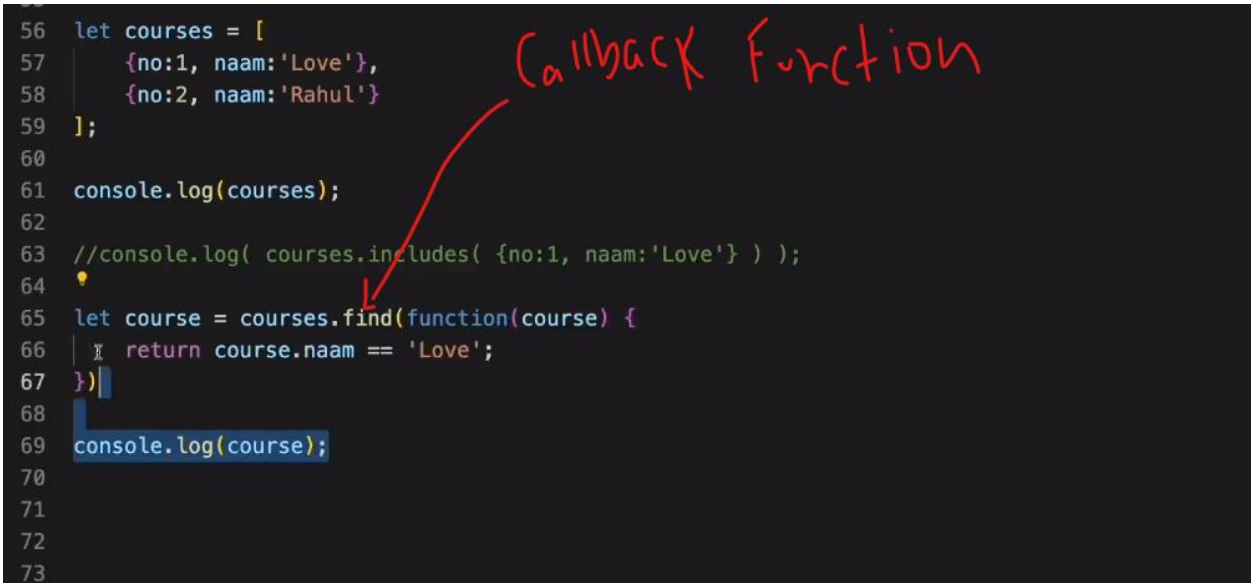
Class - 3

Callback function



```
let courses = [
    {no:1, naam:'Love'},
    {no:2, naam:'Rahul'}
];

console.log(courses);

//console.log( courses.includes( {no:1, naam:'Love'} ) );

let course = courses.find(function(course) {
    return course.naam == 'Love';
})

console.log(course);
```

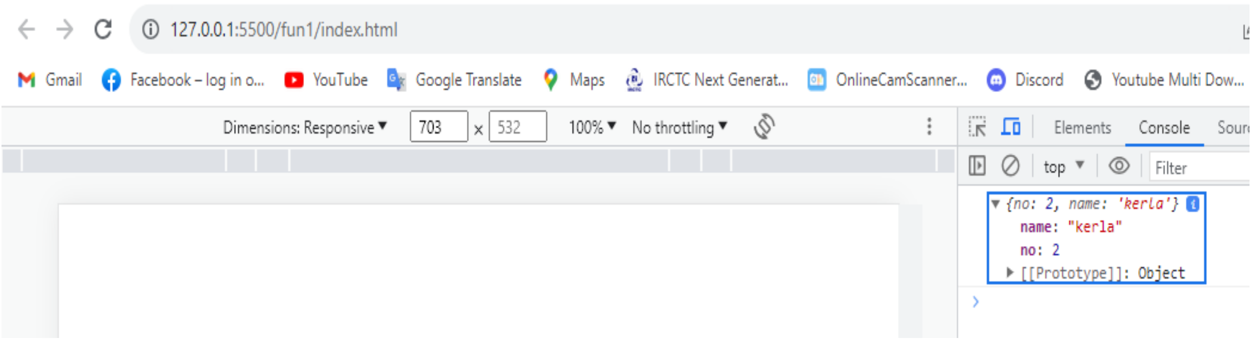
Html code :

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <script src="/fun1/index.js"></script>
</body>
</html>
```

JavaScript code :

```javascript
//callback function objct mai search karna

let city=[
    {no:1,name:'mumbai'},
    {no:2,name:'kerla'},
    {no:3,name:'goa'}
];

let cityData=city.find(function(cities){
    return cities.name=='kerla';
});

console.log(cityData);
```

output :



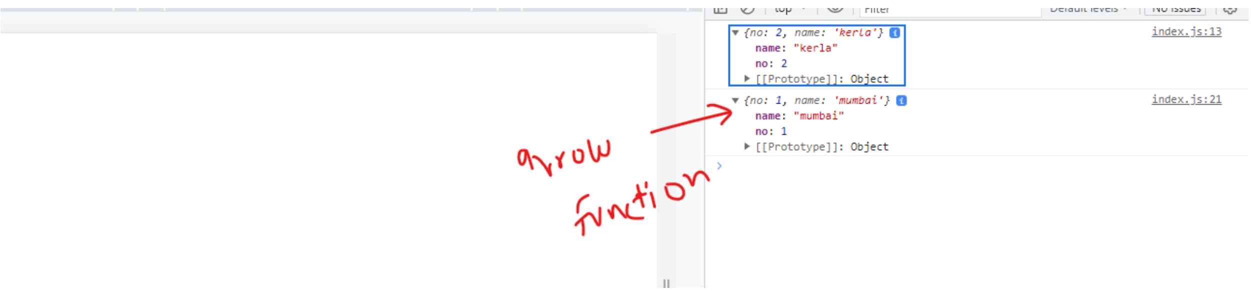
127.0.0.1:5500/fun1/index.html

Dimensions: Responsive 703 x 532 100% No throttling

