Arrays basics-

```ts
let names:string[]=['tom','riddle']
console.log(names) //[ 'tom', 'riddle' ]
console.log(names[0]) //tom

```

```ts
//using generics

let n1:Array<string>=['tom','riddle','peter']
let n2:Array<number>=[324234, -34324, 34234.234, -34324.3434]
let n3:Array<boolean>=[true, false]
let n4:Array<any>=['t', "lion", -23434.42234, 23324324.32434 ,32434, -32434, true, null, undefined]

console.log(n1) //[ 'tom', 'riddle', 'peter' ]
console.log(n2) //[ 324234, -34324, 34234.234, -34324.3434 ]
console.log(n3) //[ true, false ]
console.log(n4) //[ 't', 'lion', -23434.42234, 23324324.32434, 32434, -32434, true, null, undefined]
```

```ts
//multi type array

let values:(string | number)[]=['tom', 't', 256, -56, -56.78, 334.344]

let v2:Array<string | boolean>=['t', 'tom', true]

console.log(values) //[ 'tom', 't', 256, -56, -56.78, 334.344 ]
console.log(v2) //[ 't', 'tom', true ]
```

```ts
//first declare array then initialise.

let users:Array<string>
users=['tom', 't']

console.log(users) //[ 'tom', 't' ]
```

```ts
//iterate array using index based for loop.

// let e1:Array<string>=['tata', 't', null, undefined]
//Type 'null' is not assignable to type 'string'.ts(2322)
//Type 'undefined' is not assignable to type 'string'.ts(2322)


let e1:Array<string>=['tata', 't', 'john', 'cena']

for (let index = 0; index < e1.length; index++) {
    console.log(e1[index])
}

// tata
// t
// john
// cena
```

Prints index-

```typescript
//use for in loop.
//prints index not values.

let e1:Array<string>=['tata', 't', 'john', 'cena']

for (const key in e1) {
        console.log(key)
    }

// 0
// 1
// 2
// 3
```

Prints value

```typescript
//for in loop to print values.

let e1: Array<string> = ['tata', 't', 'john', 'cena']

for (const key in e1) {
    console.log(e1[key])
}

//      tata
// t
// john
// cena
```

Index plus value

```typescript
//for in loop to print index and values.

let e1: Array<string> = ['tata', 't', 'john', 'cena']

for (const key in e1) {
    console.log(key + " : : " + e1[key])
}

// 0 : : : tata
// 1 : : : t
// 2 : : : john
// 3 : : : cena
```