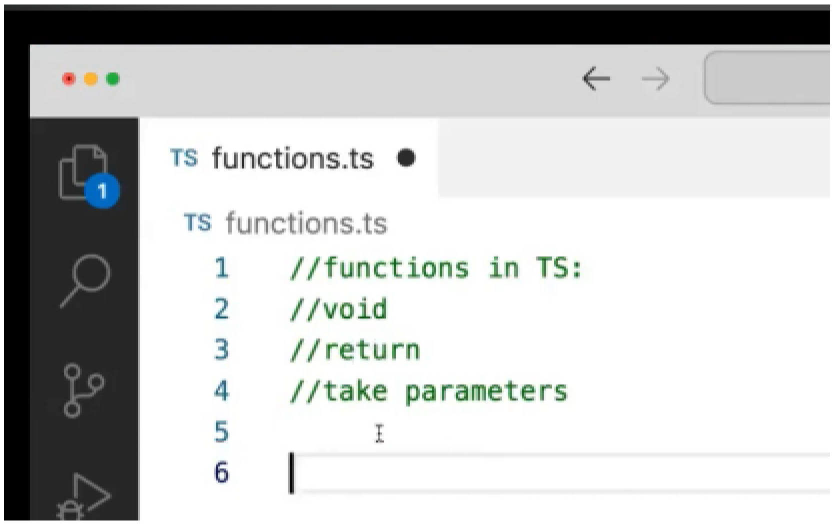
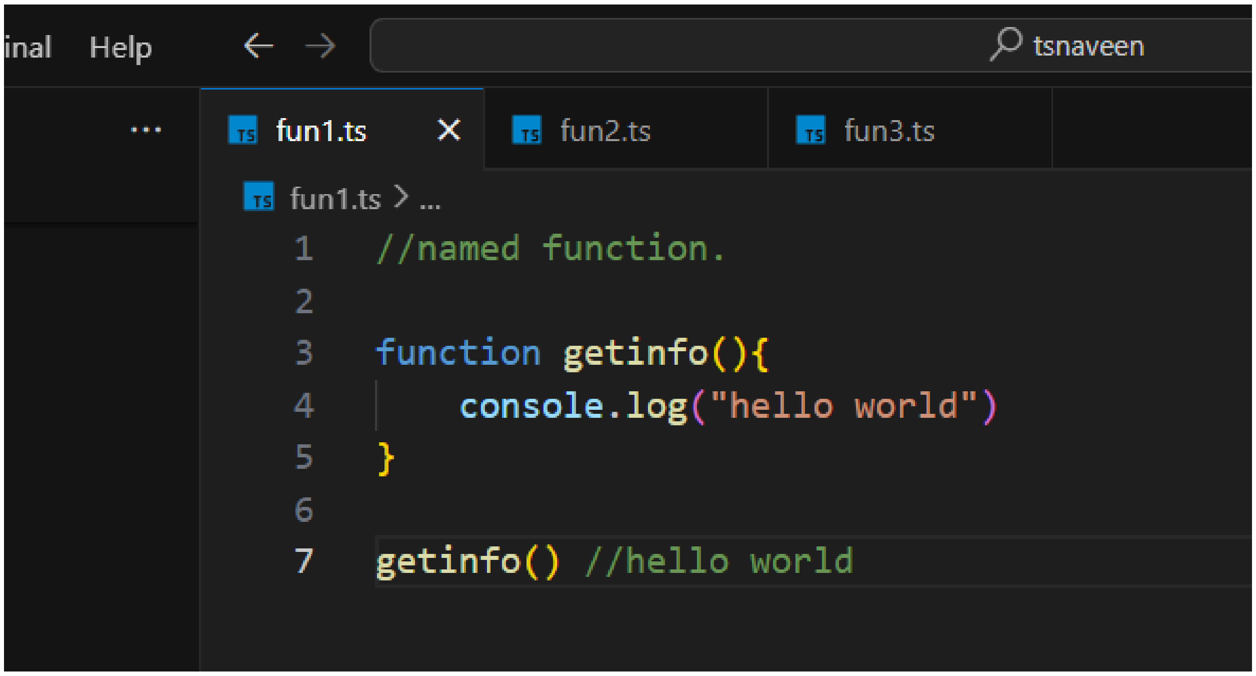


