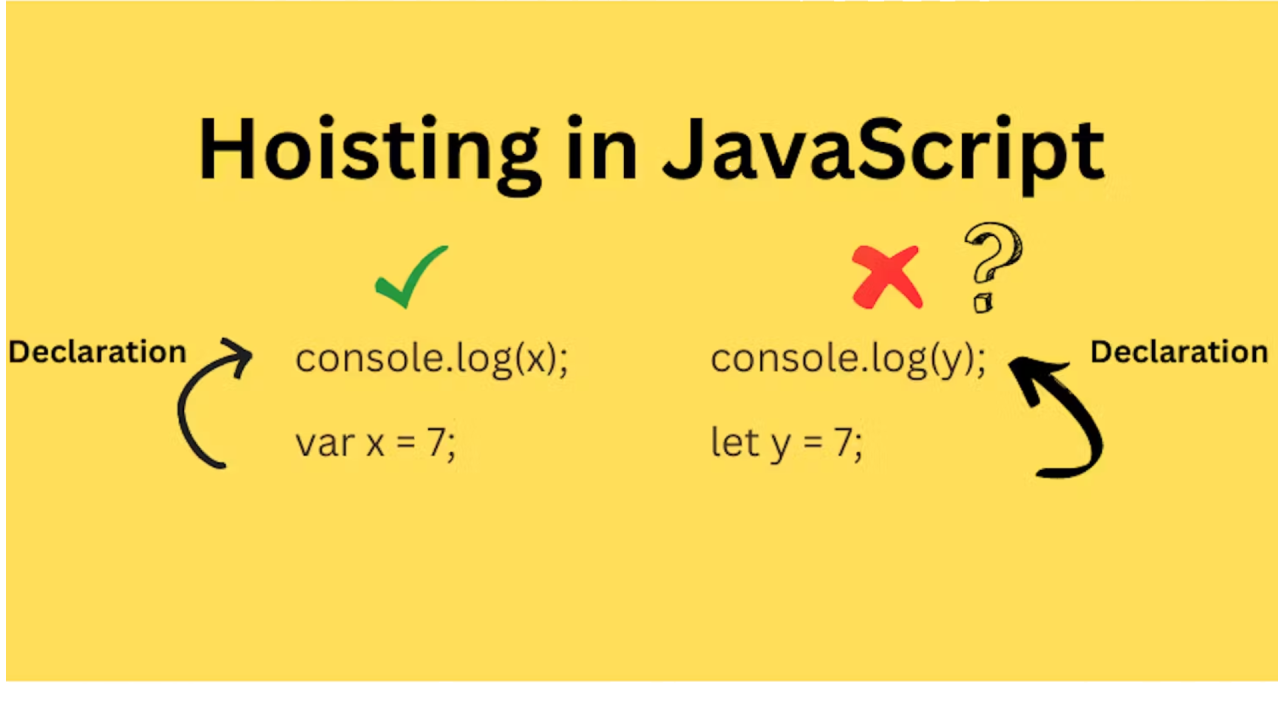
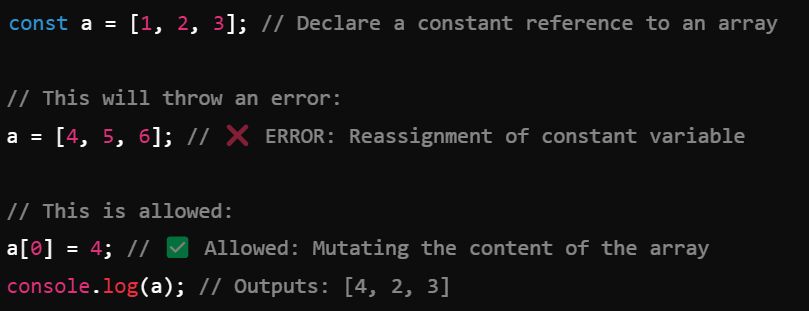
Hoisting in JavaScript is a built-in behavior that moves the declarations of functions, variables, and classes to the top of their scope before the code is executed

Here are some things to keep in mind about hoisting:

* Hoisting only moves declarations, not initializations.
* Hoisting moves declarations to the top of their scope regardless of whether the scope is global or local.
* There are generally two types of hoisting in JavaScript: variable hoisting and function hoisting.





const myMessage = 'Hello world'

console.log(myMessage)

// String a="nandhini"

=============================

const myMessage = 'Hello world'

console.log("my data is: ",myMessage)

// my data is:

================================

const myName = 'sai'

const myRoll=12

console.log("my name is: ",myName,"my roll is: ",myRoll)

console.log(my name is: ${myName} and my roll is ${myRoll})

