

## 2. Object:-

// Key value

```
const user = {  
  name: "Rohit",  
  age: 20,  
  emailId: "Rohit.mahato@gmail.com",  
  amount: 3400.  
}  
console.log(user);
```

// agar kisi specific value ko read karna ho to  
eg:- age: 20 ko read karna hai.

~~console~~

→ console.log(user.age);

agar object me kisi value ko insert karna  
hai to.

```
user.address = 1234;  
user.amount = 5000;  
console.log(user);
```

// ~~delete~~ delete karne hai to.

```
delete user.emailId;  
console.log(user);
```

Note:- All are known as CRUD operator :-  
.. Create, Read, update, ~~delete~~ Delete.

Note :- object reference ko copy karta hai:-

eg:-

```
const user = {  
  name: "Rohit",  
  age: 20,  
  emailId: "Rohitkushka@gmail",  
  amount: 3400;  
}
```

Changes  
done  
me reflect  
hoga.

```
const user2 = user;  
user2.age = 30;  
console.log(user);
```

# agar sirf keys chahiye toh:-

```
console.log(object.keys(user));
```

agar sirf values chahiye toh:-

```
console.log(object.values(user));
```

agar dono chahiye:-

```
console.log(object.entries(user));
```



Isko loop ki help se krli kar sktte hai.

```
for (let userkey in user) {  
  console.log(keys);  
}
```

or

```
for (let key in user) {  
  console.log(keys, user[key]);  
}
```

Q. Object ko destructing krna.

```
const { name, age } = user;  
console.log(name, age);
```

or

```
const arr = [10, 20, 40, 50, 11];  
const [first, second] = arr;  
console.log(first, second);
```

output (10, 20).

Q. ek function create krna ka job:

```
const user = {  
  name: "Rohit",  
  age: 20,  
  amount: 3400,  
  greeting: function() {  
    console.log("Hello world");  
    return 20;  
  }  
};
```

```
const val = user.greeting();  
console.log(val);
```

2 Nested object:-

```
const user user = {  
  name: "Rohit";  
  age: 20,  
  amount: 3400,  
  Address: {  
    city: "Bokaro",  
    State: "Jharkhand"  
  }  
}  
  
console.log(user.address.city);
```