WEB DEVELOPMENT

What happens when we type google.com?

1.The browser checks the cache for a DNS record to find the corresponding IP address of google.com.
2.DNS(Domain Name System) is a database that maintains the name of the website (URL) and the corresponding IP address.
3.In order to find the DNS record, the browser checks four caches.
First, it checks the browser cache.The browser maintains a repository of DNS records for a fixed duration for websites you have previously visited.
Second, the browser checks the OS cache. Because the OS also maintains a cache of DNS records.
Third, it checks the router cache. If it's not found on your computer, the browser would communicate with the router that maintains its' own cache of DNS records.
Fourth, it checks the ISP cache. If all steps fail, the browser would move on to the ISP.
4.If the requested URL is not in the cache, ISP's DNS server initiates a DNS query to find the IP address of the server that hosts google.com.
5.Browser initiates a TCP connection with the server.
6. Browser sends a HTTP request to the server.
7. Server handles the incoming request
8.Browser receives the HTTP response
9.Client interaction with server

How does your website works out of the internet?

GoDaddy.com/namecheap.com---> check available domain name and buy.

we will get an IP address for selected domain name

when we buy the domain name we need to tell the register to have the IP address in DNS server.

there are 2 DNS server: primary and secondary

so the web hosting company provides the DNS server in which they can have our IP address.

JAVASCRIPT

JavaScript is a very powerful client-side scripting language.
JavaScript is used mainly for enhancing the interaction of a user with the webpage.
In other words, you can make your webpage more lively and interactive, with the help of JavaScript.

JavaScript is also being used widely in game development and Mobile application development.

How to run javascript?

JavaScript cannot run on its own.
In fact, the browser is responsible for running JavaScript code.
When a user requests an HTML page with JavaScript in it, the script is sent to the browser and it is up to the browser to execute it.
The main advantage of JavaScript is that all modern web browsers support JavaScript.

Tools we need to run Javascript?

To start with, you need a text editor to write your code and a browser to display the web pages you develop.

Ex: Notepad++, Visual Studio Code, Sublime Text, Atom or any other text editor Google Chrome, Firefox, Microsoft Edge, Internet Explorer etc.

---

difference between java and javascript.

java -
object oriented programming language.
It creates applications that run in a virtual machine or browser.
Java code needs to be compiled.

Javascript -
object oriented scripting language.
The code is run on a browser only.
JavaScript code are all in the form of text.

What is JavaScript?

JavaScript is a lightweight, interpreted programming language with object-oriented capabilities that allows you to build interactivity into otherwise static HTML pages. The general-purpose core of the language has been embedded in Netscape, Internet Explorer, and other web browsers.

What are the features of JavaScript?

- It is a lightweight, interpreted programming language.
- It is designed for creating network-centric applications.
- It is complementary to and integrated with Java.
- It is an open and cross-platform scripting language.

-difference between javascript and jscript.

almost similar but

javascript - was developed by Netscape.
Jscript - reverse engineered javascript.

What are the advantages of JavaScript?

- Less server interaction − You can validate user input before sending the page off to the server. This saves server traffic, which means less load on your server.
- Immediate feedback to the visitors − They don't have to wait for a page reload to see if they have forgotten to enter something.
- Increased interactivity − You can create interfaces that react when the user hovers over them with a mouse or activates them via the keyboard.
- Richer interfaces − You can use JavaScript to include such items as drag-and-drop components and sliders to give a Rich Interface to your site visitors.

What is Callback?

A callback is a plain JavaScript function passed to some method as an argument or option. It is a function that is to be executed after another function has finished executing, hence the name 'call back'. In JavaScript, functions are objects. Because of this, functions can take functions as arguments, and can be returned by other functions.

What are the variable naming conventions in JavaScript?

The following rules are to be followed while naming variables in JavaScript:

1. You should not use any of the JavaScript reserved keyword as variable name. For example, break or boolean variable names are not valid.
2. JavaScript variable names should not start with a numeral (0-9). They must begin with a letter or the underscore character. For example, 123name is an invalid variable name but _123name or name123 is a valid one.
3. JavaScript variable names are case sensitive. For example, Test and test are two different variables.

How to create a cookie using JavaScript?

The simplest way to create a cookie is to assign a string value to the document.cookie object, which looks like this-

Syntax :

document.cookie = "key1 = value1; key2 = value2; expires = date";

How to read a cookie using JavaScript?

Reading a cookie is just as simple as writing one, because the value of the document.cookie object is the cookie. So you can use this string whenever you want to access the cookie.

- The document.cookie string will keep a list of name = value pairs separated by semicolons, where name is the name of a cookie and value is its string value.
- You can use strings' split() function to break the string into key and values.

How to delete a cookie using JavaScript?

If you want to delete a cookie so that subsequent attempts to read the cookie return nothing, you just need to set the expiration date to a time in the past. You should define the cookie path to ensure that you delete the right cookie. Some browsers will not let you delete a cookie if you don't specify the path.

What is the difference between window & document in JavaScript?

Window	Document
JavaScript window is a global object which holds variables, functions, history, location.	The document also comes under the window and can be considered as the property of the window.

How do we add JavaScript onto a web page?

if the script code is very short

```
<head>
<title>Page Title</title>
<script language="JavaScript" type="text/javascript">
  var name = "Vikas Ahlawta"
  alert(name);
</script>
</head>
```

if the code is long

```
<head>
<title>Page Title</title>
<script type="text/javascript" src="myjavascript.js"></script>
</head>
```

Simple javascript program

```
<script>
    alert("Hello World!");
</script>
```

Is JavaScript case sensitive?

yes. getElementById is not the same as getElementbyID.

Data types in javascript

There are seven data types: Boolean, Null, Undefined, Number, String, Symbol, Object. The first six are primitives, meaning data types that aren’t objects and do not have any methods.

What is an object in javascript?

Objects are the data structure in javascript that let us store data about a particular thing and helps us to track the data by using a key.

Ex;
Var person={
Name: “shobana”,
Age: 28
};

Use of isNaN function

IsNan function returns true if the argument is not a number otherwise it returns false.

What is the difference between textContent and innerHTML?

textContent is a concatenation of the text content of a node and its descendant. It is the fastest of the two. innerHTML returns the HTML. It parses the text into HTML, which makes it slower.

What is the DOM?

DOM stands for the Documentation Object Models, and it is a World Wide Web Consortium standard.It defines a standard for accessing documents, and it can be used to access and change the content of the HTML.

What are the pop-up boxes types available in JavaScript?

There are 3 pop up boxes: Alert, Confirm and Prompt.
Alert box:
An alert box is often used if you want to make sure information comes through to the user.
When an alert box pops up, the user will have to click "OK" to proceed.

window.alert(“welcome”);

Confirm box:
A confirm box is often used if you want the user to verify or accept something.
When a confirm box pops up, the user will have to click either "OK" or "Cancel" to proceed.
If the user clicks "OK", the box returns true. If the user clicks "Cancel", the box returns false.

window.confirm();

Prompt Box

A prompt box is often used if you want the user to input a value before entering a page.
When a prompt box pops up, the user will have to click either "OK" or "Cancel" to proceed after entering an input value.
If the user clicks "OK" the box returns the input value. If the user clicks "Cancel" the box returns null.

window.prompt(“Please enter your name”);

What do the break and the continue statements do?

The break statement is used to exit a current loop. On the other hand, the continue statement continues with the next announcement in the loop.

What is negative infinity in JavaScript?

The result of a negative number divided by 0.

Explain the differences between var, const and let.

1. var - Var declaration has global scope
   - Var declaration can be updated and re-declared.

Ex:

```
var a=3;
a = 8;
```

var a= 7;

let - Let declaration has block scope
- Let declaration can be updated but not re-declared.

Ex:

```
{
let a = 9;
a = 10;
let a = 8;     //not allowed
}
```

const - const declaration has block scope
- const can not be updated or re-declared.

Ex:

```
{
const a = 3;
a=9;                    //not allowed
const a=4;              //not allowed
```

Explain the difference between null, undefined or undeclared variables.

Undefined variables have been declared, but no value exists for them.
Null is a value of variables, as well as a type of object.
Undeclared variables are those declared without the var, const or let keyword.

What are global variables? What are local variables?

Global variables can be accessible anywhere in the code. This variable does not have scope.

Local variables are restricted to where it is declared. If the local variables is declared inside the function then the local variable will be accessible inside that function.
It will not be available outside the function.

How to remove white space in javascript?

```
Var a = “ AB ”;
Var b = “ CD ”;
console.log(a.trim() + b.trim());
// ABCD
```

How to replace the string in javascript?

```
Var a =”hello shobana”;
console.log(a.replace(“hello”, “hey”));
```

What is substring(), substr(), slice()?

