- What is JavaScript?
  - JavaScript is a scripting language used to develop web pages. Developed in Netscape, JS allows developers to create a dynamic and interactive web page to interact with visitors and execute complex actions. It also enables users to load content into a document without reloading the entire page

- What is the use of isNaN function?
  - sNaN() The isNaN() function determines whether a value is NaN , first converting the value to a number if necessary.

- What is negative Infinity?
  - NEGATIVE_INFINITY is a special numeric value that is returned when an arithmetic operation or mathematical function generates a negative value greater than the largest representable number in JavaScript

- Which company developed JavaScript?
  - JavaScript was created at Netscape Communications by Brendan Eich in 1995. Netscape and Eich designed JavaScript as a scripting language for use with the company's flagship web browser, Netscape Navigator.

- What are undeclared and undefined variables?

- It occurs when a variable has been declared but has not been assigned any value. Undefined is not a keyword. Undeclared: It occurs when we try to access any variable that is not initialized or declared earlier using the var or const keyword.

• What is the difference between ViewState and SessionState?

- The basic difference between these two is that the ViewState is to manage state at the client's end, making state management easy for end-user while SessionState manages state at the server's end, making it easy to manage content from this end too. ViewState: It is maintained at only one level that is page-level.

• What is === operator?

- The strict equality (===) operator checks whether its two operands are equal, returning a Boolean result. Unlike the equality operator, the strict equality operator always considers operands of different types to be different.

• How can the style/class of an element be changed?

- The class name is used as a selector in HTML which helps to give some value to the element attributes. The document. getElementById() method is used to return the element in the document with the “id” attribute and the “className” attribute can be used to change/append the class of the element.

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <h2>
        Change class name of element
    </h2>
    <button class="default"
            onclick="changeClass()"
            id="myButton">
      Click Here!
     </button><br>
    <p id="myPara">
        Old class name: default
    </p>
    <script>
        function changeClass() {
            document.getElementById('myButton').className = "changedClass";
            let button_class = document.getElementById('myButton').className;
            document.getElementById('myPara').innerHTML = "New class name: " + button_class;
        }
    </script>
</body>

</html>
```

- How to read and write a file using JavaScript?
  - The basic difference between these two is that the ViewState is to manage state at the client's end, making state management easy for end-user while SessionState manages state at the server's end, making it easy to manage content from this end too. ViewState: It is maintained at only one level that is page-level.

- What are all the looping structures in JavaScript?
  - For Loop

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <script>
        for (let i = 0; i <= 5; i++) {
            document.write("Hiii" + "<hr>")
        }
    </script>
</body>
</html>
```

- While Loop

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <script>
            i = 0
        while (i < 5) {
            document.write("hii" + "<hr>")
            i++
        }
    </script>
</body>
</html>
```

- Do While

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <script>
        i=6
        do {
            document.write("Hii"+"<hr>")
            i++
        } while (i<5);
    </script>
</body>
</html>
```

- How can you convert the string of any base to an integer in JavaScript?

  - In JavaScript parseInt() function (or a method) is used to convert the passed-in string parameter or value to an integer value itself. This function returns an integer of the base which is specified in the second argument of the parseInt() function.

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

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

<body>
    <script>
        function convertStoI() {
    let a = "100";
    let b = parseInt(a);
    console.log("Integer value is" + b);
    let d = parseInt("3 11 43");
    console.log('Integer value is ' + d);
}
    convertStoI();
</script>
</body>
</html>
```

- What is the function of the delete operator?
  - The delete operator removes a property from an object. If the property's value is an object and there are no more references to the object, the object held by that property is eventually released automatically.

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <script>
        let emp = {
        firstName: "Raj",
        lastName: "Kumar",
        salary: 40000
    }

    console.log(delete emp.salary);
    console.log(emp);

    </script>
</body>
</html>
```

➢

- What are all the types of Pop up boxes available in JavaScript?

1. Alert Box

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <button onclick="a()">XYZ</button>
    <div id="one"></div>
   <script>
    function a() {
        alert("How Are You!!")
    }
   </script>
</body>
</html>
```

➢

2. Confirm Box

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <button onclick="a()">XYZ</button>
    <div id="one"></div>
   <script>
    function a() {
        var txt
        if (confirm("How Are You!")) {
            txt="OK"
        } else {
            txt="NO"
        }
        document.getElementById("one").innerHTML=txt
    }
   </script>
</body>
</html>
```

➢

3. Prompt Box

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <button onclick="a()">XYZ</button>
    <div id="one"></div>
   <script>
    function a() {
        var txt
        var per=(prompt("How Are You!!"))
        if (per ==null || per == "") {
            txt="plz enter your name"
        } else {
            txt="Hii " +per+" How Are You"
        }
        document.getElementById("one").innerHTML=txt
    }
   </script>
</body>
</html>
```

- What is the use of Void (0)?
  - avaScript void 0 means returning undefined (void) as a primitive value. You might come across the term “JavaScript:void(0)” while going through HTML documents. It is used to prevent any side effects caused while inserting an expression in a web page.

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <p>sdsdfdsfdsfds</p>
    <a href="javascript:void(alert('warning'))">clcixk</a>
</body>
</html>
```

• How can a page be forced to load another page in JavaScript?

