[复制代码](javascript:void(0);)

function writeCode(callback) {

// do something...

callback();

// ...

}

function introduceBugs() {

// ... make bugs

}

writeCode(introduceBugs);

[复制代码](javascript:void(0);)

**A Callback Example**

[复制代码](javascript:void(0);)

// refactored findNodes() to accept a callback

var findNodes = function (callback) {

var i = 100000,

nodes = [],

found;

// check if callback is callable

if (typeof callback !== "function") {

callback = false;

}

while (i) {

i -= 1;

// complex logic here...

// now callback:

if (callback) {

callback(found);

}

nodes.push(found);

}

return nodes;

};

[复制代码](javascript:void(0);)

**Callbacks and Scope**

[复制代码](javascript:void(0);)

var myapp = {};

myapp.color = "green";

myapp.paint = function (node) {

node.style.color = **this**.color;

};

[复制代码](javascript:void(0);)

The function findNodes() does something like this:

[复制代码](javascript:void(0);)

var findNodes = function (callback) {

// ...

if (typeof callback === "function") {

callback(found);

}

// ...

};

[复制代码](javascript:void(0);)

If you call findNodes**(myapp.paint), it won’t work as expected**, because this.color will not be defined. **The object this will refer to the global object, because findNodes() is a global function**. If  findNodes() were a  method  of  an  object  called  dom (like dom.findNodes()), then this inside of the callback would refer to dom instead of the expected myapp.

**pass the callback function and in addition pass the object this callback belongs to**

[复制代码](javascript:void(0);)

findNodes(myapp.paint, myapp);

var findNodes = function (callback, callback\_obj) {

//...

if (typeof callback === "function") {

callback.call(callback\_obj, found);

}

// ...

};

[复制代码](javascript:void(0);)

**pass the method as a string**

[复制代码](javascript:void(0);)

findNodes("paint", myapp);

var findNodes = function (callback, callback\_obj) {

if (typeof callback === "string") {

callback = callback\_obj[callback];

}

//...

if (typeof callback === "function") {

callback.call(callback\_obj, found);

}

// ...

};

[复制代码](javascript:void(0);)

**Asynchronous Event Listeners**

document.addEventListener("click", console.log, false);

**Timeouts**

[复制代码](javascript:void(0);)

var thePlotThickens = function () {

console.log('500ms later...');

};

setTimeout(thePlotThickens, 500);

[复制代码](javascript:void(0);)