# JavaScript

微信小程序开发使用语言：JavaScript(前端开发)

一、什么是JavaScript

Java是高级编程语言

JavaScript是脚本语言（Script就是脚本的意思）

Java需要先编译再执行

JavaScript是浏览器解释运行

Java是强类型语言（变量与类型绑定）

JavaScript是弱类型语言（不绑定）

Java是语法敏感语言（有一点没写到就报错）

JavaScript语法不敏感（大差不差就可以）

Java运行在服务器上（运行在服务端）

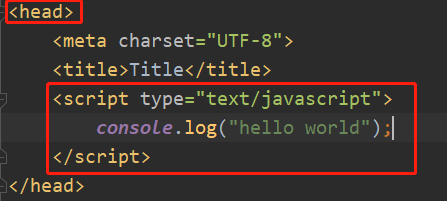
JavaScript运行在浏览器上（运行在客户端）

二、JS入门

尝试在浏览器中输出helloworld

JS可以写在行内、内部、外部

内部



JS代码作为脚本语言需要写在head标签中的script标签中

要记得在type选择语言类型”text/javascript”代码

console.log(“hello world”);

console就是控制台，这里是浏览器的控制台

log 表示以日志形式输出

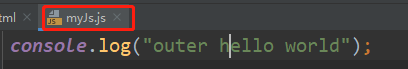
(“”) 中是具体输出的内容



外部

JS代码也能独立生成文件，后缀 .js

先生成独立的js文件



然后在页面上引用js文件



tips：script标签是一组双标签，如果不引入文件，js代码是写在双标签之间的

如果一个script标签已经用于引入外部文件了，请绝对不要在这对标签之间写任何内容（空格、回车都不要）

行内

在标签中写JS一般都是用于绑定一个事件，例如写一段JS代码在用户触发某一事件的时候执行这一段JS代码。



tips：最后一种，行内的JS，一般不会这么写，JS代码也是有一定的代码量的，几乎不可能一行就写好了。在后续的编程中一般会将JS生成一个函数在这里调用。

有的浏览器也不支持写行内的JS

变量

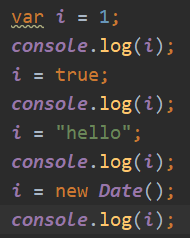
如何声明一个变量

在Java中 int i = 1; 这里的int表示i这个变量数据类型就是int类型，这就叫强类型语言，变量和类型是绑定的，一个变量在使用的周期中只有一个类型。

在JavaScript中使用关键字var声明变量

var i = 1;

var是variable缩写，表示的就是变量，也就是说这条语句声明了一个变量i，这个变量在声明过程中没有和任何类型进行绑定，在该变量中可以保存任何类型的数据



这就是弱类型，变量和数据类型不绑定，可以保存任何类型的数据

应该如何判断变量中实际存储是什么类型？

数据类型

JS是弱类型语言，不是无类型语言，它也有常用的数据类型

字符串string

数字number

布尔型boolean

对象object

数组array

日期date

空值 null

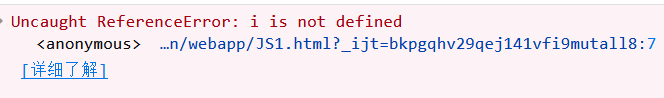
有一个获取类型的关键字 typeof(变量名) 会返回该变量中存储的值的类型

这里给大家介绍一个新的关键字

当声明了变量i再访问这个i的时候可以访问到变量中的值

这里有两个特殊情况

1、i未声明直接调用



会报错，提示 i 未定义

2、i声明未赋值



输出undefined关键字，这个词表示变量未初始化（首次赋值）

运算符

和Java中基本一致，有一点小的区别

1 / 0 = ?

java中会报错，除数不能为0

JS中会输出infinity正无穷，因为JS中number相当于是Java中整型 + 浮点型，这里就自行拓展到浮点型类型上输出了和Java中1.0 / 0一样的结果，正无穷。