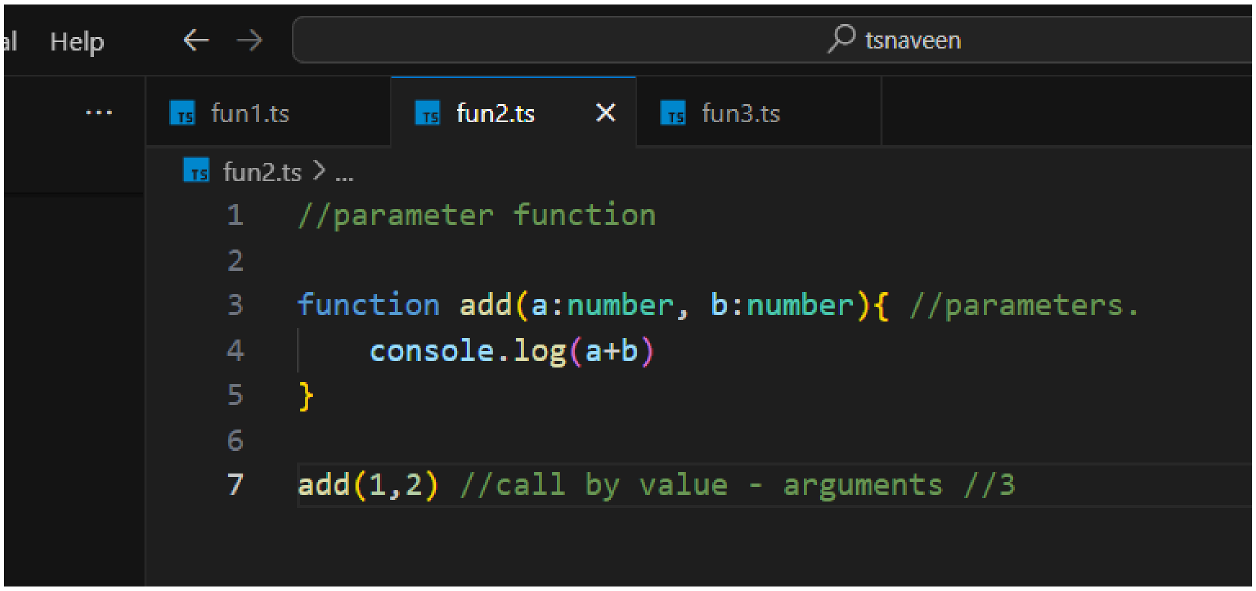
```
//functions in TS:
//void
//return
//take parameters
```

Parameterised function

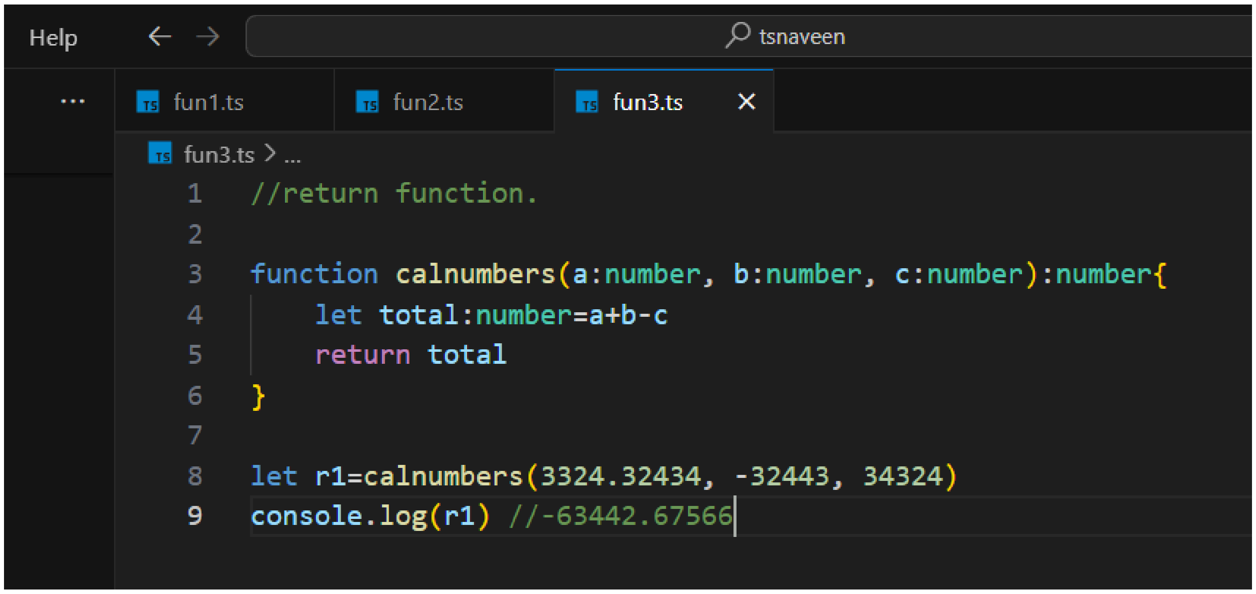


```
//named function.

function getinfo(){
    console.log("hello world")
}

getinfo() //hello world
```

