ARRAYS

In arrays you can put more than one values. In other languages array mein agar kisi nay number add kiyee tu saari values ko number hi daaalna parega, ya phirr agr string add kiye tu sari values ko string hii raknaa parega and so on matlab k data type same hona chahiyee.. magar js mein hm values ki data type ko change b rakh sktay hain like

Let a = [ 1, 4, 5, 13, “ali”, true];

Ye iss mein hm agar in values ko access karna chahein tu ye jo array mein values hoti hain ye index mein save hoti hain like jo pahli value hoti hai wo 0 index pr hoti hai 2nd value 2nd index pr hotiii haii and so on. Like agar hm iss tarah karein

Let marks = [4,88,100,50];

Console.log(Marks[0]); tu ue humein 4 print kr k de dega.

Agar hm koi aise value maangein tu array mein exist hii nai krti tu phirr humein ye undefined ka error dega.

Marks[0] = 100;

Tu ye jo 0 index prr value haii uss mein 100 dal degii tu matlab jo pahlii value 4 thiii ab uss ki jagah 100 agayeee hogii...

Methods In Array:

Push: to add something in array like marks.push(500) so this will add 500 inside marks array.

Pop : to remove something from array. Marks.pop() this will remove the last value from array. So in marks array this will remove 500 because that is in last index.

Shift: it will remove value from first position like marks.shift() so this will remove first value in marks array.

Unshift: ye starting mein yani k 0 index prr koi b number ya cheez ko add karta hai like

Marks.unshift(599); tu ye 599 ko add kardega 0 index pr

Splice():

Marks.splice(3,2) : tu iss ka matlab haii k ye ho splice hai ye b values ko array say nikalay gaaa magar hum apnii marzi say inded day sktay hain jaisay k yahan prr hm nay bataya hai k marks.splice(3,2) 3rd index say value ko nikalo orr wahan say 2 values ko nkialo matlab k agar hm eg lein marks = [4,2,5,280,292,88] tu ab yahan parr ye 280 ko orr dosri value jo haii wo 292 ko bahir kar deingi from array qk hm nay kaha hai splice ko k humein 3rd index say value ko nikalnaa haii or 2 values ko nikalna hai

Slice(): iss mein hm iss ko dengay k kahan say kahan orr kahan tuk hatana hai like for eg marls.slice(0,5); tu ab ye humein 0 sy lekar from 4th index tuk wo values hata krr de dega. Ab iss mein question ata hai k slice vs splice mein difference kiaa hai tu ye jo splice hai ye agar koi value ko delete karta hai tu wo original array say hii un values ko hata deta hai. And ye jo slice hai ye aik new copy banaatta hai matlab k agar hm ussee original array ko print karengay tu uss mein koi change naii dhike wo waisay hii rahay gaa

Iss ko agar hm koi new array mein save kr lein orr phir print kar lein tu phir hm new array ko dhaik sakengay like let newArr = marks.slice(0,3);

Tu agar hm example lein slice method kii tu given below

let numbers = [

  53, 89, 10, 82, 19, 300, 599, 45, 200, 85, 68, 19, 29, 69, 88, 100,

];

console.log(numbers);

console.log("before slice method");

let nerNum = numbers.slice(0, 5);

console.log(nerNum);

console.log("original array after slice method");

console.log(numbers);

tu iss mein hm nay aik array banai haii numbers ki. Uss k baad hm nay aik new variable liya nerNum k name say orr numbers array prr slice ka method apply kiyaa tu ye 0 index say le kr 5th position tuk un numbers ko uss numbers k array say delete kr raha hai orr jo new nerNum ka variable haii uss mein siraf un ko safe kar raha hai magar jo original numbers name ka array hai wo wohi same rahigi matlab uss original mein koi farq nai parega wahan values utnay hiii rahengay. Matlab ye jo values nikal raha hai ye aik aur copy banaye ga uss array ka or un numbers jo delete kiye hain array say un ko new array mein save kr lega. Matlab k main element yani k array change nai hota. Jo b changes hongi wo humein new variable yani k array mein milengi.