```
▼ {no: 2, name: 'kerla'}
    name: "kerla"
    no: 2
  ▶ [[Prototype]]: Object
```

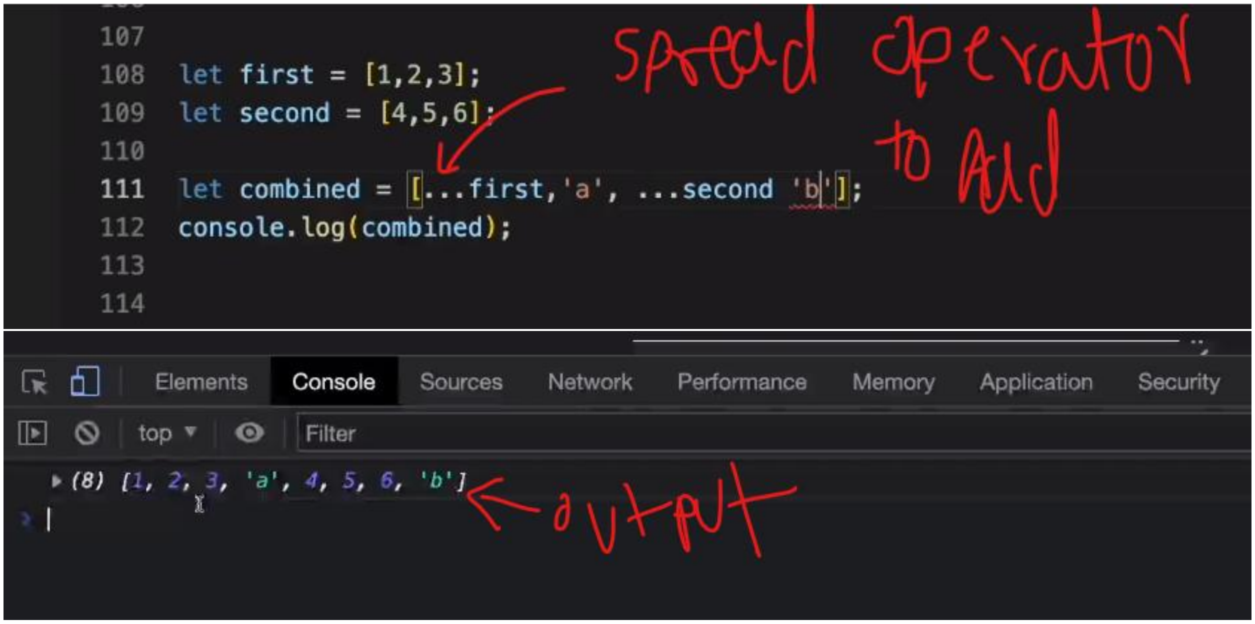
Arrow function

```
//callback function objct mai search karna

let city=[
    {no:1,name:'mumbai'},
    {no:2,name:'kerla'},
    {no:3,name:'goa'}
];

let cityData=city.find(function(cities){
    return cities.name=='kerla';
});

console.log(cityData);


//using arrow function
let cityData2=city.find(cities=>
     cities.name=='mumbai'
);

console.log(cityData2);
```



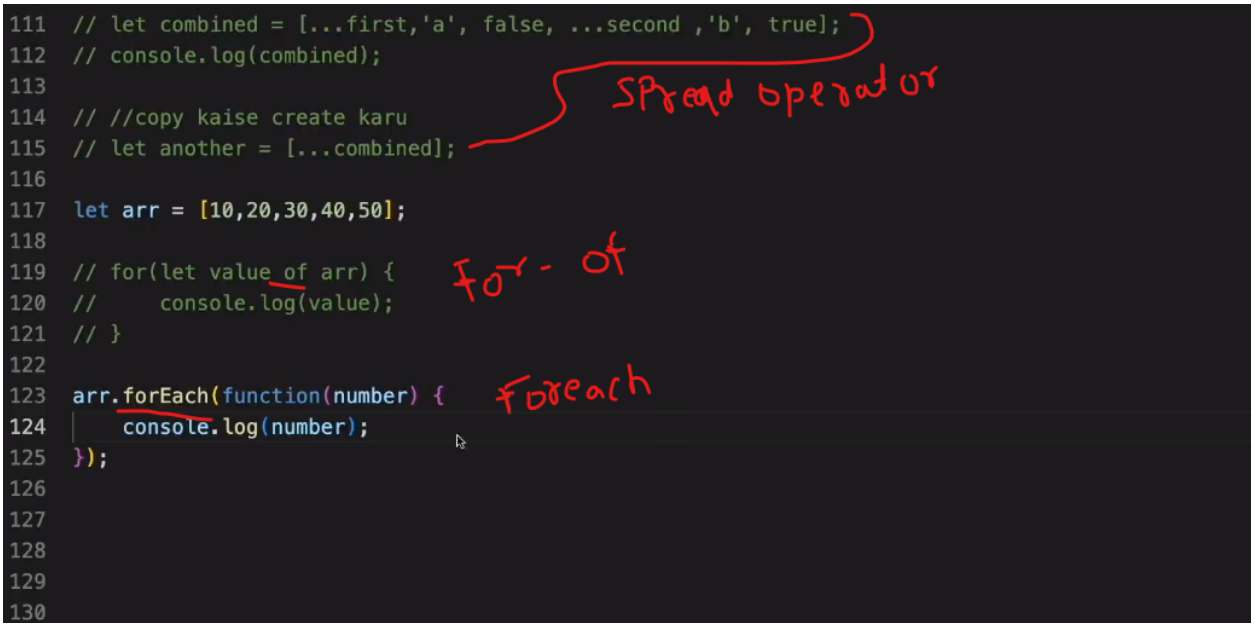
```
▼ {no: 2, name: 'kerla'}                                   index.js:13
    name: "kerla"
    no: 2
  ▶ [[Prototype]]: Object
▼ {no: 1, name: 'mumbai'}                                  index.js:21
    name: "mumbai"
    no: 1
  ▶ [[Prototype]]: Object
>
```

arrow function



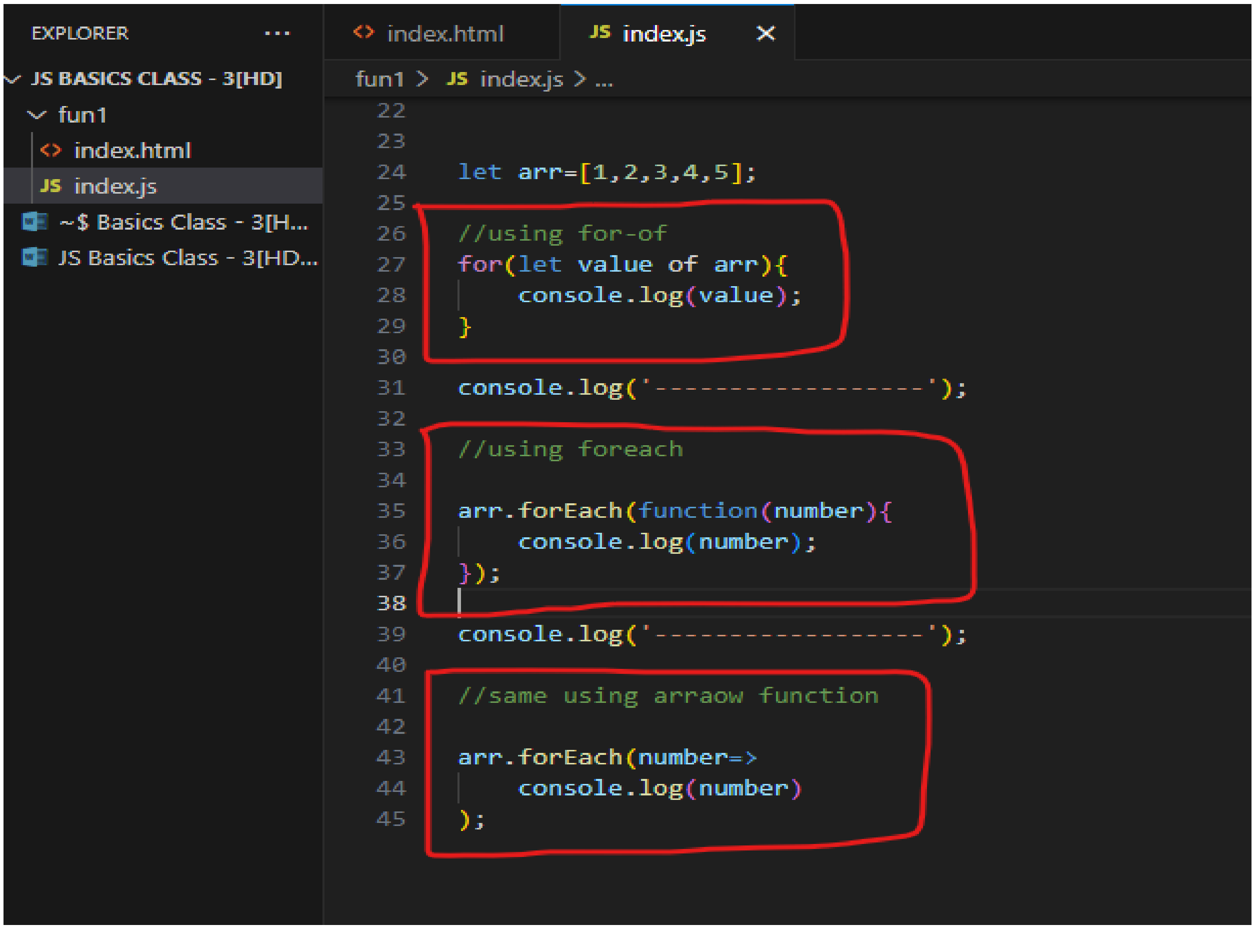
```
let first = [1,2,3];
let second = [4,5,6];