Return function

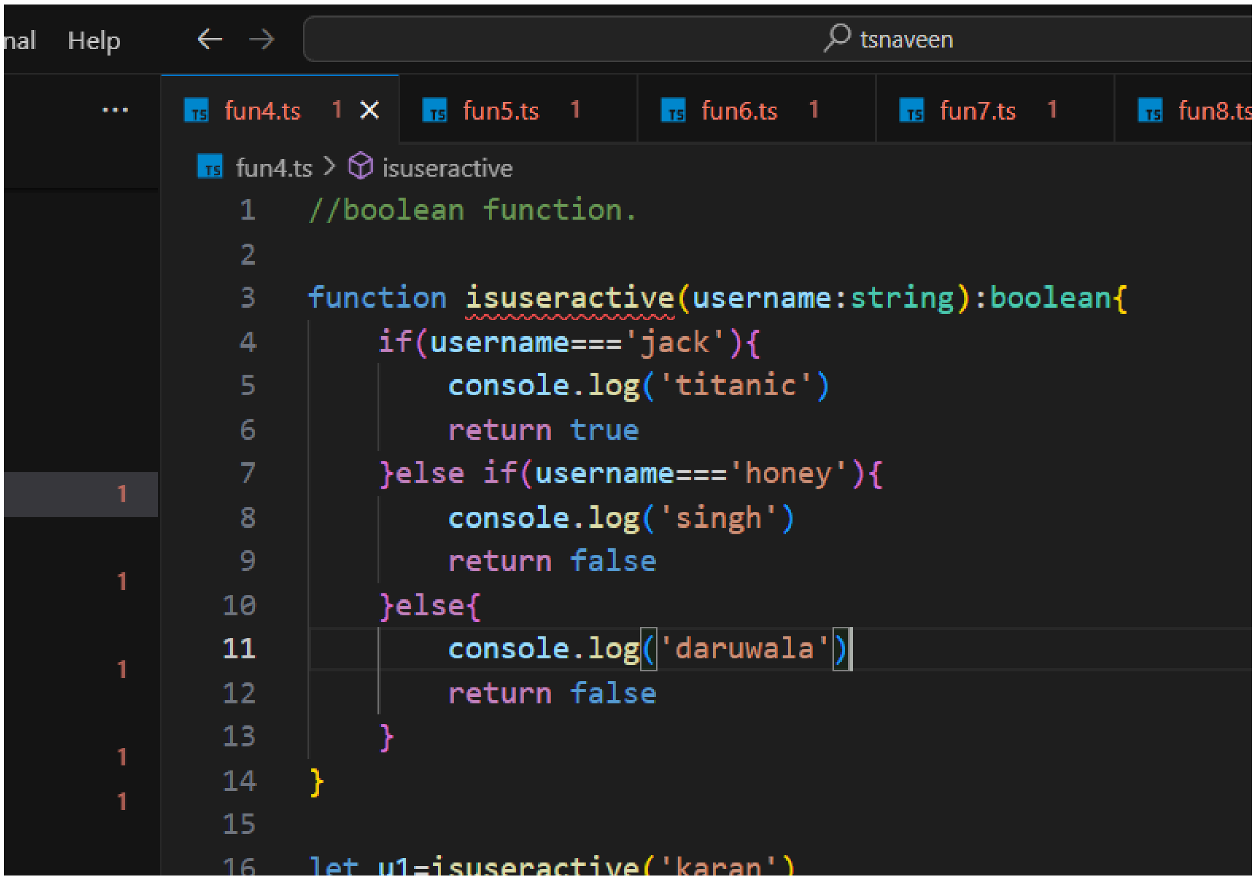


```
//parameter function

function add(a:number, b:number){ //parameters.
    console.log(a+b)
}

add(1,2) //call by value - arguments //3
```

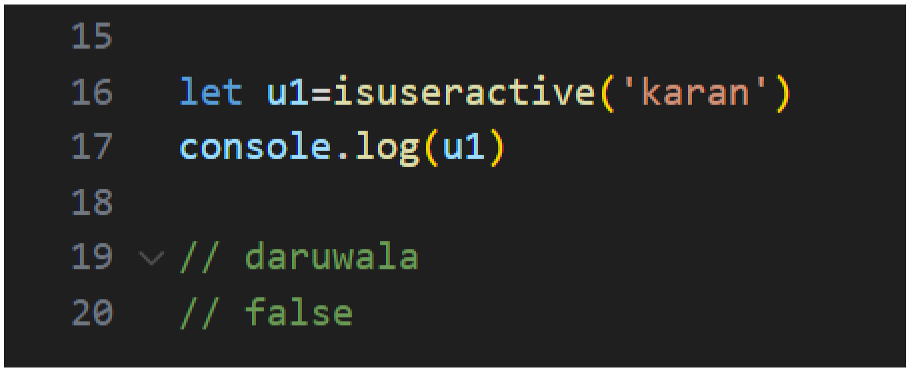


```
//return function.

function calnumbers(a:number, b:number, c:number):number{
    let total:number=a+b-c
    return total
}

let r1=calnumbers(3324.32434, -32443, 34324)
console.log(r1) //-63442.67566
```

