学习方法

1. 建立逻辑思维
   1. 多分析
   2. 理解逻辑算法
   3. 多练、多敲、多读
2. 编程习惯
3. 大胆动手
4. 保持好奇好
5. 乐观面对错误

就业能力

1. 分析能力
2. 独立思考能力
3. 程序阅读能力
4. 程序排错能力
5. 独立编写程序能力
6. 团队协作能力
7. 学习能力
8. 善于总结的能力

专业技能

ES5\String

1. 严格模式
   1. 概念：所谓严格模式，从字面上就很好理解，即更严格的
   2. Demo:  
      function m1(){  
       max = 100;  
      }  
      m1();  
      alert(max); //由于max没有var 声明，因此max会变成全局变量  
      但是在严格模式下：  
      function m1(){  
       “use strict”;  
       max = 100;  
      }  
      m1();  
      alert(max);  
      报错：Uncaught ReferenceError:max is not defined  
      注：不要轻易在全局范围开头增加”use strict”声明  
       建议在函数作用域内部使用
2. ES5新增数组操作方法（indexOf/forEach/map/reduce/filter）
   1. indexOf:查找指定元素在数组中首次出现的下标位置,如果没有查找到该元素，则返回-1
      1. indexOf(查找的元素，从哪里开始查找)
      2. demo:  
         [2,5,7,3,5].indexOf(5) //返回1
   2. forEach:遍历数组
      1. demo:  
         [“a”,”b”,”c”,”d”,”e”].forEach(function(item,index,array){  
          //item表示当前遍历的元素，index表示下标，array表示数组本身  
          console.log(item);  
         });  
         其实也可以更简单：  
         [“a”,”b”,”c”,”d”,”e”].forEach(alert);  
         注意：  
         [“a”,”b”,”c”,”d”,”e”].forEach(console.log);  
         //会出现Uncaught TypeError:Illegal invocation(…)  
         因为执行上下文非法导致的，我们需要指定一下上下文：  
         [“a”,”b”,”c”,”d”,”e”].forEach(console.log,console); //正确执行
   3. map映射，遍历-----》操作------》返回
      1. demo:  
         var newArr = [1,2,3,4,5].map(function(item,index,array){  
          return item + 2;  
         })  
         console.log(newArr);  
         结果：[3,4,5,6,7]
   4. reduce归并
      1. demo:  
         [1,2,3,4].reduce(function(pre,next,index,array){  
          //参数中的pre和next代表了之前 的处理结果和下一个元素  
          return pre+next; //return的结果会赋值给下一个函数的pre的参数  
         });  
         结果：10
   5. filter过滤
      1. demo:  
         var res = [88,69,92,77,90,96,98,99].filter(function(item,index,array){  
          return item > 80; //找出所有大于80的元素  
         });  
         结果：[88,92,90,96,98,99]
3. 字符串的两种创建方式（常量和构造函数）
   1. 常量（字面量）创建方式
      1. Var 变量名 = 字符串常量；
   2. 构造函数创建方式
      1. Var 变量名 = new String(“字符串常量”);
4. 字符串常见API（charAt\indexOf\substring\slice\split\replace）
   1. Length:字符串长度 下标从0开始
   2. charAt(下标):根据下标获取指定的字符串，返回指定位置的字符串。
   3. IndexOf(字符，start):返回指定字符串在字符串中首次出现的位置。
   4. 扩展：lastIndexOf():返回一个字符串最后出现的位置。
   5. Substring(start,end):截取指定位置的字符串（从开始位置起，到结束位置，但不包含结束位置的字符串。
   6. slice(start,end):截取指定位置的字符串（从开始位置起，到结束位置，但不包含结束位置的字符串。
   7. 扩展：substr(start,length):从指定的位置开始截取指定长度的字符串。
   8. Split(“分割位置（指定分割符）”,[指定的长度]):将一个字符串分割成数组。
   9. Replace():将字符串中的一些字符替换为另外一些字符串。返回一个替换后的新字符串，不会改变原字符串。
   10. 扩展：toLowerCase():用于把字符串转换为小写。
   11. 扩展：toUpperCase():用于将字符串转换为大写。

第二节：应用

1. 敏感词过滤
2. 密码格式要求
3. 留言过滤



第三节

1. ASCII码和字符串
   1. 4E00(19968)
   2. 9FA5(40869)
   3. 20902(回文数)
2. 字符串常见API（charCodeAt\fromCharCode）
   1. charCodeAt():根据下标获取指定字符的编码。
   2. String.fromCharCode():根据编码返回字符。

第四节：应用

1. 统计字符串中每个字符的个数？

第五节：综合应用

1. 数字字母混合验证码  
   
2. 统计每个字符出现的次数，结果显示a2、b1、c2、d1，去掉重复的字符，使结果显示abcdfgj