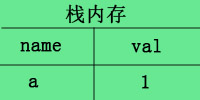
**1.基础数据类型：string number boolean null undefined symbol**

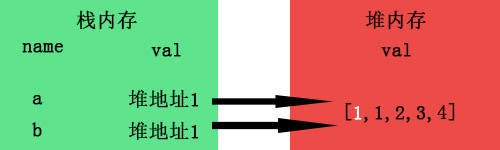
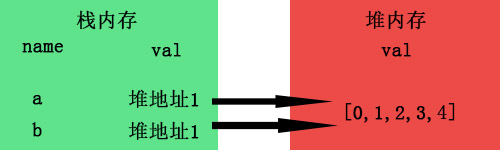
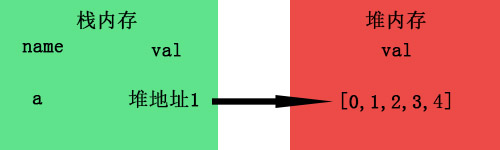
**引用（复杂）数据类型：array object 函数**

**2.深拷贝只针对复杂数据类型：**

**a.基本类型--名值存储在栈内存中,浅拷贝的两个变量各自的值存放在自己的内存，所以修改值，相应的内存数据改变**

**b.引用类型—名存在占内存中，值存放在堆内存，并且栈内存会存放一个引用的地址指向堆内存的数据**

****

**所以，改变堆内存中的数据，由于变量都指向这个堆内存地址，所以两个变量的数据都随着改变；所以实现深拷贝的原理是各自有自己的堆内存存放数据，改变时互不影响**

**3.实现深拷贝的方法：**

**a) 递归**

// 递归实现深拷贝

const deepCopy = (obj) => {

    let deepObj = Array.isArray(obj) ? [] : {}

    for (let key in obj) {

        if (obj[key] && typeof (obj[key]) === 'object') {

            deepObj[key] = deepCopy(obj[key])

        } else {

            deepObj[key] = obj[key]

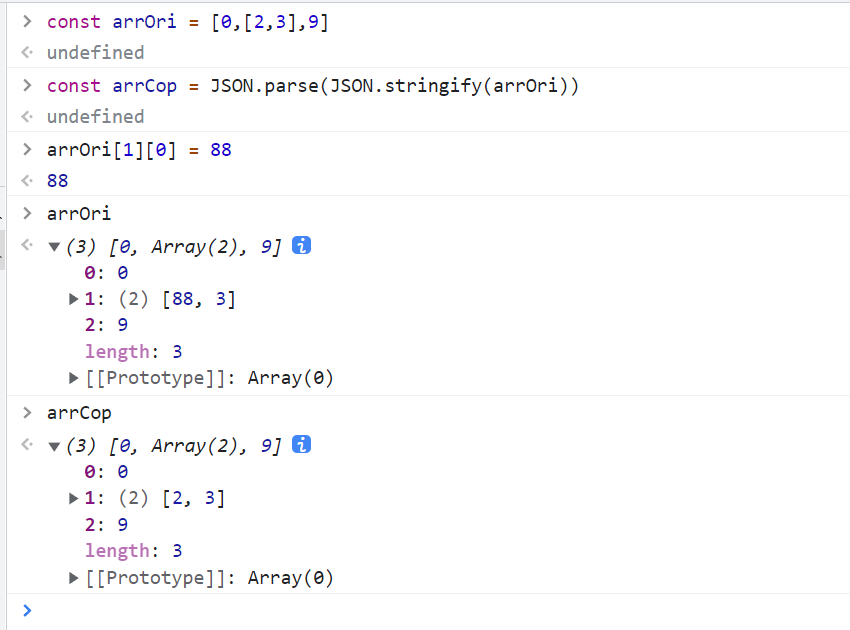
        }

    }

    return deepObj

}

**b) JSON. parse (JSON. stringify (obj))**

****