All are same methods.

```
var str = "shobana vijayakumar";
var res= str.substring(0,7);          0 is the start position 7 is the end position to extract
console.log(res);                          it will swap if first position is lesser
//shobana

var str = "shobana vijayakumar";
var res= str.substr(0,7);             method tells number of characters to return
console.log(res);
shobana

var str = "shobana vijayakumar";
var res= str.slice(0,7);              slice method will not swap the positions if first is greater
console.log(res);                        than the second.
shobana
```

What is an array in javascript?

An array is a variable that can store multiple values.

Ex:

```
Var arrVal = new Array();            ----> can add N number of value
Var arrVal = [];
Var arrVal = new Array(3);         ----> An array constructor, can store upto 3 numbers.


Var arr =[10,20,30];
console.log(arr[0]) // 10      first element
console.log(arr[arr.length-1]); //30 las element
```

Mutator array:
It will change the positions of an array
push()
pop()
unshift()
shift()
reverse()
sort() sort method will sort the strings but not numbers

To sort numbers:

```
var no=[4,7,1,9,2,0];
no.sort(function(a,b){
```

```
return a-b;
});
console.log(no);
// 0,1,2,4,7,9
```

splice()
array.splice(index, howmany, item1, ..., itemX)

Ex;

```
var fruits = ["Banana", "Orange", "Apple", "Mango"];
fruits.splice(2, 0, "Lemon", "Kiwi");
// Banana,Orange,Lemon,Kiwi,Apple,Mango
```

Non mutator array:
It will not change the position of an array.
Contains
indexOf()
lastIndexOf()

filter()
The filter() creates a new array. And populates that array with all the elements that meet the condition specified in a callback function.

Callback function syntax:

```
function callbackFunction(value,index,array)
```

How do javascript timers works?what is a drawback?

Timers allow you to execute code at a set time or repeatedly using an interval.

setTimeout(function , delay) - timer that calls a specific function after the delay.
setInterval(function, delay) - timer that calls a specific function repeatedly on the delay.
clearInterval(id) - used to stop a timer.

Timers can be tricky to use since they operate within a single thread, thus events queue up waiting to execute.

DOM event types?

The browser triggers many events.
some of the most common event types and event names:

- mouse events (MouseEvent): mousedown, mouseup, click, dblclick, mousemove, mouseover, mousewheel, mouseout, contextmenu
- touch events (TouchEvent): touchstart, touchmove, touchend, touchcancel
- keyboard events (KeyboardEvent): keydown, keypress, keyup
- form events: focus, blur, change, submit
- window events: scroll, resize, hashchange, load, unload

What is event bubbling?

Event bubbling describes the behaviour of events in child node and parent nodes in the DOM. that is the child node events are automatically passed to the parent nodes.
The benefit of this method is speed, because the code needs to traverse the DOM tree once. This is useful when you want to place more than one event listener on a DOM element since you can put just one event listener on all the elements, thus the code simplicity and reduction. One application of this is the creation of one event listener on a page's body element to respond to any click event that occurs within the pages body.

"1" + 4 + 4 = 144

1 + 8 + "7" = 97

Does javascript support foreach?

Yes. Javascript will support foreach. In javascript we will call it as for-in statement. The for-in loop will continue until the value exist in the array or object.

Ex:

```
Var arr = [1,2,3,4];
for(index in values){
console.log(values[index]);
}
```

What is function declaration and function expressions?

Function declaration:
Normal function

Hoisting will work.

function(){
Some code
}

Function expression:

Function assigned to a variable.
Hoisting will not work.

Var exp = function(){
Some code
};

What is variable hoisting in javascript?

Hoisting - means lift or raise.
In javascript hoisting means all function declarations are lift to the top of the code.

We can refer them before they are declared. Because they are raised to the top of the code.

Rule -1 :

```
hoisted();
Function hoisted(){
Var name =”laksha”;
console.log(name);
}
// laksha            function declaration gets hoisted.
```

Rule-2:

```
hoist();
Function hoist(){
console.log(name);
Var name =”shobana”;
}
// undefined            definition inside function will remain same so it won’t get hoisted.
```

Rule-3:

```
Hoist;
Var hoist = function(){
console.log(“shobana”);
}
```

//undefined function expression will not get hoisted.

What is Immediately invoked function expression in javascript? (IIFE)

The function expression is invoked at a time it is defined.

Ex:

```
var a = function(){
   console.log("shobana");
}();
//shobana

(function(){
   console.log("shobana");
})();                          -----> IIFE
//shobana
```

Closures in javascript?

A closure is a local variable for a function-kept alive after the function has returned.

EX:

```
function outer(){
   var a = 10, b=10;
function inner(){
   var c= a + b;
console.log(c);
}
return inner;
}
var func = outer();
func();
```

---

how can you create an Object in Javascript?

```
var emp = {
  name: "Zara",
  age: 10
};
```

How can you read properties of an Object in JavaScript?
You can write and read properties of an object using the dot notation as follows -

```
// Getting object properties
```

```
emp.name  // ==> Zara
emp.age   // ==> 10
// Setting object properties
emp.name = "Daisy"  // <== Daisy
emp.age  =  20      // <== 20
```

How can you create an Array in JavaScript?
You can define arrays using the array literal as follows -

```
var x = [];
var y = [1, 2, 3, 4, 5];
```

How to read elements of an array in JavaScript?
An array has a length property that is useful for iteration. We can read elements of an array as follows -

```
var x = [1, 2, 3, 4, 5];
for (var i = 0; i < x.length; i++) {
   // Do something with x[i]
}
```

How many types of functions JavaScript supports?

A function in JavaScript can be either named or anonymous.

How to define a anonymous function?

An anonymous function can be defined in similar way as a normal function but it would not have any name.

Can you pass a anonymous function as an argument to another function?

Yes! An anonymous function can be passed as an argument to another function.

What is arguments object in JavaScript?

JavaScript variable arguments represents the arguments passed to a function

---

How can you get the type of arguments passed to a function?
Using typeof operator, we can get the type of arguments passed to a function. For example -

```
function func(x){
   console.log(typeof x, arguments.length);
}
func();               //==> "undefined", 0
func(1);               //==> "number", 1
```

```
func("1", "2", "3");   //==> "string", 3
```

How can you get the total number of arguments passed to a function?

Using arguments.length property, we can get the total number of arguments passed to a function. For example -

```
function func(x){
   console.log(typeof x, arguments.length);
}
func();               //==> "undefined", 0
func(1);               //==> "number", 1
func("1", "2", "3");   //==> "string", 3
```

How can you get the reference of a caller function inside a function?

The arguments object has a callee property, which refers to the function you're inside of. For example -

```
function func() {
   return arguments.callee;
}
func();
```

Which type of variable among global and local, takes precedence over other if names are same?

A local variable takes precedence over a global variable with the same name.

What is callback?

A callback is a plain JavaScript function passed to some method as an argument or option. Some callbacks are just events, called to give the user a chance to react when a certain state is triggered.

All event listeners in javascript are callbacks.

Ex:
```
let a = function() {
    console.log("The function was called back.");
};
setTimeout(a, 2000);
```

Ex:
```
Var btn= document.getElementById(“button”);
btn.addEventListener(“click”, function(){
```

console.log(“The button was clicked.”)
});

Which built-in method returns the character at the specified index?

charAt() method returns the character at the specified index.

Which built-in method combines the text of two strings and returns a new string?

concat() method returns the character at the specified index.

Which built-in method calls a function for each element in the array?

forEach() method calls a function for each element in the array.

Which built-in method returns the index within the calling String object of the first occurrence of the specified value?

indexOf() method returns the index within the calling String object of the first occurrence of the specified value, or -1 if not found.

Which built-in method returns the length of the string?

length() method returns the length of the string.

Which built-in method reverses the order of the elements of an array?

reverse() method reverses the order of the elements of an array --the first becomes the last, and the last becomes the first.

Which built-in method sorts the elements of an array?

sort() method sorts the elements of an array.

Which built-in method returns the characters in a string beginning at the specified location?

substr() method returns the characters in a string beginning at the specified location through the specified number of characters.

Which built-in method returns the calling string value converted to lowercase?

toLowerCase() method returns the calling string value converted to lowercase.

Which built-in method returns the calling string value converted to uppercase?

toUpperCase() method returns the calling string value converted to upper case.

Which built-in method returns the string representation of the number's value?

toString() method returns the string representation of the number's value.

What are the variable naming conventions in JavaScript?

JavaScript variable names should not start with a numeral (0-9).
They must begin with a letter or the underscore character.
For example, 123test is an invalid variable name
but _123test is a valid one.

How typeof operator works?

The typeof operator evaluates to "number", "string", or "boolean" if its operand is a number, string, or boolean value and returns true or false based on the evaluation.

What typeof returns for a null value?

It returns "object".

Can you access Cookie using javascript?

JavaScript can also manipulate cookies using the cookie property of the Document object. JavaScript can read, create, modify, and delete the cookie or cookies that apply to the current web page.

How to create a Cookie using JavaScript?

The simplest way to create a cookie is to assign a string value to the document.cookie object, which looks like this -

Syntax -

document.cookie = "key1 = value1; key2 = value2; expires = date";

How to redirect a url using JavaScript?

