ES6函数允许传入默认值，函数参数是一单独作用域，函数内部也是一单独作用域，但是在参数和函数内部之间浏览器厂商处理的有问题。

var x = 1;

function foo(x, y = function() { x = 2; }) {

var x = 3;

y();

console.log(x);

}

Chrome64下

foo() // 3

x // 1

firefox48下

foo() // 2

x // 1

主要的差异在与y = function() { x = 2; }执行后，x改变的是参数作用域还是函数内部作用域。chrome是放弃了兼容ES5，但是firefox做了兼容ES5处理。

function(x,y){

var x = x;

var y = y;

}

ES6下参数里的变量不能在函数内部重新用let和const定义，但是var没问题。