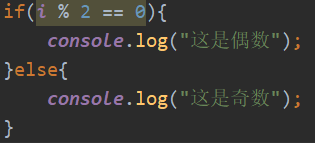
JS中的 + 可以实现字符串拼接

流程控制语句

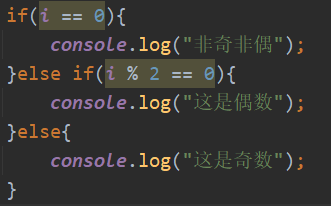
JS中的分支语句

简单if

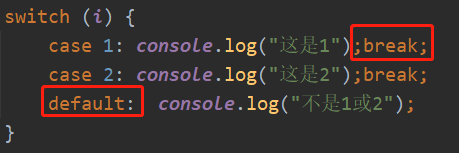
if…else



if…else if



switch



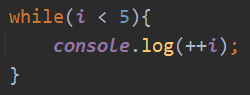
tips：和Java中类似，每个case表示一个分支，如果在分支代码中没有出现break关键字就会出现穿层现象（会执行下一个case直到运行结束或遇到break为止）

如果每一个case都没有匹配上，会进入default代码块表示默认分支。

JS中的循环语句

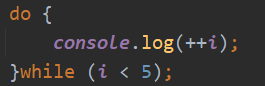
while(){

}



do{

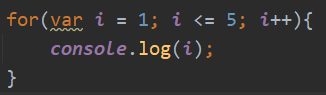
}while()



for(){

}

普通for循环

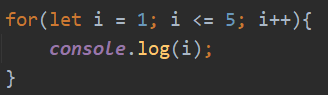


在JS代码中声明变量有两个关键字 var let

var 声明的是全局变量

let 声明的是局部变量

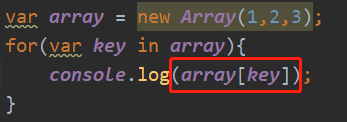
在for循环中，应该趋向于使用局部变量（仅在当前循环体中有效）



增强for循环

tips：在JS中对象的结构接近于Java中的map，属性名接近于key，对应的值接近于value。对象的增强for循环得到的是每一个key，再使用这个key获取每一个value。

在JS的增强for循环中，不能直接获取数组中的每一位，而是获取所有的下标



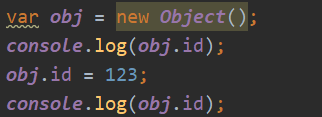
对象

Object

JS中的对象和Java中的有很大区别

在Java中由于强类型的关系，一个对象中的属性是固定的，类型中有几个属性对象中就只能有这些属性

在JS中属性可以临时增加或删除



obj这个对象在声明的时候是一个没有任何属性的空的对象，获取id的值返回的是undefined（未定义）。

在执行obj.id = 123之后，obj对象中就新增了一个属性id，并同步赋值123。

当再次访问obj.id的时候返回的就是刚刚赋值的123了。

在JS中对象的属性是很灵活的，可以随意增加。

删除属性需要使用delete关键字。

delete 对象名.属性名;