- We can use window. location property inside the script tag to forcefully load another page in Javascript. It is a reference to a Location object that is it represents the current location of the document. We can change the URL of a window by accessing it.
- Syntex

```
 script>
window.location = <Path / URL>
 </script>
```

• What are the disadvantages of using innerHTML in JavaScript?

- Disadvantages of using innerHTML property in JavaScript:

- The use of innerHTML very slow: The process of using innerHTML is much slower as its contents as slowly built, also already parsed contents and elements are also re-parsed which takes time.
- Preserves event handlers attached to any DOM elements: The event handlers do not get attached to the new elements created by setting innerHTML automatically. To do so one has to keep track of the event handlers and attach it to new elements manually. This may cause a memory leak on some browsers.
- Content is replaced everywhere: Either you add, append, delete or modify contents on a webpage using innerHTML, all contents is replaced, also all the DOM nodes inside that element are reparsed and recreated.
- Appending to innerHTML is not supported: Usually, += is used for appending in JavaScript. But on appending to an Html tag using innerHTML, the whole tag is re-parsed.
- Old content replaced issue: The old content is replaced even if object.innerHTML = object.innerHTML + 'html' is used instead of object.innerHTML += 'html'. There is no way of appending without reparsing the whole innerHTML. Therefore, working with innerHTML becomes very slow. String concatenation just does not scale when

dynamic DOM elements need to be created as the plus' and quote openings and closings becomes difficult to track.

- Can break the document: There is no proper validation provided by innerHTML, so any valid HTML code can be used. This may break the document of JavaScript. Even broken HTML can be used, which may lead to unexpected problems.
- Can also be used for Cross-site Scripting(XSS): The fact that innerHTML can add text and elements to the webpage, can easily be used by malicious users to manipulate and display undesirable or harmful elements within other HTML element tags. Cross-site Scripting may also lead to loss, leak and change of sensitive information.

- Create password field with show hide functionalities

➢

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

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>

<p>Click the radio button to toggle between password visibility:</p>

Password: <input type="password" value="FakePSW" id="myInput"><br><br>
<input type="checkbox" onclick="myFunction()">Show Password

<script>
function myFunction() {
  var x = document.getElementById("myInput");
  if (x.type === "password") {
    x.type = "text";
  } else {
    x.type = "password";
  }
}
</script>

</body>
</html>
```

- Create basic math operation in JS

- Create result

```
<html>
<head></head>
<body>
    <h2 align="center">Marksheet for Information technology</h2>
    <h3 align="center">Enter marks</h3>
    <table align="center" cellspacing="5" cellpadding="5">
        <form name="form1">
        <tr>
            <td><b>1. C Language</b></td>
            <td><INPUT type="textbox" name="txtNum1"> </br></td>
        </tr>
        <tr>
            <td><b>2. C++ Language</b></td>
            <td><INPUT type="textbox" name="txtNum2"> </br></td>
        </tr>
        <tr>
            <td><b>3. Database</b></td>
            <td><INPUT type="textbox" name="txtNum3"> </br></td>
        </tr>
        <tr>
            <td><b>4. HTML</b></td>
            <td><INPUT type="textbox" name="txtNum4"> </br></td>
        </tr>
        <tr>
            <td><b>5. CSS</b></td>
            <td><INPUT type="textbox" name="txtNum5"> </br></td>
        </tr>
        <tr>
            <td><b>6. php</b></td>
            <td><INPUT type="textbox" name="txtNum6"> </br></td>
        </tr>
        <tr>
            <td><b>7. Core Java</b></td>
            <td><INPUT type="textbox" name="txtNum7"> </br></td>
        </tr>
```

```
        <tr>
            <td colspan="2" align="center"><INPUT type="button" Value="Result" onClick='Calculate()'></td>
        </tr>
    </form>
    </table>
</body>
<script>
    var a;
    var b;
    var c;
    var d;
    var e;
    var f;
    var g;
    var result=0;
    var avg=0;
    function Calculate()
    {
     a=parseInt(form1.txtNum1.value);
     b=parseInt(form1.txtNum2.value);
     c=parseInt(form1.txtNum3.value);
     d=parseInt(form1.txtNum4.value);
     e=parseInt(form1.txtNum5.value);
     f=parseInt(form1.txtNum6.value);
     g=parseInt(form1.txtNum7.value);
     result=a+b+c+d+e+f+g;
     avg = result/7;
    document.write("The Total is  :"+ "<b>" + result  +  "<b>" + " / 350" + "</br>");
    document.write("The Percentage is  :"+ "<b>" + avg  +  "<b>" + "</br>");
    }
    </script>
</html>
```

- Create a slider using JavaScript
  - ➢ CSS

