Module 8 : JavaScript Essentials & Advanced

Q.1) Using console.log() print out the following statement: The quote 'There is no exercise better for the heart than reaching down and lifting people up.' by John Holmes teaches usto help one another. Using console.log() print out the following quote by Mother Teresa:

Ans.1)

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>
    <script>
        console.log("The quote 'There is no exercise better for the heart than reaching down and lifting people up.' by John Holmes teaches us to help one another.");

        console.log("Mother Teresa once said, \"If you can't feed a hundred people, then feed just one.\"");

    </script>
</body>

</html>
```

Output:

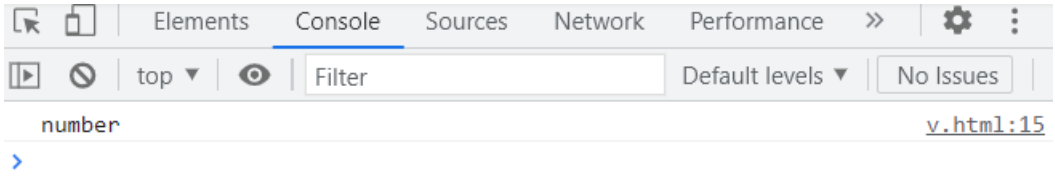
```
The quote 'There is no exercise better for the heart than reaching down and lifting people up.' by John Holmes teaches us to help one another.   conso.html:13
Mother Teresa once said, "If you can't feed a hundred people, then feed just one."   conso.html:15
```

Q.2) Check if typeof '10' is exactly equal to 10. If not make it exactly equal?

Ans.2)

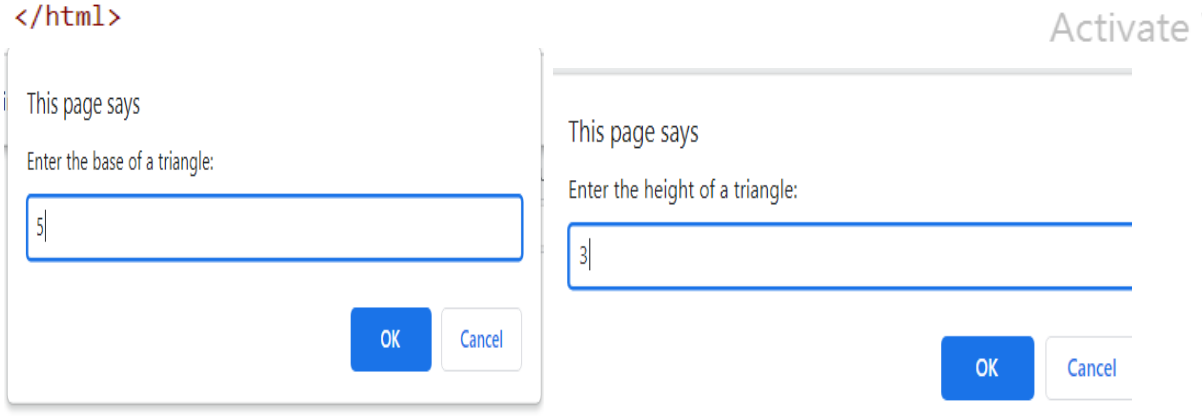
```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>
    <script>
        var num=10;
        if(num===10){
            console.log(typeof(num));
        }
    </script>
</body>

</html>
```

Output:



Q.3) Write a JavaScript Program to find the area of a triangle?
Ans.3)

```
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>
    <script>
        const baseValue = prompt('Enter the base of a triangle: ');
        const heightValue = prompt('Enter the height of a triangle: ');

        // calculate the area
        const areaValue = (baseValue * heightValue) / 2;

        document.write(
            `The area of the triangle is ${areaValue}`
        );
    </script>
</body>

</html>
```



Output: The area of the triangle is 7.5

Q.4) Write a JavaScript program to calculate days left until next Christmas?
Ans.4)

```
<script>
    // Get the current date
    let today = new Date();

    // Get the year of the current date
    let christmasYear = today.getFullYear();

    // Check if the current date is
    // already past by checking if the month
    // is December and the current day
    // is greater than 25
    if (today.getMonth() == 11 &&
        today.getDate() > 25) {

        // Add an year so that the next
        // Christmas date could be used
        christmasYear = christmasYear + 1;
    }

    // Get the date of the next Christmas
    let christmasDate =
        new Date(christmasYear, 11, 25);

    // Get the number of milliseconds in 1 day
    let dayMilliseconds =
        1000 * 60 * 60 * 24;

    // Get the remaining amount of days
    let remainingDays = Math.ceil(
        (christmasDate.getTime() - today.getTime()) /
        (dayMilliseconds)
    );

    // Write it to the page
    document.write("There are " + remainingDays +
        " days remaining until Christmas.");
</script>
```

Output:

Program to calculate days left until next Christmas using JavaScript?

There are 271 days remaining until Christmas.

Q.5) What is Condition Statement?

Ans.5) In A JavaScript we have three forms of statement.

- if statement
- if else statement
- else if statement

1)if statement

- The if statement evaluated the condition inside the ().
- if the condition is evaluated to true , the code inside the body of if executed the code is

not executed

```
<script>
   var age=10;
   if(age<18){
    document.write("Adult");
   }
</script>
```

Adult

2) if else statement:

- The else statement is used to specify a block of code to be executed if a condition is not met.