表示从这个对象中，移除指定的属性



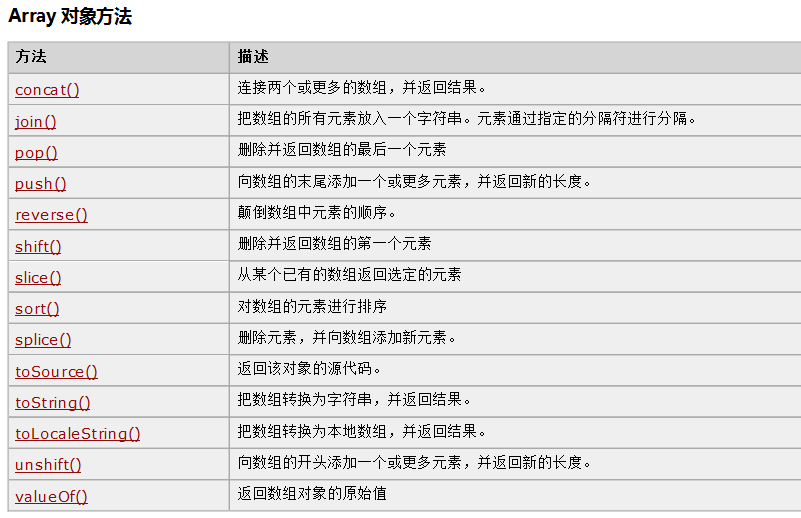
在JS中对象的属性也可以随意删除。

数组

JS中的数组声明方式



JS提供的针对数组的方法



tips：

push + pop

push在数组的末尾新增一个元素

pop 从数组的末尾弹出一个元素

同端进出，是出栈入栈操作

push + shift

push在数组的末尾新增一个元素

shift从数组的头部弹出一个元素

异端进出，是出队入队操作

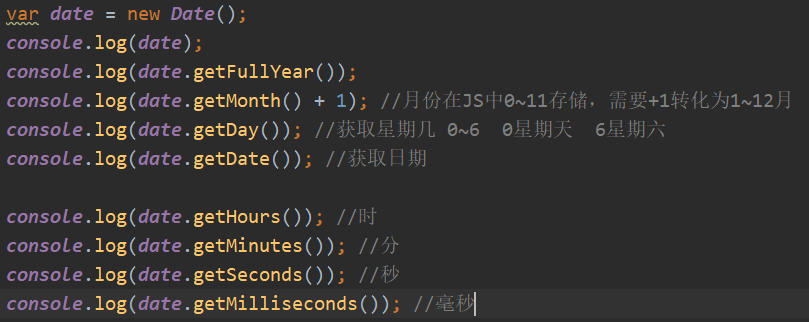
结合上面的方法和这里的队栈操作发现，JS中的Array基本等同于Java中数组 + 集合。

日期

var date = new Date();

常用方法

1、根据时间域获取时间



在JS中没有SimpleDateFormat这样的格式化的类，所以要输出yyyy-MM-dd这种格式的时间的时候，就需要自己写代码一个个取值并拼成目标字符串了。

tips：请一定要注意JS中的month是0-11的，在使用的时候务必+1

2、如何设置时间



可以指定时间域设置时间值

3、整体赋值

getTime 获取Date对象表示的时间点的毫秒数

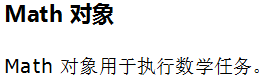
setTime 设置当前Date对象存储的时间点的毫秒数

可以一次性指定一个时间点



还可以在构造方法中就传入毫秒数，生成的date对象在一开始就表示传入的时间点

Math数学对象



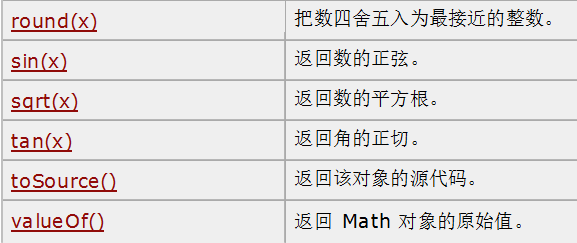
提供了大量的计算用的常量和方法

常量

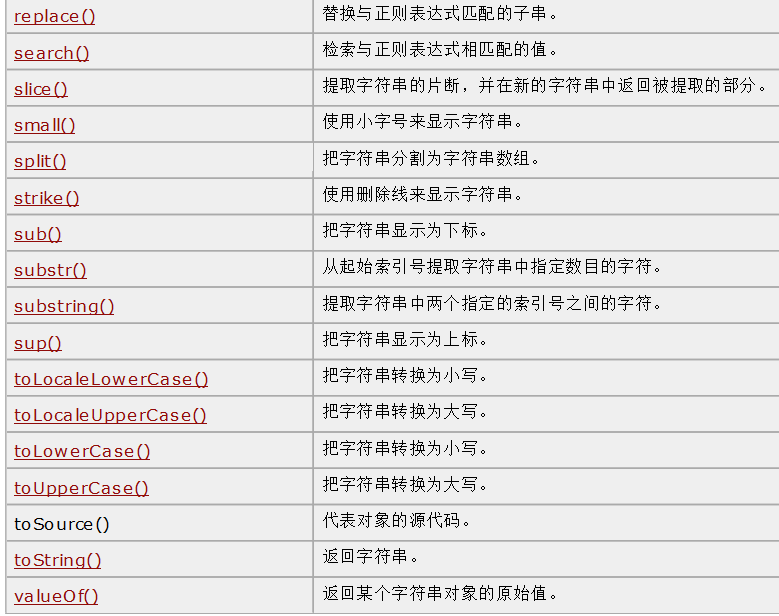


方法





字符串

查看手册发现，大部分的方法都和Java中的String类的方法是相似的

正则表达式

在Java中正则是以字符串的形式出现的，“\\d{10}”

