1.History of javascript

In 1990 , the internet was new and was becoming famous. In 1993, Mosaic was one of the best browsers available at that time.
Netscape launched their browser netscape navigator.
Then the giant Microsoft launched the internet explorer which was free .
Then Netscape wanted to develop a website with which the users could interact. Not just a static page but a page with components and interactions.

Netscape partnered with Brendan Eich . He created javascript in just 10 days in May 1995. Javascript was originally named Mocha. Then renamed it to Livescript but to market the language well they changed the name to Javascript as java was well known by then.

In 1996, Microsoft developed Jscript which is a language inspired by javascript for internet explorer.This led to the standardization of JavaScript by the European Computer Manufacturers Association (ECMA), resulting in the first version of the ECMAScript standard in 1997.
Node.js is an engine or environment which can be used to build an application. Current version of ecmascript is ecmascript12. Today, JavaScript is one of the most widely used programming languages in the world, powering different kind of applications

2.Data Types in javascript 1:
Datatypes in js is divided into two parts : Primitive and Object

Primitive:
Number: Any kind of number(integer, floating, exponential)
 Several operations can be performed on number like addition, subtraction, multiplication, division, modulus, power etc
Safe range: 9007199254740991 (will not lose any data)

3.Data Types in javascript 2:
Strings: Storing a group of character or text
Several operations can be performed on strings.
Concatenation:
Let a=”hi”
Let b=”everyone”
console.log(a+b)--> hi everyone
console.log(“hi”+`”everyone”`)--> hi “everyone”
\n will print the output in next line

Boolean: true/false
console.log(5>6)--> false

Null: absence of a value
Let a=null
console.log(typeof a)-->object

Undefined:
Let x;
console.log(x)--> undefined
console.log(typeof x)-->undefined

4.Type Conversion & coercion:

Type Conversion: This is an explicit conversion done by the developer where one type is converted into another type.
let num = String(42);
console.log(num, typeof num)--> 42 string

let str = Number("23");
console.log(str, typeof str)--> 23 string

Type Coercion: This is an implicit conversion done by javascript where it converts one data type to other according to the context.
let result = "5" + 3;
console.log(result)--> "53"
When + operator is used with string , it concatenates

let result = "5" - 3;
console.log(result)--> 2
Operator only performs subtraction. So string "5" is converted to number

5.Arithmetic operators: Performs operations like addition, subtraction, multiplication, division, modulus, power etc
Let a=7
Let b=2
console.log(a+b)-->9
console.log(a-b)-->5
console.log(a/0)--> infinity
console.log(a*b)-->14
console.log(a%b)--> 1
console.log(a++)-->7
console.log(++a)-->8
Let a=true
Let b=true
console.log(a+b)-->2

6.Relational Operators: Used to compare two operands
console.log(5>6)-->false
console.log(10>6)-->true
console.log(z>a)-->true
console.log("6">4)-->true, converts the"6" into number
console.log("1"==1)-->true

console.log(“1”===1)-->false

7.Logical Operators:
Logical AND (&&):
let x = 5;
let y = 10;
console.log(x > 3 && y < 15); →true
console.log(x > 6 && y < 15); → false
Logical OR (||):
console.log(x > 3 ||y < 6);-->true
console.log(x > 6 && y < 15);-->true
Logical NOT (!):
console.log(!(x))-->false
console.log(!(x>4))--false

8.Ternary Operators: 1st expression is the condition, 2nd expression is the output is the 1st expression is true, 3rd expression will be the output if the 1st expression is false
Let a=100
Ley b=99
console.log(a>b? a:b)-->100

9.Template Literal: Template literals in JavaScript are a way to work with strings that allow for easier formatting, and creating multi-line strings.
Let x=6;
Let y=5;
Let result=x*y;
console.log(`The product of &{x} and ${y} is ${result});-->The product of 6 and 5 is 30

console.log(`hi
Everyone
I’m Vaishali`)-->
hi
everyone
I’m Vaishali