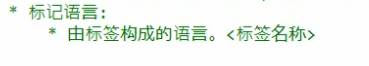
#html

概念：hyper text markup language 超文本标记语言

超链接：就是点一下，可以连接到某个内容页面；多个链接组织在一起就成了网状文本。

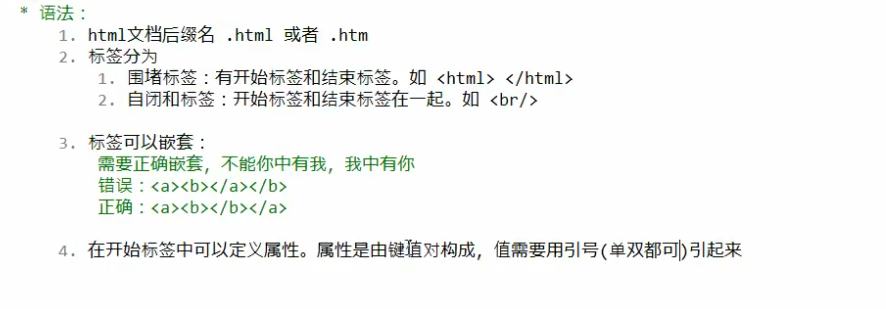


、

如：html xml等

标记语言不是编程语言，标记语言是没有逻辑性的，写上就会执行。

#快速入门



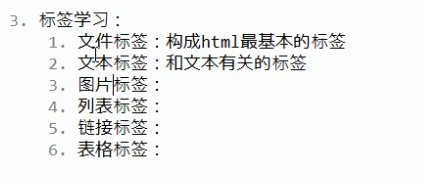
Helloworld.html

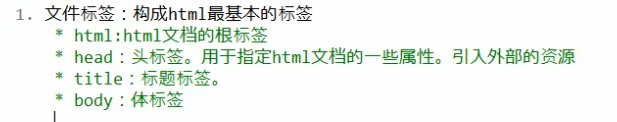


换行helloworld：



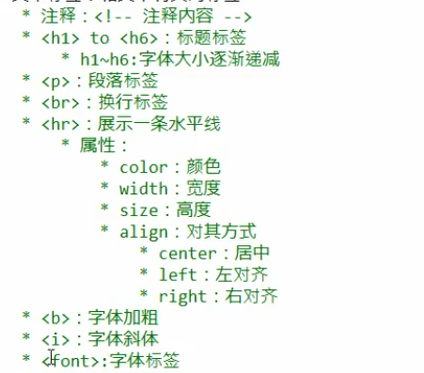
#标签





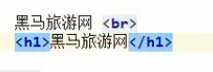
这些标签不用自己记，idea新建html file自己已经写好了。

2.文本标签



<br />换行

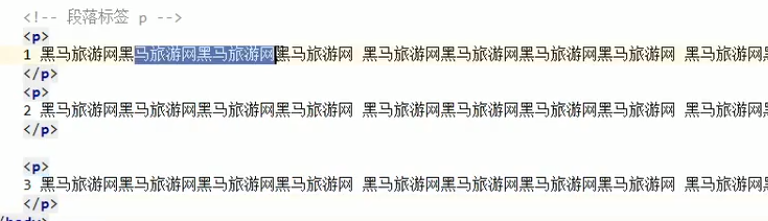
<hi>标题：





H1-6：字体逐渐减小

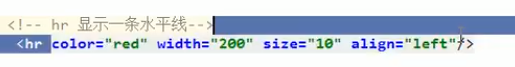
<p>：分段落标签。



把一段抱进一个p标签

<hr>：显示一条水平线





可以修改属性（align是对齐方式），但是不推荐，将来用js或css专门改属性。Html就是用来展示的。