It is very simple to do a page redirect using JavaScript at client side. To redirect your site visitors to a new page, you just need to add a line in your head section as follows -

```
<head>
<script type="text/javascript">
<!--
   window.location="http://www.newlocation.com";
```

//-->
</script>
</head>

How to print a web page using javascript?

JavaScript helps you to implement this functionality using print function of window object. The JavaScript print function window.print() will print the current web page when executed.

What is Date object in JavaScript?

The Date object is a datatype built into the JavaScript language. Date objects are created with the new Date( ).

Once a Date object is created, a number of methods allow you to operate on it. Most methods simply allow you to get and set the year, month, day, hour, minute, second, and millisecond fields of the object, using either local time or UTC (universal, or GMT) time.

What is Number object in JavaScript?

he Number object represents numerical date, either integers or floating-point numbers. In general, you do not need to worry about Number objects because the browser automatically converts number literals to instances of the number class.

Syntax -

Creating a number object -

var val = new Number(number);
If the argument cannot be converted into a number, it returns NaN (Not-a-Number).

What is purpose of onError event handler in JavaScript?

The onerror event handler provides three pieces of information to identify the exact nature of the error -

Error message - The same message that the browser would display for the given error.

URL - The file in which the error occurred.

Line number - The line number in the given URL that caused the error.

Arrow functions?

Its a short way to write function expressions.

```
const greet = ()=>{
console.log("hello");
}
greet();
```

Imports and exports?

Using Imports and exports we can split our code into multiple files.
Imports allow taking only some specific variables or methods of a file. We can import methods or variables that are exported by a module.

```
//index.js

import name,age from './person';

console.log(name);
console.log(age);

//person.js

let name ='Sharad', occupation='developer', age =26;

export { name, age};
```

Undeclared variables and undefined variables.

Undeclared variables are which do not exist in the program, it will give a runtime error.
Ex:
```
console.log(a);      //undeclared
```

undefined variables are declared in the program but not given any value to it.
it will return undefined.
Ex:
```
var a;
console.log(a);   // undefined.
```

Code for adding new elements:

```
<html>
<head>
<title>t1</title>
<script type="text/javascript">
        function addNode() { var newP = document.createElement("p");
        var textNode = document.createTextNode(" This is a new text node");
        newP.appendChild(textNode);
document.getElementById("firstP").appendChild(newP); }
```

</script> </head>
<body> <p id="firstP">firstP<p> </body>
</html>

Global variables:

Global variables are available throughout the program. They have no scope.
Var keyword is used to declare a local variable or object. if var keyword is
omitted, a global variable is declared.

Ex:

globalVariable = "test";

there will be clash of variable names of local and global scope.so it is difficult to debug.

Alert box displays only one button which is the OK button.
Confirm box displays two buttons namely ok and cancel.
Prompt box allows the user to enter input by providing a text box.

This keyword in javascript?

This keyword refers to the object from where it was called.

Timers in javascript

It is used to execute a piece of code at a set time and interval of time.
this is done by using functions like
setTimeout(function, delay) // start a timer that calls a function after the mentioned delay
setInterval(function, delay) // repeatedly execute the given function in the mentioned delay
clearInterval(id) //function instructs the timer to stop.

Timers are operated within a single thread, and thus events might queue up, waiting to be executed.

ViewState:

Specific to a page in session

SessionState:

Specific to user specific data that can be accessed across all pages in the web application.

== and ===

== checks equality only

=== checks for equality and type

---

submit a form in javascript:

```
document.form[0].submit();
```

Javascript supports automatic type conversion. it’s a common way used by developers.

---

how to access the value of a textbox using JavaScript?

```
<!DOCTYPE html>
<html>
<body>
Full name: <input type="text" id="txtFullName"

name="FirstName" value="Vikas Ahlawat">
</body>
</html>
```

code:

```
var name = document.getElementById('txtFullName').value;
alert(name);
```

Assign a Value to the Variable

```
var name = "John";
```

---

comments

```
// line comments
/*   */ block comments
```

How will you get the Checkbox status whether it is checked or not?

```
var status = document.getElementById('checkbox1').checked;
alert(status);
```

will return true or false

---

how to create arrays in javascript?

```
var names = new Array();
Add Elements in Array:-
names[0] = "Vikas";
names[1] = "Ashish";
```

names[2] = "Nikhil";

If an array with name as "names" contain three elements, then how will you print the third element of this array?

Print third array element document.write(names[2]);

How do you submit a form using JavaScript?

Ans:Use document.forms[0].submit();

What does isNaN function do?

Ans: It returns true if the argument is not a number.

code:
```
document.write(isNaN("Hello")+ "<br>");      //true
document.write(isNaN("2013/06/23")+ "<br>"); //true
document.write(isNaN(123)+ "<br>");  //false
```

What is the use of Math Object in JavaScript?

The math object provides you properties and methods for mathematical constants and functions.

```
var x = Math.PI; // Returns PI
var y = Math.sqrt(16); // Returns the square root of 16
var z = Math.sin(90); //Returns the sine of 90
```

What do you understand by this keyword in JavaScript?

this refers to the owner of the function

```
var person = {
   firstName: "John",
   lastName : "Doe",
   id       : 5566,
   fullName : function() {
      return this.firstName + " " + this.lastName;
   }
};
```
---
what does "1" + 2 + 4?

124

---
what does 3 + 4 + "7"?

77
---
how do you change the style on any element using javascript?

document.getElementById("myText").style.fontsize = "10";
---
how to read and write a file using JavaScript?

Using JavaScript extensions
Using a web page and Active X objects
---
loops in JS

for
while
do-while

Variable typing in Javascript?

Variable typing is used to assign a number to a variable and the same variable can be assigned to a string.
Ex:
i=10;
i="string";
---
convert the string of any base to integer in JavaScript?

The parseInt() function parses a string and returns an integer.

EX:
parseInt("4F",16);
---
how to detect the operating system on the client machine?

the navigator.platform string (property) should be used.

NULL in javascript

The NULL value is used to represent no value or no object. It implies no object or null string, no valid boolean value, no number and no array object.
---
function of delete operator?

The delete keyword is used to delete the property as well as its value.

```
var student= {age:20, batch:"ABC"};
delete student.age;
```

---

escape characters:

```
document.write "I m a "good" boy"
document.write "I m a \"good\" boy"
```

JavaScript Cookies?

Cookies are the small test files stored in a computer and it gets created when the user visits the websites to store information that they need.

Example could be User Name details and shopping cart information from
the previous visits.

JavaScript has concept level scope?

No. JavaScript does not have concept level scope. The variable declared inside the function has scope inside the function.

---

the disadvantage of using innerHTML in JavaScript?

Content is replaced everywhere

---

generic objects can be created?

Generic objects can be created as:

```
var I = new object();
```

---

the use of type of operator?

'Typeof' is an operator which is used to return a string description of the type of a variable.

Which keywords are used to handle exceptions?

Try… Catch---finally is used to handle exceptions in the JavaScript

```
Try{
        Code
}
Catch(exp){
        Code to throw an exception
```

}
Finally{
Code runs either it finishes successfully or after catch
}
---
keyword is used to print the text in the screen?

document.write("Welcome") is used to print the text – Welcome in the screen.
---
the use of blur function?

Blur function is used to remove the focus from the specified object.

How to find operating system in the client machine using JavaScript?

The 'Navigator.appversion' is used to find the name of the operating system in the client machine.
---
the different types of errors in JavaScript?

There are three types of errors:

Load time errors:
Errors which come up when loading a web page like improper syntax errors are known as Load time errors and it generates the errors dynamically.

Run time errors: Errors that come due to misuse of the command inside the HTML language.

Logical Errors: These are the errors that occur due to the bad logic performed on a function which is having different operation.

How are object properties assigned?

Properties are assigned to objects in the following way -

obj["class"] = 12;
or

obj.class = 12;

---
undefined value in JavaScript?

Undefined value means the Variable used in the code doesn't exist.
---
the use of Void(0)?

Void(0) is used to prevent the page from refreshing and parameter "zero" is passed while calling.

Void(0) is used to call another method without refreshing the page.

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

The following code has to be inserted to achieve the desired effect:

```
<script language="JavaScript" type="text/javascript" >

<!-- location.href="http://newhost/newpath/newfile.html"; //--></script>
```

What is the data type of variables of in JavaScript?

All variables in the JavaScript are object data types.

---

does javascript support foreach loop?

javascript 1.6 support foreach loop.

---

object in javascript? example?