let combined = [...first,'a', ...second 'b'];
console.log(combined);
```

```
▸ (8) [1, 2, 3, 'a', 4, 5, 6, 'b']
```

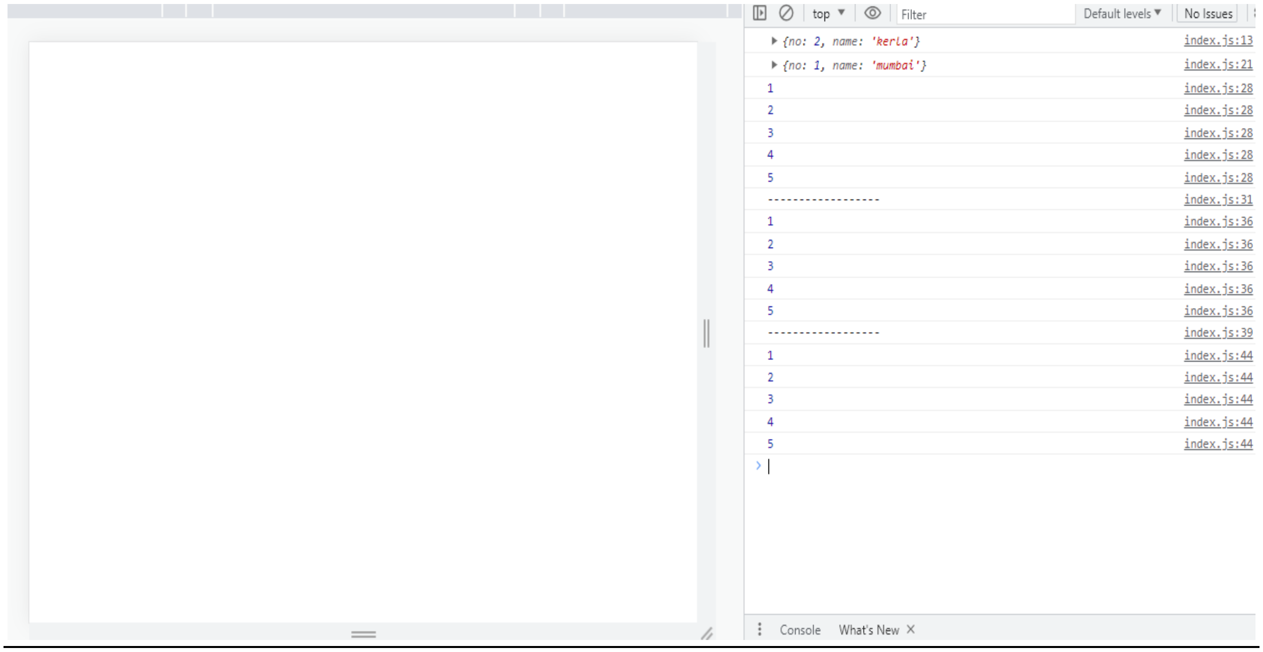


```
// let combined = [...first,'a', false, ...second ,'b', true];
// console.log(combined);

// //copy kaise create karu
// let another = [...combined];

let arr = [10,20,30,40,50];

// for(let value of arr) {
//     console.log(value);
// }

arr.forEach(function(number) {
    console.log(number);
});
```

Using foreach

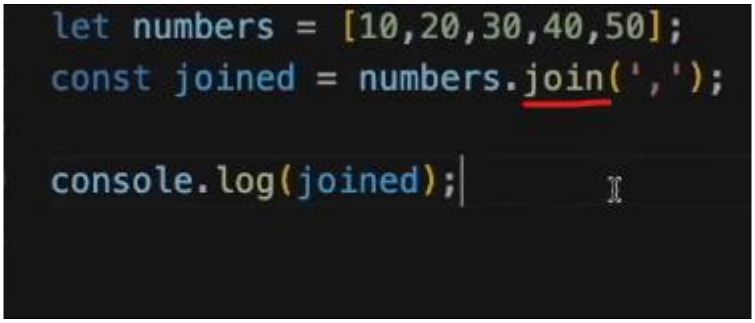


```javascript


let arr=[1,2,3,4,5];

//using for-of
for(let value of arr){
    console.log(value);
}

console.log('-----------------');

//using foreach

arr.forEach(function(number){
    console.log(number);
});

console.log('-----------------');

//same using arraow function

arr.forEach(number=>
    console.log(number)
);
```

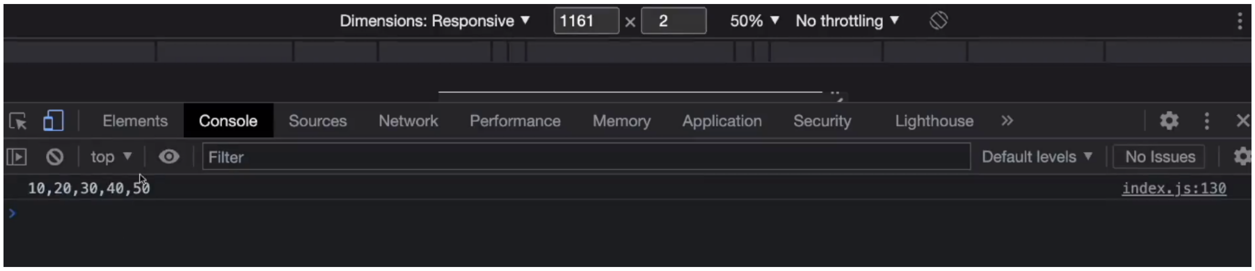


