Level-1 Questions

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

Explanation: Even numbers are those divisible by 2. We can check this using the modulus operator % which gives the remainder. If a number % 2 == 0, it is even.

For Loop Code:

javascript
Copy code

```
// Using For Loop
for (let i = 0; i <= 100; i++) {
    if (i % 2 === 0) {
        console.log(i);
    }
}
```

While Loop Code:

javascript
Copy code

```
// Using While Loop
let i = 0;
while (i <= 100) {
    if (i % 2 === 0) {
        console.log(i);
    }
    i++;
}
```

Output:

Copy code

```
0 2 4 6 8 10 ... 100
```

---

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

Explanation: Odd numbers are those not divisible by 2, meaning if a number % 2 != 0, it's odd.

For Loop Code:

javascript
Copy code

```
// Using For Loop
for (let i = 0; i <= 100; i++) {
    if (i % 2 !== 0) {
        console.log(i);
    }
}
```

While Loop Code:

javascript
Copy code
```
// Using While Loop
let i = 0;
while (i <= 100) {
    if (i % 2 !== 0) {
        console.log(i);
    }
    i++;
}
```

Output:

Copy code
```
1 3 5 7 9 11 ... 99
```

---

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

Explanation: Prime numbers are numbers greater than 1 that have only two divisors: 1 and themselves.

For Loop Code:

javascript
Copy code
```
// Helper function to check if a number is prime
function isPrime(num) {
    if (num < 2) return false;
    for (let i = 2; i <= Math.sqrt(num); i++) {
        if (num % i === 0) return false;
    }
    return true;
}

// Using For Loop
for (let i = 0; i <= 100; i++) {
    if (isPrime(i)) {
        console.log(i);
    }
}
```

While Loop Code:

javascript
Copy code
```
// Using While Loop
let i = 0;
while (i <= 100) {
    if (isPrime(i)) {
        console.log(i);
```

```
    }
    i++;
}
```

Output:

Copy code
2 3 5 7 11 13 ... 97

---

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

Explanation: We will initialize a sum variable and add each number from 0 to 100 to this variable.

For Loop Code:

javascript
Copy code

```
let sum = 0;
for (let i = 0; i <= 100; i++) {
    sum += i;
}
console.log("Sum of all numbers:", sum);
```

While Loop Code:

javascript
Copy code

```
let sum = 0;
let i = 0;
while (i <= 100) {
    sum += i;
    i++;
}
console.log("Sum of all numbers:", sum);
```

Output:

yaml
Copy code
Sum of all numbers: 5050

---

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

Explanation: We will maintain two sums, one for even numbers and one for odd numbers, and add them accordingly.

For Loop Code:

javascript

Copy code
```
let evenSum = 0, oddSum = 0;
for (let i = 0; i <= 100; i++) {
    if (i % 2 === 0) {
        evenSum += i;
    } else {
        oddSum += i;
    }
}
console.log("Sum of even numbers:", evenSum);
console.log("Sum of odd numbers:", oddSum);
```

While Loop Code:

javascript
Copy code
```
let evenSum = 0, oddSum = 0, i = 0;
while (i <= 100) {
    if (i % 2 === 0) {
        evenSum += i;
    } else {
        oddSum += i;
    }
    i++;
}
console.log("Sum of even numbers:", evenSum);
console.log("Sum of odd numbers:", oddSum);
```

Output:

yaml
Copy code
```
Sum of even numbers: 2550
Sum of odd numbers: 2500
```

---

6. Write a program that calculates the factorial of a given positive integer

Explanation: Factorial is the product of an integer and all the integers below it. For example, factorial of 5 is 5 * 4 * 3 * 2 * 1.

For Loop Code:

javascript
Copy code
```
let number = 5; // You can change this number
let factorial = 1;

for (let i = 1; i <= number; i++) {
    factorial *= i;
}
```

```
console.log(`Factorial of ${number} is ${factorial}`);
```

While Loop Code:

```
javascript
Copy code
let number = 5; // You can change this number
let factorial = 1;
let i = 1;

while (i <= number) {
    factorial *= i;
    i++;
}

console.log(`Factorial of ${number} is ${factorial}`);
```

Output:

```
csharp
Copy code
Factorial of 5 is 120
```

Level-2 Questions

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

Explanation: We will use a for loop to iterate from 0 to 20 and print each number on a new line.