Object refers to a compound value where we can set the properties which holds their own values and of any type.

```
var man = new Object();
man.name = 'Vikas Ahlawat';
man.living = true;
man.age = 27;
```

How you will add function as a property in a JavaScript object? Give an example.

```
var man = new Object();
man.name = 'Vikas Ahlawat';
man.living = true;
man.age = 27;
man.getName = function() { return man.name;}
console.log(man.getName());
```

What is the similarity between the 1st and 2nd statement?

```
1st:- var myString = new String('male');l // An object.
2nd:- var myStringLiteral = 'male'; // Primitive string value, not an object.
```

What will be the output of the following statements?

```
var myString = 'Vikas' // Create a primitive string object.
var myStringCopy = myString; // Copy its value into a new variable.
var myString = null; // Manipulate the value
console.log(myString, myStringCopy);
```

Ans: // Logs 'null Vikas'

What would be the output of the following statements?

```
var object1 = { same: 'same' };
var object2 = { same: 'same' };
console.log(object1 === object2);   //false
```

JavaScript does not care that they are identical and of the same object type.
When comparing complex objects, they are equal only when they reference the same object.

What would be the output of the following statements?

```
var object1 = { same: 'same' };
var object2 = object1;
console.log(object1 === object2);   //true
```

What is this?

```
var myArray = [[[]]];
```

Three dimensional array.

Name any two JavaScript functions which are used to convert non numeric values
into numbers?

Number()
parseInt()
parseFloat()

```
var n1 = Number("Hello world!");      //NaN
var n2 = Number("");                   //0
var n3 = Number("000010");             //10
var n4 = Number(true);                   //1
var n5 = Number(NaN);                  //NaN
```

Does JavaScript Support automatic type conversion, If yes give example.

```
var s = '5';
var a = s*1;
```

```
var b = +s;
typeof(s); //"string"
typeof(a); //"number"
typeof(b); //"number"
```

Javascript array

```
var students = ["John", "Ann", "Kevin"];
or
var students = new Array("John", "Ann", "Kevin");
```

Javascript array methods:

length property --> If you want to know the number of elements in an array, you can use the length property.
prototype property --> If you want to add new properties and methods, you can use the prototype property.
reverse method --> You can reverse the order of items in an array using a reverse method.
sort method --> You can sort the items in an array using sort method.
pop method --> You can remove the last item of an array using a pop method.
shift method --> You can remove the first item of an array using shift method.
push method --> You can add a value as the last item of the array.

What is the 'Strict' mode in JavaScript and how can it be enabled?

The purpose of "use strict" is to indicate that the code should be executed in "strict mode". With strict mode, you can not, for example, use undeclared variables.

You can use strict mode in all your programs. It helps you to write cleaner code, like preventing you from using undeclared variables.

Ex:

```
"use strict";
x = 3.14;       // This will cause an error because x is not declared
```

What is the way to get the status of a CheckBox?

```
alert(document.getElementById('checkbox1').checked);
```

If the CheckBox will be checked, this alert will return TRUE.

Explain window.onload and onDocumentReady?

The onload function is not run until all the information on the page is loaded. This leads to a substantial delay before any code is executed.

onDocumentReady loads the code just after the DOM is loaded. This allows early manipulation of the code

---

how can a value be appended to an array?

A value can be appended to an array in the given manner -

arr[arr.length] = value;

---

some native functions?

alert()
confirm()
prompt()
Date()
parseInt("123")
setTimeout(functionname, 5000) ---> executes the mentioned function after the mentioned delay

---

browser properties and methods?

event
history
location
status
alert()
close()
confirm()
focus()

DOM - allows us to access manipulate the content of a document.
it is the programming interface for HTML and XML pages.
provides the structure map of the document as well as a set of methods to interface with the elements contained.

Each element in the page is referred to as a node.
The DOM is the collection of nodes like element nodes, attribute node, text node.

polyfills - it provides older browser with modern features and normalize the functionality.

Accessing DOM nodes

getElementByTagName() ---> var foo = document.getElementByTagName("p");

getElementById() ---> var foo = document.getElementById("beginner").innerHTML;

getElementByClassName() ---> document.getElementByClassName("col");

querySelectorAll() ---> var sidebar = document.querySelectorAll("input[type='text']");

Accessing an attribute value

getAttribute()

```
var bigImage = document.getElementById("lead-image");
alert(bigImage.getAttribute("src"));
```

manipulating nodes

setAttribute()

```
var bigImage = document.getElementById("lead-image");
bigImage.setAttribute("src", "pic.jpg");
```

innerHTML

```
var introDiv = document.getElementById("intro");
introDiv[0].innerHTML = "<p>This is our intro</p>";
```

style

```
document.getElementById("intro").style.backgroundcolor = "blue";
```

adding and removing elements

createElement() ----> var newDiv = document.createElement("div");
createTextNode() ---> var ourText = document.createTextNode("This is out text");

appendChild()
insertBefore()
replaceChild()
removeChild()

---

what is close()?

It is used to close the current window.

window.close()

What will happen if an infinite while loop is run in Javascript?

The program will crash the browser.

DOM mouse events

onclick
ondblclick
mousemove
mousedown
mouseover
mouseout
mouseup

What is Javascript BOM?

BOM stands for “Browser Object Model” that allows Javascript to ‘talk’ to the browser,
no standards, modern browsers implement similar
BOMS – window, screen, location, history, navigator, timing, cookies.

How does your website works out of the internet?

GoDaddy.com/namecheap.com---> check available domain name and buy.

we will get an IP address for selected domain name

when we buy the domain name we need to tell the register to have the IP address
in DNS server.

there are 2 DNS server: primary and secondary

so the web hosting company provides the DNS server in which they can have our IP
address.

JAVASCRIPT

JavaScript is a very powerful client-side scripting language.
JavaScript is used mainly for enhancing the interaction of a user with the webpage.
In other words, you can make your webpage more lively and interactive,
with the help of JavaScript.
JavaScript is also being used widely in game development and
Mobile application development.

How to run javascript?

JavaScript cannot run on its own.
In fact, the browser is responsible for running JavaScript code.

When a user requests an HTML page with JavaScript in it,
the script is sent to the browser and it is up to the browser to execute it.
The main advantage of JavaScript is that all modern web browsers support JavaScript.

Tools we need to run Javascript?

To start with, you need a text editor to write your code
and a browser to display the web pages you develop.

Ex: Notepad++, Visual Studio Code, Sublime Text, Atom or any other text editor
Google Chrome, Firefox, Microsoft Edge, Internet Explorer etc.
---
difference between java and javascript.

java - object oriented programming language.
Javascript - client-side scripting language.
---
difference between javascript and jscript.

almost similar but

javascript - was developed by Netscape.
Jscript - reverse engineered javascript.

How do we add JavaScript onto a web page?

if the script code is very short

```
<head>
<title>Page Title</title>
<script language="JavaScript" type="text/javascript">
  var name = "Vikas Ahlawta"
  alert(name);
</script>
</head>
```

if the code is long

```
<head>
<title>Page Title</title>
<script type="text/javascript" src="myjavascript.js"></script>
</head>
```

Simple javascript program

```
<script>
```

```
    alert("Hello World!");
</script>
```

Is JavaScript case sensitive?

yes. getElementById is not the same as getElementbyID.

---

types in javascript

string
number
boolean
function
object
null
undefined

Use of isNaN function

IsNan function returns true if the argument is not a number
otherwise it returns false.

Closures in javascript?

A closure is the local variables for a function-kept alive after the function has returend.
Closure is when a function is able to remember and access its lexical scope even
when the function is executing outside its lexical scope.
A clsoue is a function having access to the parent scope, even after the
parent function has closed.

---

how can you create an Object in Javascript?

```
var emp = {
   name: "Zara",
   age: 10
};
```

How can you read properties of an Object in JavaScript?
You can write and read properties of an object using the dot notation as follows -

```
// Getting object properties
emp.name  // ==> Zara
emp.age   // ==> 10
// Setting object properties
emp.name = "Daisy"  // <== Daisy
emp.age  =  20      // <== 20
```

How can you create an Array in JavaScript?
You can define arrays using the array literal as follows -

```
var x = [];
var y = [1, 2, 3, 4, 5];
```

How to read elements of an array in JavaScript?
An array has a length property that is useful for iteration. We can read elements of an array as follows -

```
var x = [1, 2, 3, 4, 5];
for (var i = 0; i < x.length; i++) {
   // Do something with x[i]
}
```

How many types of functions JavaScript supports?

A function in JavaScript can be either named or anonymous.

How to define a anonymous function?

An anonymous function can be defined in similar way as a normal function
but it would not have any name.

Can you pass a anonymous function as an argument to another function?

Yes! An anonymous function can be passed as an argument to another function.

What is arguments object in JavaScript?

JavaScript variable arguments represents the arguments passed to a function
---
How can you get the type of arguments passed to a function?
Using typeof operator, we can get the type of arguments passed to a function. For example -

```
function func(x){
   console.log(typeof x, arguments.length);
}
func();               //==> "undefined", 0
func(1);               //==> "number", 1
func("1", "2", "3");   //==> "string", 3
```

How can you get the total number of arguments passed to a function?
Using arguments.length property, we can get the total number of arguments passed to a function. For example -

```
function func(x){
   console.log(typeof x, arguments.length);
}
func();               //==> "undefined", 0
func(1);               //==> "number", 1
func("1", "2", "3");   //==> "string", 3
```

How can you get the reference of a caller function inside a function?
The arguments object has a callee property, which refers to the function you're inside of. For example -

```
function func() {
   return arguments.callee;
}
func();
```

Which type of variable among global and local, takes precedence
 over other if names are same?