```
▸ {no: 2, name: 'kerla'}          index.js:13
▸ {no: 1, name: 'mumbai'}         index.js:21
1                                 index.js:28
2                                 index.js:28
3                                 index.js:28
4                                 index.js:28
5                                 index.js:28
-----------------                 index.js:31
1                                 index.js:36
2                                 index.js:36
3                                 index.js:36
4                                 index.js:36
5                                 index.js:36
-----------------                 index.js:39
1                                 index.js:44
2                                 index.js:44
3                                 index.js:44
4                                 index.js:44
5                                 index.js:44
>
```

Array join



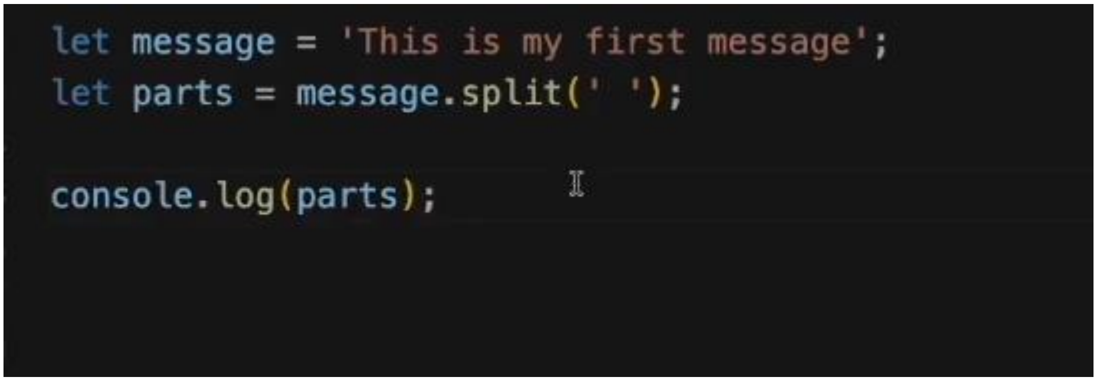
```
let numbers = [10,20,30,40,50];
const joined = numbers.join(',');

console.log(joined);
```



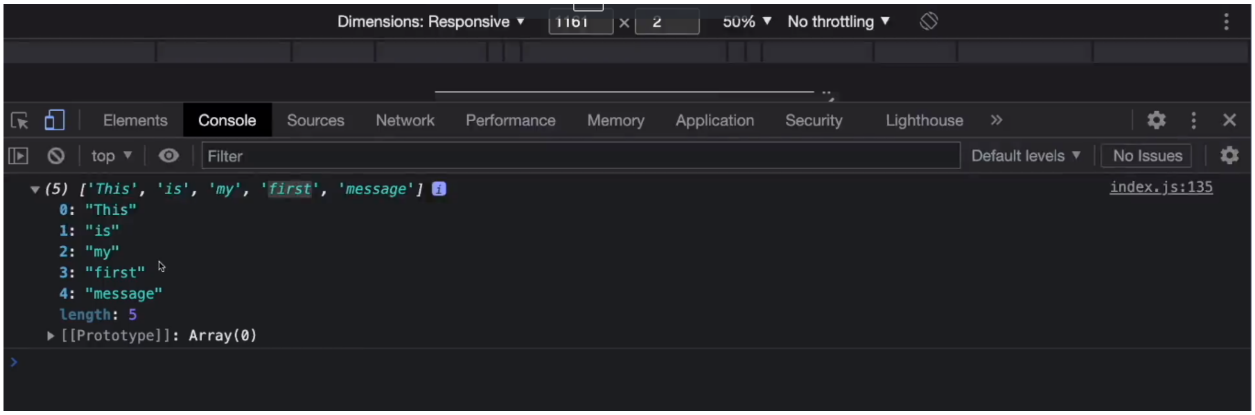
Dimensions: Responsive ▼ 1161 × 2 50% ▼ No throttling ▼

```
10,20,30,40,50                                                    index.js:130
```

Split: split wala method ek array create kar deta hai



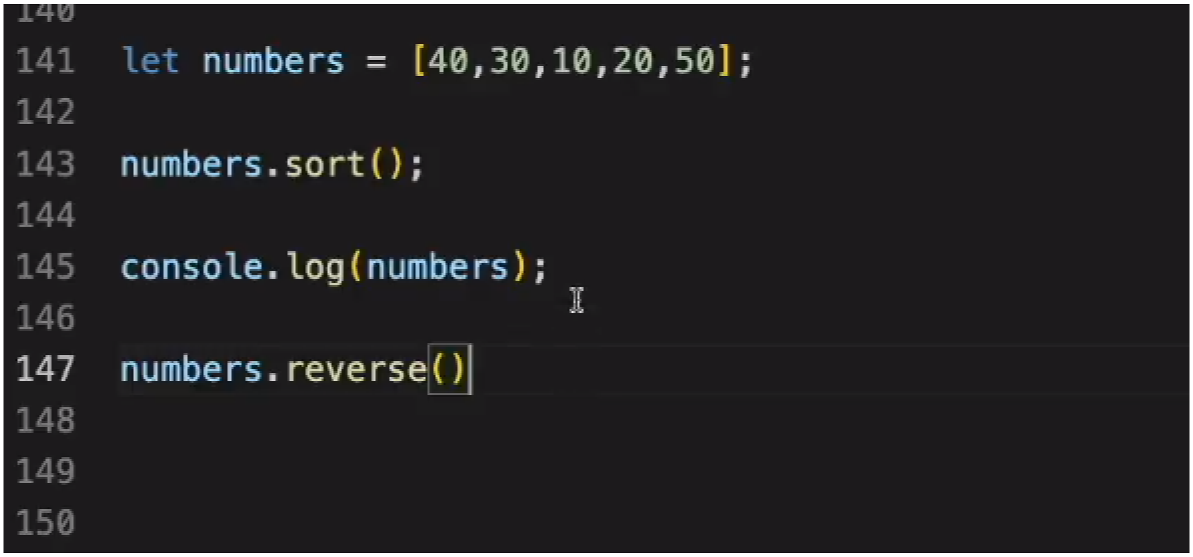
```
let message = 'This is my first message';
let parts = message.split(' ');

console.log(parts);
```



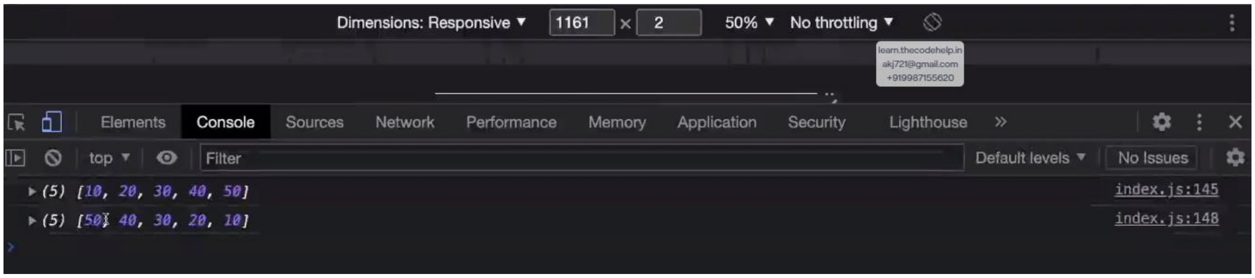
Dimensions: Responsive ▼ 1161 × 2 50% ▼ No throttling ▼