For Loop Code:

```
javascript
Copy code
// Using For Loop
for (let i = 0; i <= 20; i++) {
    console.log(i);
}
```

Output:

```
Copy code
0
1
2
3
4
5
6
7
8
9
```

10
11
12
13
14
15
16
17
18
19
20

---

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

Explanation: We will iterate through the numbers from 3 to 29, checking if a number is odd and printing it.

For Loop Code:

javascript
Copy code
```
// Using For Loop
for (let i = 3; i <= 29; i++) {
    if (i % 2 !== 0) {
        console.log(i);
    }
}
```

Output:

Copy code
3
5
7
9
11
13
15
17
19
21
23
25
27
29

---

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

Explanation: We'll start from 12 and decrement down to -14, checking if the number is even before printing it.

For Loop Code:

javascript
Copy code
```
// Using For Loop
for (let i = 12; i >= -14; i--) {
    if (i % 2 === 0) {
        console.log(i);
    }
}
```

Output:

diff
Copy code
```
12
10
8
6
4
2
0
-2
-4
-6
-8
-10
-12
-14
```

---

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

Explanation: We will start from 50 and decrement to 20, printing numbers that are multiples of 3.

For Loop Code:

javascript
Copy code
```
// Using For Loop
for (let i = 50; i >= 20; i--) {
    if (i % 3 === 0) {
        console.log(i);
    }
}
```

Output:

Copy code
```
48
45
42
39
36
```

33
30
27
24
21

---

5. 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:

- **Print each element of the array to a new line.**

Explanation: We will iterate through the array and print each element.

Code:

javascript
Copy code
```
const str = 'LaunchCode';
const arr = [1, 5, 'LC101', 'blue', 42];

// Print each element of the array
for (let i = 0; i < arr.length; i++) {
   console.log(arr[i]);
}
```

Output:

Copy code
1
5
LC101
blue
42

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

Explanation: We will iterate through the string from the last character to the first.

Code:

javascript
Copy code
```
// Print each character of the string in reverse order
for (let i = str.length - 1; i >= 0; i--) {
   console.log(str[i]);
}
```

Output:

mathematica
Copy code
e
d
o
C
h
c
n
au
L

---

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

Explanation: We'll create two arrays, one for even numbers and one for odd numbers, and categorize each element accordingly.

Code:

javascript
Copy code
```
const numbers = [2, 3, 13, 18, -5, 38, -10, 11, 0, 104];
const evens = [];
const odds = [];

// Sort the numbers into evens and odds
for (let i = 0; i < numbers.length; i++) {
    if (numbers[i] % 2 === 0) {
        evens.push(numbers[i]);
    } else {
        odds.push(numbers[i]);
    }
}

// Print the arrays
console.log("Evens:", evens);
console.log("Odds:", odds);
```

Output:

makefile
Copy code
```
Evens: [ 2, 18, 38, -10, 0, 104 ]
Odds: [ 3, 13, -5, 11 ]
```

---

7. Define three variables for the shuttle: the starting fuel level, the number of astronauts aboard, and the altitude the shuttle reaches.

Explanation: We will use while loops to gather user input for the fuel level and the number of astronauts. Then we will monitor fuel levels and altitude.

Code:

javascript
Copy code

```
let fuelLevel;
let astronauts;
let altitude = 0;

// Prompt for starting fuel level
while (true) {
    fuelLevel = parseInt(prompt("Enter starting fuel level (between 5000 and 30000):"));
    if (fuelLevel > 5000 && fuelLevel < 30000) {
        break;
    }
}

// Prompt for number of astronauts
while (true) {
    astronauts = parseInt(prompt("Enter number of astronauts (1 to 7):"));
    if (astronauts >= 1 && astronauts <= 7) {
        break;
    }
}

// Monitor fuel status and altitude
while (fuelLevel >= 100) {
    fuelLevel -= 100 * astronauts;
    altitude += 50;
    console.log(`Current fuel level: ${fuelLevel}, Altitude: ${altitude} km`);
}
```

Output (example simulation):

yaml
Copy code

```
Enter starting fuel level (between 5000 and 30000): 15000
Enter number of astronauts (1 to 7): 5
Current fuel level: 149500, Altitude: 50 km
Current fuel level: 149400, Altitude: 100 km
...
```

(Note: The outputs will vary depending on the fuel level and astronaut input.)