// my data is: sai my roll is: 12

=============================================

let reassign redeclare function scope

var reassign no block scope

const no no block scope

=================================

const myName = 'sai'

myName='ram'

console.log(`my name is: ${myName} `)

// my data is: sai my roll is: 12 // err

======================================

var myName = 'sai'

myName='ram'

console.log(`my name is: ${myName} `)

// my data is: sai my roll is: 12

=============================================

let myName = 'sai'

myName='ram'

console.log(`my name is: ${myName} `)

// my data is: sai my roll is: 12

====================================

var myName = 'sai'

function disp(){

myName = 'ram'

console.log('im inside the fun: ',myName)

if(true){

myName = 'nandhini'

console.log('im inside the block: ',myName)

}

}

disp()

console.log('im inside the fun: ',myName)

============================

const myName='outside'

function disp(){

const myName = 'inside func'

console.log('im inside the fun: ',myName)

if(true){

const myName = 'inside the block'

console.log('im inside the block: ',myName)

}

}

disp()

console.log('im outside the function: ',myName)

======================================

var myName='1'

function disp(){

if(false){

var myName = '2'

}

console.log('im inside the block: ',myName)

}

disp()

console.log('im outside the function: ',myName)

=========================

let myName='1'

function disp(){

if(true){

let myName = '2'

}

console.log('im inside the block: ',myName)

}

disp()

console.log('im outside the function: ',myName)

==============================

//var b=10

function disp() {

//let a=10

b=10

}

disp()

//console.log(a);

console.log(b);

============================

const person={

roll:23,

stuName:"sai"

}

person.stuName="ram"

console.log(person.stuName);

const per=[12,'nan',90,89]

const a=[23,1,4,6]

/\*

{

0:23

1:1

2:4

3:6

}

\*/

a[0]=45

console.log(a);

========================== a='sai'

b="ram"

c=`hai erwer

weerewrewrewrw werwerwerwe werwerewrewr `

console.log(a);

console.log(b);

console.log(c);

===============================

console.log("Hello, World!");a = 12;

b = 32432423442343242342342222222222n;

c = Infinity;

d = 10 / 0;

e = 5 < Infinity;

d = 'hai' \* 3;

f=NaN

console.log(a);

console.log(b);

console.log(c);

console.log(d);

console.log(e);

console.log(d);

console.log(f);

=========================================

let a= false

let b= true

console.log(a)

console.log(b)

console.log (typeof (a))

let a=[12,4,"sai",23.4]

let datas=[100,200,300]

let person={name:'reena', roll:5}

console.log(a)

a.push(datas,800,person)

console.log(a)

console.log(a[4])

a[4].push(400)

console.log(a)

============================

let a=[12,4,"sai",23.4]

let datas=[100,200,300]

let person={name:'reena', roll:5}

console.log(a)

a.unshift(datas,800,person)

a.unshift(800)

console.log(a)

console.log(a[0])

=========================================

let a=[12,4,"sai",23.4]

let datas=[100,200,300]

let person={name:'reena', roll:5}

a.push(datas,800,person)

//res=a.pop()

res=a[4].pop()

console.log("result" ,res)

console.log(a)

===========================================

let a=[12,4,"sai",23.4]

res=a.shift()

console.log("result" ,res)

console.log(a)

**Day 2**  
  
const person={

name:'sai',

roll:34,

gender:'male'

}

let {name,roll,gender}=person

console.log(name," ",roll," ",gender)

console.log(person)

======================================

const person={

n:'sai',

r:34,

g:'male'

}

let {n:name,r:roll,g:gender}=person

console.log(name," ",roll," ",gender)

console.log(person)

====================================

const person={

n:'sai',

r:34,

g:'male'

}

let {n:name,r:roll,g:gender, city='mdu'}=person // also assign default value

console.log(name," ",roll," ",gender, city)

console.log(person)

=================================

const person={

name:'sai',

roll:34,

gender:'male',

city:'cbe'

}

let {name,...restdatas}=person

console.log(name,restdatas)

==================================

const person={

name:'sai',

roll:34,

gender:'male'

}

let {name,roll,gender}=person

console.log(name," ",roll," ",gender)

console.log(person)

======================================

const person={

n:'sai',

r:34,

g:'male'

}

let {n:name,r:roll,g:gender}=person

console.log(name," ",roll," ",gender)

console.log(person)

====================================

const person={

n:'sai',

r:34,

g:'male'

}

let {n:name,r:roll,g:gender, city='mdu'}=person // also assign default value

console.log(name," ",roll," ",gender, city)

console.log(person)

=================================

const person={

name:'sai',

roll:34,

gender:'male',

city:'cbe'

}

let {name,...restdatas}=person

console.log(name,restdatas)

==================================