```
▼(5) ['This', 'is', 'my', 'first', 'message']                     index.js:135
    0: "This"
    1: "is"
    2: "my"
    3: "first"
    4: "message"
    length: 5
  ▶[[Prototype]]: Array(0)
```

Sort

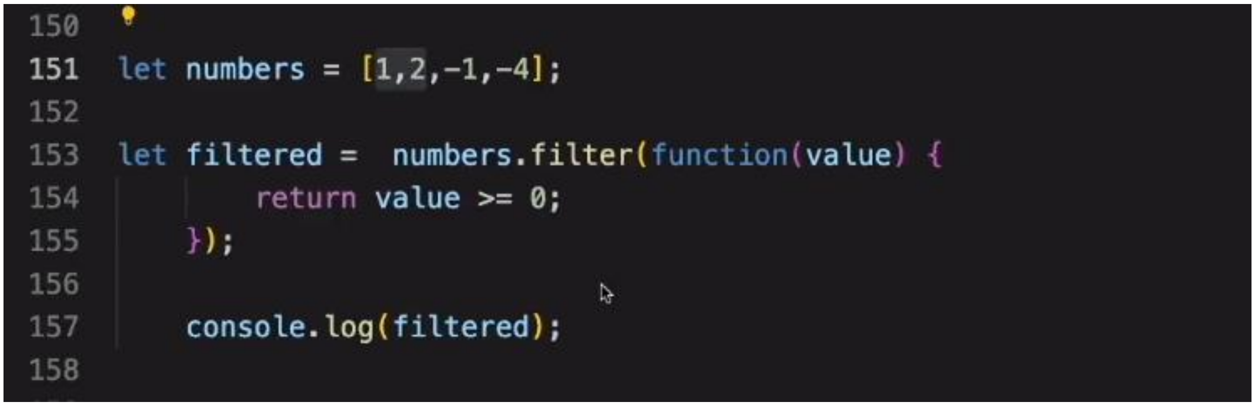


```
let numbers = [40,30,10,20,50];

numbers.sort();

console.log(numbers);

numbers.reverse()
```

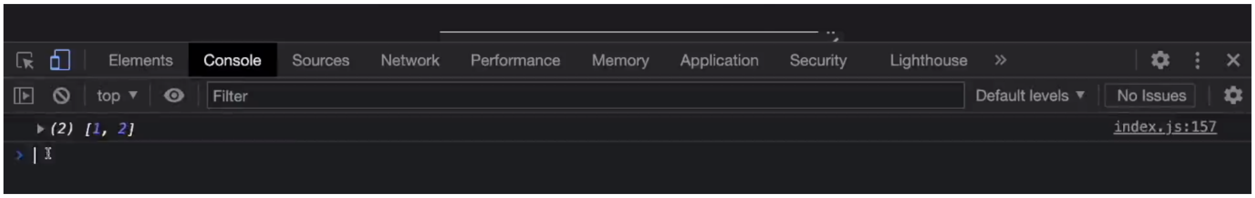


```
▸(5) [10, 20, 30, 40, 50]                                   index.js:145
▸(5) [50, 40, 30, 20, 10]                                   index.js:148
```

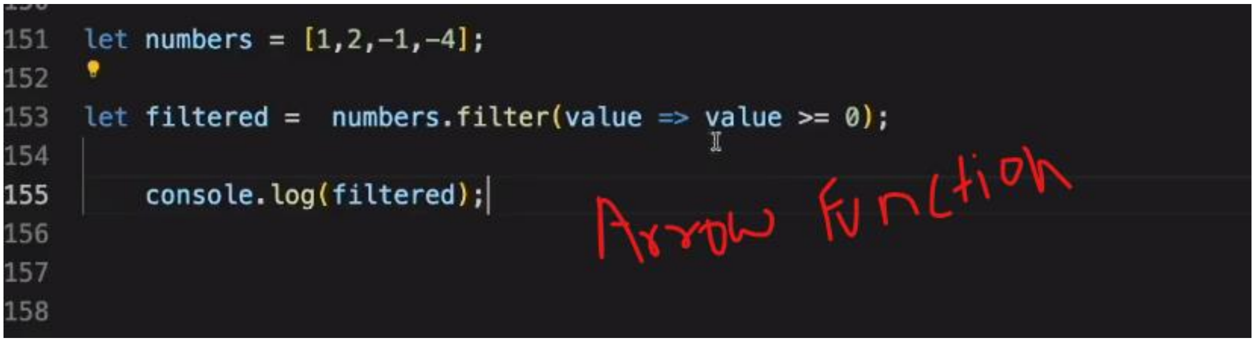
Filter



```
let numbers = [1,2,-1,-4];

let filtered =  numbers.filter(function(value) {
        return value >= 0;
    });

    console.log(filtered);

```

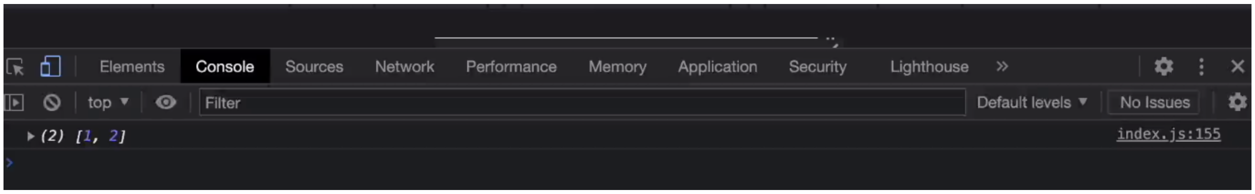


```
▸(2) [1, 2]                                                index.js:157
>
```



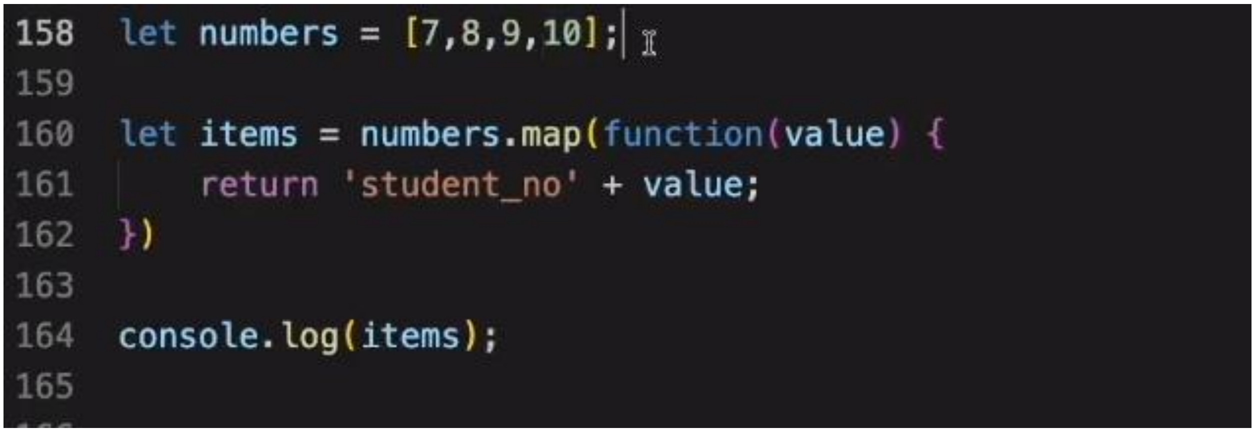
```
let numbers = [1,2,-1,-4];