<b>:加粗。对称标签（包进去）

<i>:斜体。对称标签

3图片标签

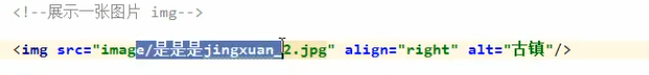
<img>：图片展示

首先需要把图片文件夹放入工程内



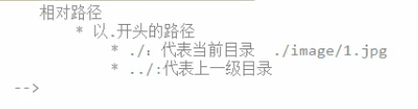
Src是必须属性：即图片的来源

Alt属性：若图片加载失败，可以显示alt字段



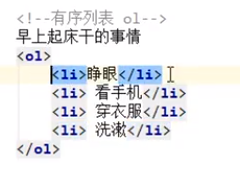
Tips：所有属性都是双引号括起来的字符串

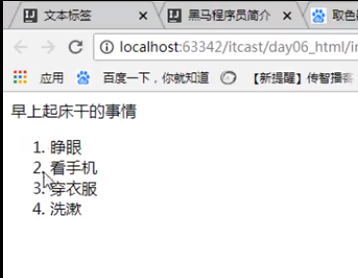
Src也可以使用相对路径来寻转：



4.列表标签

Ol：有序列表 li：列表中的一项



效果：

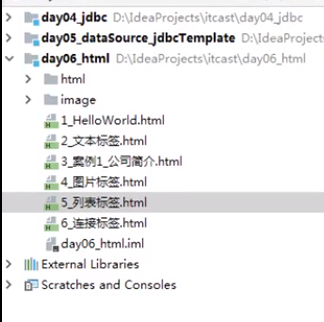
Ul：无序列表 li：列表项

#连接标签

<a>：定义一个超链接 对称标签。必须定义一个属性：href。决定点击后跳转到哪里。

另一个属性：target。 Target：\_self,在本页面打开；target=\_blank,在新页面打开。



最后一行：./是指当前目录下；

可以套娃，把你之前写的页面作为href属性



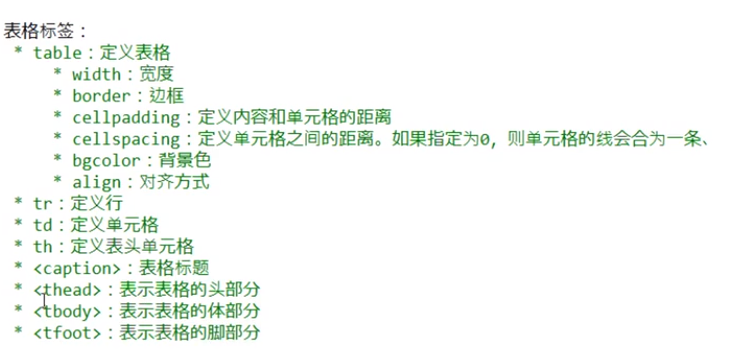
双重套娃：点击图片可以跳转

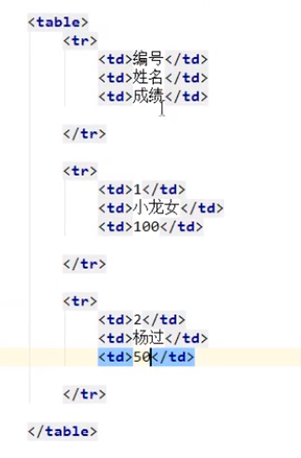
#div和span标签（其实是和css配合使用的，css可以控制这两个标签中的文本样式）