A local variable takes precedence over a global variable with the same name.

What is callback?
A callback is a plain JavaScript function passed to some method as an argument or option. Some callbacks are just events, called to give the user a chance to react when a
 certain state is triggered.

Which built-in method returns the character at the specified index?

charAt() method returns the character at the specified index.

Which built-in method combines the text of two strings and returns a new string?

concat() method returns the character at the specified index.

Which built-in method calls a function for each element in the array?

forEach() method calls a function for each element in the array.

Which built-in method returns the index within the calling String object
of the first occurrence of the specified value?

indexOf() method returns the index within the calling String object o
f the first occurrence of the specified value, or -1 if not found.

Which built-in method returns the length of the string?

length() method returns the length of the string.

Which built-in method reverses the order of the elements of an array?
reverse() method reverses the order of the elements of an array --
the first becomes the last, and the last becomes the first.

Which built-in method sorts the elements of an array?
sort() method sorts the elements of an array.

Which built-in method returns the characters in a string beginning at the s
pecified location?
substr() method returns the characters in a string beginning at the specified
location through the specified number of characters.

Which built-in method returns the calling string value converted to lower case?
toLowerCase() method returns the calling string value converted to lower case.

Which built-in method returns the calling string value converted to upper case?
toUpperCase() method returns the calling string value converted to upper case.

Which built-in method returns the string representation of the number's value?
toString() method returns the string representation of the number's value.

What are the variable naming conventions in JavaScript?

JavaScript variable names should not start with a numeral (0-9).
They must begin with a letter or the underscore character.
For example, 123test is an invalid variable name
but _123test is a valid one.

How typeof operator works?
The typeof operator evaluates to "number", "string", or "boolean" if its
operand is a number, string, or boolean value and returns true or false
based on the evaluation.

What typeof returns for a null value?
It returns "object".

Can you access Cookie using javascript?
JavaScript can also manipulate cookies using the cookie property of the
Document object. JavaScript can read, create, modify, and delete the cookie or
cookies that apply to the current web page.

How to create a Cookie using JavaScript?
The simplest way to create a cookie is to assign a string value to the
document.cookie object, which looks like this -

Syntax -

document.cookie = "key1 = value1; key2 = value2; expires = date";

How to redirect a url using JavaScript?

his is very simple to do a page redirect using JavaScript at client side. To redirect your site visitors to a new page, you just need to add a line in your head section as follows -

```
<head>
<script type="text/javascript">
<!--
   window.location="http://www.newlocation.com";
//-->
</script>
</head>
```

How to print a web page using javascript?

JavaScript helps you to implement this functionality using print function of window object. The JavaScript print function window.print() will print the current web page when executed.

What is Date object in JavaScript?

The Date object is a datatype built into the JavaScript language. Date objects are created with the new Date( ).

Once a Date object is created, a number of methods allow you to operate on it. Most methods simply allow you to get and set the year, month, day, hour, minute, second, and millisecond fields of the object, using either local time or UTC (universal, or GMT) time.

What is Number object in JavaScript?

he Number object represents numerical date, either integers or floating-point numbers. In general, you do not need to worry about Number objects because the browser automatically converts number literals to instances of the number class.

Syntax -

Creating a number object -

var val = new Number(number);

If the argument cannot be converted into a number, it returns NaN (Not-a-Number).

What is purpose of onError event handler in JavaScript?

The onerror event handler provides three pieces of information to identify the exact nature of the error -

Error message - The same message that the browser would display for the given error.

URL - The file in which the error occurred.

Line number - The line number in the given URL that caused the error.

Arrow functions?

Its a short way to write function expressions.

```
const greet = ()=>{
console.log("hello");
}
greet();
```

Imports and exports?

Using Imports and exports we can split our code into multiple files.
Imports allow taking only some specific variables or methods of a file. We can import methods or variables that are exported by a module.

```
//index.js

import name,age from './person';

console.log(name);
console.log(age);

//person.js

let name ='Sharad', occupation='developer', age =26;

export { name, age};
```

Negative infinity

It is a number in javascript which can be derived by dividing negative number by zero.

Undeclared variables and undefined variables.

Undeclared variables are which do not exist in the program, it will give a runtime error.
Ex:

```
console.log(a);      //undeclared
```

undefined variables are declared in the program but not given any value to it.
it will return undefined.
Ex:
var a;
console.log(a); // undefined.

Code for adding new elements:

```
<html>
<head>
<title>t1</title>
<script type="text/javascript">
        function addNode() { var newP = document.createElement("p");
        var textNode = document.createTextNode(" This is a new text node");
        newP.appendChild(textNode);
document.getElementById("firstP").appendChild(newP); }
</script> </head>
<body> <p id="firstP">firstP<p> </body>
</html>
```

Global variables:

Global variables are available throughout the program. They have no scope.
Var keyword is used to declare a local variable or object. if var keyword is
omitted, a global variable is declared.

Ex:

```
globalVariable = "test";
```

there will be clash of variable names of local and global scope.
so its difiicult to debug.

Alert box displays only one button which is the OK button.
Confirm box displays two buttons namely ok and cancel.
Prompt box allows the user to enter input by providing a text box.

This keyword in javascript?

This keyword refers to the object from where it was called.

Timers in javascript

It is used to execute a piece of code at a set time and interval of time.
this is done by using functions like
setTimeout(function, delay) // start a timer that calls a function after the mentioned delay

setInterval(function, delay) // repeatedly execute the given function in the mentioned delay
clearInterval(id) //function instructs the timer to stop.

Timers are operated within a single thread, and thus events might queue up, waiting to be executed.

ViewState:

Specific to a page in session

SessionState:

Specific to user specific data that can be accessed across all pages in the web application.

== and ===

== checks equality only
=== checks for equality and type

---

submit a form in javascript:

```
document.form[0].submit();
```

Javscript supports automatic type conversion. its a common way used by developers.

---

how to access the value of a textbox using JavaScript?

```
<!DOCTYPE html>
<html>
<body>
Full name: <input type="text" id="txtFullName"

name="FirstName" value="Vikas Ahlawat">
</body>
</html>
```

code:

```
var name = document.getElementById('txtFullName').value;
alert(name);
```

Assign a Value to the Variable

```
var name = "John";
```

---

comments

// line comments
/* */ block comments

How will you get the Checkbox status whether it is checked or not?

var status = document.getElementById('checkbox1').checked;
alert(status);

will return true or false
---
how to create arrays in javascript?

var names = new Array();
Add Elements in Array:-
names[0] = "Vikas";
names[1] = "Ashish";
names[2] = "Nikhil";

If an array with name as "names" contain three elements, then how will you print the third element of this array?

Print third array element document.write(names[2]);

How do you submit a form using JavaScript?

Ans:Use document.forms[0].submit();

What does isNaN function do?

Ans: It returns true if the argument is not a number.

code:
```
document.write(isNaN("Hello")+ "<br>");      //true
document.write(isNaN("2013/06/23")+ "<br>"); //true
document.write(isNaN(123)+ "<br>");  //false
```

What is the use of Math Object in JavaScript?

The math object provides you properties and methods
for mathematical constants and functions.

```
var x = Math.PI; // Returns PI
var y = Math.sqrt(16); // Returns the square root of 16
var z = Math.sin(90); //Returns the sine of 90
```

What do you understand by this keyword in JavaScript?

this refers to the owner of the function

```
var person={
   firstName:"shobana",
lastName:"Vijayakumar",
age:28,
fullName:function(){
return this.firstName+" "+this.lastName;
}
};
console.log(person.fullName());
```

shobana Vijayakumar
---
what does "1" + 2 + 4?

124
---
what does 3 + 4 + "7"?

77
---
how do you change the style on any elemeng using javascript?

```
document.getElementById("myText").style.fontsize = "10";
```
---
how to read and write a file using JavaScript?

Using JavaScript extensions
Using a web page and Active X objects
---
loops in JS

for
while
do-while

Variable typing in Javascript?

Variable typing is used to assign a number to a variable and
the same variable can be assigned to a string.
Ex:
i=10;

i="string";
---
convert the string of any base to integer in JavaScript?

The parseInt() function parses a string and returns an integer.

EX:
parseInt("4F",16);
---
how to detect the operating system on the client machine?

the navigator.platform string (property) should be used.

NULL in javascript

The NULL value is used to represent no value or no object.
It implies no object or null string, no valid boolean value, no number
and no array object.
---
function of delete operator?

The delete keyword is used to delete the property as well as its value.

var student= {age:20, batch:"ABC"};
delete student.age;
---
escape characters:

document.write "I m a "good" boy"
document.write "I m a \"good\" boy"

JavaScript Cookies?