Boolean

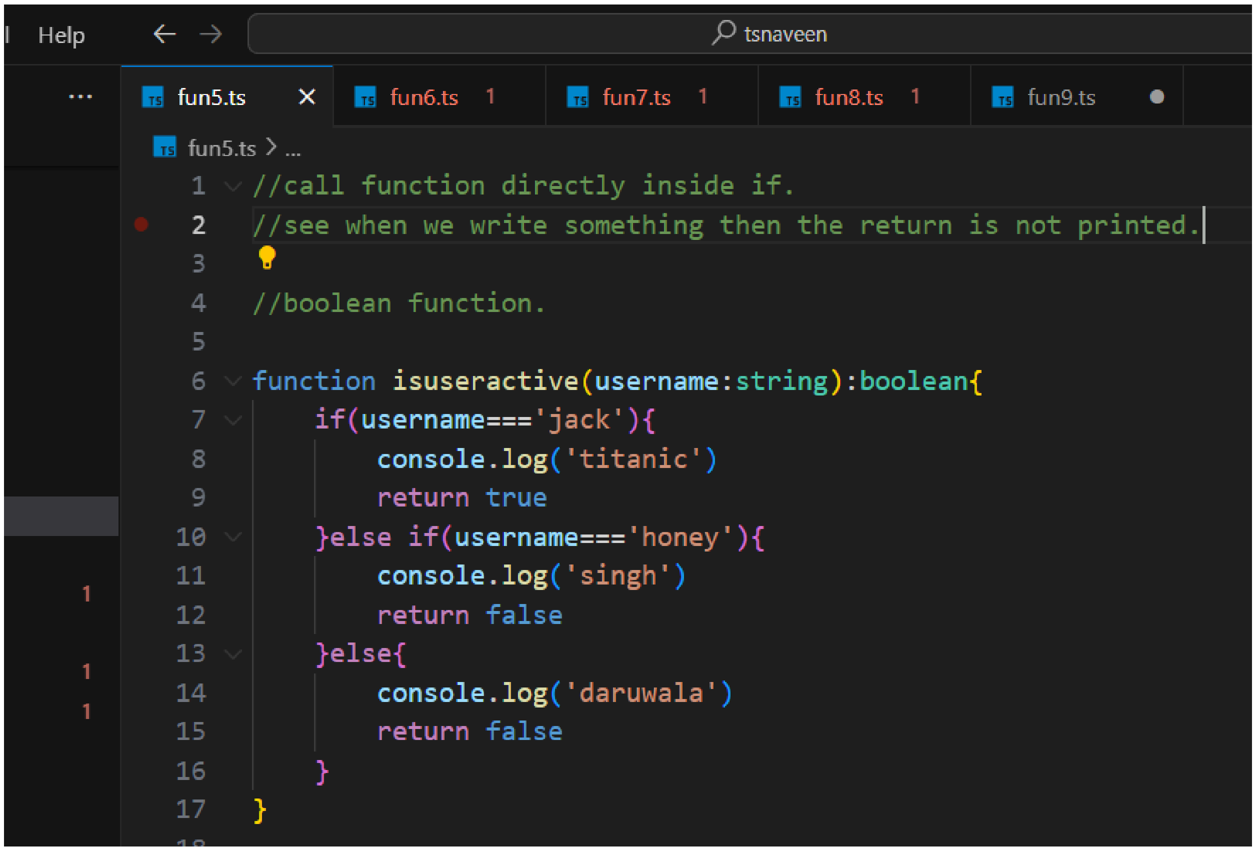


```
//boolean function.

function isuseractive(username:string):boolean{
    if(username==='jack'){
        console.log('titanic')
        return true
    }else if(username==='honey'){
        console.log('singh')
        return false
    }else{
        console.log('daruwala')
        return false
    }
}

let u1=isuseractive('karan')
```

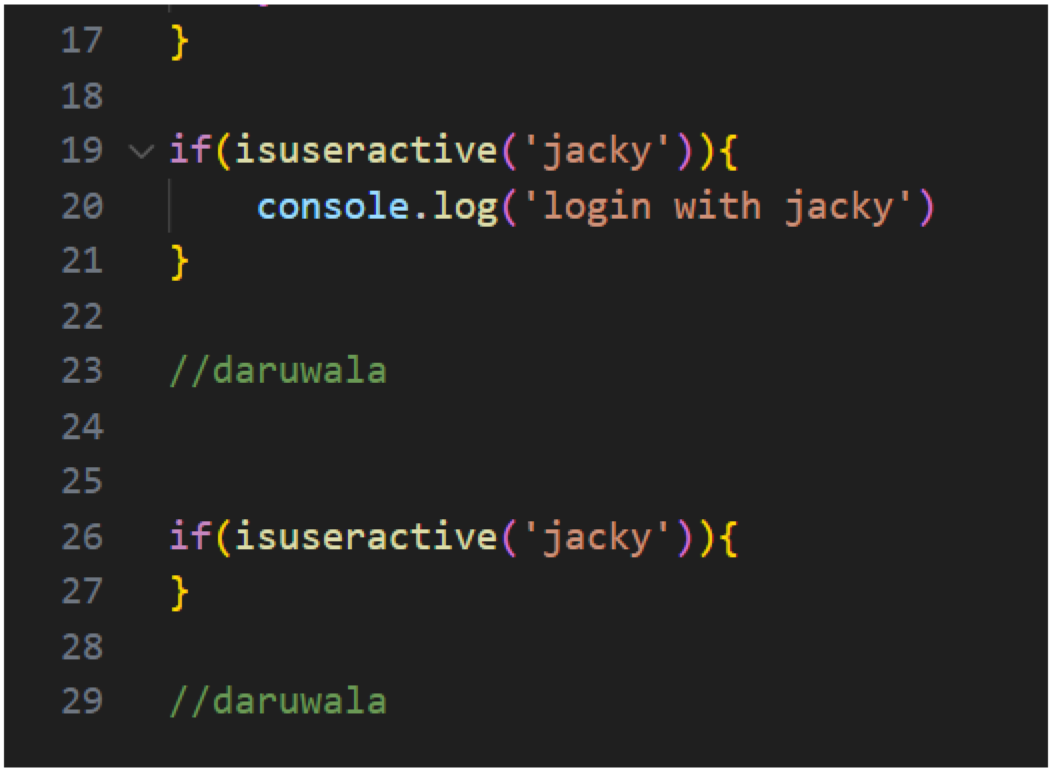


```

let u1=isuseractive('karan')
console.log(u1)

// daruwala
// false
```