let filtered =  numbers.filter(value => value >= 0);

    console.log(filtered);



```

Arrow function

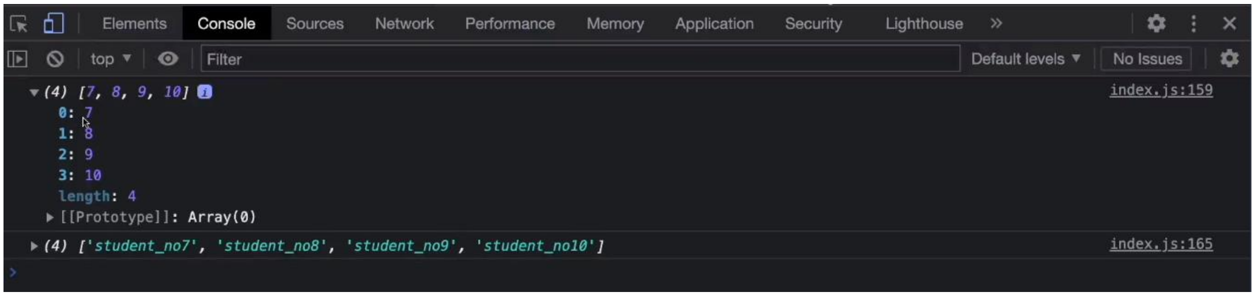


```
▸(2) [1, 2]                                                index.js:155
>
```

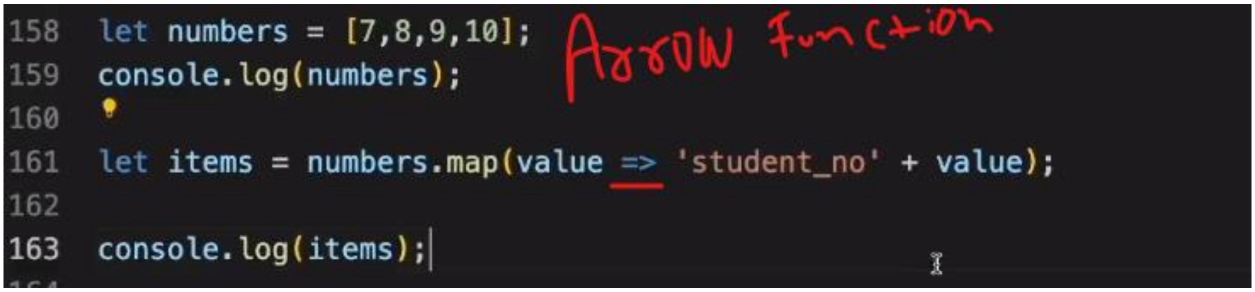
Mapping array



```
let numbers = [7,8,9,10];

let items = numbers.map(function(value) {
    return 'student_no' + value;
})

