

es5:

```
function fn() {}  
let fn = function() {}
```

es6:

`() //函数参数=>{} //函数体`

`let fn = ()=>{} // 没有参数时括号不能省略，有且只有一个参数时，括号可以省略`

```
let fn = name=>{  
  console.log(name)  
}
```

`let sum = (x=1,y=2,z=3) => x+y+z // 函数体中是表达式时可以省略括号，且返回值是表达式的值`

```
console.log(sum())
```

`let sum = (x=1,y=2,z=3) => ({x:x,y:y,z:z}) // 函数体需要返回对象时，在大括号外加小括号即可`

```
console.log(sum())
```

记不住时最好都不要省略花括号

```
let test = {  
  name:'test',  
  say:function() {  
    console.log(this.name)  
  }  
}  
test.say()
```

es5:this 执行时指向的位置（调用这个函数的对象）

es6:this 写代码的位置指向的位置

箭头函数是在函数定义的时候this所在的对象，

普通函数是谁调用 this 指向谁