数组实例的find方法，用于找出第一个符合条件的数组成员。它的参数是一个回调函数，所有数组成员依次执行该回调函数，直到找出第一个返回值为true的成员，然后返回该成员。如果没有符合条件的成员，则返回undefined。

数组实例的findIndex方法的用法与find方法非常类似，返回第一个符合条件的数组成员的位置，如果所有成员都不符合条件，则返回-1。

[1, 5, 10, 15].find(function(value, index, arr) {

return value > 9;

})

[1, 5, 10, 15].findIndex(function(value, index, arr) {

return value > 9;

}) // 2

需要注意的是，在find和findIndex中，回调函数的参数只是用于条件判断或其它表达式，不能作为返回值。对于find只返回值value，findIndex只返回索引index，哪怕强制用return arr都不可能改变返回的值，实际上需要的是return返回的bollean值。

[1, 5, 10, 15].findIndex(function(value, index, arr) {

if(value > 9){

return value;

}

}) //这里不会返回10，返回的任然是2

find和findIndex接收第二个参数，用来绑定判断时的this。

var person= {name: 'John', age: 10};

[10, 12, 26, 15].find(function f(v,i){

// 这里的return值没有实际意义

if(v > this.age && i>2){

return true //改变这里的Bollean值会改变find的结果 为true时等于15，为false时等于 undefinded.

}else{

return false;

};

}, person);