21:17

// // 46 Loops.js:21:17

// // 48 Loops.js:21:17

// // 50 Loops.js:21:17

// // 52 Loops.js:21:17

// // 54 Loops.js:21:17

// // 56 Loops.js:21:17

// // 58 Loops.js:21:17

// // 60 Loops.js:21:17

// // 62 Loops.js:21:17

// // 64 Loops.js:21:17

// // 66 Loops.js:21:17

// // 68 Loops.js:21:17

// // 70 Loops.js:21:17

// // 72 Loops.js:21:17

// // 74 Loops.js:21:17

// // 76 Loops.js:21:17

// // 78 Loops.js:21:17

// // 80 Loops.js:21:17

// // 82 Loops.js:21:17

// // 84 Loops.js:21:17

// // 86 Loops.js:21:17

// // 88 Loops.js:21:17

// // 90 Loops.js:21:17

// // 92 Loops.js:21:17

// // 94 Loops.js:21:17

// // 96 Loops.js:21:17

// // 98 Loops.js:21:17

// // 100

// // Use for loop & while-loop to iterate from 0 to 100 and print only odd numbers

// console.log("Odd numbers using  loop :")

// for(let i = 0; i<100;i++){

//     if(i % 2 !== 0){

//         console.log(i)

//     }

// }

// o/p =>

//     Odd numbers using  loop : Loops.js:83:9

// 1 Loops.js:86:17

// 3 Loops.js:86:17

// 5 Loops.js:86:17

// 7 Loops.js:86:17

// 9 Loops.js:86:17

// 11 Loops.js:86:17

// 13 Loops.js:86:17

// 15 Loops.js:86:17

// 17 Loops.js:86:17

// 19 Loops.js:86:17

// 21 Loops.js:86:17

// 23 Loops.js:86:17

// 25 Loops.js:86:17

// 27 Loops.js:86:17

// 29 Loops.js:86:17

// 31 Loops.js:86:17

// 33 Loops.js:86:17

// 35 Loops.js:86:17

// 37 Loops.js:86:17

// 39 Loops.js:86:17

// 41 Loops.js:86:17

// 43 Loops.js:86:17

// 45 Loops.js:86:17

// 47 Loops.js:86:17

// 49 Loops.js:86:17

// 51 Loops.js:86:17

// 53 Loops.js:86:17

// 55 Loops.js:86:17

// 57 Loops.js:86:17

// 59 Loops.js:86:17

// 61 Loops.js:86:17

// 63 Loops.js:86:17

// 65 Loops.js:86:17

// 67 Loops.js:86:17

// 69 Loops.js:86:17

// 71 Loops.js:86:17

// 73 Loops.js:86:17

// 75 Loops.js:86:17

// 77 Loops.js:86:17

// 79 Loops.js:86:17

// 81 Loops.js:86:17

// 83 Loops.js:86:17

// 85 Loops.js:86:17

// 87 Loops.js:86:17

// 89 Loops.js:86:17

// 91 Loops.js:86:17

// 93 Loops.js:86:17

// 95 Loops.js:86:17

// 97 Loops.js:86:17

// 99

//while loop

// console.log("Odd numbers using while loop:");

// let i = 0;

// while (i <= 100) {

//     if (i % 2 !== 0) {

//         console.log(i);

//     }

//     i++;

// }

// Use for loop & while-loop to iterate from 0 to 100 and print only prime numbers

//for loop

//  console.log("prime number using for loop :")

//  for(let i = 2; i < 100; i++){

//     let isprime = true;

//     for(let j = 2; j < Math.sqrt(i); j++){

//         if(i % j == 0) {

//             isprime = false;

//             break;

//         }

//     }

//     if(isprime) {

//         console.log(i)