Cookies are the small test files stored in a computer and it gets
created when the user visits the websites to store information that they need.
Example could be User Name details and shopping cart information from
the previous visits.

JavaScript has concept level scope?

No. JavaScript does not have concept level scope. The variable
declared inside the function has scope inside the function.
---
the disadvantage of using innerHTML in JavaScript?

Content is replaced everywhere

What is break and continue statements?

Break statement exits from the current loop.

Continue statement continues with next statement of the loop.
---
the two basic groups of dataypes in JavaScript?

They are as –

Primitive
Reference types.
Primitive types are number and Boolean data types.
Reference types are more complex types like strings and dates.
---
generic objects can be created?

Generic objects can be created as:

var I = new object();
---
the use of type of operator?

'Typeof' is an operator which is used to return a string description
of the type of a variable.

Which keywords are used to handle exceptions?

Try… Catch---finally is used to handle exceptions in the JavaScript

```
Try{
        Code
}
Catch(exp){
        Code to throw an exception
}
Finally{
        Code runs either it finishes successfully or after catch
}
```
---
keyword is used to print the text in the screen?

document.write("Welcome") is used to print the text – Welcome in the screen.
---
the use of blur function?

Blur function is used to remove the focus from the specified object.

---

variable typing?

Variable typing is used to assign a number to a variable and then assign string to the same variable. Example is as follows:

```
i= 8;
i="john";
```

How to find operating system in the client machine using JavaScript?

The 'Navigator.appversion' is used to find the name of the operating system in the client machine.

---

the different types of errors in JavaScript?

There are three types of errors:

Load time errors:
Errors which come up when loading a web page like improper syntax errors are known as Load time errors and it generates the errors dynamically.
Run time errors: Errors that come due to misuse of the command inside the HTML language.
Logical Errors: These are the errors that occur due to the bad logic performed on a function which is having different operation.

How are object properties assigned?

Properties are assigned to objects in the following way -

```
obj["class"] = 12;
```

or

```
obj.class = 12;
```

---

undefined value in JavaScript?

Undefined value means the

Variable used in the code doesn't exist
Variable is not assigned to any value
Property doesn't exist

---

types of Pop up boxes available in JavaScript?

Alert
Confirm and
Prompt
---
the use of Void(0)?

Void(0) is used to prevent the page from refreshing and parameter "zero" is passed while calling.

Void(0) is used to call another method without refreshing the page.

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

The following code has to be inserted to achieve the desired effect:

<script language="JavaScript" type="text/javascript" >

<!-- location.href="http://newhost/newpath/newfile.html"; //--></script>

What is the data type of variables of in JavaScript?

All variables in the JavaScript are object data types.
---
does javascript support foreach loop?

javascript 1.6 support foreach loop.
---
object in javascript? example?

Object refers to a compound value where we can set the properties which holds their own values and of any type.

```
var man = new Object();
man.name = 'Vikas Ahlawat';
man.living = true;
man.age = 27;
```

How you will add function as a property in a JavaScript object? Give an example.

```
var man = new Object();
man.name = 'Vikas Ahlawat';
man.living = true;
man.age = 27;
man.getName = function() { return man.name;}
console.log(man.getName());
```

What is the similarity between the 1st and 2nd statement?

1st:- var myString = new String('male');l // An object.
2nd:- var myStringLiteral = 'male'; // Primitive string value, not an object.

What will be the output of the following statements?

```
var myString = 'Vikas' // Create a primitive string object.
var myStringCopy = myString; // Copy its value into a new variable.
var myString = null; // Manipulate the value
console.log(myString, myStringCopy);
```
Ans: // Logs 'null Vikas'

What would be the output of the following statements?

```
var object1 = { same: 'same' };
var object2 = { same: 'same' };
console.log(object1 === object2);   //false
```

JavaScript does not care that they are identical and of the same object type.
When comparing complex objects, they are equal only when they reference the same object.

What would be the output of the following statements?

```
var object1 = { same: 'same' };
var object2 = object1;
console.log(object1 === object2);   //true
```

What is this?

```
var myArray = [[[]]];
```

Three dimensional array.

Name any two JavaScript functions which are used to convert nonnumeric values
into numbers?

Number()
parseInt()
parseFloat()

```
var n1 = Number("Hello world!");      //NaN
var n2 = Number("");                   //0
var n3 = Number("000010");             //10
var n4 = Number(true);                   //1
```

```
var n5 = Number(NaN);                    //NaN
```

Does JavaScript Support automatic type conversion, If yes give example.

```
var s = '5';
var a = s*1;
var b = +s;
typeof(s); //"string"
typeof(a); //"number"
typeof(b); //"number"
```

Javascript array

```
var students = ["John", "Ann", "Kevin"];
```
or
```
var students = new Array("John", "Ann", "Kevin");
```

Javascript array methods:

length property --> If you want to know the number of elements in an array, you can use the length property.
prototype property --> If you want to add new properties and methods, you can use the prototype property.
reverse method --> You can reverse the order of items in an array using a reverse method.
sort method --> You can sort the items in an array using sort method.
pop method --> You can remove the last item of an array using a pop method.
shift method --> You can remove the first item of an array using shift method.
push method --> You can add a value as the last item of the array.

What is the 'Strict' mode in JavaScript and how can it be enabled?

Strict Mode adds certain compulsions to JavaScript.
Under the strict mode, JavaScript shows errors for a piece of codes, which did not show an error before,
but might be problematic and potentially unsafe.
```
function myfunction() {
    "use strict";
    var v = "This is a strict mode function";
}
```

What is the way to get the status of a CheckBox?

```
alert(document.getElementById('checkbox1').checked);
```

If the CheckBox will be checked, this alert will return TRUE.

Explain window.onload and onDocumentReady?

The onload function is not run until all the information on the page is loaded. This leads to a substantial delay before any code is executed.

onDocumentReady loads the code just after the DOM is loaded. This allows early manipulation of the code

---

how can a value be appended to an array?

A value can be appended to an array in the given manner -

arr[arr.length] = value;

---

some native functions?

alert()
confirm()
prompt()
Date()
parseInt("123")
setTimeout(functionname, 5000) ---> executes the mentioned function after the mentioned delay

---

browser properties and methods?

event
history
location
status
alert()
close()
confirm()
focus()

DOM - allows us to access manipulate the content of a document.
it is the programming interface for HTML and XML pages.
provides the structure map of the document as well as a set of methods to interface with the elements contained.

Each element in the page is referred to as a node.
The DOM is the collection of nodes like element nodes, attribute node, text node.

polyfills - it provides older browser with modern features and normalize the functionality.

Accessing DOM nodes

getElementByTagName() ---> var foo = document.getElementByTagName("p");

getElementById() ---> var foo = document.getElementById("beginner").innerHTML;

getElementByClassName() ---> document.getElementByClassName("col");

querySelectorAll() ---> var sidebar = document.querySelectorAll("input[type='text']");

Accessing an attribute value

getAttribute()

```
var bigImage = document.getElementById("lead-image");
alert(bigImage.getAttribute("src"));
```

manipulating nodes

setAttribute()

```
var bigImage = document.getElementById("lead-image");
bigImage.setAttribute("src", "pic.jpg");
```

innerHTML

```
var introDiv = document.getElementById("intro");
introDiv[0].innerHTML = "<p>This is our intro</p>";
```

style

```
document.getElementById("intro").style.backgroundcolor = "blue";
```

adding and removing elements

createElement() ----> var newDiv = document.createElement("div");
createTextNode() ---> var ourText = document.createTextNode("This is out text");

appendChild()
insertBefore()
replaceChild()
removeChild()

---

what is close()?

It is used to close the current window.

window.close()

What will happen if an infinite while loop is run in Javascript?

The program will crash the browser.

DOM mouse events

onclick
ondblclick
mousemove
mousedown
mouseover
mouseout
mouseup

What is Javascript BOM?

BOM stands for “Browser Object Model” that allows Javascript to ‘talk’ to the browser,
no standards, modern browsers implement similar
BOMS – window, screen, location, history, navigator, timing, cookies.

Explain event loop, binding, bubbling, capturing, delegation and propagation

Event bubbling:
When an event happens on an element, it first runs the handlers on it, then on its parent, then all the way up on other ancestors.
Ex:

```
<form onclick="alert('form')">FORM
  <div onclick="alert('div')">DIV
    <p onclick="alert('p')">P</p>
  </div>
</form>
```

A click on the inner <p> first runs onclick:

1. On that <p>.
2. Then on the outer .
3. Then on the outer <form>.
4. And so on upwards till the document object.

Explain why the following doesn't work as an IIFE: function foo(){ }();. What needs to be changed to properly make it an IIFE?

To make it IIFE

```
(function foo(){
})();
```

Can you describe the main difference between a.forEach loop and a .map() loop and why you would pick one versus the other?