```
<script>
    let a = 5;
    let b = 2;
    if (a == b) {
        document.write("Hello World")
    } else {
        document.write("Condition is not met")
    }
</script>
```

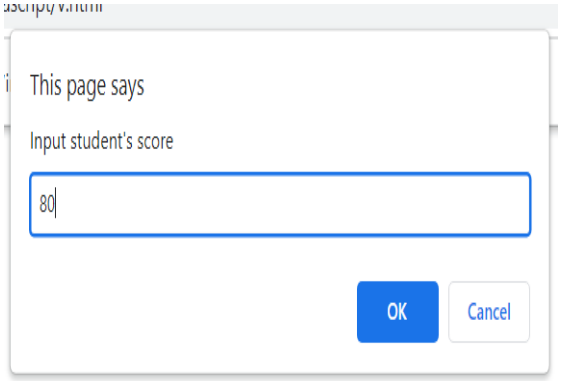
Output:

Condition is not met

3) else if statement:

The else if statement is used to specify a new condition to be tested if the first condition is not met. This can be used when there's more than one condition to be tested.

```
<script>
    let score = prompt("Input student's score");

    if (score >= 70) {
        document.write("This Student has an A")
    } else if (score >= 55 && score < 70) {
        document.write("This Student has a B")
    } else if (score >= 50 && score < 70) {
        document.write("This Student has a C")
    } else if (score < 50) {
        document.write("This Student has an F")
    } else {
        document.write("Invalid Score")
    }
</script>
```



Output:

This Student has an A

Q.6) Find circumference of Rectangle formula: C = 4 * a?

Ans.6)

```html
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1
    <title>Document</title>
</head>

<body>

    <script>

        // Javascript program to find
        // Circumference of a square
        function Circumference(a)
        {
            return 4 * a;
        }

        // Driver Code
            let a = 5;
            document.write("Circumference of"
                +" a square is "
                +Circumference(a));

        </script>


</body>

</html>
```

Output:

Circumference of a square is 20

Q.7)WAP to convert years into days and days into years?

Ans.7)

```html
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>

    <script>
        function yearsToDays(years) {
            return years * 365; // Assuming 1 year has 365 days
        }

        function daysToYears(days) {
            return days / 365; // Assuming 1 year has 365 days
        }

        // Example usage:
        var years = 5;
        var days = 730; // 2 years * 365 days/year = 730 days

        document.write(years + " years is equal to " + yearsToDays(years) + " days."+"<br>");
        document.write(days + " days is equal to " + daysToYears(days) + " years.");


    </script>
```

Output:

5 years is equal to 1825 days.
730 days is equal to 2 years.

Q.8)Convert temperature Fahrenheit to Celsius? (Conditional logic Question).

Ans.8)

```html
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>


    <script>
        function fahrenheitToCelsius(fahrenheit) {
            // Check if the input is a valid number
            if (typeof fahrenheit !== 'number' || isNaN(fahrenheit)) {
                return "Invalid input. Please provide a valid number.";
            }

            // Perform the conversion
            var celsius = (fahrenheit - 32) * 5 / 9;

            return celsius;
        }

        // Example usage
        var fahrenheitTemperature = 70;
        var celsiusTemperature = fahrenheitToCelsius(fahrenheitTemperature);
        document.write(fahrenheitTemperature + "°F is equal to " + celsiusTemperature.toFixed(2) + "°C");

    </script>
```

Output:
70°F is equal to 21.11°C

Q.9) Write a JavaScript exercise to get the extension of a filename.?
Ans.9)

```html
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>


    <script>
        // program to get the file extension

        function getFileExtension(filename) {

            // get file extension
            const extension = filename.split('.').pop();
            return extension;
        }

        // passing the filename
        const result1 = getFileExtension('module.js'+"<br>");
        document.write(result1);

        const result2 = getFileExtension('module.txt');
        document.write(result2);
    </script>


</body>

</html>
```

Output:

js
txt

Q.10)What is the result of the expression (5 > 3 && 2 < 4)?

Ans.10)

```html
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>


    <script>
        const a=5;
        const b=2;
        document.write(a>3 && b<4);
    </script>


</body>

</html>
```

Output:

true

Q.11)What is the result of the expression (true && 1 && "hello")?
Ans.11)

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>


    <script>
        document.write(true && 1 && "hello");
    </script>


</body>

</html>
```

Output:

hello

Q.12) What is the result of the expression true && false || false && true?
Ans.12)

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>


    <script>
        const a=5;
        const b=2;
       document.write(true && false || false && true);
    </script>


</body>

</html>
```

Output:
false

Q.13) Check Number Is Positive or Negative in JavaScript?

Ans.13) The Math.sign() method returns whether a number is negative, positive or zero.

If the number is positive, this method returns 1.
If the number is negative, it returns -1.
If the number is zero, it returns 0.

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>

    <p id="demo"></p>

    <script>
    document.getElementById("demo").innerHTML =  Math.sign(3);
    </script>



</body>

</html>
```

Output:

1

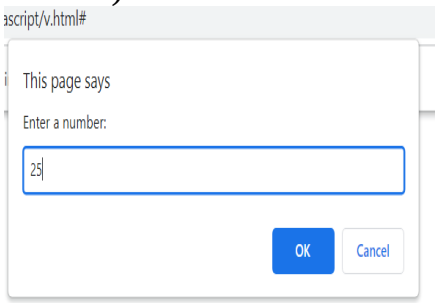
Q.14) Find the Character Is Vowel or Not ?
Ans.14)

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0
    <title>Document</title>
</head>

<body>
    <script>
        var ch='e';
        if(ch=='a' || ch=='e' || ch=='i' || ch=='o' || ch=='u')
          document.write("<b>"+ch+"</b>" + " is a Vowel");
        else if(ch=='A' || ch=='E' || ch=='I' || ch=='O' || ch=='U')
          document.write("<b>"+ch+"</b>" + " is a Vowel");
        else
          document.write("<b>"+ch+"</b>" + " is a Consonant");
        </script>
</body>

