

yield表达式:

yield表达式没有返回值

```
function * gen() {  
    let val;  
    val = yield 1;  
    console.log(val);  
}  
  
let g = gen();  
g.next()  “ ”  
g.next() undefined
```

next() 返回值:

```
console.log(g.next());  
▼ {value: undefined, done: false} ⓘ  
  done: false  
  value: undefined  
  __proto__: Object
```

Generator函数内星号 \* 表示嵌套Generator

next() 的返回值

done: 是否执行完成

value: yield后的表达式的值（没有星号 \* ），yield后的可遍历对象的第几项的值（有星号 \* ）

next() 传参:

next(10) 参数控制yield的返回值

```
function * gen() {  
    let val = 1;  
    console.log(val); 1  
    val = yield [1, 2, 4, 6];  
    console.log(val); 20  
}  
  
let g = gen();  
console.log(g.next(10));  
console.log(g.next(20));
```

提前结束：

1、`g.return(100)` 返回值为参数100

2、

```
console.log(g.next())  
console.log(g.next())  
console.log(g.next())  
console.log(g.next())  
g.throw(new Error('ss'))  
console.log(g.next())
```