Level-3 Questions

1. Write a JS code to print a pattern using a for loop.

Pattern:

Copy code
```
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
```

Explanation: We will use nested loops. The outer loop will iterate over the rows, and the inner loop will print numbers from 1 to the current row number.

Code:

javascript
Copy code
```
// Print the pattern
for (let i = 1; i <= 8; i++) {
    let row = '';
    for (let j = 1; j <= i; j++) {
        row += j + ' ';
    }
    console.log(row.trim());
}
```

Output:

Copy code
```
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
```

---

2. Write a JS code to print Square Star Pattern.

Pattern:

markdown
Copy code
```
*****
*****
*****
*****
*****
```

Explanation: We'll use a nested loop where the outer loop represents rows and the inner loop prints stars for each row.

Code:

javascript
Copy code
```
const size = 5; // Size of the square

// Print square star pattern
for (let i = 0; i < size; i++) {
    let row = '';
    for (let j = 0; j < size; j++) {
        row += '*';
    }
    console.log(row);
}
```

Output:

markdown
Copy code
```
*****
*****
*****
*****
*****
```

---

3. Write a JS code to print Hollow Square Pattern.

Pattern:

markdown
Copy code
```
*****
*   *
*   *
*   *
*****
```

Explanation: In this pattern, the first and last rows are filled with stars, while the middle rows have stars at the start and end, with spaces in between.

Code:

javascript
Copy code
```
const size = 5; // Size of the hollow square

// Print hollow square star pattern
for (let i = 0; i < size; i++) {
```

```
    let row = '';
    for (let j = 0; j < size; j++) {
        if (i === 0 || i === size - 1 || j === 0 || j === size - 1) {
            row += '*';
        } else {
            row += ' ';
        }
    }
    console.log(row);
}
```

Output:

markdown
Copy code
*****
*   *
*   *
*   *
*****

---

4. Write a JS code to print Right Triangle Pattern in JavaScript.

Pattern:

markdown
Copy code
*
**
***
****
*****

Explanation: The outer loop controls the number of rows, while the inner loop prints stars equal to the current row number.

Code:

javascript
Copy code

```
const height = 5; // Height of the triangle

// Print right triangle star pattern
for (let i = 1; i <= height; i++) {
    let row = '';
    for (let j = 1; j <= i; j++) {
        row += '*';
    }
    console.log(row);
}
```

Output:

markdown
Copy code
```
*
**
***
****
*****
```

---

5. Write a JS code to print Left Triangle Pattern in JavaScript.

Pattern:

markdown
Copy code
```
*****
****
***
**
*
```

Explanation: Here, we start from the maximum number of stars and decrease by one star for each subsequent row.

Code:

javascript
Copy code
```
const height = 5; // Height of the triangle

// Print left triangle star pattern
for (let i = height; i >= 1; i--) {
    let row = '';
    for (let j = 1; j <= i; j++) {
        row += '*';
    }
    console.log(row);
}
```

Output:

markdown
Copy code
```
*****
****
***
**
*
```

---

6. Write a JS code to print Downward Triangle Star Pattern.

Pattern:

markdown
Copy code
```
*********
********
*******
*****
***
*
```

Explanation: This pattern starts from a certain number of stars and decreases by 2 stars each time.

Code:

javascript
Copy code
```
const height = 5; // Half the height

// Print downward triangle star pattern
for (let i = height * 2 - 1; i >= 1; i -= 2) {
    let row = '';
    for (let j = 1; j <= i; j++) {
        row += '*';
    }
    console.log(row);
}
```

Output:

markdown
Copy code
```
*********
********
*******
*****
***
*
```

---

7. Write a JS code to print Pyramid Pattern.

Pattern:

markdown
Copy code
```
    *
   ***
  *****
 *******
```

****

Explanation: The pyramid pattern requires spaces before the stars to center them properly. The number of spaces decreases while the number of stars increases.

Code:

javascript
Copy code
```
const height = 5; // Height of the pyramid

// Print pyramid star pattern
for (let i = 1; i <= height; i++) {
    let row = ' '.repeat(height - i); // Adding spaces
    for (let j = 1; j <= (2 * i - 1); j++) { // Stars
        row += '*';
    }
    console.log(row);
}
```

Output:

markdown
Copy code
```
    *
   ***
  *****
 *******
*********
```

---

8. Write a JS code to print reverse Pyramid Pattern.

Pattern:

markdown
Copy code
```
*********
 *******
  *****
   ***
    *
```

Explanation: This pattern is similar to the pyramid but inverted. The stars decrease while the spaces increase.