</html>
```

Output:

e is a Vowel

Q.15) Write to check whether a number is negative, positive or zero?
Ans.15)




```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>
    <script>
        // check if the number is positive, negative or zero
        const number = prompt("Enter a number: ");

        if (number >= 0) {
            if (number == 0) {
                document.write("The number is zero");
            } else {
                document.write("The number is positive");
            }
        } else {
            document.write("The number is negative");
        }
        </script>
</body >

</html >
```

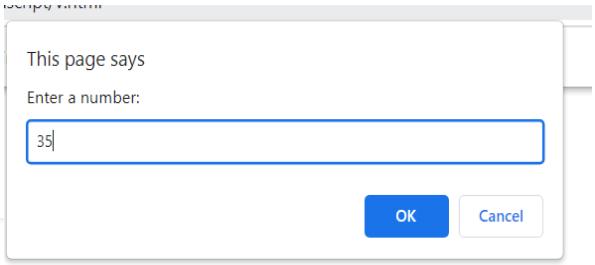
Output:

The number is positive

Q-16) Write to find number is even or odd using ternary operator in JS?
Ans.16)

```html
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>
    <script>
        // program to check if the number is even or odd
        // take input from the user
        const number = prompt("Enter a number: ");

        //check if the number is even
        if (number % 2 == 0) {
            document.write("The number is even.");
        }

        // if the number is odd
        else {
            document.write("The number is odd.");
        }
    </script>
</body>
```

Output:



The number is odd.

Q.17) Write find maximum number among 3 numbers using ternary operator in JS?

Ans.17)

```html
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>
    <script>

        // Javascript program to find largest among three
        // numbers using ternary operator

        // Variable declaration
        var n1 = 5, n2 = 10, n3 = 15, max;

        // Largest among n1, n2 and n3
        max = (n1 > n2) ? (n1 > n3 ? n1 : n3) : (n2 > n3 ? n2 : n3);

        // Print the largest number
        document.write("Largest number among "
            + n1 + ", " + n2 + " and "
            + n3 + " is " + max + ".");


    </script>

</body>
```

Output:

Largest number among 5, 10 and 15 is 15.

Q.18) Write to find minimum number among 3 numbers using ternary operator in JS?

Ans.18)

```html
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>
    <script>

        // Javascript program to find smallest among three
        // numbers using ternary operator

        // Variable declaration
        var n1 = 5, n2 = 10, n3 = 15, max;

        // Largest among n1, n2 and n3
        max = (n1 < n2) ? (n1 < n3 ? n1 : n3) : (n2 < n3 ? n2 : n3);

        // Print the largest number
        document.write("Smallest number among "
            + n1 + ", " + n2 + " and "
            + n3 + " is " + max + ".");


    </script>

</body>

</html>
```

Output:

Smallest number among 5, 10 and 15 is 5.

Q.19)Write to find the largest of three numbers in JS?
Ans.19)

```html
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>
    <script>

        // declaring the variables
        let num1 = 30;
        let num2 = 70;
        let num3 = 15;
        let largestNum;

        if (num1 > num2 && num1 > num3) {
            largestNum = num1;
        }
        else if (num2 > num1 && num2 > num3) {
            largestNum = num2;
        }
        else {
            largestNum = num3;
        }

        document.write("The largest num is:"+largestNum);

    </script>

</body>

</html>
```

Output:

The largest num is:70

Q.20) Write to show?

i. Monday to Sunday using switch case in JS?

```
<body>
    <h2>Monday to Sunday</h2>

    <p id="demo"></p>

    <script>
        let day;
        switch (new Date().getDay()) {
            case 0:
                day = "Sunday";
                break;
            case 1:
                day = "Monday";
                break;
            case 2:
                day = "Tuesday";
                break;
            case 3:
                day = "Wednesday";
                break;
            case 4:
                day = "Thursday";
                break;
            case 5:
                day = "Friday";
                break;
            case 6:
                day = "Saturday";
        }
        document.getElementById("demo").innerHTML = "Today is " + day;
    </script>
```

Output:

Monday to Sunday

Today is Saturday

ii. Vowel or Consonant using switch case in JS?

```html
<body>
    <table>
        <tr>
            <td> <input type="text" name="a" id="first" placeholder="Enter character" /> </td>
        </tr>
        <tr>
            <td> <button onclick="vowel()">Submit</button> </td>
        </tr>
    </table>
    <div id="num"></div>
</body>

<script type="text/javascript">
    function vowel() {
        var ch;
        ch = document.getElementById("first").value;
        switch (ch) {
            case 'a':
            case 'e':
            case 'i':
            case 'o':
            case 'u':
            case 'A':
            case 'E':
            case 'I':
            case 'O':
            case 'U':
                document.getElementById("num").innerHTML = "vowel character";
                break;
            default: document.getElementById("num").innerHTML = "Consonant character";
                break;
        }
    }
</script>
```

Output:

(Conditional looping logic Question)

Q.21) What are the looping structures in JavaScript? Any one Example?

Ans.21) we use loops to perform repeated actions. For example - if you are assigned a task of printing numbers from 1 to 100, it will be very hectic to do it manually, loops help us automate such tasks.

- ❖ types of loops:
- ➢ for loop
- ➢ for in loop
- ➢ for of loop
- ➢ while loop
- ➢ do-while loop

The for loop:

the syntax of a for loop something like this.

for(statement1;statement2;statement3)
{
//code to be executed;
}
example:

```
<body>
    <script>
        for (i = 1; i <= 5; i++) {
            document.write("The number is:" + i + "<br/>")
        }
    </script>
</body>
```

Output:
The number is:1
The number is:2
The number is:3
The number is:4
The number is:5

Q.22) Write a print 972 to 897 using for loop in JS?
Ans.22)

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>
    <script>
        let i;
        for (i = 972; i>=897; i=i-1) {
            document.write(i + "<br/>")
        }
    </script>
</body>

</html>
```

```
972
971
970
969
968
967
966
965
964
963
962
961
960
959
958
957
956
955
954
953
952
951
950
949
948
947
946
945
944
943
942
941
940
939
```

Q.23) Write to print factorial of given number?