Array iteration has 8 methods:

The forEach() method calls a function once for each array element.
the forEach() method doesn't actually return anything.
Ex:

```
var txt = "";
var numbers = [45, 4, 9, 16, 25];
Var new = numbers.forEach(function (value, index, array) {
  txt = txt + value + "<br>";
});
```

Result: 45,4,9,16,25

****

The map() method creates a new array by performing a function on each array element.
The map() utilizes return values and actually returns a new Array of the same size.
Ex:

```
var numbers1 = [45, 4, 9, 16, 25];
var numbers2 = numbers1.map(function(value, index, array) {
  return value * 2;
});
```

Result: 90, 8, 18, 32, 50

****

The filter() method creates a new array with array elements that passes a test.
Ex:

```
var numbers = [45, 4, 9, 16, 25];
var over18 = numbers.filter(function(value, index, array) {
  return value > 18;
});
```

Result: 45, 25

****

The reduce() method runs a function on each array element to produce (reduce it to) a single value.
It takes 4 arguments
Ex:

```
var numbers1 = [45, 4, 9, 16, 25];
var sum = numbers1.reduce(function(total, value, index, array) {
```

```
  return total + value;
});
```
Result: 90

****

The some() method check if some array values pass a test.

Ex:

```
var numbers = [45, 4, 9, 16, 25];
var someOver18 = numbers.some(function(value, index, array) {
  return value > 18;
});
```
Result: true

****

The every() method check if all array values pass a test.

Ex:

```
var numbers = [45, 4, 9, 16, 25];
var allOver18 = numbers.every(function(value, index, array) {
  return value > 18;
});
```
Result: false

****

The find() method returns the value of the first array element that passes a test function.

Ex:

```
var numbers = [4, 9, 16, 25, 29];
var first = numbers.find(function(value, index, array) {
  return value > 18;
});
```
****

The indexOf() method searches an array for an element value and returns its position.

Ex:

```
var fruits = ["Apple", "Orange", "Apple", "Mango"];
var a = fruits.indexOf("Apple");
```
Result: 0

****

Array.lastIndexOf() is the same as Array.indexOf(), but searches from the end of the array.

Ex:

```
var fruits = ["Apple", "Orange", "Apple", "Mango"];
var a = fruits.lastIndexOf("Apple");
```
Result: 2

What's a typical use case for anonymous functions?

Anonymous functions are function expressions, and thus we can pass them as variables and return functions. When used as IIFE, they are used to maintain scope.

Since Anonymous Functions are function expressions rather than the regular function declaration which are statements. Function expressions are more flexible. We can assign functions to variables, object properties, pass them as arguments to other functions, and even write a simple one line code enclosed in an anonymous functions.

```
var squaredArray = inputArray.map(function(x) {
 return x * x;
});
```

Another typical example would be an anonymous function used by popular frameworks used as IIFE (Immediately Invoked Function Expression).

```
(function() { })();
```

Difference between: function Person(){}, var person = Person(), and var person = new Person()?

function Person(){ → it's a named function
}

var person = Person() → It's a function expression

var person = new Person() → function constructor

Create a for loop that iterates up to 100 while outputting "fizz" at multiples of 3, "buzz" at multiples of 5 and "fizzbuzz" at multiples of 3 and 5

```
for (i=1; i<=100; i++){
  if(i%3==0 & i%5==0){
    document.write("fizzbuzz");
  }else if(i%5==0){
    document.write("buzz");
  }else if(i%3==0){
    document.write("fizz");
  }else {
  document.write(i);
  }
  document.write('<br/>');
}
```

Who is the author of JavaScript Language?

Brendan Eich- in may 1995.

What is the best book for learning JavaScript and why? - @answer-good-js-books--github

You dont know js

What is the type of NaN? How to check if a value is NaN?

Type of NaN is number.
isNaN(“shobana”) → true
isNaN(2) → false

What the reason that window.window === window return true? - @doc--mdn

The window property of a window object points to the window object itself. Thus, the following expressions all return the same window object:

window.window
window.window.window
window.window.window.window

What is the outcome of the JavaScript calculation? 1/0 = ?

infinity

Describe a few ways to communicate between a server and a client. Describe how a few network protocols work at a high level (IP, TCP, HTTP/S/2, UDP, RTC, DNS, etc.)

- Internet protocol
- TCP: A set of rules that governs the delivery of data over the internet or other network that uses the Internet Protocol, and sets up a connection between the sending and receiving computers.
- HTTP: Defines how messages are formatted and transmitted and what actions web servers and browsers should take in response to various commands
- HTTPS: The secure version of HTTP. All communications between the site and your computer are encrypted.
- HTTP2: A major revision of HTTP
- UDP: Alternative protocol to TCP used primarily for establishing low-latency and loss tolerating connections between applications on the internet.
- DNS: Domain name server. Translate list of names into ip addresses. Makes addresses easy to remember.

My website is slow. Walk me through diagnosing and fixing it. What are some performance optimizations people use, and when should they be used?

1. Minimize HTTP Requests by Combining CSS/JS Files, use queries to load only what is needed, reducing the no of images,
2. Compressing an image
3. Browser caching
4. Remove unused files/scripts.

What is jQuery?

jQuery is a JavaScript Library.
jQuery greatly simplifies JavaScript programming.

The jQuery library contains the following features:

- HTML/DOM manipulation
- CSS manipulation
- HTML event methods
- Effects and animations
- AJAX
- Utilities

Will jQuery work in all browsers?

The jQuery team knows all about cross-browser issues, and they have written this knowledge into the jQuery library. jQuery will run exactly the same in all major browsers.

Is there a difference between a function and a method?

Method and a function are the same, with different terms.
A method is a procedure or function in object-oriented programming.
A function is a group of reusable code which can be called anywhere in your program. This eliminates the need for writing the same code again and again. It helps programmers in writing modular codes.

What is the difference between a parameter and an argument?

A parameter is a variable in a method definition. When a method is called, the arguments are the data you pass into the method's parameters.

Parameter is what's given in the function declaration/definition.
Argument is what's passed when calling the function.

What is an API?

An application-programming interface (API) is a set of programming instructions and standards for accessing a Web-based software application or Web tool.

APIs just allow applications to communicate with one another.

What is ECMAScript?

ECMAScript is a Standard for a scripting languages. Languages like Javascript are based on the ECMAScript standard. ECMA Standard is based on several originating technologies, the most well known being JavaScript (Netscape) and JScript (Microsoft). ECMA means European Computer Manufacturer's Association.

Have you worked with Photoshop as a front end developer?
For what purpose?

I have used adobe photoshop for making logos for my project.

What software versioning system have you used? Could you describe
how a workflow would look like in this system?

There are lot of versioning systems out there like apache subversion, concurrent version system, revision control system… but so far i have used only GIT.

Workflow of git will be
Clone from the remote repo to local repo
Then commit the code to the local repo
Then push to the remote repo

How do you stay up to date with technology news?

I have signed up for new feature alerts in google cloud and amazon web services.
So that i get to know the latest developments. and
I follow www.cnet.com and www.techcrunch.com for latest news.

When would you use document.write()?

document.write() writes a string of text to a document .
But it goes on and overwrites on entire document.

What's the difference between .call(), .apply(), .bind()?

call()
Call's arguments are separated by commas
Ex;

```
var person={
firstName:"Shobana",
lastName:"Vijayakumar",
age: 28
};
```

```
function personDetails(message){
return message+" "+this.firstName+" "+this.lastName;
}
personDetails.call(person, "Hello");
"Hello Shobana Vijayakumar"
```

apply()
Apply accepts arguments as an Array

```
var person={
firstName:"Shobana",
lastName:"Vijayakumar",
age: 28
};
function personDetails(message){
return message+" "+this.firstName+" "+this.lastName;
}
personDetails.call(person,["Hello"]);
"Hello Shobana Vijayakumar"
```

bind()
bind() is the only method out of the three that returns a new function altogether. It does not call the function.
Ex:

```
var person={
   firstName: "Shobana",
lastName: "Vijayakumar",
age: 28,
fullName: function(){
return this.firstName +" "+ this.lastName;
}
};
var personDetails= person.fullName.bind(person);
personDetails();
"Shobana Vijayakumar"
```

Explain Function.prototype.bind.

bind is a method on the prototype of all functions in JavaScript. It allows you to create a new function from an existing function.

What's the difference between an "attribute" and a "property"?

Attributes are defined by HTML. Properties are defined by DOM.

- Attributes are referring to additional information of an object.

<meta name="viewport" content="width=device-width"> <!-- name and content are attributes -->

- Properties are the values associated with a JavaScript object.

person.age, person.name.

Difference between document load event and document DOMContentLoaded event?

DOMContentLoaded — the whole document (HTML) has been loaded.
load — the whole document and its resources (e.g. images, iframes, scripts) have been loaded.

Make this work: duplicate([1,2,3,4,5]); // [1,2,3,4,5,1,2,3,4,5]

```
var array=[1,2,3,4,5];
function duplicate(){
  return array.concat(array);
}
var duplicateArray= duplicate();
console.log(duplicateArray);
```

VM491:6 (10) [1, 2, 3, 4, 5, 1, 2, 3, 4, 5]

Why is it called a Ternary expression, what does the word "Ternary" indicate?