Calling function directly inside if-



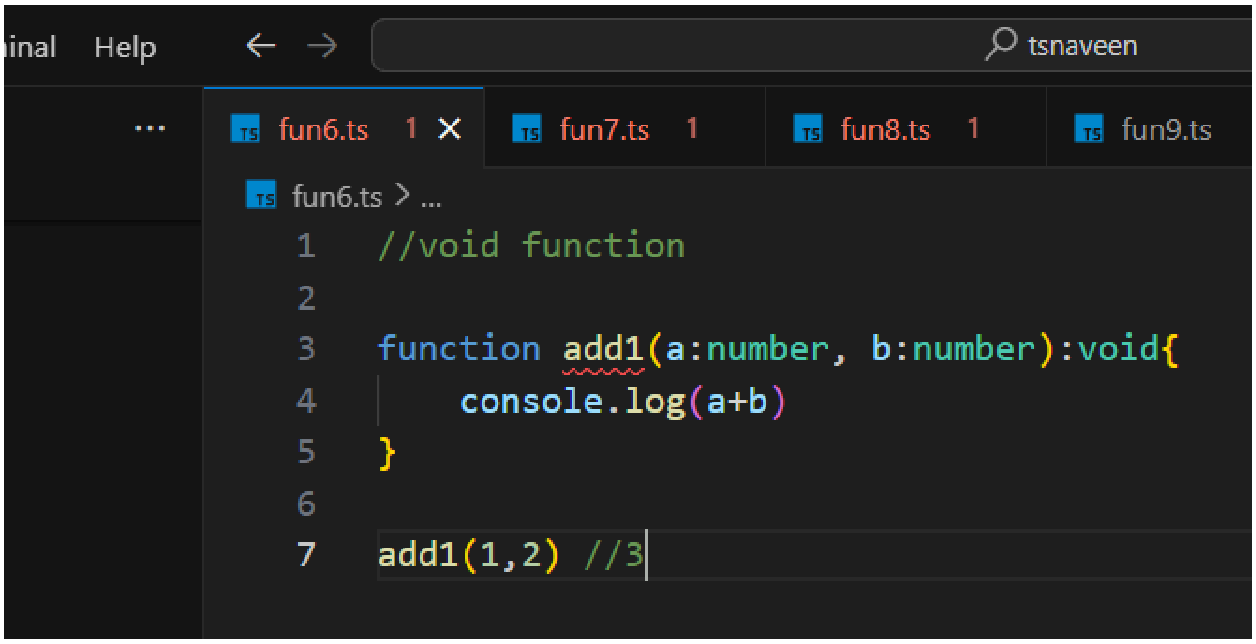
```
//call function directly inside if.
//see when we write something then the return is not printed.

//boolean function.

function isuseractive(username:string):boolean{
    if(username==='jack'){
        console.log('titanic')
        return true
    }else if(username==='honey'){
        console.log('singh')
        return false
    }else{
        console.log('daruwala')
        return false
    }
}
```