//     }

//  }

//  o/p ==>

//     prime number using for loop : Loops.js:160:10

// 2 Loops.js:171:17

// 3 Loops.js:171:17

// 4 Loops.js:171:17

// 5 Loops.js:171:17

// 7 Loops.js:171:17

// 9 Loops.js:171:17

// 11 Loops.js:171:17

// 13 Loops.js:171:17

// 17 Loops.js:171:17

// 19 Loops.js:171:17

// 23 Loops.js:171:17

// 25 Loops.js:171:17

// 29 Loops.js:171:17

// 31 Loops.js:171:17

// 37 Loops.js:171:17

// 41 Loops.js:171:17

// 43 Loops.js:171:17

// 47 Loops.js:171:17

// 49 Loops.js:171:17

// 53 Loops.js:171:17

// 59 Loops.js:171:17

// 61 Loops.js:171:17

// 67 Loops.js:171:17

// 71 Loops.js:171:17

// 73 Loops.js:171:17

// 79 Loops.js:171:17

// 83 Loops.js:171:17

// 89 Loops.js:171:17

// 97

//while loop

// console.log("prime number using  while loop :")

// let i = 2;

// while (i <= 100) {

//     let isPrime = true;

//     let j = 2;

//     while (j <= Math.sqrt(i)) {

//         if (i % j === 0) {

//             isPrime = false;

//             break;

//         }

//         j++;

//     }

//     if (isPrime) {

//         console.log(i);

//     }

//     i++;

// }

// o/p ==>

  //  prime number using  while loop : Loops.js:209:9

// 2 Loops.js:222:17

// 3 Loops.js:222:17

// 5 Loops.js:222:17

// 7 Loops.js:222:17

// 11 Loops.js:222:17

// 13 Loops.js:222:17

// 17 Loops.js:222:17

// 19 Loops.js:222:17

// 23 Loops.js:222:17

// 29 Loops.js:222:17

// 31 Loops.js:222:17

// 37 Loops.js:222:17

// 41 Loops.js:222:17

// 43 Loops.js:222:17

// 47 Loops.js:222:17

// 53 Loops.js:222:17

// 59 Loops.js:222:17

// 61 Loops.js:222:17

// // 67 Loops.js:222:17

// 71 Loops.js:222:17

// 73 Loops.js:222:17

// 79 Loops.js:222:17

// 83 Loops.js:222:17

// 89 Loops.js:222:17

// 97

// Use for loop & while-loop  to iterate from 0 to 100 and print the sum of all numbers.

//for loop

// console.log("Sum of all numbers using for loop:");

// let sum = 0;

// for (let i = 0; i <= 100; i++) {

//     sum += i;

// }

// console.log(sum);

///while loop

// console.log("Sum of all numbers using while loop:");

// let sum = 0;

// let i = 0;

// while (i <= 100) {

//     sum += i;

//     i++;

// }

// console.log(sum);

// Use for loop & while-loop to iterate from 0 to 100 and print the sum of all evens and the sum of all odds

//foor loop

// console.log("Sum of evens and odds using for loop:");

// let sumEven = 0;

// let sumOdd = 0;

// for (let i = 0; i <= 100; i++) {

//     if (i % 2 === 0) {

//         sumEven += i;

//     } else {

//         sumOdd += i;

//     }

// }

// console.log("Sum of evens:", sumEven);

// console.log("Sum of odds:", sumOdd);

// o/p==>

//     Sum of evens and odds using for loop: Loops.js:284:9

// Sum of evens: 2550 Loops.js:294:9

// Sum of odds: 2500

// console.log("Sum of evens and odds using while loop:");

// let sumEven = 0;

// let sumOdd = 0;

// let i = 0;

// while (i <= 100) {

//     if (i % 2 === 0) {

//         sumEven += i;

//     } else {

//         sumOdd += i;

//     }

//     i++;

// }

// console.log("Sum of evens:", sumEven);

// console.log("Sum of odds:", sumOdd);

// Sum of evens and odds using while loop: Loops.js:303:9

// Sum of evens: 2550 Loops.js:315:9

// Sum of odds: 2500 Loops.js:316:9

// ​

// Write a program that calculates the factorial of a given positive integer

// function factorial(n) {

//     if (n === 0 || n === 1) {

//         return 1;

//     }

//     let result = 1;

//     for (let i = 2; i <= n; i++) {

//         result \*= i;

//     }

//     return result;

// }

// let number = 5;  // You can change this value to calculate the factorial of any positive integer

// console.log("Factorial of " + number + " is: " + factorial(number));

// o/p ==>

// Factorial of 5 is: 120

level-2

// Construct for loops that accomplish the following tasks:

// Print the numbers 0 - 20, one number per line.