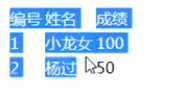
Span不换行 div换行（每个div占满一行）

Div有一个属性叫id，可以用id区分一个页面内不同的div

#表格标签







属性：

：表格有边界线分开

Tips：每个标签的属性其实没必要记住，上html手册自己查就好啦

#JavaScript

#概述

JavaScript首先和java没有半毛钱关系。

Js实际上由三部分脚本语言组成：ecmascript+bom+dom

#与html结合方式

内部js：

在html文件中写js代码

用一个Script标签包住，内部就可以写js代码

外部js：

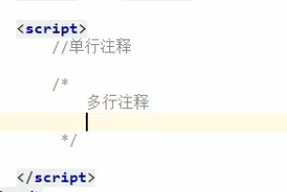
专门一个js文件里面写js代码

在html文件中，

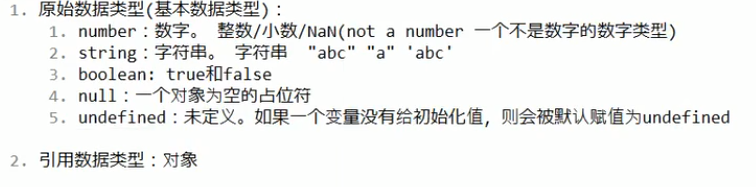
总结：



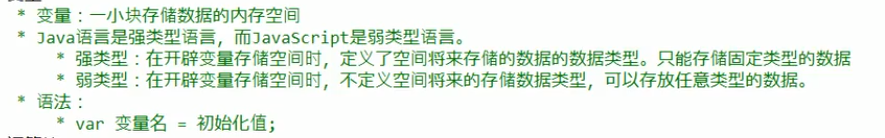
#js注释



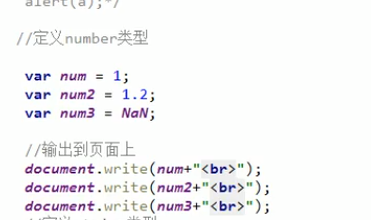
#js中的数据类型



#js是一个弱类型的语言



举例：

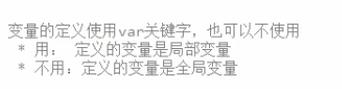


既然定义变量都是用var定义的，我咋知道那个变量是啥类型呢？



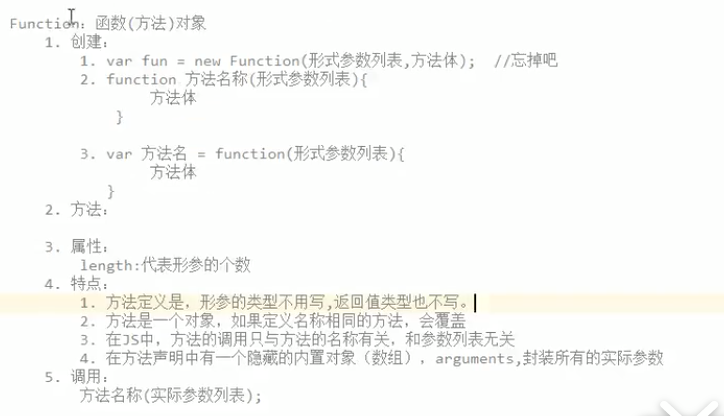
用typeof即可

如何定义全局变量：

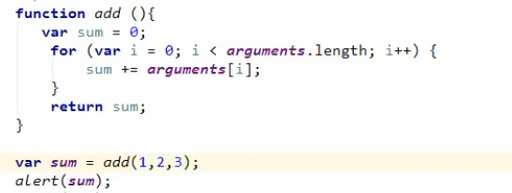


#function对象

Js中，一个方法就是一个对象（一般的面向对象语言，方法应该是对象的一部分）

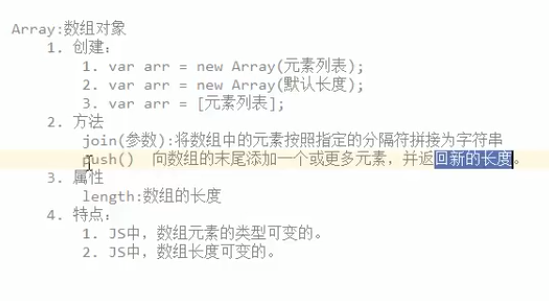


关于特点，这个函数可以很好的说明：