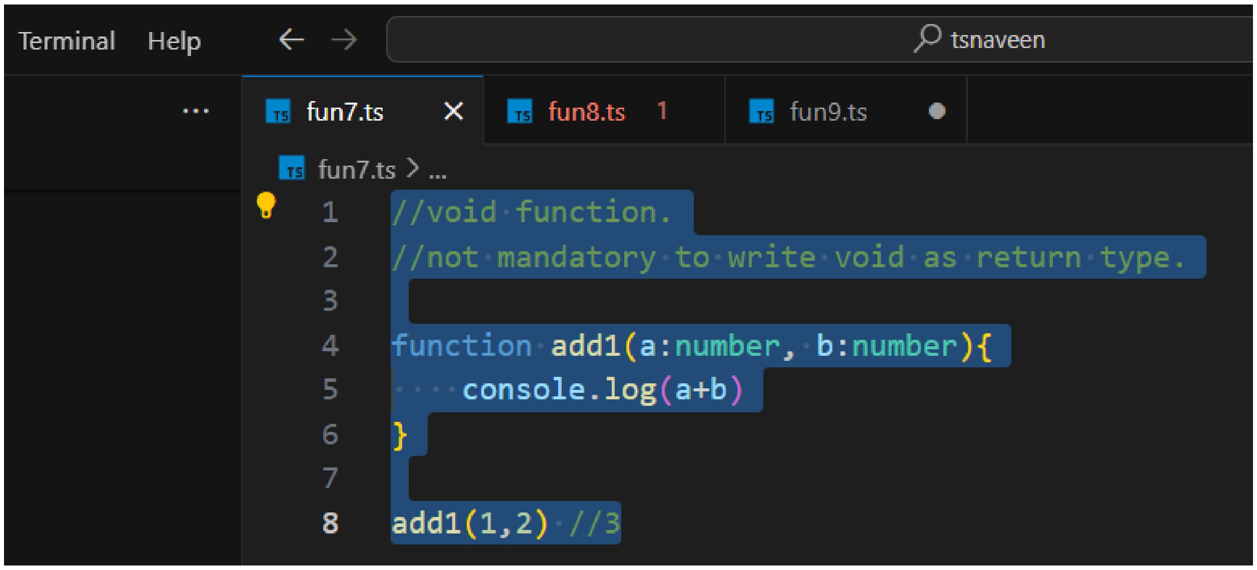
```
}

if(isuseractive('jacky')){
    console.log('login with jacky')
}

//daruwala


if(isuseractive('jacky')){
}

//daruwala
```

Void function

Not mandatory to write void.

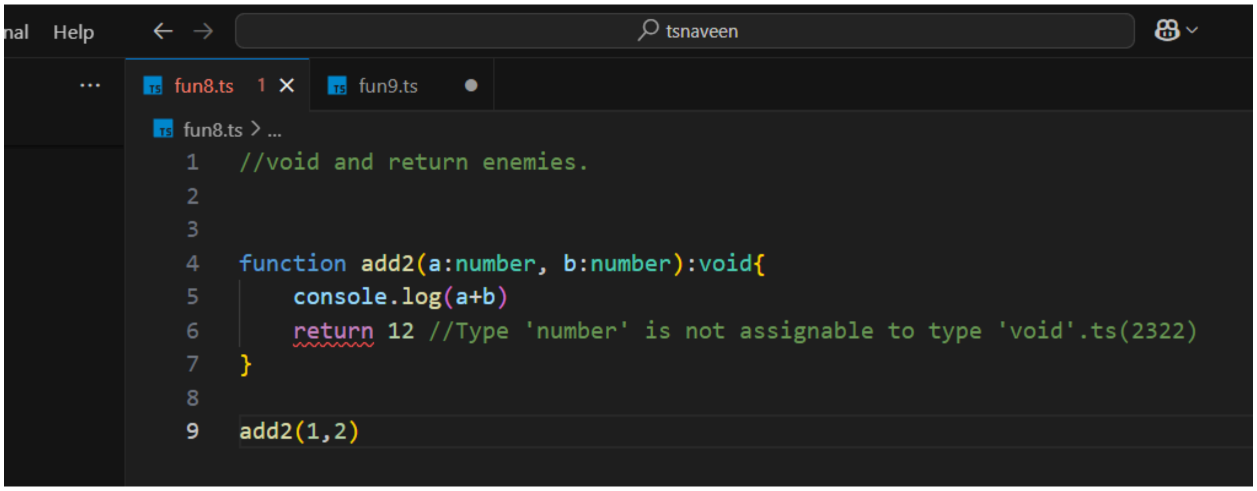


```
//void function

function add1(a:number, b:number):void{
    console.log(a+b)
}

add1(1,2) //3
```