“Ternary” indicates three, and a ternary expression accepts three operands, the test condition, the “then” expression and the “else” expression. Ternary expressions are not specific to JavaScript and I’m not sure why it is even in this list.

```
 (1 === 1) ? “True" : "False";
```

Explain the differences on the usage of foo between function foo() {} and var foo = function() {}

function foo() {} → it is a normal function
var foo = function() {} → Its a function expression

Explain the difference between synchronous and asynchronous functions.

Synchronous: Step wise execution. Next line executed after first. Asynchronous: Execution moves to next step before first is finished.

Explain the difference between mutable and immutable objects.

Mutable objects are those whose state is allowed to change over time. An immutable value is the exact opposite — after it has been created, it can never change. Strings and Numbers are inherently immutable in javascript.

What language constructions do you use for iterating over object properties and array items?

for loop, for..in, for each..in, map, reduce

What tools and techniques do you use debugging JavaScript code?

Web/Browser console using console.log.

Explain what a single page app is and how to make one SEO-friendly.

A single-page application (SPA) is a web application or web site that fits on a single web page with the goal of providing a more fluid user experience similar to a desktop application. In a SPA, either all necessary code — HTML, JavaScript, and CSS — is retrieved with a single page load, or the appropriate resources are dynamically loaded and added to the page as necessary, usually in response to user actions.

Why would you use something like the load event? Does this event have disadvantages? Do you know any alternatives, and why would you use those?

The load event fires at the end of the document loading process. At this point, all of the objects in the document are in the DOM, and all the images, scripts, links and sub-frames have finished loading. To execute anything post document load, we fire these events.

Why is it, in general, a good idea to leave the global scope of a website as-is and never touch it?

The primary reason why global variables are discouraged in javascript is because, in javascript all code share a single global namespace, also javascript has implied global variables i.e. variables which are not explicitly declared in local scope are automatically added to global namespace. Relying too much on global variables can result in collisions between various scripts on the same page.

What's the difference between host objects and native objects?

Host — objects provided by environment like window, document by browser.
Native — object in an ECMAScript implementation whose semantics are fully defined by this specification rather than by the host environment. Eg. Array, String etc.

What are some of the advantages/disadvantages of writing JavaScript code in a language that compiles to JavaScript?

CoffeeScript. Pros/Cons: Syntactic sugar, readable code, and use of good patterns vs debugging and compilation issues.

Write a simple function to tell whether 2 is passed as parameter or not?

function isTwoPassed(){

```
  var args = Array.prototype.slice.call(arguments);
  return args.indexOf(2) != -1;
}

isTwoPassed(1,4) //false
isTowPassed(5,3,1,2) //true
```

How could you use Math.max to find the max value in an array?

```
var numbers = [1, 2, 3, 4];
Math.max.apply(null, numbers) // 4
```

How could you set a prefix before everything you log? for example, if you log('my message') it will log: "(app) my message"

```
function log(){
  var args = Array.prototype.slice.call(arguments);
  args.unshift('(app)');
  console.log.apply(console, args);
}

log('my message'); //(app) my message
log('my message', 'your message'); //(app) my message your message
```

Does JavaScript pass parameter by value or by reference?

Primitive type (string, number, etc.) are passed by value and objects are passed by reference

```
var num = 10,
    name = "Addy Osmani",
    obj1 = {
      value: "first value"
    },
          obj2 = {
     value: "second value"
          },
          obj3 = obj2;

function change(num, name, obj1, obj2) {
    num = num * 10;
    name = "Paul Irish";
    obj1 = obj2;
    obj2.value = "new value";
}
```

What advantage is there for using the arrow syntax for a method?

```
let foo = function() {
 console.log('BAZ')
}
let bar = () => {
 console.log('BAZ')
}
foo() // BAZ
bar() // BAZ
```

Avantages:

1. It reduces a lot of code and makes it more readable.
2. The greatest advantage of having contextual “this”, no longer need to “bind” functions any more.
3. Most modern browsers support Arrow Functions out of the box. Although I will still advice using a transpiler like Bable to polyfill for backward compatibility.

Why you might want to create static class members?

The static keyword defines a static method for a class. Static methods aren't called on instances of the class. Instead, they're called on the class itself. These are often utility functions, such as functions to create or clone objects.

```
class ClassWithStaticMethod {
  static staticMethod() {
    return 'static method has been called.';
  }
}

console.log(ClassWithStaticMethod.staticMethod());
```

What is the difference between event bubbling and capturing?

Event Bubbling:
When an event happens on an element, it first runs the handlers on it, then on its parent, then all the way up on other ancestors.

```
<style>
  body * {
    margin: 10px;
    border: 1px solid blue;
  }
</style>
```

```
<form onclick="alert('form')">FORM
  <div onclick="alert('div')">DIV
    <p onclick="alert('p')">P</p>
  </div>
</form>
```

A click on the inner <p> first runs onclick:

1. On that <p>.
2. Then on the outer .
3. Then on the outer <form>.
4. And so on upwards till the document object.

So if we click on <p>, then we'll see 3 alerts: p → div → form.
The process is called "bubbling", because events "bubble" from the inner element up through parents like a bubble in the water.

Event Capturing:
There's another phase of event processing called "capturing". It is rarely used in real code, but sometimes can be useful.
The standard DOM Events describes 3 phases of event propagation:

1. Capturing phase – the event goes down to the element.
2. Target phase – the event reached the target element.
3. Bubbling phase – the event bubbles up from the element.

How does prototypal inheritance work, and how is it different from classical inheritance? (this is not a useful question IMO, but a lot of people like to ask it)

Class Inheritance:
A class is like a blueprint — a description of the object to be created. Classes inherit from classes and create subclass relationships: hierarchical class taxonomies.

Ex:

```
class GuitarAmp {
}
class BassAmp extends GuitarAmp {
}
class ChannelStrip extends BassAmp {
}
```

Prototypal inheritance:

Whenever we create an object in javascript. It will create a default prototype(_proto_) property. That prototype property will be linked to the another object. That object's prototype property will be linked to another object. This is the prototype chain.

We can access the properties method of all the objects with 1 object.

Can you give an example for destructuring an object or an array?

```
// we have an array with the name and surname
let arr = ["Ilya", "Kantor"]

// destructuring assignment
let [firstName, surname] = arr;

alert(firstName); // Ilya
alert(surname);  // Kantor
```

How will you push something on an array?

```
Var array = [‘a’,’b’,’c’,’d’,’e’];

//To add in the end
array.push(“end”);
//To add in the start
array.unshift(“end”);
//To remove in the end
array.pop();
//To remove in the start
array.shift();
```

In ES6

```
Var array = [‘a’,’b’,’c’,’d’,’e’];
array = [“start”,...array, “end”];
```

How to create a private variable in javascript?

```
function getName(){
  var name="Shobana";
  return function(){
    return name;
  }
}
var getMyName = getName();
console.log(getMyName());

// Shobana
```

```
var a = 4;
function outer(){
  var a = 3;
```

```
  function inner(){
    var a = 2;
    console.log(a);
  }
inner();
}
outer();

//2
```

What is the output?

```
console.log(typeof typeof 1);

// string
```

What is higher order function?
What is the extent of your experience with Promises and/or their polyfills?
What are the pros and cons of using Promises instead of callbacks?
What is event loop? What is the difference between call stack and task queue?
ES6 Template Literals offer a lot of flexibility in generating strings, can you give an example?
Can you give an example of a curry function and why this syntax offers an advantage?
How could you implement cache to save calculation time for a recursive fibonacci function?
How could you cache execution of any function?
If you need to implement the following chaining with call back, how will you implement it?
How could you implement moveLeft animation?
How would you implement currying for any functions?

What are the differences between ES6 class and ES5 function constructors?
Do you lint your code? If so, which linters do you regularly use?
What is an AJAX request?
What is JSON?
What is an async response?
What does JSON stand for? What is it used for?
How would you explain an API request?
What APIs have you worked with?
How would you send out/convert to JSON format?
What is the difference between GET and POST when specifying an AJAX request?
Have you worked with Photoshop as a front end developer?
For what purpose?
What was the hardest project you ever worked on? Why was it difficult?
If it is an application/algorithm can you explain how it works?

What is REST, and why do people use it?
Explain Ajax in as much detail as possible.
What are the advantages and disadvantages of using Ajax?
Explain how JSONP works (and how it's not really Ajax).
What do you think of AMD vs CommonJS?
How do you organize your code? (module pattern, classical inheritance?)
What's the difference between feature detection, feature inference, and using the UA string?
Have you ever used JavaScript templating? If so, what libraries have you used?
Why is extending built-in JavaScript objects not a good idea?
Explain the same-origin policy with regards to JavaScript.
What is Big O notation, and why is it useful?
What are the benefits of using spread syntax and how is it different from rest syntax?