```html
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <style>
        * {
            box-sizing: border-box
        }
        body {
            font-family: Verdana, sans-serif;
            margin: 0
        }
        .mySlides {
            display: none
        }
        img {
            vertical-align: middle;
        }
        /* Slideshow container */
        .slideshow-container {
            max-width: 1000px;
            position: relative;
            margin: auto;
        }
        /* Next & previous buttons */
        .prev,
        .next {
            cursor: pointer;
            position: absolute;
            top: 50%;
            width: auto;
            padding: 16px;
            margin-top: -22px;
            color: white;
            font-weight: bold;
            font-size: 18px;
```

```
    transition: 0.6s ease;
    border-radius: 0 3px 3px 0;
    user-select: none;
}
/* Position the "next button" to the right */
.next {
    right: 0;
    border-radius: 3px 0 0 3px;
}
/* On hover, add a black background color with a little bit see-through */
.prev:hover,
.next:hover {
    background-color: rgba(0, 0, 0, 0.8);
}
/* Caption text */
.text {
    color: #f2f2f2;
    font-size: 15px;
    padding: 8px 12px;
    position: absolute;
    bottom: 8px;
    width: 100%;
    text-align: center;
}
/* Number text (1/3 etc) */
.numbertext {
    color: #f2f2f2;
    font-size: 12px;
    padding: 8px 12px;
    position: absolute;
    top: 0;
}
/* The dots/bullets/indicators */
```

```
    .dot {
        cursor: pointer;
        height: 15px;
        width: 15px;
        margin: 0 2px;
        background-color: #bbb;
        border-radius: 50%;
        display: inline-block;
        transition: background-color 0.6s ease;
    }
    .active,
    .dot:hover {
        background-color: #717171;
    }
    /* Fading animation */
    .fade {
        animation-name: fade;
        animation-duration: 1.5s;
    }
    @keyframes fade {
        from {
            opacity: .4
        }

        to {
            opacity: 1
        }
    }
    /* On smaller screens, decrease text size */
    @media only screen and (max-width: 300px) {
        .prev,
        .next,
        .text {
            font-size: 11px
        }
    }
</style>
```

➢ HTML

```
</head>
<body>
    <div class="slideshow-container">
        <div class="mySlides fade">
            <div class="numbertext">1 / 3</div>
            <img src="https://www.w3schools.com/w3css/img_snow_wide.jpg" style="width:100%">
            <div class="text">Caption Text</div>
        </div>
        <div class="mySlides fade">
            <div class="numbertext">2 / 3</div>
            <img src="https://upload.wikimedia.org/wikipedia/commons/thumb/e/ef/Great_nature_landscape_Pakistan.jpg/
            1200px-Great_nature_landscape_Pakistan.jpg?20160530195450"
                style="width:100%">
            <div class="text">Caption Two</div>
        </div>
        <div class="mySlides fade">
            <div class="numbertext">3 / 3</div>
            <img src="https://images.unsplash.com/photo-1549558549-415fe4c37b60?auto=format&fit=crop&q=80&w=1000&ixlib=rb-4.0.3&
            ixid=M3wxMjA3fDB8MHxleHBsb3JlLWZlZWR8NXx8fGVufDB8fHx8fA%3D%3D"
                style="width:100%">
            <div class="text">Caption Three</div>
        </div>
        <a class="prev" onclick="plusSlides(-1)">❮</a>
        <a class="next" onclick="plusSlides(1)">❯</a>
    </div>
    <br>
    <div style="text-align:center">
        <span class="dot" onclick="currentSlide(1)"></span>
        <span class="dot" onclick="currentSlide(2)"></span>
        <span class="dot" onclick="currentSlide(3)"></span>
    </div>
```

- JAVA SCRIPT

```
    <script>
        let slideIndex = 1;
        showSlides(slideIndex);
        function plusSlides(n) {
            showSlides(slideIndex += n);
        }
        function currentSlide(n) {
            showSlides(slideIndex = n);
        }
        function showSlides(n) {
            let i;
            let slides = document.getElementsByClassName("mySlides");
            let dots = document.getElementsByClassName("dot");
            if (n > slides.length) { slideIndex = 1 }
            if (n < 1) { slideIndex = slides.length }
            for (i = 0; i < slides.length; i++) {
                slides[i].style.display = "none";
            }
            for (i = 0; i < dots.length; i++) {
                dots[i].className = dots[i].className.replace(" active", "");
            }
            slides[slideIndex - 1].style.display = "block";
            dots[slideIndex - 1].className += " active";
        }
    </script>
</body>

</html>
```

-