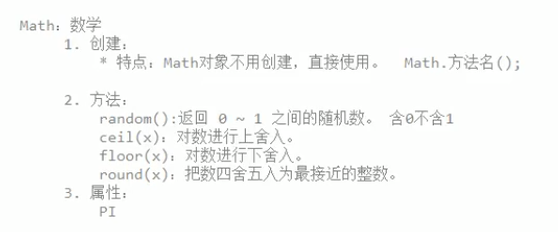
实际参数列表可以随便写任意数量的数，都会装入arguments数组中。

#js中的数组array

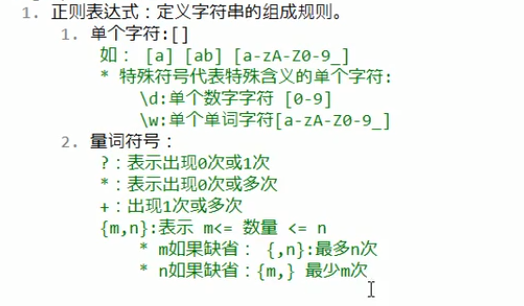


Tips：其实js中的各种对象，去JavaScript手册查看就行

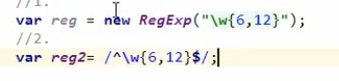
#math对象



#regexp正则表达式







^$是开始结束符

第一个方法其实是不对的，因为字符串中的/默认是转义符

正确方法：



#global类（主要使用里面的解码编码方法）



还有一个常用方法：



#dom

功能：控制html文档的内容

最关键的一句代码：获取页面标签（element）对象



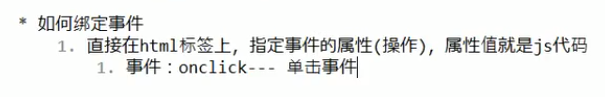
获取了对象之后，要对对象进行操作：



获取对象之后，直接点它的属性；这个对象有哪些属性，查手册

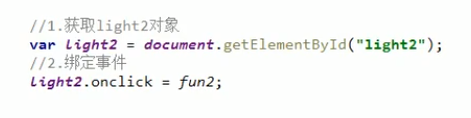
#事件（简单学习）





Like this：

当然了，最好是把绑定事件和获取对象分开写：



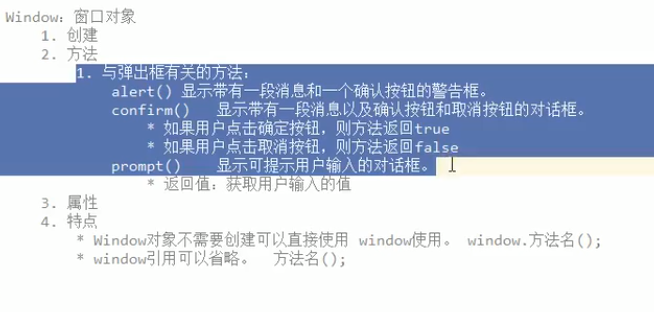
#bom（browser object model）

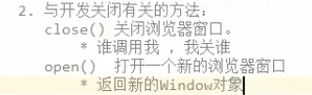
首先来看bom中的对象：





Window对象：



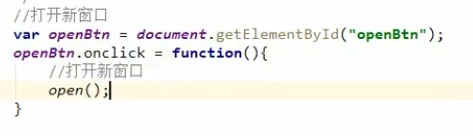


小例子：通过确定按钮打开一个新网页

首先定义一个按钮：



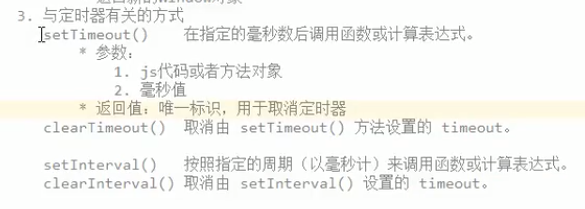
获取按钮这个控件的属性



Open（）可以加参数：网站域名；open（）的返回值就是打开的网页。