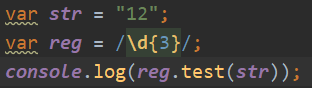
在JS中正则是一个独立的数据类型

RegExp 规则表达式 -> 正则表达式



在JS中可以定义一个正则格式的对象，在赋值的时候使用 /表达式/

常用方法是test，判断正则和入参中的字符串是否匹配，如果匹配返回true，否则返回false



在JS中正则和Java中的略有不同

1、匹配的起始和结束

在正则中使用 ^ 表示字符串的开始 用 $ 表示字符串的结束

/^\d{3}$/

表示字符串中从开始到结束只有三个数字

/^\d{3}/

表示字符串以三个数字为开头

2、元字符

. 表示一个任意字符

字符本身 直接出现在正则中的字符表示它自己

\d 表示一个数字（0~9）

\D 表示一个非数字（可以是任何字符除了数字）

\w 表示一个字典字符（a-zA-Z0-9\_）

\W 表示一个非字典字符

\s 表示一个空白字符（回车、制表符、不显示的特殊符）

\S 表示一个非空白字符

3、枚举

[abc] 表示一个字符，字符可能是abc中的任意一个

[a-z] 表示一个字符，字符是a-z这个连续范围中的任意一个

[0-27-9] 表示一个字符，字符在0-2 或7-9两个区间中的任意一个字符

[.] . 本身是元字符表示一个任意字符，如果写在枚举区间，才能表示小数点本身这个字符

4、量词

{m} 表示一个整体出现m次，不能多不能少

{m, n} 表示整体出现至少m次，最多n次

{m, } 表示整体至少出现m次，上不封顶

\* 表示整体出现0或多次

+ 表示整体出现1或多次

? 表示整体出现最多一次（0或1次）

5、分组

( ) 可以将一端正则写到括号中，表示他们是一个整体

6、或

| 这个符号前后的表达式有一个生效即可

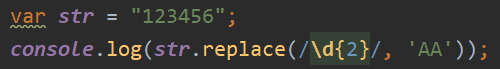
7、修饰符

在正则表达式的最后添加的新的内容，表示匹配规则的设置

这些修饰符在match方法上更加明显

g 如果不加g的话，正则只会匹配一次，匹配上了就结束

加上g，正则会匹配多次，直到字符串的末尾为止（全局匹配）

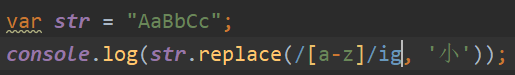


输出的是AA3456，匹配一次12之后就结束了，不再向后匹配



输出的是AAAAAA，匹配到了12 34 56三组结果，直到字符串结束才停止。

i 忽略大小写匹配



如果不加i，只能匹配小写的a-z，加上了之后a-z和A-Z都能匹配

修饰符可以不止加一个，例如这里就复用了i和g，表示不区分大小写的全局匹配

小练习：

1、已知str是QQ邮箱，请给出它的正则表达式

324134134@qq.com

/^\d{6,11}@qq[.]com$/

这题的考点就是 . 的匹配，. 本身是元字符表示任意一个字符，必须放在[]中还原本意才能判断

2、已知str是用户输入的日期（只有日），请给出它的正常表达

01-31

/^(0[1-9])|([12]\d)|(3[01])$/

函数function

1、JS中的预设函数

弹框

1、消息框

alert("hello");

tips：在JS中调用函数，类似于Java中调用方法，使用函数的名字来调用，在（）中输入入参，参数之间用逗号隔开

2、确认框

confirm(“消息”);

这个方法是有返回值的，当用户点击确定的时候，返回true，否则返回false

3、输入框

prompt(“消息”);