Ans.23)

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>
    Enter Num: <input id="num">
    <button onclick="show()">Factorial</button>
    <input id="answer">
    <script>
        function show() {

            var i, no, fact;
            fact = 1;
            no = Number(document.getElementById("num").value);
            for (i = 1; i <= no; i++) {
                fact = fact * i;
            }
            document.getElementById("answer").value = fact;
        }
    </script>



</body>
```

Output:

Q.24)Write to print Fibonacci series up to given numbers?

Ans.24)

```
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>
    <script>
        // program to generate fibonacci series up to n terms

        // take input from the user
        const number = parseInt(prompt('Enter the number of terms: '));
        let n1 = 0, n2 = 1, nextTerm;

        document.write('Fibonacci Series:'+"<br>");

        for (let i = 1; i <= number; i++) {
            document.write(n1+"<br>");
            nextTerm = n1 + n2;
            n1 = n2;
            n2 = nextTerm;
        }
    </script>
</body>
```

Output:

Fibonacci Series:
0
1
1
2
3
5

Q.25) Write to print number in reverse order e.g.: number = 64728 ---> reverse =82746 in JS?

Ans.25)

```
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>
    <script>
        let num1 = 64728;
        let result = num1.toString().split('').reverse().join('');
        document.write(result);

    </script>
</body>
```

Output:

82746

Q.26) Write a program make a summation of given number (E.g., 1523 Ans: -11) in JS?

Ans.26)

```
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>
    <script>
        function sumOfDigit(num) {
            return num.toString().split("")
                .reduce((sum, digit) =>
                    sum + parseInt(digit), 0);
        }

        document.write("The summation of given number:",sumOfDigit(1523));


    </script>
</body>

</html>
```

Output:

The summation of given number:11

Q.27) Write a program you have to make a summation of first and last Digit.
(E.g., 1234 Ans: - 5) in JS?

Ans.27)

```
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initi
    <title>Document</title>
</head>

<body>
    <script>
        var num = 1234;
        var firstdigit;
        var lastdigit=num%10;
        var len = num.toString().length - 1;
        firstdigit = parseInt(num / Math.pow(10, len));
        var sum=firstdigit+lastdigit;
        document.write("firstdigit:"+firstdigit+"<br>")
        document.write("lastdigit:"+lastdigit+"<br>")

        document.write("sum of number:"+sum);

    </script>
</body>
```

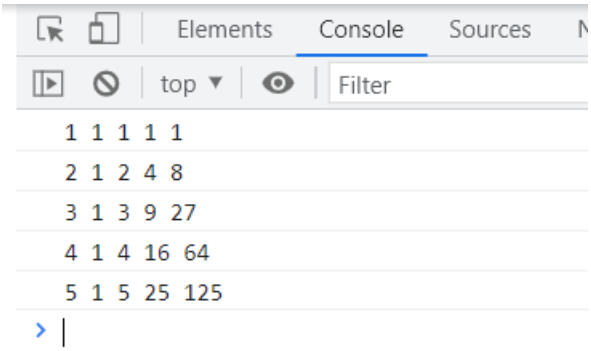
Output:

```
firstdigit:1
lastdigit:4
sum of number:5
```

Q.28) Use console.log() and escape characters to print the following pattern in JS?

1 1 1 1 1
2 1 2 4 8
3 1 3 9 27
4 1 4 16 64
5 1 5 25 125

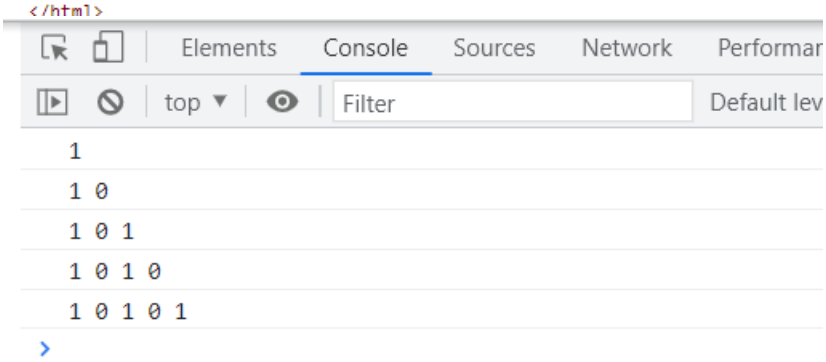
```
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>
    <script>
        console.log("1 1 1 1 1")
        console.log("2 1 2 4 8")
        console.log("3 1 3 9 27")
        console.log("4 1 4 16 64")
        console.log("5 1 5 25 125")
    </script>
</body>

</html>
```



Q.29) Use pattern in console.log in JS?

1) 1
   1 0
   1 0 1
   1 0 1 0
   1 0 1 0 1

```
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>
    <script>
        for (let i = 1; i <= 5; i++) {
            let row = '';
            for (let j = 1; j <= i; j++) {
                if (j % 2 === 0) {
                    row += '0 ';
                } else {
                    row += '1 ';
                }
            }
            console.log(row);
        }