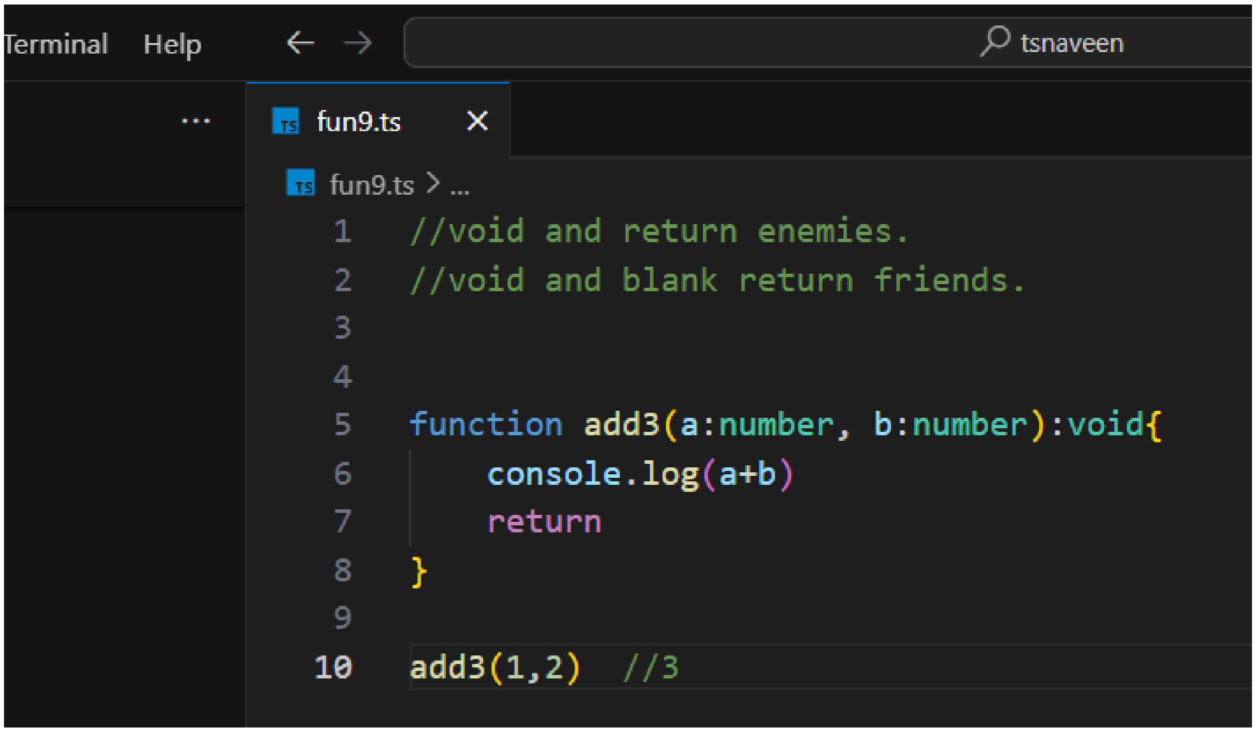
```
//void function.
//not mandatory to write void as return type.

function add1(a:number, b:number){
    console.log(a+b)
}

add1(1,2) //3
```

Void and return enemies

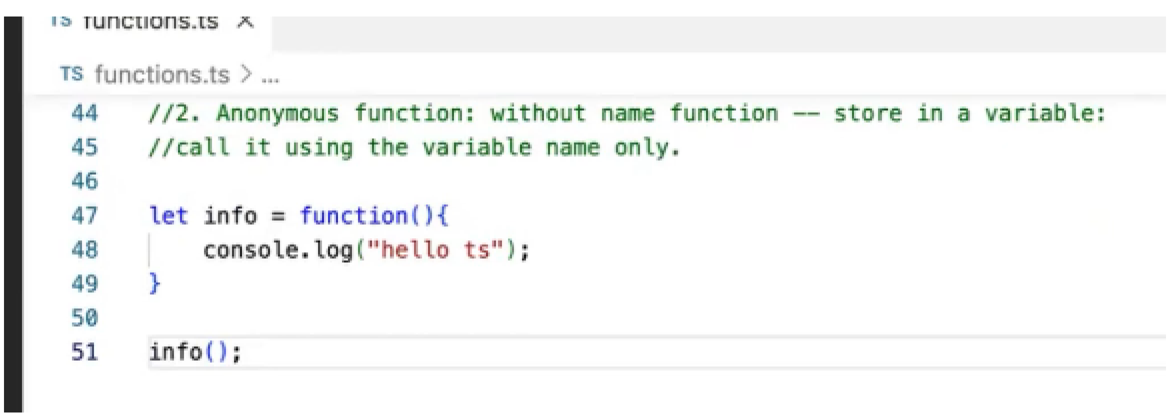


```
//void and return enemies.


function add2(a:number, b:number):void{
    console.log(a+b)
    return 12 //Type 'number' is not assignable to type 'void'.ts(2322)
}

add2(1,2)
```