Code:

javascript
Copy code
```
const height = 5; // Height of the pyramid

// Print reverse pyramid star pattern
```

```
for (let i = height; i >= 1; i--) {
    let row = ' '.repeat(height - i); // Adding spaces
    for (let j = 1; j <= (2 * i - 1); j++) { // Stars
        row += '*';
    }
    console.log(row);
}
```

Output:

markdown
Copy code
```
*********
 *******
  *****
   ***
    *
```

---

9. Write a JS code to print Diamond Pattern in JavaScript.

Pattern:

markdown
Copy code
```
    *
   ***
  *****
 *******
*********
 *******
  *****
   ***
    *
```

Explanation: The diamond pattern consists of a pyramid followed by an inverted pyramid.

Code:

javascript
Copy code
```
const height = 5; // Half the height of the diamond

// Print diamond star pattern
for (let i = 1; i <= height; i++) {
    let row = ' '.repeat(height - i);
    for (let j = 1; j <= (2 * i - 1); j++) {
        row += '*';
    }
    console.log(row);
}
```

```
for (let i = height - 1; i >= 1; i--) {
    let row = ' '.repeat(height - i);
    for (let j = 1; j <= (2 * i - 1); j++) {
        row += '*';
    }
    console.log(row);
}
```

Output:

markdown
Copy code

```
    *
   ***
  *****
 *******
*********
 *******
  *****
   ***
    *
```

Level-2 Questions

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

Explanation: We will use a for loop that starts from 0 and goes up to 20, printing each number.

Code:

javascript
Copy code

```
// Print numbers from 0 to 20
for (let i = 0; i <= 20; i++) {
    console.log(i);
}
```

Output:

Copy code

```
0
1
2
3
4
5
6
7
8
9
10
```

11
12
13
14
15
16
17
18
19
20

---

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

Explanation: This code uses a for loop that starts at 3 and goes up to 29, incrementing by 2 to get only odd numbers.

Code:

javascript
Copy code
```
// Print odd values from 3 to 29
for (let i = 3; i <= 29; i += 2) {
    console.log(i);
}
```

Output:

Copy code
3
5
7
9
11
13
15
17
19
21
23
25
27
29

---

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

Explanation: Here, we will use a for loop that starts from 12 and decrements by 2 until it reaches -14.

Code:

javascript

Copy code
```
// Print even numbers from 12 down to -14
for (let i = 12; i >= -14; i -= 2) {
    console.log(i);
}
```

Output:

diff
Copy code
```
12
10
8
6
4
2
0
-2
-4
-6
-8
-10
-12
-14
```

---

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

Explanation: We will iterate from 50 to 20, checking if each number is a multiple of 3 using the modulus operator.

Code:

javascript
Copy code
```
// Print multiples of 3 from 50 down to 20
for (let i = 50; i >= 20; i--) {
    if (i % 3 === 0) {
        console.log(i);
    }
}
```

Output:

Copy code
```
48
45
42
39
36
33
30
```

27
24
21

---

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

Variables:

javascript
Copy code
```
const launchCode = 'LaunchCode';
const arr = [1, 5, 'LC101', 'blue', 42];
```

Explanation: This code uses a for loop to iterate through the array and print each element.

Code:

javascript
Copy code
```
// Print each element of the array
for (let i = 0; i < arr.length; i++) {
    console.log(arr[i]);
}
```

Output:

Copy code
```
1
5
LC101
blue
42
```

---

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

Explanation: We'll iterate over the string in reverse using a for loop and print each character.

Code:

javascript
Copy code
```
// Print each character of the string in reverse order
for (let i = launchCode.length - 1; i >= 0; i--) {
    console.log(launchCode[i]);
}
```

Output:

```
r
```

Copy code
e
d
o
c
n
au
L

---

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

Explanation: This code initializes two empty arrays, evens and odds, then checks each number to see if it's even or odd.