    </script>
</body>

</html>
```

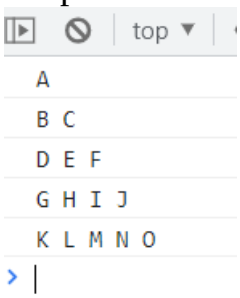


2)A
B C
D E F

G H I J
K L M N O

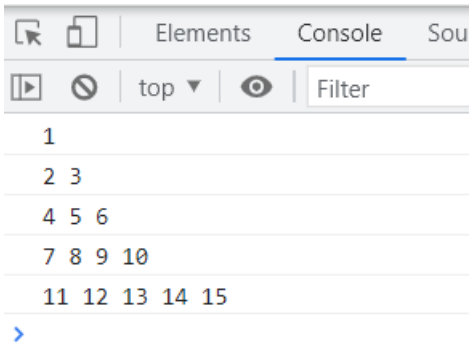
```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>
    <script>
        let alphabet = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ';
        let index = 0;

        for (let i = 1; i <= 5; i++) {
            let row = '';
            for (let j = 0; j < i; j++) {
                row += alphabet[index] + ' ';
                index++;
            }
            console.log(row.trim());
        }

    </script>
</body>

</html>
```

Output:



3) 1
2 3
4 5 6
7 8 9 10
11 12 13 14 15

```html
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, in
    <title>Document</title>
</head>

<body>
    <script>
        let num = 1;

        for (let i = 1; i <= 5; i++) {
            let row = '';

            // Adding numbers
            for (let j = 1; j <= i; j++) {
                row += num + ' ';
                num++;
            }
            console.log(row);
        }

    </script>
</body>

</html>
```



4) *

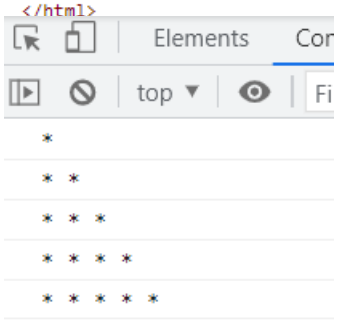
* *

* * *

* * * *

* * * * *

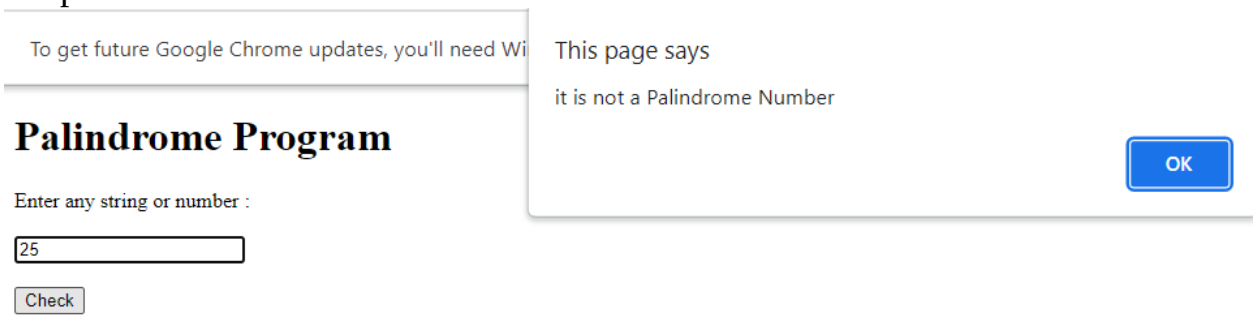
```
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge"
    <meta name="viewport" content="width=device-width, i
    <title>Document</title>
</head>

<body>
    <script>
        for (let i = 1; i <= 5; i++) {
            let row = '';
            for (let j = 1; j <= i; j++) {
                row += '* ';
            }
            console.log(row);
        }


    </script>
</body>

</html>
```



Q.30) Accept 3 numbers from user using while loop and check each numbers palindrome?

```
<body>
    <h1> Palindrome Program </h1>
    <div class="palindrome">
        <label> Enter any string or number : </label> <br> <br>
        <input id="palindrome"> <br> <br>
        <button type="button" onclick="palindrome()"> Check </button>
    </div>

    <script>
        function palindrome() {
            var a, b, no, temp = 0;
            no = Number(document.getElementById("palindrome").value);
            b = no;
            while (no > 0) {
                a = no % 10;
                no = parseInt(no / 10);
                temptemp = temp * 10 + a;
            }
            if (temp == b) {
                alert("It is a Palindrome Number");
            }
            else {
                alert("it is not a Palindrome Number");
            }
        }
    </script>
</body>

</html>
```

Output:



Q.31) day Write a JavaScript Program to display the current and time in the following format. Sample Output: Today is Friday. Current Time is 12 PM: 12 : 22 2 ?

Ans.31)

```
<script>
    var today = new Date();
    var day = today.getDay();
    var daylist = ["Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"];
    document.write("Today is: " + daylist[day] + "."+"<br>");
    var hour = today.getHours();
    var minute = today.getMinutes();
    var second = today.getSeconds();
    var prepand = (hour >= 12) ? " PM " : " AM ";
    hour = (hour >= 12) ? hour - 12 : hour;
    if (hour === 0 && prepand === ' PM ') {
        if (minute === 0 && second === 0) {
            hour = 12;
            prepand = ' Noon';
        } else {
            hour = 12;
            prepand = ' PM';
        }
    }
    if (hour === 0 && prepand === ' AM ') {
        if (minute === 0 && second === 0) {
            hour = 12;
            prepand = ' Midnight';
        } else {
            hour = 12;
            prepand = ' AM';
        }
    }
    document.write("Current Time: " + hour + prepand + " : " + minute + " : " + second);
</script>
```

Output:

Today is: Saturday.
Current Time: 6 PM : 51 : 4

Q.32) Write a JavaScript program to get the current date?

Ans.32)

```
<body>
    <script>
        let date=new Date().toDateString();
        document.write(date);
    </script>
</body>