console.log(items);
```

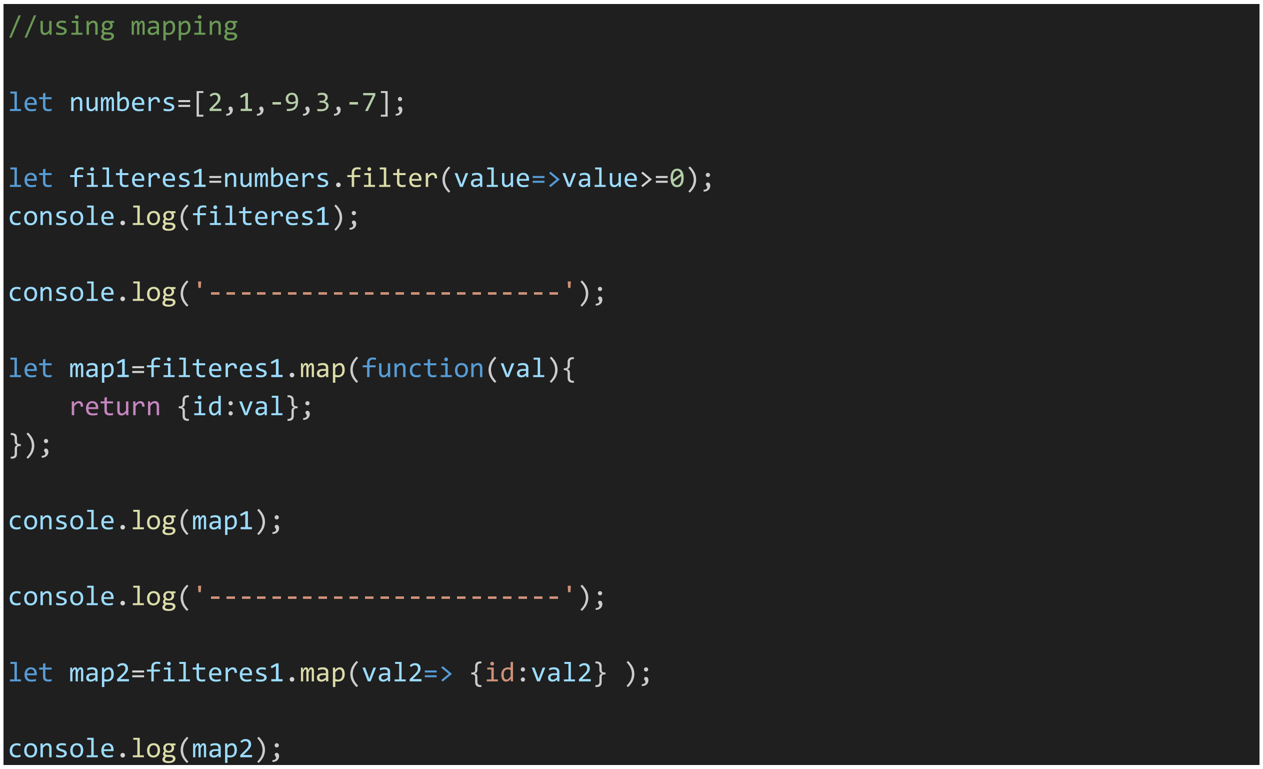


```
▼(4) [7, 8, 9, 10]                                                    index.js:159
    0: 7
    1: 8
    2: 9
    3: 10
    length: 4
  ▶[[Prototype]]: Array(0)
▶(4) ['student_no7', 'student_no8', 'student_no9', 'student_no10']   index.js:165
>
```

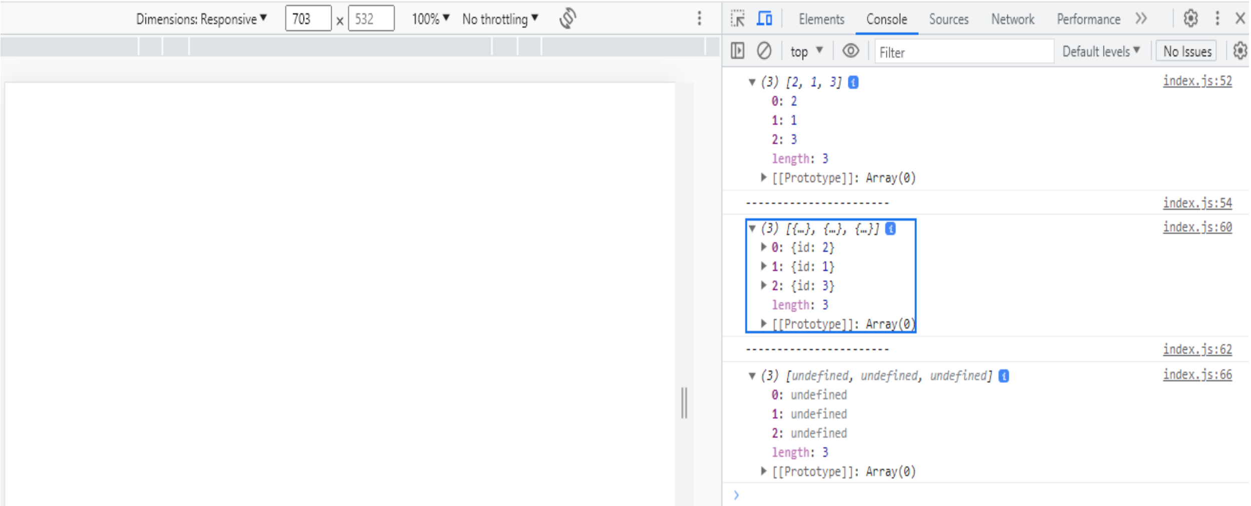


```
let numbers = [7,8,9,10];
console.log(numbers);

let items = numbers.map(value => 'student_no' + value);

console.log(items);
```

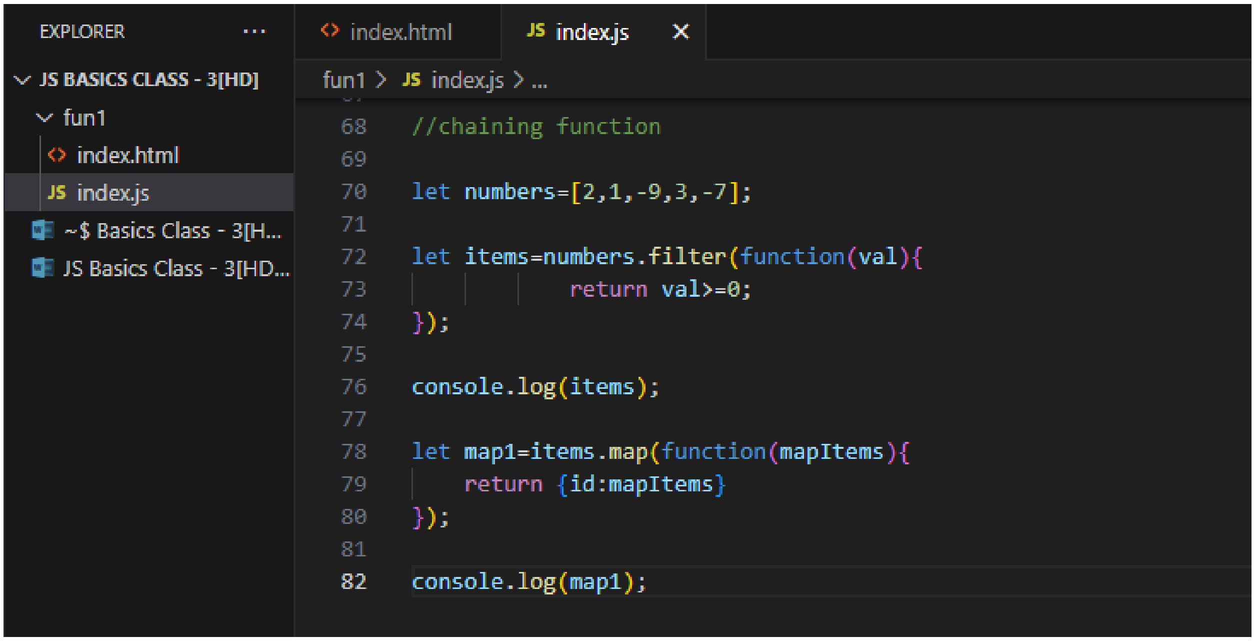
Arrow function



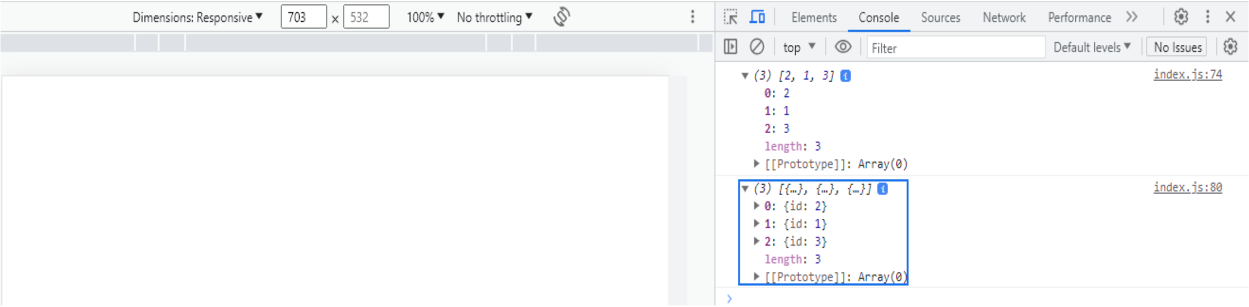
```
//using mapping

let numbers=[2,1,-9,3,-7];

let filteres1=numbers.filter(value=>value>=0);
console.log(filteres1);

console.log('-----------------------');

let map1=filteres1.map(function(val){
    return {id:val};
});

console.log(map1);

console.log('-----------------------');

let map2=filteres1.map(val2=> {id:val2} );