Code:

javascript
Copy code
```
const numbers = [2, 3, 13, 18, -5, 38, -10, 11, 0, 104];
const evens = [];
const odds = [];

// Sort numbers into evens and odds
for (let i = 0; i < numbers.length; i++) {
    if (numbers[i] % 2 === 0) {
        evens.push(numbers[i]);
    } else {
        odds.push(numbers[i]);
    }
}

// Print the arrays
console.log("Evens:", evens);
console.log("Odds:", odds);
```

Output:

makefile
Copy code
```
Evens: [ 2, 18, 38, -10, 0, 104 ]
Odds: [ 3, 13, -5, 11 ]
```

---

8. Define three variables for the starting fuel level, number of astronauts, and altitude. Construct while loops to validate user input.

Explanation: We'll prompt the user for input and validate the conditions using while loops.

Code:

javascript

Copy code
```
let fuelLevel = 0;
let astronauts = 0;
let altitude = 0;

// Prompt for starting fuel level
while (fuelLevel <= 5000 || fuelLevel >= 30000) {
    fuelLevel = parseInt(prompt("Enter starting fuel level (between 5000 and 30000):"));
}

// Prompt for number of astronauts
while (astronauts < 1 || astronauts > 7) {
    astronauts = parseInt(prompt("Enter number of astronauts (1 - 7):"));
}

// Monitor fuel status and altitude
while (fuelLevel > 0) {
    fuelLevel -= astronauts * 100; // Decrease fuel level
    altitude += 50; // Increase altitude
    console.log(`Current fuel level: ${fuelLevel}, Current altitude: ${altitude} km`);

    if (fuelLevel < 100) break; // End if not enough fuel for next boost
}
```

Note: The above code uses prompt(), which is typically used in a browser environment.

Level-1 Questions

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

Explanation: We will use both a for loop and a while loop to print even numbers between 0 and 100. An even number is defined as a number that is divisible by 2.

Using For Loop:

javascript
Copy code
```
console.log("Even numbers using for loop:");
for (let i = 0; i <= 100; i++) {
    if (i % 2 === 0) { // Check if the number is even
        console.log(i);
    }
}
```

Output:

arduino
Copy code
```
Even numbers using for loop:
0
```

2
4
6
8
10
12
14
16
18
20
22
24
26
28
30
32
34
36
38
40
42
44
46
48
50
52
54
56
58
60
62
64
66
68
70
72
74
76
78
80
82
84
86
88
90
92
94
96
98
100

Using While Loop:

javascript
Copy code
```
console.log("Even numbers using while loop:");
let j = 0;
while (j <= 100) {
    if (j % 2 === 0) { // Check if the number is even
        console.log(j);
    }
    j++;
}
```

Output:

arduino
Copy code
```
Even numbers using while loop:
0
2
4
6
8
10
12
14
16
18
20
22
24
26
28
30
32
34
36
38
40
42
44
46
48
50
52
54
56
58
60
62
64
66
68
70
```

72
74
76
78
80
82
84
86
88
90
92
94
96
98
100

---

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

Explanation: Similarly, we will iterate from 0 to 100 and print only odd numbers, which are numbers that are not divisible by 2.

Using For Loop:

javascript
Copy code

```
console.log("Odd numbers using for loop:");
for (let i = 0; i <= 100; i++) {
    if (i % 2 !== 0) { // Check if the number is odd
        console.log(i);
    }
}
```

Output:

arduino
Copy code

```
Odd numbers using for loop:
1
3
5
7
9
11
13
15
17
19
21
23
25
27
```

29
31
33
35
37
39
41
43
45
47
49
51
53
55
57
59
61
63
65
67
69
71
73
75
77
79
81
83
85
87
89
91
93
95
97
99

Using While Loop:

javascript
Copy code

```
console.log("Odd numbers using while loop:");
let k = 1;
while (k <= 100) {
    if (k % 2 !== 0) { // Check if the number is odd
        console.log(k);
    }
    k++;
}
```

Output:

arduino
Copy code
Odd numbers using while loop:
1
3
5
7
9
11
13
15
17
19
21
23
25
27
29
31
33
35
37
39
41
43
45
47
49
51
53
55
57
59
61
63
65
67
69
71
73
75
77
79
81
83
85
87
89
91
93
95
97

---

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

Explanation: A prime number is a natural number greater than 1 that cannot be formed by multiplying two smaller natural numbers. We will check each number from 2 to 100 to see if it is prime.