</html>
```

Output:

Sat Mar 30 2024

Q.33) Write a JavaScript program to compare two objects?
Ans.33)

```
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Comparing two objects</title>
</head>

<body>
    <script>
        const Fruit1 = { fruit: 'kiwi' }; // Creating Fruit1 object
        const Fruit2 = { fruit: 'kiwi' }; // Creating Fruit2 object

        // Performing JSON.Stringify and === operator
        document.write(JSON.stringify(Fruit1) === JSON.stringify(Fruit2));

    </script>
</body>
```

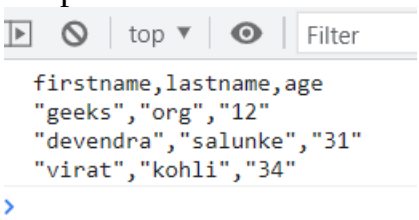
Output:

true

Q.34) Write a JavaScript program to convert an array of objects into CSV string?
Ans.34)

```
<script>
    const objectToCsv = function (data) {
        const csvRows = [];
        const headers = Object.keys(data[0]);
        csvRows.push(headers.join(','));

        // Loop to get value of each objects key
        for (const row of data) {
            const values = headers.map(header => {
                const val = row[header]
                return `"${val}"`;
            });
            csvRows.push(values.join(','));
        }
        return csvRows.join('\n');
    };
    const data = [{
        "firstname": "geeks",
        "lastname": "org",
        "age": 12
    },
    {
        "firstname": "devendra",
        "lastname": "salunke",
        "age": 31
    },
    {
        "firstname": "virat",
        "lastname": "kohli",
        "age": 34
    },
    ];
    const csvData = objectToCsv(data);
    console.log(csvData);
</script>
```

Output:



Q.35) Write a JavaScript program to capitalize first letter of a string?

Ans.35)

```
<body>
    <script>
        function capitalizeFLetter() {
            let string = 'welcome';
            document.write(string[0].toUpperCase() +
                string.slice(1));
        }
        capitalizeFLetter()

    </script>
```

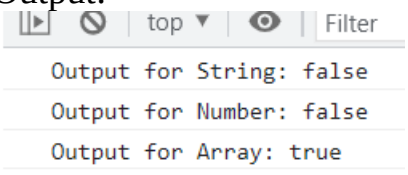
Output:

Welcome

Q. 45)Write a JavaScript program to determine if a variable is array?

```
<script>
    function checkArray() {
        let str = 'This is a string';
        let num = 25;
        let arr = [10, 20, 30, 40];

        let ans = Array.isArray(str);
        console.log("Output for String: " + ans);

        ans = Array.isArray(num);
        console.log("Output for Number: " + ans);

        ans = Array.isArray(arr);
        console.log("Output for Array: " + ans);
    }

    checkArray();


</script>
```

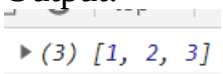
Output:



Q.46)Write a JavaScript program to clone an array?

```
<body>
    <script>
        const originalArray = [1, 2, 3];
        const clonedArray = Array.from(originalArray);
        console.log(clonedArray);

    </script>


</body>

</html>
```

Output:



Q.47)What is the drawback of declaring methods directly in JavaScript objects?
A-47) One potential drawback of declaring methods directly in JavaScript objects is that it can lead to increased memory usage, especially when you have multiple instances of the object.

```html
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>
    <script>
        // Initialize an object
        const employees = {
            boss: 'Michael',
            secretary: 'Pam',
            sales: 'Jim',
            accountant: 'Oscar'
        };

        // Get the keys of the object
        const keys = Object.keys(employees);

        console.log(keys);
    </script>
```

Output:

```
(4) ['boss', 'secretary', 'sales', 'accountant']
```

Q.48) Print the length of the string on the browser console using console.log()?
Ans.48)

```html
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>
    <script>
        const myString = "bluebells";
        console.log(myString);
        console.log(myString.length);

    </script>


</body>

</html>
```

Output:

```
bluebells
9
>
```

Q.49 Change all the string characters to capital letters using toUpperCase() method?
Ans.49)

```
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>
    <script>
        let text = "Hello World!";
        let result = text.toUpperCase();
        document.write(result);

    </script>
```

Output:

HELLO WORLD!

Q.50)What is the drawback of declaring methods directly in JavaScript objects?
Ans.50)

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>
    <script>
        function Person(first, last, age, eye) {
            this.firstName = first;
            this.lastName = last;
            this.age = age;
            this.eyeColor = eye;
        }

        const myFather = new Person("John", "Doe", 50, "blue");
        const myMother = new Person("Sally", "Rally", 48, "green");

        document.write("My father is " + myFather.age + ". My mother is " + myMother.age);

    </script>


</body>
```

Output:

My father is 50. My mother is 48

Q.51) Write a JavaScript program to get the current date.
Expected Output: mm-dd-yyyy, mm/dd/yyyy or dd-mm-yyyy, dd/mm/yyyy?
Ans.51)

```html
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>
    <script>
        let today = new Date();
        console.log(today);

        let dd = today.getDate();
        let mm = today.getMonth() + 1;

        let yyyy = today.getFullYear();

        if (dd < 10) {
            dd = '0' + dd;
        }
        if (mm < 10) {
            mm = '0' + mm;
        }
        today = dd + '/' + mm + '/' + yyyy;

        console.log(today);

    </script>
</body>
```

Output:

```
Sat Mar 30 2024 20:57:26 GMT+0530 (India Standard Time)
30/03/2024
```

Q.52) Use indexOf to determine the position of the first occurrence of a in 30 Days Of JavaScript?