Reverse: this will reverse the elements of array like marks.reverse()

Sort: matlab k jo values hain un ko ascending ya descending order mein krna tu kr skay hain like for eg iss ko use krnay k liyee, ye humeisha aik function accept krta hai orr uss k parameter mein hm 2 variables dengay jin ka name koi b ho skta hai. Orr return karega wo variables. Tu for eg hm nay aik variable ka name rakh diya first,and dosray ka rakha second. Agar return mein hm nay rakhaa k first-second tu ye jo array haii ye humein ascending mein kr dega orr agar hm nay return mein second – first likh diyaa tu ye humein array ko descending mein kr dega like eg given below

let numbers = [

  53, 89, 10, 82, 19, 300, 599, 45, 200, 85, 68, 19, 29, 69, 88, 100,

];

let nnn = numbers.sort(function(first,second){

    return first-second;

});

This will make array in ascending order.

Ye jahan prr function ka use ho raha haii yahan prr hm fat arrow function ka b use kr sktay hain agar chahein tu =🡺 imp

forEach:

let numbers = [

  53, 89, 10, 82, 19, 300, 599, 45, 200, 85, 68, 19, 29, 69, 88, 100,

];

let nnn = numbers.sort(function(first,second){

    return second-first;

});

This will make array in descending order

forEach,Sort,Reduce,map,filter, yo jo saaray method hain ye jab hm Array pr lagayengay tu ye aik function accept kartay hain.

let primeNumbers = [2, 3, 5, 7, 11, 13, 17];

primeNumbers.forEach(function(val){

    return

});

Ye hm nay array banai hai. Ab hm uss array pr aik method apply kartay hain forEach ka

Ye jo forEach ka jo method hai ye jab array prr lagatay hain tu ye har aik value k liye chalay ga orr return mein hm uss k sath kuch b karwa sktay hain matlab k array mein pahlii value 2 hai tu ye jo function ki val mein ye 2 ajaye ga orr ab hm return mein uss 2 k sath jo b karwana chahein tu wo krwa sktay hain.. iss tarah 2nd value k liye ye forEach apply hoga yahan prr 3 hai tu ye 3 k liye forEach apply hoga orr ye jo 3 value hai ye forEach k val mein ajaye giii orr hm return mein uss k sth jo b krwana chahein wo kr sktay hain and so on.

let primeNumbers = [2, 3, 5, 7, 11, 13, 17];

primeNumbers.forEach(function(val){

    console.log(val);

});

Ab ye jo hm nay likhii hai hm nay bss siraf return krnay k bajaye uss ko print kr diyaa jo k siraf simply aik aik ki taraf jaye ga orr aik aik ko print karta rahega.

Simply jo forEach haii ye bsss ye krta haii k har aik liyee ye function jo chalaao yani k jo sari values hain array mein har aik k liyee iss function ko chalaaao..

let primeNumbers = [2, 3, 5, 7, 11, 13, 17];

primeNumbers.forEach(function(val){

    console.log(val + 5);

});

Tu ab ye ye karega k har aik value jo b array mein hai ye sab mein 5 ko add karta rahega. Orr sath mein print b kar dega.

Map: map ham nay tub use krna hai jab apko aik new array banana hai pichlay array k data k basis par. Map ko dikhtay hiii aik new blank array apnay dimagh mein bana liyaa karo.

Jab b hm ko koi aisa case nazar ajaye k aik array say new array banega and wo new array kuch values ko rakhega tub hm map use karengay… .