请看例子：





#window中的location对象

方法：



Href：网页当前的地址





#dom

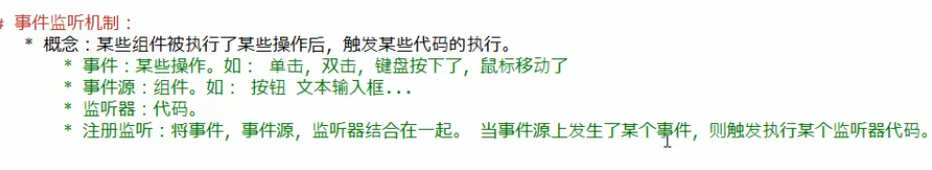


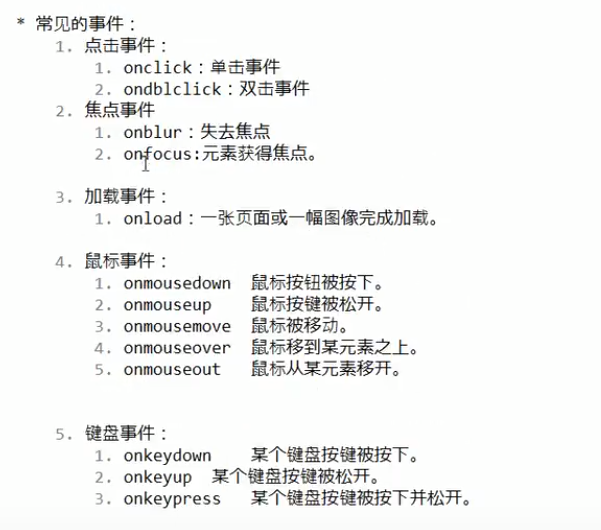


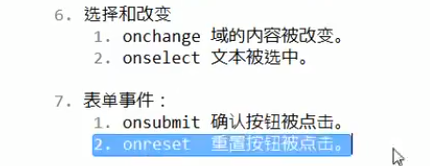
我们现在主要学html dom

（暂时不学）

#事件（dom event）

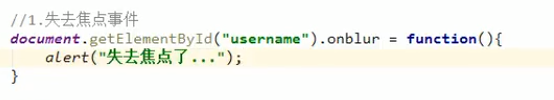






1. 焦点丢失

（文本框）

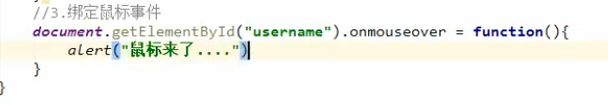


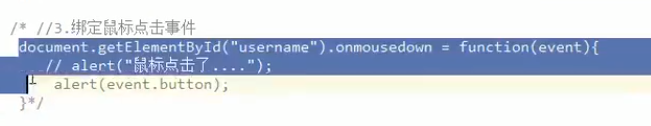
1. 加载完成事件



Onload（）可以写在script前面

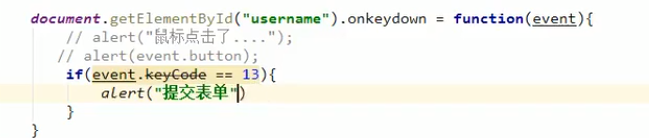
3.鼠标事件



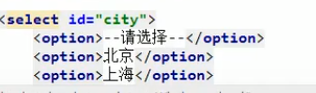


Event这个参数：当你点击鼠标时，其实就产生了一个事件；这个事件有一个整数返回值，代表不同的按键（012分别是鼠标左中右键；13是回车）

Like this：



做一个简单的下拉选择列表：

</select>

接上这个：



实现效果：当你改变选择栏中的选项时，提示

Onsubmit实现校验提交：

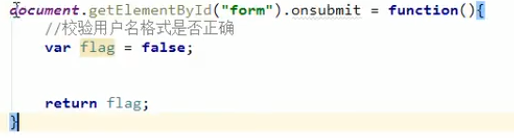
首先创建表格：



可以看到，input的属性必须是submit，你写个button按钮把名字改成submit没用；

表格form的属性action是指表单提交后提交到哪里，#是默认值。

然后校验：



这么写的话，永远都是不提交。但是你要明白，必须返回一个Boolean值。