// for(let i =0; i<=20; i++){

//   console.log(i)

// }

// o/p

// 0 Loops.js:355:11

// 1 Loops.js:355:11

// 2 Loops.js:355:11

// 3 Loops.js:355:11

// 4 Loops.js:355:11

// 5 Loops.js:355:11

// 6 Loops.js:355:11

// 7 Loops.js:355:11

// 8 Loops.js:355:11

// 9 Loops.js:355:11

// 10 Loops.js:355:11

// 11 Loops.js:355:11

// 12 Loops.js:355:11

// 13 Loops.js:355:11

// 14 Loops.js:355:11

// 15 Loops.js:355:11

// 16 Loops.js:355:11

// 17 Loops.js:355:11

// 18 Loops.js:355:11

// 19 Loops.js:355:11

// 20

// Print only the ODD values from 3 - 29, one number per line.

// for(let i =3; i<=29; i +=2) {

//    console.log(i)

//   }

//   o/p

//   3 Loops.js:385:12

// 5 Loops.js:385:12

// 7 Loops.js:385:12

// 9 Loops.js:385:12

// 11 Loops.js:385:12

// 13 Loops.js:385:12

// 15 Loops.js:385:12

// 17 Loops.js:385:12

// 19 Loops.js:385:12

// 21 Loops.js:385:12

// 23 Loops.js:385:12

// 25 Loops.js:385:12

// 27 Loops.js:385:12

// 29

// Print the EVEN numbers 12 down to -14 in descending order, one number per line.

//  for(let i =12; i>= -14; i-= 2) {

//     console.log(i)

//  }

//  o/p

//  12 Loops.js:408:13

// 10 Loops.js:408:13

// 8 Loops.js:408:13

// 6 Loops.js:408:13

// 4 Loops.js:408:13

// 2 Loops.js:408:13

// 0 Loops.js:408:13

// -2 Loops.js:408:13

// -4 Loops.js:408:13

// -6 Loops.js:408:13

// -8 Loops.js:408:13

// -10 Loops.js:408:13

// -12 Loops.js:408:13

// -14

// Print the numbers 50 down to 20 in descending order, but only if the numbers are multiples of 3.

// for(let i = 50; i>= 20; i--) {

//   if(i % 3 == 0){

//     console.log(i)

//   }

//  }

//  o/p ===>

//   48 Loops.js:436:13

// 45 Loops.js:436:13

// 42 Loops.js:436:13

// 39 Loops.js:436:13

// 36 Loops.js:436:13

// 33 Loops.js:436:13

// 30 Loops.js:436:13

// 27 Loops.js:436:13

// 24 Loops.js:436:13

// 21

// Initialize two variables to hold the string 'LaunchCode' and the array [1, 5, 'LC101', 'blue', 42], then construct for loops to accomplish the following tasks:

// let array = [1, 5, 'LC101', 'blue', 42];

// for (let i = 0; i < array.length; i++) {

//   console.log(array[i]);

// }

// o/p ===>

//   1 Loops.js:463:11

// 5 Loops.js:463:11

// LC101 Loops.js:463:11

// blue Loops.js:463:11

// 42

// let string = 'LaunchCode';

// for (let i = string.length - 1; i >= 0; i--) {

//   console.log(string[i]);

// }

// o/p===>

//   e Loops.js:478:11

// d Loops.js:478:11

// o Loops.js:478:11

// C Loops.js:478:11

// h Loops.js:478:11

// c Loops.js:478:11

// n Loops.js:478:11

// u Loops.js:478:11

// a Loops.js:478:11

// L

// Print each element of the array to a new line.