方法会返回用户输入的内容，如果用户什么都没有输入直接点确定，返回的是一个空字符串“”，如果用户点击取消返回null。

数值转化

parseInt 将字符串转为整数

tips：应转尽转

这个方法会尽可能的从字符串中读出数字，如果字符串中存在非数字的字符，它会从第一个字符开始转化到第一个非数字字符结束，将读到的内容转为数字。

12A3 -> 12

在parseInt中小数点也是非数字字符，也会导致取值结束。

12.3 -> 12

如果输入的内容中没有任何可转的内容，会返回NaN表示这不是一个有效数字

abc -> NaN

parseFloat 将字符串转为浮点数

parseFloat方法和parseInt是类似的，区别是parseFloat可以接收一个小数点

超过一个的小数点都视为非数字字符会导致取值结束

12.3 -> 12.3

12.3.4 -> 12.3

小练习：请尝试完成以下功能

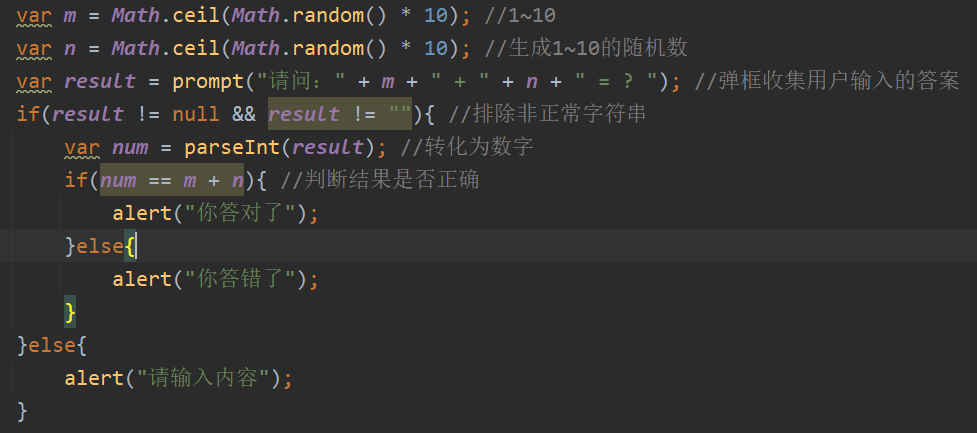
在页面上使用输入框给用户出一道10以内的加减法，用户在输入框中输入答案，在JS中判断用户输入的是否正确，并弹框显示

m + n = ?

m和n应该是使用Math.random生成10以内的两个数字

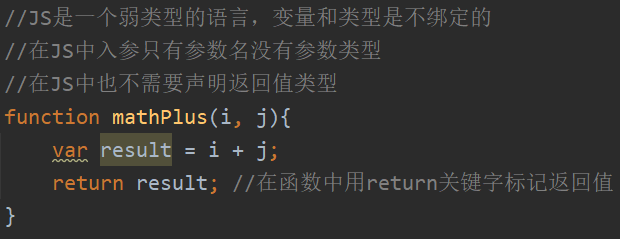
用户输入一个字符串，parseInt转为数字

判断用户输入的数字和m+n的和是否相同，如果相同提示答对了，否则提示答错了



2、自定义函数 function

方法函数



小练习：

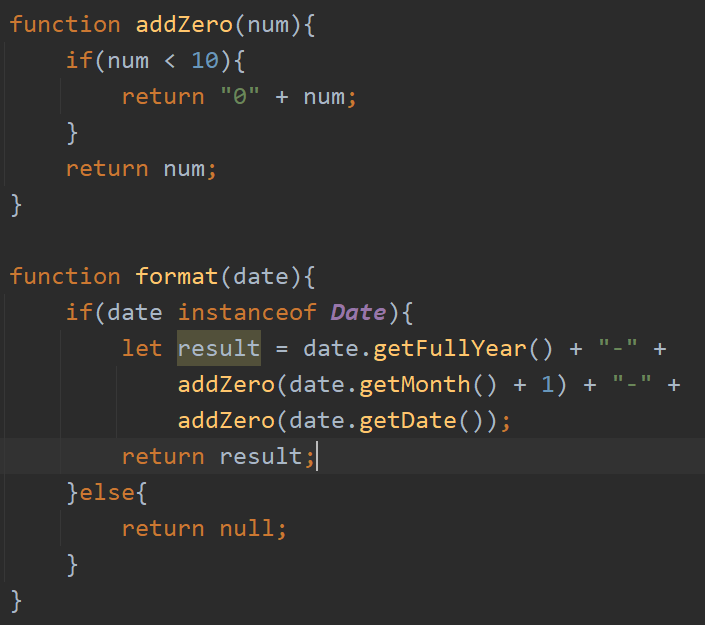
尝试仿写以下SimpleDateFormat的功能，写一个函数入参是date类型的时间，返回YYYY-MM-DD的字符串

