

## Iterators

```
//iterator
// array, object ai gula k loop diya iteratr korte gele onek shomoi hang hoiya jai performns er ekta issue ache ai jonno
// js looping/iterate machanism k e bar kore diyache baire. Mane object akare amder k diya diyase. So amra amader iccha
// moto iterator toiri kore nite pari & iterable o toiri korte pari

//What is Iterable & Iterator?.....

//Example 1. normal way..
let array = ["😘", "😍", "😊"];

for (i = 0; i < array.length; i++){
  console.log(array[i]);
}

// Example 1. object er jonno...
let object = {
  kiss: "😘",
  love: "😍",
  smile: "😊",
};

let entries = Object.entries(object);
//entries property: Returns an array of key/values of the enumerable properties of an object
//It's like: [["kiss", "😘"], ["love", "😍"], ["smile", "😊"]]

for(i = 0; i < entries.length; i++){
  console.log(entries[i]);
}

//for of loop
let array = ["😘", "😍", "😊"];

for(let element of array) {
  console.log(element);
  //output: 😘 😍 😊 ....line by line
}

let names = "JavaScript";
for(let element of names){
  console.log(element);
  //output: j a v a S c r i p t ....line by line
}

//kisher upor base kore decision nai iterable naki iterable na??
//number, object ai gulo iterable na
//iterable mane jake iterate ekora jai / jar moddo diya jawa jai
//ek ek ta step k bola hoi iteration like kiss, love, smile / r ja machanism er modda diya jai sheta ta iterator

let array = ["😘", "😍", "😊"];
console.log(array);
//array er prototype er modda iterator ase tai amra array k iterate korte pari

let object = {
  kiss: "😘",
  love: "😍",
  smile: "😊",
};
console.dir(object);
//object er protor modda iterator nai tai she object k iterate korte pare na
```

```
//Iterable Protocol // Iterable hote hole ki ki rule flow korte hobe?  
// 2 ta Rule:  
// 1. Must have a Symbol.iterator function property  
// 2. The function must return an Iterator
```

```
// 2. The function must return an Iterator....here is the prove  
let array = ["😞", "😏", "😁"];  
let iterator = array[Symbol.iterator](); //aita ekta function tai call kora holo ().  
console.dir(iterator); //Iterator er Prototype er bitor next name ekta method/function ache  
console.log(iterator.next()); //output: {value: "😞", done: false}  
console.log(iterator.next()); // {value: "😏", done: false}  
console.log(iterator.next()); // {value: "😁", done: false}  
console.log(iterator.next()); //Output: {value: undefined, done: true}
```

```
/**  
//Iterator er bitor next name ekta method/function ache
```

```
//JS ye for of loop builtin iterator call kore kore disse & ami chile Symbol.iterator k nije o implement korte pari karo  
n eta array er ekta property
```

```
//jokhon ami next() k call korsir first element ta pull kore niya ashse take pura array travers korte hoi nai. So ami  
jokhon next() k call korsir er por ami amr onnao code likhe onek pore giya iterator.next() kore porer element nite parbo.  
So array, list, string je tai hok & joto boro e hok ami travars korte parbo amr iccha moto koto tuko jabo koto tuko dek  
habo amr iccha moto. But for loop a amra aita korte parbo na. So aita amk ekta pause behavior dicche.
```

```
// Iterator Protocol/niom  
//Iterator hote hole Rule:  
//1. Object hote hobe  
//2. next() method implement korte hobe  
//next() must return an object with done: boolean and a value  
let array = ["😞", "😏", "😁"];  
let iterator = array[Symbol.iterator]();  
console.dir(iterator);
```

```
//Example 1.  
//change spread operator  
// Nijera ekta iterator toiri korbo...
```

```
"JavaScript"[Symbol.iterator]().next() //ekta String ter Symbol.iterator (2nd bracket diya likte hoi) tar ekta function  
likhe fellam & shai function er modda ekta next function ache liklam and call kore dilam () 1st bracket diya.  
//output: {value: "J", done: false} amra ekta object passi, variable a store kore output dekte pari or console a code  
tuko dile e hobe.
```

```
// spread operator
console.log(..."hello"); //Array sign diya likhesi [] //string iterable //Je kono iterable er element gulo k ekta ekta
kore baire bar kore niya ashe & notun array return kore spread operator. Aita spread operator er behavior.
//Output: (5) ["h", "e", "l", "l", "o"]

console.log(...124); //output: Uncaught TypeError: 124 is not iterable
//number k iterate kora jai na

//Spread operator js er builtin behavior
//Amra kokhono global behavior change korbo na but shiker jonno example..
//Nijera ekta iterator toiri korbo jeta spread operator er bihavor change kore dibe.....Finally let's do..
console.log(String); //ai boro string er modda symbol.iterator ase amra aita k over write kore dibo tahole spread operato
r er bihavor change hoiya jabe

//Symble k excess korte hole [] diya korte hoi
String.prototype[Symbol.iterator] = function() { //So akhon amder k iterator er protocol flow korte hobe
  let count = this.length;
  return { //protocol 1. object hote hobe so ekta obj return korte hobe
    next() { //protocol 2. ekta next method implement korte hobe //so ekta next() function raklam
      //next function ekta obj return korte hobe tar modda done & value thakte hobe
      //mane return korbo {done: false/true, value: 'JS'}
      //implementation akhon amr hate ja kori
      //iterate er khatre amra ek ekta single unit/piece niya chinta korsi
      //string er length porjonto iterat ekorte hobe
      //So string er length upore define kore ashi
      //akhon check korbo
      if(count >0) {
        count --;
        return {done: false, value: 'JS'};
      }
      return { done: true};
    },
  };
};

console.log(..."hello"); //Output: (5) ["JS", "JS", "JS", "JS", "JS"]
//aita global behavior change kore amra aita kokhono korbo na
```

```

//Example 2.
//our own custom iterable & iterator

// custom iterable banai
function range(start, end, step) { //range function ta spread operator er khatre use korbo
  let current = start;
  return { //aita amr iterable
    [Symbol.iterator]: function() {
      return { //aituko iterator
        next(){
          // if(current <= end) {
          //   return { //return age kore falici so ai jaigai rich korbe na
          //     done: false,
          //     value: current
          //   };
          //   current += step; //return age kore falici so ai jaigai rich korbe na //but ++ to ai jagai e
          //   korte hobe mane pore korte hoi
          // }

          //uporer code tuko
          let result; //so ekta variable niya nilam jar modda result ta rakbo
          if(current <= end) {
            result = { //genarator korle ai tuto ai vabe korte hobe na
              done: false,
              value: current
            };
            current += step;
            return result;
          }
          return {
            done: true,
          };
        },
      };
    },
  };
}

//spread operator je kono iterable er khatre e kaj kore
//[...range(1, 30, 2)] age e likhe nilam call

console.log([...range(1, 30, 2)]);

```