// let array = [1, 5, 'LC101', 'blue', 42];

// for (let i = 0; i < array.length; i++) {

//   console.log(array[i]);

// }

// Print each character of the string---in reverse order---to a new line

// let string = 'LaunchCode';

// for (let i = string.length - 1; i >= 0; i--) {

//   console.log(string[i]);

// }

// Construct a for loop that sorts the array [2, 3, 13, 18, -5, 38, -10, 11, 0, 104] into two new arrays:

// let numbers = [2, 3, 13, 18, -5, 38, -10, 11, 0, 104];

// let evens = [];

// let odds = [];

// for (let i = 0; i < numbers.length; i++) {

//   if (numbers[i] % 2 === 0) {

//     evens.push(numbers[i]);

//   } else {

//     odds.push(numbers[i]);

//   }

// }

// console.log('Evens:', evens);

// console.log('Odds:', odds);

// o/p ===>

//   Evens:

// Array(6) [ 2, 18, 38, -10, 0, 104 ]

// ​

// 0: 2

// ​

// 1: 18

// ​

// 2: 38

// ​

// 3: -10

// ​

// 4: 0

// ​

// 5: 104

// ​

// length: 6

// ​

// <prototype>: Array []

// Loops.js:529:9

// Odds:

// Array(4) [ 3, 13, -5, 11 ]

// ​

// 0: 3

// ​

// 1: 13

// ​

// 2: -5

// ​

// 3: 11

// ​

// length: 4

// ​

// <prototype>: Array []

// Define an empty evens array to hold the even numbers and an odds array for the odd numbers.

// let evens = [];

// let odds = [];

// let numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]; // ex array

// for (let i = 0; i < numbers.length; i++) {

//   if (numbers[i] % 2 === 0) {

//     evens.push(numbers[i]);

//   } else {

//     odds.push(numbers[i]);

//   }

// }

// console.log('Evens:', evens);

// console.log('Odds:', odds);

// o/p ===>

//   Evens:

// Array(5) [ 2, 4, 6, 8, 10 ]

// Loops.js:588:9

// Odds:

// Array(5) [ 1, 3, 5, 7, 9 ]

// In the loop, determine if each number is even or odd, then put that number into evens or odds, as appropriate.

// Print the arrays to confirm the results. Print evens first. Example: console.log(evens);

// let evens = []; // Array to hold even numbers

// let odds = [];  // Array to hold odd numbers

// let numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]; // Example array

// // Loop through the numbers array

// for (let i = 0; i < numbers.length; i++) {

//   if (numbers[i] % 2 === 0) {

//     evens.push(numbers[i]);  // Add even numbers to the evens array

//   } else {

//     odds.push(numbers[i]);   // Add odd numbers to the odds array

//   }

// }

// // Print the results

// console.log('Evens:', evens);

// console.log('Odds:', odds);

// //Define three variables for the socialprachar for the starting fuel level, another for the number of astronauts aboard, and the third for the altitude the shuttle reaches.

// let fuelLevel;       // Variable the starting fuel level

// let astronauts;      // Variable the number of astronauts aboard

// let altitude = 0;    // Variablethe altitude the shuttle reaches

//Construct while loops to do the following:

//Prompt the user to enter the starting fuel level. The loop should continue until the user enters a positive value greater than 5000 but less than 30000.

// let fuelLevel;

// while (true) {

//   fuelLevel = parseInt(prompt("Enter the starting fuel level (between 5000 and 30000):"), 10);

//   //  fuel level is valid

//   if (fuelLevel > 5000 && fuelLevel < 30000) {

//     break; // Exit the loop

//   }

//   console.log("Invalid input. Please enter a number greater than 5000 and less than 30000.");

// }

// o/p ====>

//   Invalid input. Please enter a number greater than 5000 and less than 30000. Loops.js:648:11

​

///Use a second loop to query the user for the number of astronauts (up to a maximum of 7). Validate the entry by having the loop continue until the user enters an integer from 1 - 7.

// let astronauts;

// while (true) {

//   astronauts = parseInt(prompt("Enter the number of astronauts (1 to 7):"), 10);

//  // number of astronauts

//   if (astronauts >= 1 && astronauts <= 7) {

//     break; // Exit the loop value

//   }

//    // value vaild number

//   console.log("Invalid input. Please enter an integer between 1 and 7.");

// }

///Use a final loop to monitor the fuel status and the altitude of the shuttle. Each iteration, decrease the fuel level by 100 units for each astronaut aboard. Also, increase the altitude by 50 kilometers. (Hint: The loop should end when there is not enough fuel to boost the crew another 50 km, so the fuel level might not reach 0).