1、判断一下是不是Date类型的对象

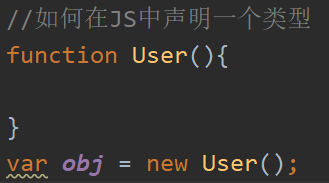
入参 instanceof Date 如果返回true就是的

2、转化格式（getFullYear + getMonth+1 + getDate）

3、返回字符串的值



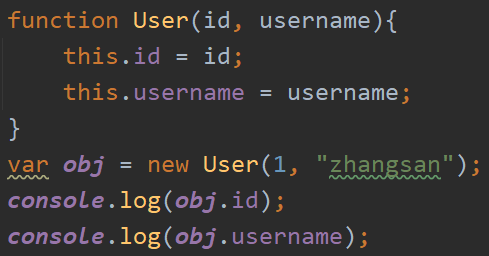
类函数



在之前学习Object的时候和大家提过，JS中对象的属性是不固定的，可以随意的添加和删除。

这里类函数也不能固定属性，只能是在一开始的时候，给出一些预设属性。

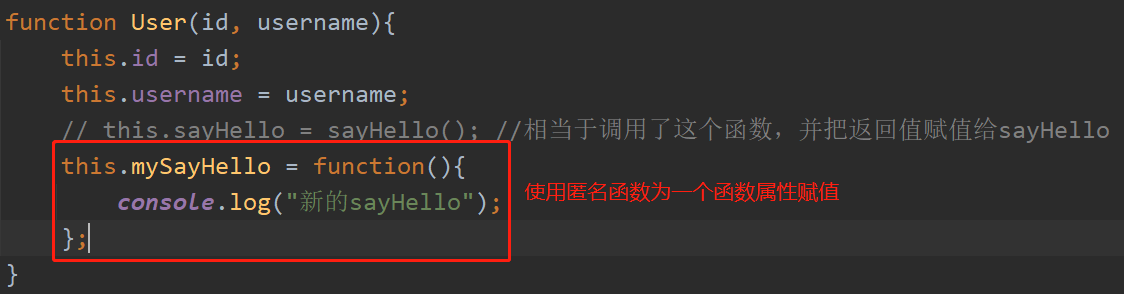
写法类似于Java中的构造方法



如果要预设一些方法，将函数写到类中，可以在这个类对象上调用函数

匿名函数

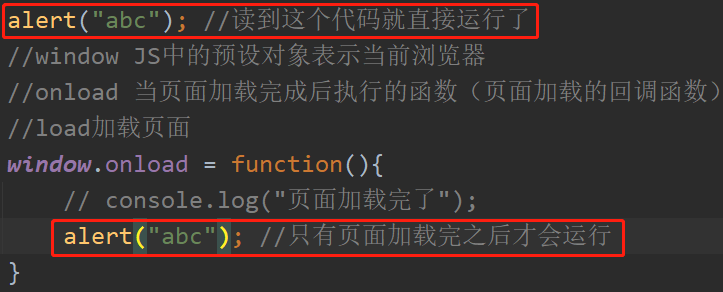
当需要为一个函数进行赋值的时候，使用一个已经存在名称的函数是不明智，可以选择不给函数命名，将它变成一个匿名函数进行赋值