Ans.52)

```html
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>
    <script>
        var sentence= "30 Days of Javascript";
        var position=sentence.indexOf('a');
        document.write("The first occurence of 'a' is at position:"+position);
    </script>
</body>
```

Output:

```
The first occurence of 'a' is at position:4
```

Q.53) Use lastIndexOf to determine the position of the last occurrence of a in 30 Days Of JavaScript?

Ans.53)

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>

<body>
    <script>
        var sentence= "30 Days of Javascript";
        var position=sentence.lastIndexOf('a');
        document.write("The last occurence of 'a' is at position:"+position);
    </script>
</body>
```

Output:

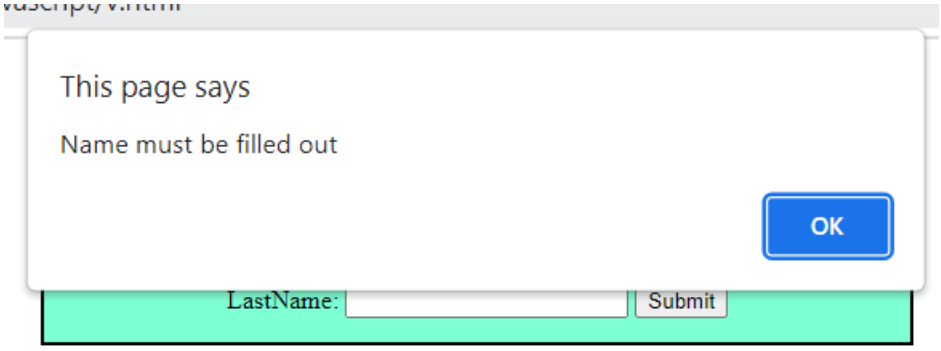
The last occurence of 'a' is at position:14

Q.54)Form Validation in JS?

Ans.54) Form validation is the process of verifying that the data entered into an HTML form is accurate, complete, and meets the specified criteria.

```
        display: block;
        align-items: center;
        padding-bottom: 15px;
        background-color: aquamarine;
      }
    h2{
      text-align: center;
    }
    .formcontent{
      text-align: center;
    }
    </style>
</head>

<body>
    <script>
        function validateForm() {
            let x = document.forms["myForm"]["fname"].value;
            if (x == "") {
                alert("Name must be filled out");
                return false;
            }
        }
    </script>
    <div class="container-fluid">
        <h2>JavaScript Validation</h2>

        <form name="myForm" action="" onsubmit="return validateForm()" method="post" class="formcontent">
            FirstName: <input type="text" name="fname">
            <input type="submit" value="Submit"><br><br>
            LastName: <input type="text" name="lname">
            <input type="submit" value="Submit">
        </form>
    </div>
</body>
```

Output:



Q.55)Form in Email, number, Password, Validation?

Ans.55)

```
<h2>Registration Form</h2>
<form id="registrationForm" onsubmit="return validateForm()" class="formcontent">
  <label for="email">Email:</label>
  <input type="email" id="email" name="email" required><br><br>
  <label for="phone">Phone Number:</label>
  <input type="tel" id="phone" name="phone" pattern="[0-9]{10}" required><br><br>
  <label for="password">Password:</label>
  <input type="password" id="password" name="password" required><br><br>
  <input type="submit" value="Submit">
</form>

<script>
  function validateForm() {
    var email = document.getElementById('email').value;
    var phone = document.getElementById('phone').value;
    var password = document.getElementById('password').value;
    // Email validation
    var emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;
    if (!emailRegex.test(email)) {
      alert("Please enter a valid email address");
      return false;
    }
    // Phone number validation
    var phoneRegex = /^\d{10}$/;
    if (!phoneRegex.test(phone)) {
      alert("Please enter a 10-digit phone number");
      return false;
    }
     // Password validation
    if (password.length < 6) {
      alert("Password must be at least 6 characters long");
      return false;
    }
    return true; // Form submission allowed
  }
</script>
```

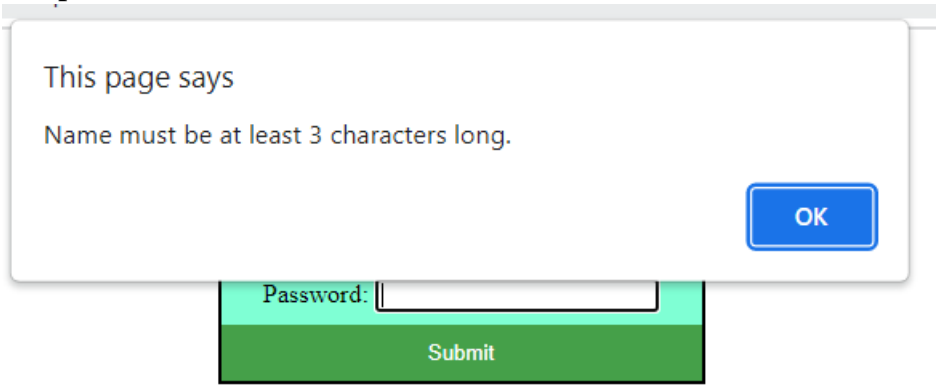
Output:

Q.56)Dynamic Form Validation in JS?

Ans.56)

```
        margin-top: 8px;
    }

    input[type='submit']:hover {
        background-color: #45a049;
    }
</style>
/head>

body>
<form id='myForm'>
    <h2>Dynamic Form</h2>
    <label for='txtName'>Name:</label>
    <input type='text' id='txtName'><br><br>
    <label for='txtEmail'>Email:</label>
    <input type='email' id='txtEmail'><br><br>
    <label for='txtPassword'>Password:</label>
    <input type='password' id='txtPassword'><br>
    <input type='submit' value='Submit'>
</form>
<script>
    document.getElementById('myForm').addEventListener('submit', function (event) {
        event.preventDefault();