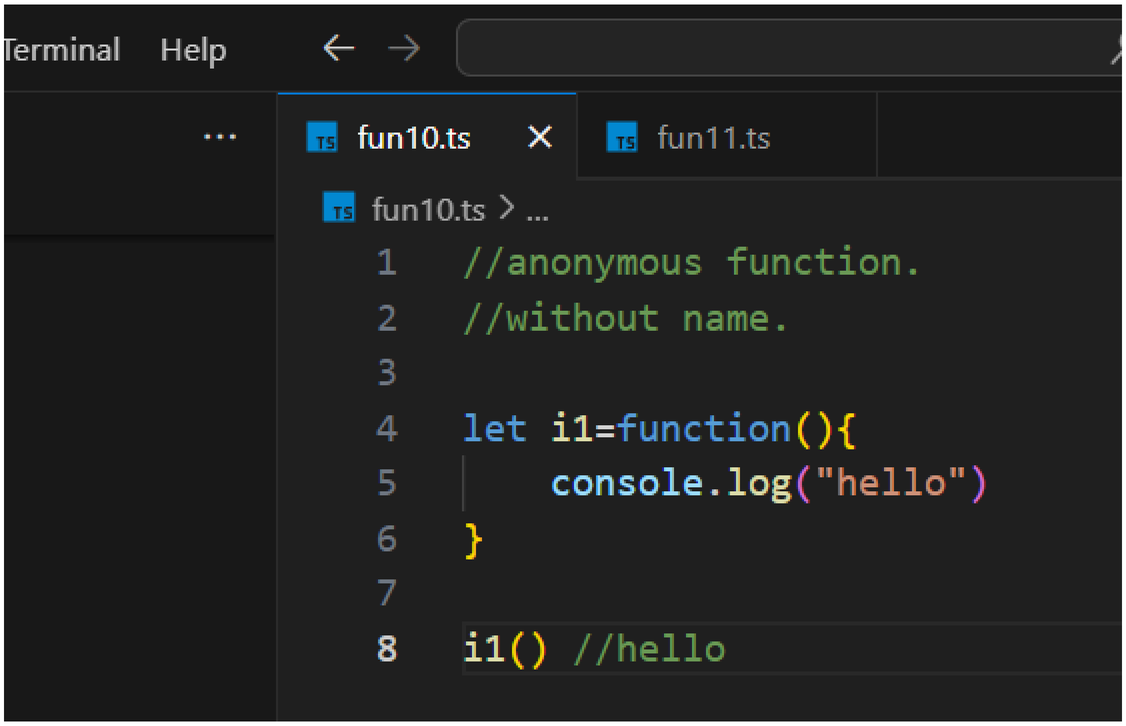
Void and blank return



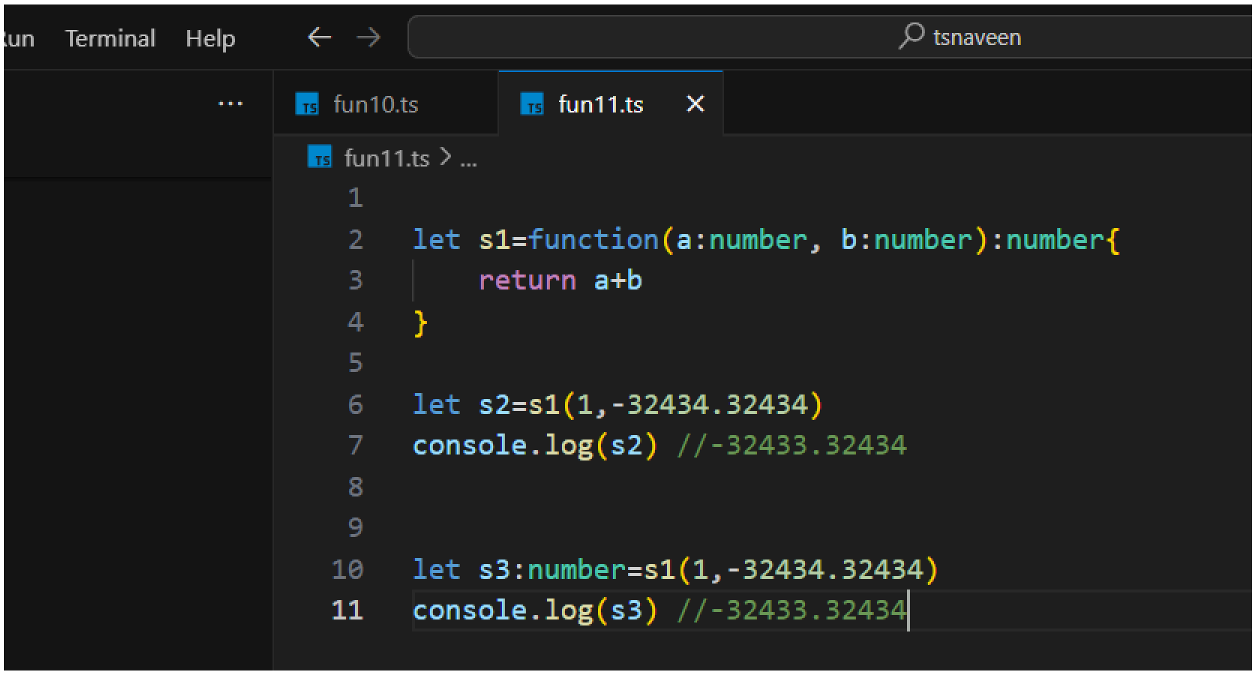
```
//void and return enemies.
//void and blank return friends.


function add3(a:number, b:number):void{
    console.log(a+b)
    return
}

add3(1,2)  //3
```



```
//2. Anonymous function: without name function -- store in a variable:
//call it using the variable name only.

let info = function(){
    console.log("hello ts");
}

info();
```



```
//anonymous function.
//without name.

let i1=function(){
    console.log("hello")
}

i1() //hello
```



```typescript

let s1=function(a:number, b:number):number{
    return a+b
}

let s2=s1(1,-32434.32434)
console.log(s2) //-32433.32434


let s3:number=s1(1,-32434.32434)
console.log(s3) //-32433.32434
```