Jaisay for eg....

let primeNumbers = [2, 3, 5, 7, 11, 13, 17];

primeNumbers.map(function(val){

    return 12;

});

Tu ye jo map hai ye jo new blank array banay gii ye uss mein jiss array pr hm map karwa rahay hain un k members k hisaab say matlab utni baar chalega. Yani k yahan prime number array mein 7 members hain tu jo new blank array hoga uss mein b 7 members hongay. Tu jab ye method chalega ye prmie numbers array mein jo pahlii value hai wo 2 haii tu ye uss k liyee new black array mein 12 ko return kr dega phirr 2nd member k liye chalegaa jo k 3 haii tu ye jo map hai ye wo jo new blank array tha uss mein 3 ki jagah 12 daal dega qk hm nay return karwaya hi 12 hai phir map 5 k liye chalega orr uss ki jagah jo new blank array hai uss mein b 12 daal dega qk hm nay return krwaya hi 12 hai tu iss tarah ye primenumbers mein 7 members hain tu ye map in 7 members k liye chalega orr jo new blank array haii uss mein seven baar 12 daal dega qk hm nay return hii 12 krwaya hai. Ye jo new array banega wo hm kisi variable mei safe kr k phirr print krwa sktay hain like

let primeNumbers = [2, 3, 5, 7, 11, 13, 17];

let blankyArr = primeNumbers.map(function(val){

    return 12;

});

Console.log(blankyArr);

Tu agar hm iss ko ab print karein tu humein [12,12,12,12,12,12,12] milega.

Waisay forEach and map same hi hain magar jo major difference hai wo ye hai k hm agar forEach mein return kuch na karein phirr b koi masla nai hai magar map mein humein must return karna parega jo b return krwana chahein magar return must hai

Magar agar return na karein tu phirr uss mein un saray members mein undefined,undefined,undefined and so on. ye add ho jayega.

let ppp = primeNumbers.map(function (val) {

  if (val > 10) return val;

 else return "Small";

});

console.log(ppp);

ye ab yahan prr ye karega k primeNumber k array pr map chalega tu map tu har aik value pr chalta hai. Hm nay function mein aik condition lagai hai k agar jo map mein jo val mein value save hue hai agar wo >10 hai tu uss ko return kr do agar naii hai greater tu simply uss ki jagah “small” ko return kr do. Or iss ko hm nay ppp mein save kr liyaa hai prr log pr print kr diyaa hai.

Filter : jisss tarah hm nay kahaa thaa k jab b map dhaiko tu dimagh mein aik new array bana lo isse tarah ye filter b jab dhaiko tu dimagh mein aik new array soch lo qk ye aik new array banaye ga map ki tarah.

Is mein map ki tarah return karna must hai.

Tu jiss tarah hm map mein likhtay thay aik value return k baad tu wo value uss new array mein chali jati tu filter mein hum true ya false likhengay return k baad. Matlab ye filter b map ya foreach ki tarah hii har aik value ki tarah chalta hai tu jiss value k ooper ye chlega tu agar hm nay return likha hoga tu wo wala jo element thaa uss array mein wo new array mein chala jayega orr agar hm nay false liha hoga tu wo value jo hai wo new array mein nai jaye gi..

let primeNumbers = [2, 3, 5, 7, 11, 13, 17];

primeNumbers.filter(function(val){

    return true;

});

Tu ab iss mein ye jo filter method hai ye foreach ya map ki tarah pahlay value k ooper chalega jo yaha pr 2 hai tu hm nay return mein true likhaa hai tu ye jo 2 hai wo new array mein store ho jaye ga. Iss tarah ye phir second value k ooper chalega jo k yahan prr 3 hai, hm nay return mein true likhaa hai tu b new array mein store ho jaye ga. tu iss tarah ye saray values k ooper chalega tu jahan prr true likhaa hoga wo store ho jaye ga new array mein orr jahan false wo store nai hoga.