        // add this code
        const name = document.getElementById('txtName').value;
        if (name.length < 3) {
            alert('Name must be at least 3 characters long.');
            return;
        }
    });
</script>
```

Output:



Q.57) how many type of JS Event? How to use it?
Ans.57) JavaScript has Event to provide a dynamic interface to a webpage. These events are
hooked to element in the Document object model.

- ❖ Common JavaScript Events
- ➢ Mouse Events: Event Performed. Event Handler. Description. click. onclick. ...
- ➢ Keyboard Events: Event Performed. Event Handler. Description. Keydown & Keyup. ...
- ➢ Form Events: Event Performed. Event Handler. Description. focus. ...
- ➢ Window or Document Events: Event Performed. Event Handler. Description. load.

- ❖ onclick Event Type : This is the most frequently used event type which occurs when a user clicks the left button of his mouse. You can put your validation, warning etc., against this event type.

```
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Form Validation</title>
</head>

<body>
    <h1>JavaScript HTML Events</h1>
    <h2>The onclick Attribute</h2>

    <button onclick="document.getElementById('demo').innerHTML=Date()">The time is?</button>

    <p id="demo"></p>


</body>

</html>
```

Output:

JavaScript HTML Events

The onclick Attribute

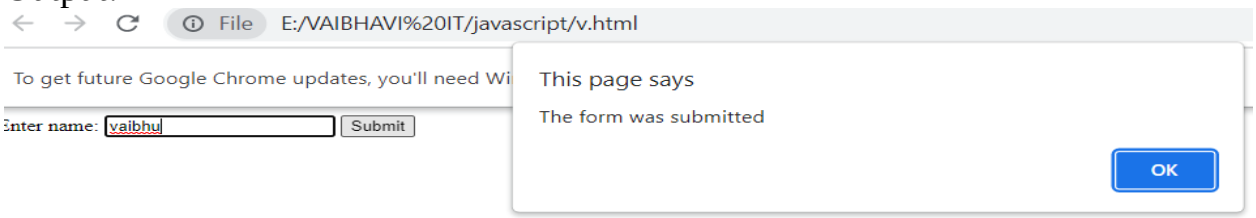
The time is?

Sat Mar 30 2024 22:47:01 GMT+0530 (India Standard Time)

- ❖ onsubmit Event type :onsubmit is an event that occurs when you try to submit a form. You can put your form validation against this event type.

```
    <title>Form Validation</title>

</head>

<body>
    <form action="" onsubmit="myFunction()">
        Enter name: <input type="text" name="fname">
        <input type="submit" value="Submit">
    </form>

    <script>
        function myFunction() {
            alert("The form was submitted");
        }
    </script>
</body>

</html>
```

Output:



Q.60) What is Bom vs Dom in JS?
Ans.60)

Document Object Model (DOM)	Browser Object Model (BOM)
It mainly focuses on the structure of the displayed document.	It mainly focuses on browser-specific functionality.
It facilitates a standardized interface to access and modify the elements and content of an HTML or XML document.	It allows JavaScript to interact with browser features beyond the scope of manipulating the document structure.
When an HTML document gets loaded in the browser, then it becomes a document object.	In this case, the window object will be created automatically by the browser.

Document Object Model (DOM)	Browser Object Model (BOM)
It facilitates access and manipulation, along with dynamically updating the structure, content, and styling of the web document.	It facilitates the different functionality for governing the browser window, handling the navigation, managing history, and accessing browser-related information.
It provides direct access control to the content of the web document, along with permitting the traversal and modification of its elements and attributes.	It doesn't have any access to the content of the web document directly.

Q.61) Array vs object defences in JS?
Ans.61)

Array	Object
Arrays are a special type of variable that is *also* mutable and can *also* be used to store a list of values.	Objects represent a special data type that is mutable and can be used to store a collection of data.
We use arrays whenever we want to create and store a list of multiple items in a single variable.	Objects are used to represent a "thing" in your code. That could be a person, a car, a building, a book, a character in a game — basically anything that is made up or can be defined by a set of characteristics.
Arrays use zero-based indexing, so the first item in an array has an index of 0, the second item an index of 1, and so on.	Properties in objects can be accessed, added, changed, and removed by using either **dot** or **bracket** notation.

Q.62) Split the string into an array using split() Method?
Ans.62)

```
<body>
    <script>
        const str = 'The quick brown fox jumps over the lazy dog.';

        const words = str.split(' ');
        document.write(words[3]);
    </script>
</body>

</html>
```

Output:

fox

Q.63) Check if the string contains a word Script using includes() method?
Ans.63)

```
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Form Validation</title>

</head>

<body>
    <script>
        let text = "Hello world, welcome to the universe.";
        let result = text.includes("world");
        document.write(result);
    </script>
</body>

</html>
```

Output:

true

Q.64) Change all the string characters to lowercase letters using to Lowercase () Method.
Ans.64)

```
<body>
    <script>
        let text = "Hello world, welcome to the universe.";
        let result = text.toLowerCase();
        document.write(result);
    </script>
</body>

</html>
```

Output:

hello world, welcome to the universe.

Q.65) What is Character at index 15 in ’30 Days of JavaScript’ string? Use char At () method.
Ans.65)

```
<body>
    <script>
        let text = "30 Days of JavaScript";
        let result = text.charAt(15);
        document.write(result);
    </script>
</body>

</html>
```

Output:

S

Q.66) copy to one string to another string in JS?
Ans.66)

```
<body>
    <script>
        let str1 = "Hello";
        let str2 = str1.slice();
        document.write(str2);

    </script>
</body>
```

Output:

Hello