Using For Loop:

javascript
Copy code

```
console.log("Prime numbers using for loop:");
for (let num = 2; num <= 100; num++) {
    let isPrime = true; // Assume the number is prime
    for (let i = 2; i <= Math.sqrt(num); i++) { // Check for factors
        if (num % i === 0) {
            isPrime = false; // Not a prime number
            break;
        }
    }
    if (isPrime) {
        console.log(num);
    }
}
```

Output:

arduino
Copy code

```
Prime numbers using for loop:
2
3
5
7
11
13
17
19
23
29
31
37
41
43
47
53
59
61
67
71
73
```

79
83
89
97

Using While Loop:

javascript
Copy code

```
console.log("Prime numbers using while loop:");
let num = 2;
while (num <= 100) {
    let isPrime = true; // Assume the number is prime
    let i = 2;
    while (i <= Math.sqrt(num)) { // Check for factors
        if (num % i === 0) {
            isPrime = false; // Not a prime number
            break;
        }
        i++;
    }
    if (isPrime) {
        console.log(num);
    }
    num++;
}
```

Output:

arduino
Copy code

```
Prime numbers using while loop:
2
3
5
7
11
13
17
19
23
29
31
37
41
43
47
53
59
61
67
71
```

73
79
83
89
97

---

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

Explanation: We'll accumulate the sum of all numbers from 0 to 100 using both loop types.

Using For Loop:

javascript
Copy code
```
let sumForLoop = 0;
for (let i = 0; i <= 100; i++) {
   sumForLoop += i; // Add the current number to the sum
}
console.log("Sum of all numbers using for loop:", sumForLoop);
```

Output:

arduino
Copy code
```
Sum of all numbers using for loop: 5050
```

Using While Loop:

javascript
Copy code
```
let sumWhileLoop = 0;
let j = 0;
while (j <= 100) {
   sumWhileLoop += j; // Add the current number to the sum
   j++;
}
console.log("Sum of all numbers using while loop:", sumWhileLoop);
```

Output:

arduino
Copy code
```
Sum of all numbers using while loop: 5050
```

---

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

Explanation: We will maintain two sums: one for even numbers and one for odd numbers, iterating from 0 to 100.

Using For Loop:

javascript
Copy code
```
let evenSumFor = 0;
let oddSumFor = 0;
for (let i = 0; i <= 100; i++) {
    if (i % 2 === 0) {
        evenSumFor += i; // Sum of even numbers
    } else {
        oddSumFor += i; // Sum of odd numbers
    }
}
console.log("Sum of evens using for loop:", evenSumFor);
console.log("Sum of odds using for loop:", oddSumFor);
```

Output:

arduino
Copy code
```
Sum of evens using for loop: 2550
Sum of odds using for loop: 2500
```

Using While Loop:

javascript
Copy code
```
let evenSumWhile = 0;
let oddSumWhile = 0;
let k = 0;
while (k <= 100) {
    if (k % 2 === 0) {
        evenSumWhile += k; // Sum of even numbers
    } else {
        oddSumWhile += k; // Sum of odd numbers
    }
    k++;
}
console.log("Sum of evens using while loop:", evenSumWhile);
console.log("Sum of odds using while loop:", oddSumWhile);
```

Output:

arduino
Copy code
```
Sum of evens using while loop: 2550
Sum of odds using while loop: 2500
```

---

6. Write a program that calculates the factorial of a given positive integer.

Explanation: A factorial of a positive integer nnn is the product of all positive integers less than or equal to nnn. We can calculate it using a loop.

Using For Loop:

javascript
Copy code

```
function factorial(n) {
    let result = 1;
    for (let i = 1; i <= n; i++) {
        result *= i; // Multiply the current number
    }
    return result;
}

const number = 5; // Change this to calculate factorial of any positive integer
console.log(`Factorial of ${number} using for loop:`, factorial(number));
```

Output:

arduino
Copy code

```
Factorial of 5 using for loop: 120
```

Using While Loop:

javascript
Copy code

```
function factorialWhile(n) {
    let result = 1;
    let i = 1;
    while (i <= n) {
        result *= i; // Multiply the current number
        i++;
    }
    return result;
}

console.log(`Factorial of ${number} using while loop:`, factorialWhile(number));
```

Output:

```
Factorial of 5 using while loop: 120
```