

Objects

There are Two types for defining object

1. **Object literal**
2. **Object constructor**

note - agr object literal se bnega to singleton nhi bnega lekin constructor se bnega too singleton bnega

1. Object literals

```
let obj = {  
  key : value  
  name: "gaurav",  
  "Muje access":"kro",  
  designation: "software engineer",  
}
```

how we access value of object

```
1 = console.log(obj.name)  
2 = console.log(obj["name"])  
3 = 4th line ko sirf console.log(obj["muje access"])  
se hi access kr skte ha
```

How we update object values

```
obj.name = "Don Hu may"  
console.log(obj.name)  
  
Ouput  
let obj = {  
  key    value  
  name: "Don Hu may",  
  surname: "sondhiya",  
  "Muje access":"kro",  
  designation: "software engineer",  
  place: "Banglore",
```

```
}
```

obj.freeze()

Agr muje object ki value ko update hone se rokna ha to may use kruga freeze method to phir object ki value update nhi hogi.

```
Object.freeze(obj)
Object.freeze(object ka naam)
```

```
object.surname="may hu gangster"
console.log(object.surname)
output is sondhiya because object freeze ha
```

-Interview question Symbol ko as a key kese store kre

```
const name= Symbol("key1")
```

```
let obj1= {
  [name]:"may key ka value hu"
}
console.log(obj1)
output => [Symbol(key1)]: 'may key ka value hu'
```

Object ke andr function kese define kre or kese access krna ha

```
let obj3= {
  name: function(){
    console.log("gaurav sondhiya")
  }
}

console.log(obj3.name)
output => [Function:name]
```

```
console.log(obj3.name())  
output => gaurav sondhiya
```

This keyword

sirf object ke andr jo Function create hota ha uspr hi work krta ha

```
let obj= {  
  name: "khalnayak",  
  fullname:function (){  
    console.log(`${this.name} sondhiya`)  
  }  
}  
  
console.log(obj.fullname())  
Output-  
khalnayak sondhiya
```

This

Object ke andr jo object create hota ha usmay this keyword work nhi krta ha

```
let obj = {  
  name:"khalnayak",  
  fullname:{  
    surname:this.name  
  }  
}  
  
console.log(obj.fullname.surname)  
Output-  
undefined
```

Construtor object

when we create construtor object that only time singleton is created it's not applicable for literal object

const gaurav = new Object() **singleton object**

```
const obj= {}
```

Nonsingleton object or Object literals

Object create using New keyword

```
const gaurav = new Object()  
gaurav.surname = "sondhiya"  
gaurav.designation= "sde"
```

```
console.log(gaurav)
```

OutPut

```
{  
  surname: 'sondhiya',  
  designation: 'sde'  
}
```

```
let obj = {  
  name: "gaurav",  
  surname: "sondhiya",  
  "Muje access": "kro",  
  designation: "software engineer",  
  place: "Banglore",  
  floorgino : {  
    isfloor: {  
      secondfloor: {  
        thirdfloor: {  
          fourthfloor: "bs yhi tak hu may"  
        }  
      }  
    }  
  }  
}
```

How to access fourthfloor value

```
console.log(obj.floorgino.isfloor.secondfloor.thirdfloor.fourthfloor)
```

Question mark "?"

Help krta error define ke liya mention kiya to undefined aayega nhi too error aayega

```
console.log(obj.floorgino.istfloor?.secondfloor.thirdfloor.fourthfloor)
```

Merge 2 or more Object in Under one Object

Object.assign({},obj)

```
let obj1 = {
  1:"dd",
  2: "gg",
  3:"aa"
}

let obj2={
  4:"qq",
  5:"rr",
  6:"ii",
}

let obj3 =Object.assign({},target object,source object),

**curly bracket- new object ke andr saare obj merger krte ha**

excutiion-

let obj3 =Object.assign({},obj1,obj2)
console.log(obj3)

output -
{ '1': 'dd', '2': 'gg', '3': 'aa', '4': 'qq', '5': 'rr', '6': 'ii' }
```

But instead we use **Spread Operator** for merging(always use for merging this method)

```
let obj3 = {...obj1,...obj2}
console.log(obj3)
{ '1': 'dd', '2': 'gg', '3': 'aa', '4': 'qq', '5': 'rr', '6': 'ii' }
```

Method for creating array using object key and values

1. Object.keys()
2. Object.values()

3. Object.entries()

example = 1. Object.keys()

```
let obj1 = {  
  1:"dd",  
  2: "gg",  
  3:"aa"  
}  
console.log(Object.keys(obj1))  
OUTPUT -  
[ '1', '2', '3' ]
```

example= 2. Object.values()

```
let obj1 = {  
  1:"dd",  
  2: "gg",  
  3:"aa"  
}  
console.log(Object.values(obj1))  
OUTPUT =  
[ 'dd', 'gg', 'aa' ]
```

example= 3. Object.entries()

```
let obj1 = {  
  1:"dd",  
  2: "gg",  
  3:"aa"  
}  
console.log(Object.entries(obj1))  
OUTPUT =  
[ [ '1', 'dd' ], [ '2', 'gg' ], [ '3', 'aa' ] ]
```

HasOwnProperty

object ke andr ye values ha ki nhi ye btata ha true or false may

```
let obj1 = {  
  1:"dd",  
  2: "gg",  
  3:"aa"
```

```
}
console.log(obj1.hasOwnProperty(may iss object may hu kya))
output
false
```

How to access array of object

```
let obj =
[
    {id:"ga"},
    {id:"dd"},
    {id:"ee"},
    {id:"yy"},
]

let ans= obj[3].id
console.log(obj)
```

Important topic **Object de-structure**

```
let obj = {
    name: "gaurav",
    surname: "sondhiya",
    "Muje access":"kro",
}

method 1-
agr muje surname baar-baar use krna ha to may
let {surname} = obj
ye use kruga issay muje
obj.surname likhna nhi padega baar-baar.

console.log(surname)
output -
sondhiya
-----

method 2- or short krskte ha
let obj = {
    name: "gaurav",
    surname: "sondhiya",
    "Muje access":"kro",
}
agr muje surname bhi short krke likhna ha to ese use kruga
let {surname:s} = obj
ab muje srif s mention krna hoga jb bhi muje surname use krna ho to
```

```
console.log(s)
```

```
output -  
sondhiya
```

Delete Keywords how to use and how to remove value using delete

```
const user = {  
  name:"gaurav",  
  age:23,  
  "may hu don":true  
}  
  
delete user.name  
console.log(user)  
  
output  
{ age: 23, 'may hu don': true }
```

Computed Properties

```
const user= "firstname"  
const name = "Gaurav"  
  
const obj = {  
  [user]:name //key change already assign ke hissab se change hogya ha  
}  
console.log(obj)  
  
Output  
{ firstname: 'Gaurav' }
```

For IN LOOP

```
let obj = {  
  name: "gaurav",  
  surname: "sondhiya",  
  designation: "software engineer",  
  place: "Banglore"  
}  
  
1- for (key in obj){  
  console.log(key)  
}
```


output 1

name
surname
designation
place

```
2- for (key in obj){  
    console.log(obj[key])  
}
```

**** Output****

gaurav
sondhiya
software engineer
Banglore