Ab agar hm aik or eg lein

primeNumbers.filter(function (val) {

  if (val > 5) return true;

  return false;

});

Iss mein ye ye karega k jo value primeNumber k array mein hai hm nay uss par filter ka method apply kr liyaa hai orr uss mein condition likhii hai k wo jo value agar >5 hai tu uss ko (true) yani k store kr do new array mein warna (false) na store karo

Reduce : jab b humein ye kaha jaye k ye array haii iss mein jo values likhi haiin in ko reduce karo aik single value mein yani k add kr do minus kr do magar last ka jo ans hoga wo single value mein hoga tu phirr hm reduce method ka use karengay. For eg

let ans = primeNumbers.reduce(function (accumulator, val) {

  return accumulator + val;

}, 0);

Ham nay primenumbers k array pr reduce method ko apply kiyaa tu ye reduce method b function accept krta hai. Is function mein 2 variables ko accept krta hai names koi b ho sktay hain. Ye jo pahlii value hai yahan prr jiss ka name accumulator hai ye ho hai ye value ko yaad kr k rakhtaa hai, orr ye jo 2nd position prr ho hai yani k val ye jo ye method chalega har aik new value par tu ye wo wali value uss mein aati jaye gi, orr ye jo last mein hm nay 0 lagaya hai ye starting value hai accumulator ki, k starting mein value kiaa hogi accumulator ki tu hm nay yahan prr 0 define kr di.

let primeNumbers = [2, 3, 5, 7, 11, 13, 17];

let ans = primeNumbers.reduce(function (accumulator, val) {

  return accumulator + val;

}, 0);

Tu ab ye jo method hai ye primeNumbers array pr chalega tu uss mein jo function hai uss mein 2 variables hain. Tu yahan prr accumulator ki value 0 hogi qk hm nay last mein define ki hai. Ye hm kuch b define kr sktay hain. Tu jab method chalega val men 2 ki value aye gii qk array mein 1st value 2 hai tu ye return karega 0+2 ak accumulator ki value 0 thi orr val mein 2 hai tu ye return kr lega 2. Ab ye 2 value accumulator mein save ho jaye gi tu ab accumulator ki value hai 2. Now ab ye method again 2nd value k liye chalega jo k yahan prr 3 hai tu val mein 3 hai ab orr accumulator mein tu 2 tha tu ye return karega 2 + 3 jo k 5 ho jaye ga orr now ye 5 ab accumulator mein save ho jaye ga tu ab accumulator ki value 5 hai. Ab again method 3rd value k liye chalega jo k yahan pr 5 hai tu ab val mein 5 ajaye ga tu ye return karega 5 + 5 (qk accumulator mein pahlay say 5 tha orr val mein ab 5 hai) ye ab dedega 10 ab ye 10 accumulator mein save ho jaye ga tu ab accumulator ki value 10 hai. Tu iss tarah ye saray array k members k liyee chalega orr iss tarah return krta rahega orr saray members ko add kr lega orr wo value accumulator mein save ho jaye gi orr return kr degi. Us value ko hm kisi or variable mein b safe kr sktay hain. Or phir print.

Another example:

let scores = [45, 89, 72, 60, 100, 34, 88];

let scoresAns = scores.reduce(function (accumulator, val) {

  return accumulator + val;

}, 0);

console.log(scoresAns);

Find: tu find ye krta haii k jo b pahlay banda find ko array say milega jo k condition ko satisfy karega jo k hm nay retun mein dy raki ho tu wo use value ko direct return kar dega. Jaisay k for eg.

marks.find(function (val) {

  return val === 19;

});

Tu ye find method ye karega k jo b iss ko 19 milega tu wo uss ko directly return kr dega. Orr ye 19 iss ko jiss index pr b milega matlab for eg humaray array mein 4, 5 bar 19 hain tu iss ko pahlaa 19 jiss position prrr milega tu ye oss ko bss return kr dega baki ko chor dega. iss ko hm koi or variable mein b save kr sktay hain like

let newmrks = marks.find(function (val) {

  return val === 19;

});

Another Example

let nnn = marks.find(function (val) {

  return val \* 10 === 100;

});

Tu ab hm nay condition ye lagai haii k ye find method jo haii wo ye karega k jo marks array haii uss mein aisi value dhoonday gaa k agar hm uss ko 10 k sath multiply kr lein orr wo 100 k barabar hon. Tu ye uss value ko return kr dega.

