25-> function sumValues(x,y,z){

Return x+y+z;

}

Console.log(sumValues(…[1,2,3]))//6 spread operator do array spread-1,2,3;

sumValues([…1,2,3]); sumValues([…[…1,2,3]]); sumValues([1,2,3]);

26-> const name = ‘code step by step.’; // is main do operation perform ho rhe hai -javascript operation work left to right.

Console.log(!typeof name===’object’);//false -> !typeof name=false, false ===’object’

Console.log(!typeof name===’string’);//false !typeof name=string === ‘string’ = ! (negation)always return opposite value.

Console.log(!typeof name === false);//true , first operation return false and false ===false= true

27->const name = “Javed Khan”;

Const age = 21;

Console.log(isNaN(name));//true

Console.log(isNaN(name)); //false

28->let person = {name:”Anil”};

Object.seal(person);//seal lagane se aur key value add nhi kara sakte hai jo present hai usko modifiye kr sakte ho.

Like person.name = “Khan”;

Console.log(person);

29-> let data = [2,4,3,54];//remove first element

Data.shift();

Console.log(data);

30-> let data = [2,4,35,34];//remove last element

Data.pop;

Console.log(data);

31-> check no odd or even

Console.log(a%2);//if divide ==0; odd

32-> let data = {

Name:”javed”

}

Delete data.name;

Console.log(data); // return blank object.

33-> let data = “true”;

Covert data to Boolean false value;

Console.log(!data);//false

34-> let data = “true”;

//convert data to Boolean true value.

Console.log(!!data);//first negation return false second negation return true;

35-> difference between map and foreach function

Map function kuch na kuch return karega but foreach kuch b return nhi karega.

36->Let data = [“sss”,”ddd”,”dfdf”];

Delete data[1]; console.log(data); [“sss”, empty, “dfdf”] //jab b koyi value delete hoti hai wo empty space create krte hai null and undefine kuch na kuch value leta hai.

37-Let data = [“sss”,”ddd”,”dfdf”];

Delete data[1]; console.log(data.length);//3

38-> let a = [1,2,3]; let b = [1,2,3];

Let c = […a,..b] console.log(c)//[1,2,3,1,2,3];//ye dono value print karega//object case main kabi b ek object k ander same name ki key dobar nhi aa sakti hai lekin ek array k ander same name ki value aa sakti hai.

39->let c =3\*\*3; console.log(c)// 27 iska mtlab hai 3 ka squre 3;

40->let a=2; setTimeout(()=>{

Console.log(a);//100

},0);

a=100;//jab b ham log setTimeout, setIntervel etc use karte hai to ye krta hai ek stack banta hai javascript k ander is ko api or special function bol dete hai inko sabse bad main pick krta hai ye bolta hai pehale sara code execute hoga uske bad special function call hote hai.

40-> let a=2; let A=30; console.log(A);//30 both are different variable wen can put different value in let case;

41->let A10 = “like”; console.log(A10);// let 10A = “value” console.log(10A);//give error bcz javascript k ander ek rule hai jo b variable starting main number nhi ho skta hai.

42->let a = “like”; let b=`like`; console.log(a===b);//true;

42-> let a=1; let c=2; console.log(--c === a);//true bcz c decrement

43->let a=1; let b=1; let c=2; console.log(a===b===--c);//false bcz 1==1 == true ===1 ; compare with Boolean === Number; bcz left to right javascript

44->console.log(3\*3);//9 console.log(3\*\*3);//27 console.log(3\*\*\*3);//error bcz 3 star kabi b kam nhi krta hai.

45->console.log(a); var a; undefined // ham log variable ko bad main declare kr sakte h pehale use kr sakte h; error isliye aaya kyu ki is main value nhi thi var case main.

Undefined ka mtlab hota hia declare to kr diya hai lekin us main kuch value nhi dali hai.

46->console.log(b);//error no defined. declare nhi kiya hai

47->console.log([[[[]]]]);//ye nested array print kare ga pehale array k ander ek array hai second k ander dusra array hai dusre k ander tisra array hai. Last wale k ander 0 element hai.

48->How to find OS name -> Javascript main ek property hoti hain navigator.platform

49->let for= 100;//bcz for reserverd key word hota hai.

50-> function fruit(){

Console.log(name);//undefine //hosting main pehale use karsakte hai bad main declare kr sakte hai var k case main

Console.log(price);//error yaha pr hosting ka concept aata hai . hosting bolta hai let k case main pehale let ko declare kro bad main use kro. Hosting ka rule in javascript;

Var name=”apple”; let price=20;

}

Fruit();

51-> for (var i=0; i<3; i++){

setTimeout(()=>console.log(i),1);//3 //bcz pehali bar for loop chla, ye special type k function api k ander chale jata hai aur sabse bad main call hote hai. Tab i ki value 3 ho chuki thi i ek global variable ban chuka tha kyu ki jab var k pehale koyi bracket nhi hota h tab ye ek gobal variable ban jata hai.

}

52-> for (let i=0; i<3; i++){

setTimeout(()=>console.log(i),1)//0,1,2 bcz let ka scope block level tak hota hia. Jab b for loop chala her bar ek different i bana jab setTimeout chala to is main different i gya is liye ye different i print ka rega.

}

53-> console.log(+true);//1 jab b ham kisi b value se pehal + lagte hai to wo number main convert ho jata hai.

Console.log(typeof +true);//number

54->console.log(!“anil”);//false console.log(typeof(“Javed Khan”));//string

55-> let data = “size”; const bird = { size:”small”};

console.log(bird[data])//small bcz jab b ek variable use krna hota hai object k sath main to [] brackets use krni hoti hai. Size pass kiya to ye size ban gya like -bird[size]=small

console.log(bird[“size”])//small agr string pass karoge to b chalega.

console.log(bird.size);// small

console.log(bird.data);//undefine

56->var x; var x=10; console.log(x);/10 ek bar nhi x ko 50 bar b declared karege to b last wali value print ka rega.

57->let x; let x=10; console.log(x);//error x ko re-declared nhi kr sakte hai.

58-> let a =3; let b= new Number(3);

console.log(a==b);//true //console.log(a===b);//false bcz b typeof object bcz a ka type number hai and b ka type object hai.

59->let name; nmae= {};//typo! console.log(name);//yaha pr mile ga blank; console.log(nmae);//{}

60-> function fruit(){ console.log(“woof!”); }

fruit.name = “apple”; console.log(fruit());//woof! //if we add extra property its run fine.

61->