匿名函数最常用的就是给JS中预设的一些回调函数进行赋值

回调函数：当一个事件完成之后执行的函数

例如：



结合延时任务和定时任务完成页面时钟的开发

尝试完成一个在网页上能实时更新的时钟

思路：写一个代码在页面上展示当前时间，每隔1s中再次执行这个代码更新时间

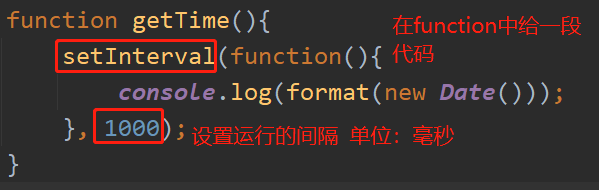
1、学习DOM Document Object Model编程

将Document文档中（网页上）的标签当做一个Object对象进行建模Model

2、学习如何每秒执行一次代码

延时任务和定时任务

定时任务



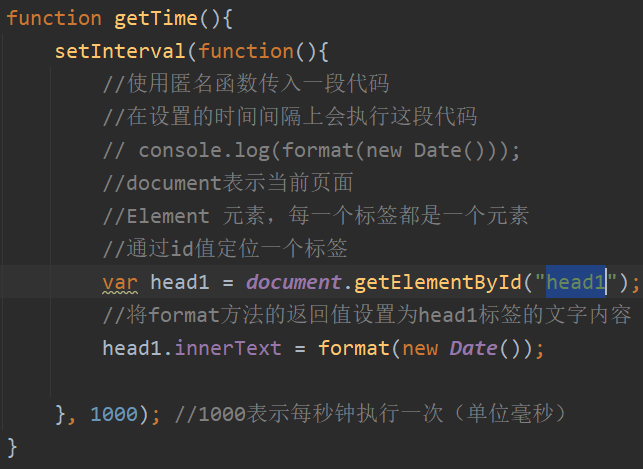
延时任务



tips：周期任务和延时任务使用场景有哪些？

1、例如股票网站、投票网站这些需要固定时间刷新数据的网站就可以用周期任务，每隔一段时间调用刷新数据的方法

2、例如在收验证码的时候，为了防止用户不停的点击获取验证码，一般都有一分钟的冷却时间，可以使用延时任务，点击后锁定按钮，延时1分钟后解锁它。



如何手动停止周期任务和延时任务？

1、周期任务和延时任务的方法是有返回值的，返回的是任务的id





这里收集到的i就是他的id值

2、停止任务的方法是



将id值传入这个方法，就可以停止id对应的周期任务

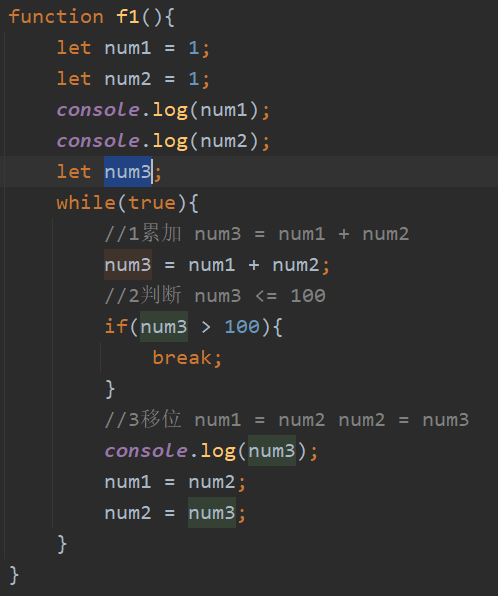


可以停止id对应的延时任务

家庭作业：

1、请尝试在JS中编写一个函数，调用后在控制台上输出100以内的费布拉奇数列

1 1 2 3 5……



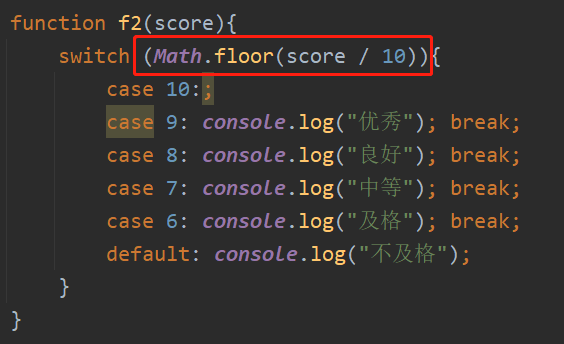
2、已知用户在注册时使用的用户名只能是英文字符和数字，且要求在6~12位之间，请写出合适的正则表达式





3、请尝试编写一个函数，入参是学生的成绩（0~100），根据参数的值进行分级输出，如果成绩在100~90之间，在控制台上输出优秀。89~80之间，输出良好。79~70之间，输出中等。69~60之间，输出及格。60以下输出不及格。