// let fuelLevel = 25000;

// let astronauts = 4;

// let altitude = 0;

// // Final loop

// while (fuelLevel >= astronauts \* 100) {

//   // Decrease the fuel level

//   fuelLevel -= astronauts \* 100;

//   // Increase

//   altitude += 50;

//   console.log("Current fuel level:", fuelLevel);

//   console.log("Current altitude:", altitude, "km");

// }

// // Final status

// console.log("Shuttle has stopped at an altitude of:", altitude, "km");

level-3

//Write a JS code to print a pattern using for loop

// /\* 1

//    1 2

//    1 2 3

//    1 2 3 4

//    1 2 3 4 5

//    1 2 3 4 5 6

//    1 2 3 4 5 6 7

//    1 2 3 4 5 6 7 8 \*/

  //  for (let i = 1; i <= 8; i++) {

  //   let line = '';

  //   for (let j = 1; j <= i; j++) {

  //     line += j + ' ';

  //   }

  //   console.log(line.trim());

  // }

// write a js code to print Square Star Pattern .

//  const n = 5; // Size of the square

//  for (let i = 0; i < n; i++) {

//   let line = '';

//   for (let j = 0; j < n; j++) {

//     line += '\* ';

//   }

//   console.log(line.trim());

//  }

// write a js code to print  Hollow Square Pattern .

// const n = 5; // Size of the square

// for (let i = 0; i < n; i++) {

//   let line = '';

//   for (let j = 0; j < n; j++) {

//     if (i === 0 || i === n - 1 || j === 0 || j === n - 1) {

//       line += '\* ';

//     } else {

//       line += '  ';

//     }

//   }

//   console.log(line.trim());

// }

// // write a js code to print  Right Triangle Pattern in Javascript.

// const n = 5;

// for (let i = 1; i <= n; i++) {

//   let line = '';

//   for (let j = 1; j <= i; j++) {

//     line += '\* ';

//   }

//   console.log(line.trim());

// }

// // write a js code to print  left Triangle Pattern in Javascript.

// const n = 5;

// for (let i = 1; i <= n; i++) {

//   let line = '';

//   for (let j = 1; j <= n; j++) {

//     if (j <= i) {

//       line += '\* ';

//     } else {

//       line += '  ';

//     }

//   }

//   console.log(line.trim());

// }

// // write a js code to print  Downward Triangle Star Pattern

// const n = 5;

// for (let i = n; i >= 1; i--) {

//   let line = '';

//   for (let j = 1; j <= i; j++) {

//     line += '\* ';

//   }

//   console.log(line.trim());

// }

// // write a js code to print  Pyramid Pattern

// const n = 5;

// for (let i = 1; i <= n; i++) {

//   let line = '';

//   for (let j = 1; j <= n - i; j++) {

//     line += '  '; // Spaces

//   }

//   for (let j = 1; j <= 2 \* i - 1; j++) {

//     line += '\* ';

//   }

//   console.log(line.trim());

// }

// // write a js code to print reverse Pyramid Pattern

// const n = 5;

// for (let i = n; i >= 1; i--) {

//   let line = '';

//   for (let j = 1; j <= n - i; j++) {

//     line += '  '; // Spaces

//   }

//   for (let j = 1; j <= 2 \* i - 1; j++) {

//     line += '\* ';

//   }

//   console.log(line.trim());

// }

// // write a js code to print Diamond Pattern in Javascript

// const n = 5; // Number of rows (half the height of the diamond)

// for (let i = 1; i <= n; i++) {

//   let line = '';

//   for (let j = 1; j <= n - i; j++) {

//     line += '  ';

//   }

//   for (let j = 1; j <= 2 \* i - 1; j++) {

//     line += '\* ';

//   }

//   console.log(line.trim());

// }

// for (let i = n - 1; i >= 1; i--) {

//   let line = '';

//   for (let j = 1; j <= n - i; j++) {

//     line += '  ';

//   }

//   for (let j = 1; j <= 2 \* i - 1; j++) {

//     line += '\* ';

//   }

//   console.log(line.trim());

// }