Tu iss tarah hm koi b condition laga sktay hain.

Some : ye method ye krta hai k ye find krta hai array mein jaiasy k for eg humara array hai marks ka or humein dhoondna hai k iss mein koi aisay marks hain ya nai k jo k 85 say ooper hon tu hm nay ye condition dedi. Tu ab agar ye condition kisi aik value prr b true ho gayee uss array mein yani k koi aik value b 85 say ooper huee tu ye true return kr dega bs. Orr agar na ho tu false return kr dega. For eg

let markky = marks.some(function (val) {

  return val > 200;

});

Tu ye ye karega k marks array mein doonday ga k koi value aisee hai ya naii jo k 200 say ooper ho. Tu agar ooper ho koi value tu true return kr dega agar nai tu false.

Every: this is fully opposite of some method. Tu ye ye dhoonday ga k iss array mein jo sari value hain wo uss condition ko follow kr rai hain ya nai. Agr kr rai hain tu true return kr dega agar nai tu false.

let am = marks.every(function(val){

    return val>300;

});

Tu ab ye false return karega qk jo marks array haii iss mein sari value tu 300 say greater nai hain tu ye false return kr dega.

De Structureing and spread operator :

Tu for eg humaray pass aik array haii orr hm uss say kuch value chahtay hain k doosray variable mein copy kr lein tu hm spread operator ka use kr sktay hain.

let marks = [

  53, 89, 10, 82, 19, 300, 599, 45, 200, 85, 68, 19, 29, 69, 88, 100,

];

let [aaa,bbb] = marks;

tu ab iss mein hm nay ye kiyaa hai k jo marks ka array hain uss k jo pahlay 2 values hain wo pahli value aaa mein copy ho jaye gi or doosri value bbb value save ho jaye gii. Uss ko dhaiknay k liye hm print kr sktay hain. Ab agar for eg hm chahtay hain k 3rd and 4th value ko chor dein orr jo 5th value hai marks array mein uss ko kisi variable mei save kr lein tu uss k liye hm ye karengay

let marks = [

  53, 89, 10, 82, 19, 300, 599, 45, 200, 85, 68, 19, 29, 69, 88, 100,

];

let [aaa,bbb, , ,kkk] = marks;

tu ab ye ye karega k 3rd ko b chor dega 4th ko b chor dega or jo 5th value hogi marks array mein uss ko copy kr lega kkk variable mein.

So simply this is called de structuring.ye thaa de structuring ka basic sa example.

Spread Operator: for eg hm nay aik whole array ko copy krna hai kisi doosray array mein tu wo hm kaisay karengay tu agar hm iss tarah karein tu

Let abc = [4,4,37.4,47,98,8]

Let def = abc;

Tu ye ghalat hoga qk def mein siraf abc array ka aik reference chala jaye ga matlabb k agar hm uss k baad def mein koi change karein b tu wo abc mein b hongi tu ab hum nay kiss tarah sahii say copy krna haii tu uss k liyee hm use kr sktay hain spread operator ka.

let xyz = [ 78,27,83,69,119,84];

// let xyz2 = xyz; wrong method to copy

let xyz2 = [...xyz];

tu ye ho ... hain ye spread operator hain. Function mein hm iss ko rest operator kahtay hain. Yahan prr spread operator. Tu jab jab ... laga lein orr phirr uss array ka name jiss ko hm copy krna chahtay hain tu ye sari k sari copy ho jaye gi matlab k wo uss tarah siraf reference copy nai hoga fully copy ho jaye gi. Tu agar phir hum xyz2 mein koi changes karein b tu xyz ko koi farq nai paregi..

matlab agar hum koi pop, push ya matlab jo b copy honay walay array k sth karein tu wo pahlay walay ko koi farq ni parega.