

Web 前端第二次作业

2013013310 徐京傲 xujingao13@gmail.com

一. 基础练习 1

`var foo = function() {}` 定义的 `foo` 为一局部变量，只有在被调用的时候才会被初始化。只有在定义之后才可以使用它。

`function foo() {}` 定义的 `foo` 为一全局变量，编译后的函数声明和他的赋值都会被提前。函数声明过程在整个程序执行之前的预处理就完成了，所以只要处于同一个作用域，就可以访问到，在定义之前也可以调用它。

二. 基础练习 2

函数的返回值为 “function”

函数返回 `f()` 的类型，在传递的 `function()` 作用域内有整数型的 `bar`、函数型的 `bar` 和字符型的 `bar`。但由于返回在之前，所以只有整数型和函数型定义方式在编译后会被提前，又由于覆盖作用，函数型后定义，所以最终 `bar` 的类型为函数型。整个函数返回 `function`。

三. 基础练习 3

函数的返回值为 “undefined”，因为在返回的时候调用了匿名函数，此时匿名函数内部的 `this` 指针指向的是 `windows`，`windows` 并没有 `baz` 变量，所以返回的是 `undefined`。

若要输出为 “number” 则需要明确作用域，可以将 `foo.bar()` 中的返回值改为 `foo.baz` 或者将下一个函数的返回值改为 “`return typeof foo.bar()`”。

```
var foo = {  
  bar: function() {return this.baz; },  
  baz: 2  
};  
(function(){  
  return typeof foo.bar();  
})(foo.bar);
```

```
var foo = {  
  bar: function() {return foo.baz; },  
  baz: 2  
};  
(function(){  
  return typeof arguments[0]();  
})(foo.bar);
```

四. 基础练习 4

`a[1]` 为 “undefined”，`a[2]` 为 “5”。执行 “`a=b`” 之后 `a` 和 `b` 指向相同的区域，这块区域内只有 “`b[2]`” 有定义为 5。

五. 基础练习 5

`a` 内的顺序为：-1, -2, -3, 1, 1048575, 2, 3, “test”。

将 `sort` 函数的判断大小规则进行修改如下就可以实现数组中的元素完全按照 ASCII 由小到大排序：

```
var a = (new Array("test",1,2,3,-1,-2,-3,0xFFFF));  
a.sort(function(a,b){ //自定义函数排序  
  var a1 = parseInt(a);  
  var b1 = parseInt(b);  
  if(a1 < b1){  
    return -1;  
  }else if(a1 > b1){  
    return 1;  
  }  
  return 0;  
});  
console.log(a);
```

六. 进阶练习 1

循环读入每一个数据，如果这个数为数值或者为可以转化为数值的字符串，则将该变量转化为对应的数值相加即可。

```
function sum(){
    var length = arguments.length;
    var sum = 0;

    for(var i = 0; i < length; i++){
        //判断是否为整数
        if(typeof arguments[i] === "number")
            sum = Add(sum, arguments[i]);
        //判断是否为可以转化为整数的字符串
        if(typeof arguments[i] === "string"){
            if(!isNaN(arguments[i])){
                var temp = parseFloat(arguments[i]);
                sum = Add(sum, temp);
            }
        }
    }
    return sum;
}

//加法函数 解决浮点数运算问题
function Add(arg1, arg2) {
    var r1, r2, m;
    try {
        r1 = arg1.toString().split(".")[1].length;
    } catch (e) {
        r1 = 0;
    }
    try {
        r2 = arg2.toString().split(".")[1].length;
    } catch (e) {
        r2 = 0;
    }
    m = Math.pow(10, Math.max(r1, r2));
    return (arg1 * m + arg2 * m) / m;
}
```

七. 进阶练习 2

关键字排序，修改 sort 函数的比较大小规则即可

```
function student_sort(stu_arr){
    var sorted_arr;
    sorted_arr = stu_arr;
    sorted_arr.sort(function(a,b){
        if(a.age < b.age)
            return -1;
        else if(a.age > b.age)
            return 1;
        else{
            if(a.score > b.score)
                return -1;
            else if(a.score < b.score)
                return 1;
            else{
                var temp1 = a.name.substr(0, 1);
                var temp2 = b.name.substr(0, 1);
                if(temp1 < temp2)
                    return -1;
                else return 1;
            }
        }
    })
    return sorted_arr;
}
```

八. Bonus

主要是用 js 语言进行对象的构造;对于 search 方法可以用上课讲到的“in”

方法，可以快速判断“age”是不是在要搜索的对象当中。

九. Github 链接

<https://github.com/xujingao13/xujingao13.github.io/tree/master/hw>

2