要求使用switch结构实现分支



4、请预习DOM编程

## DOM编程

之前学习的都是单纯的JS代码，从这里开始尝试使用JS代码修改页面上的HTML标签

什么是DOM

Document Object Model

将文档（网页）中的每一个标签当做一个对象（Object）建模的模型。

学习DOM之后就可以将页面上的标签作为JS中的一个对象来访问

1、定位

如何使用JS获取页面上的一个标签（一个DOM对象）

id

var dom = document.getElementById(标签的id值);

Element 元素（标签）

getElementById 根据id属性值获取元素（标签）

tips：根据id获取的是一个元素（id是唯一的不可重复）

标签

document.getElementsByTagName(标签名);

类

document.getElementsByClassName(类名);

名字

document.getElementsByName(标签的name属性名);

根据name取值主要是在表单中使用

tips：根据标签名、类名、名字获取都是HTMLCollection集合，他们都可能一次性获取多个标签，所以返回的是包含所有结果的集合对象

2、值

1、标签之间的文字信息

innerText





2、标签之间的子标签

innerHTML





tips：innerText和innerHTML区别是什么？

区别是参数的类型不同

在innerText中使用的数据类型是CDATA character data 字符数据

在innerHTML中使用的数据类型是PCDATA Parsed character data 经过解释翻译的字符数据

当使用innerText赋值的时候，JS不管你输入的是什么内容，统统当做纯文本，怎么输入，就怎么输出





innerText不会对入参进行解释翻译（CDATA）

使用innerHTML赋值的时候，JS会先将字符串当做html代码进行解释翻译，然后将得到的结果展示在页面上





innerHTML是先解释翻译再输出（PCDATA）

3、表单元素value属性值

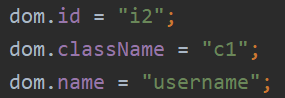




在表单元素中，例如输入框、密码框、多行文本框等用户手写的内容都可以用value进行取值。但是对单选、多选、下拉菜单就较为复杂了。

3、属性

能否使用JS代码修改标签上的属性的值



直接在dom对象后使用属性名即可访问属性的值（可取值，可赋值）

4、CSS样式



先使用style这个属性访问标签的行内样式表

再在这个属性中使用CSS的属性名为它赋值

小练习：尝试使用刚刚学过的知识，在网页上实现一个调色盘

提供三个单行文本框，分别收集红绿蓝三种颜色的色号

提供一个按钮，当用户点击这个按钮后，收集三个输入框中的值，并调出颜色

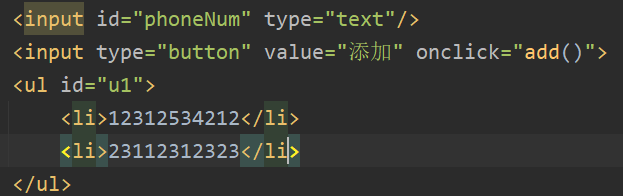
提供一个Div，将调出颜色设置为div的背景色，做颜色的展示





5、添加和删除

小练习：做一个网页版的电话簿，用户在输入框中输入手机号，点击添加按钮后将手机号保存到无序列表中，并清空输入框等待下次输入





tips：上面演示的两种方式在后续的编程中都有出现，且遇上较复杂的页面，可能第二种（拼接字符串）还简单一点。

如何在页面上创建一个新的标签

var dom = document.createElement(“标签名”);

createElement方法可以在JS中创建一个新的dom对象（这个对象不存在与页面上，只在JS代码中）

a.append(b);

a和b都是dom对象，表示将b放入a的当前所有内容之后（追加进去）

尝试在刚刚电话簿上添加删除功能，要求是在每一个号码的后面添加一个删除按钮，点击后删除此行号码。

如何删除一个标签

先获取它的dom对象，然后再对象上调用remove方法，就可以移除这个标签



这是真的从网页上彻底移除一个标签，是一个不可逆的删除操作

小练习：尝试完成一个省市的级联菜单

页面上有两个下拉框，第一个下拉框中选择省份信息，根据第一个框中选择的省份在第二个下拉框中展示该省中的城市名。

江苏：南京、苏州、无锡

四川：成都、绵阳、广元

山东：济南、青岛、淄博

在DOM编程中，没有找到合适的清空所有子标签的方法，所以使用



直接用空字符串覆盖已有的所有内容，实现清空的效果。