console.log(map2);
```



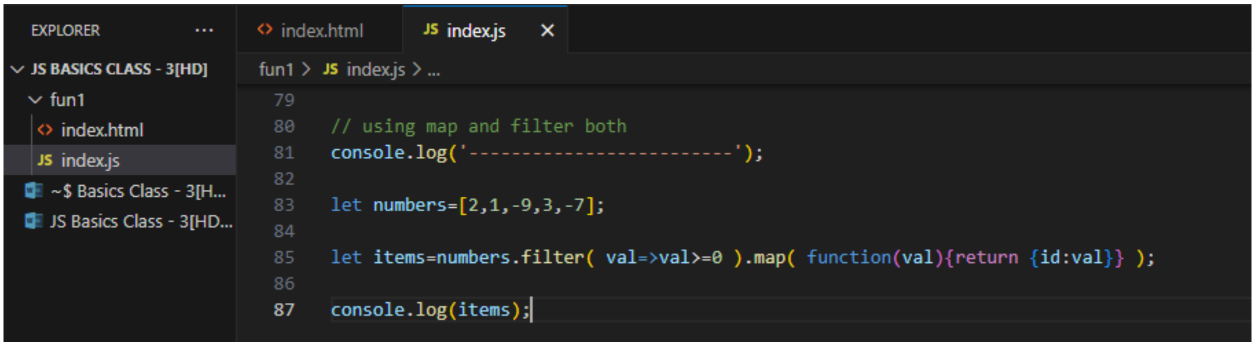
```
▼ (3) [2, 1, 3] ℹ                                      index.js:52
    0: 2
    1: 1
    2: 3
    length: 3
  ▶ [[Prototype]]: Array(0)
-----------------------                                index.js:54
▼ (3) [{…}, {…}, {…}] ℹ                                index.js:60
  ▶ 0: {id: 2}
  ▶ 1: {id: 1}
  ▶ 2: {id: 3}
    length: 3
  ▶ [[Prototype]]: Array(0)
-----------------------                                index.js:62
▼ (3) [undefined, undefined, undefined] ℹ              index.js:66
    0: undefined
    1: undefined
    2: undefined
    length: 3
  ▶ [[Prototype]]: Array(0)
›
```



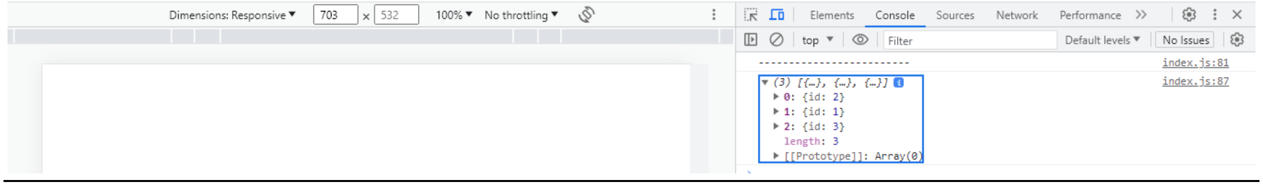
```javascript
//chaining function

let numbers=[2,1,-9,3,-7];

let items=numbers.filter(function(val){
            return val>=0;
});

console.log(items);

let map1=items.map(function(mapItems){
    return {id:mapItems}
});

console.log(map1);
```



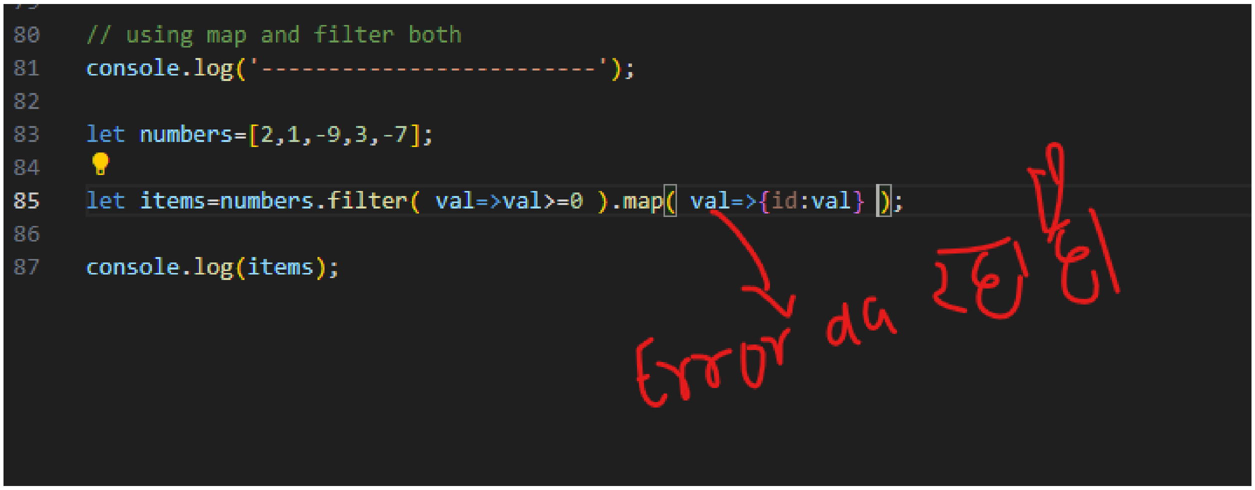
```
▼ (3) [2, 1, 3]                                   index.js:74
    0: 2
    1: 1
    2: 3
    length: 3
  ▶ [[Prototype]]: Array(0)
▼ (3) [{…}, {…}, {…}]                             index.js:80
  ▶ 0: {id: 2}
  ▶ 1: {id: 1}
  ▶ 2: {id: 3}
    length: 3
  ▶ [[Prototype]]: Array(0)
>
```



```
// using map and filter both
console.log('------------------------');

let numbers=[2,1,-9,3,-7];

let items=numbers.filter( val=>val>=0 ).map( function(val){return {id:val}} );

console.log(items);
```

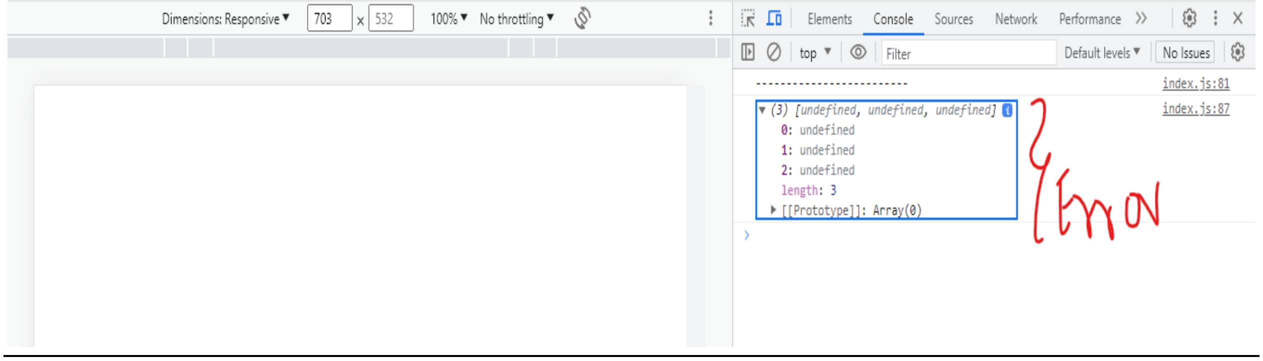


```
------------------------                         index.js:81
▾ (3) [{…}, {…}, {…}]                            index.js:87
  ▸ 0: {id: 2}
  ▸ 1: {id: 1}
  ▸ 2: {id: 3}
    length: 3
  ▸ [[Prototype]]: Array(0)
```



```
// using map and filter both
console.log('------------------------');

let numbers=[2,1,-9,3,-7];

let items=numbers.filter( val=>val>=0 ).map( val=>{id:val} );

console.log(items);
```

Error da ਏਥੇ ਹੈ



```
------------------------                         index.js:81
▾ (3) [undefined, undefined, undefined]          index.js:87
    0: undefined
    1: undefined
    2: undefined
    length: 3
  ▸ [[Prototype]]: Array(0)
```

{ Error