

jquery 写的少，做的多，链式编程 隐式迭代 解决js的兼容问题

dom中

window.onload事件 只可以写一个. jquery中可以写好多

页面中所有的内容加载完毕后才触发, 标签, 图片, 文字内容....

```
$(window).load(function () {  
    console.log("啊哈哈, 我真的好帅1");  
});
```

页面中基本的标签加载后就执行

```
$(document).ready(function() {  
    console.log("啊哈哈, 我真的好帅1");  
});  
jQuery(function() {  
    console.log("啊哈哈, 我真的好帅1");  
});  
$(function() {  
    console.log("啊哈哈, 我真的好帅1");  
});
```

2: 对象转化

** jQuery对象---->DOM对象:*

** 1.*

** \$("#btn")[0]---->DOM对象*

** 2.*

** \$("#btn").get(0)---DOM对象*

//DOM对象转jQuery对象, 只需要把DOM对象放在\$(dom对象)--->jQuery对象

3: `$("#dv").text("这是一个有颜色的div");`

//相当于DOM中的innerText或者是textContent

```
$("#dv").css("backgroundColor", "yellow");
```

//设置元素的样式

4: 选择器:

基本的 (5个): 类 id 标签

标签类选择器

多条件选择器 中间是 ,

//通过多条件选择器获取元素, 统一设置背景颜色

```
$("#btn").click(function () {  
    $("span, p, li, div").css("backgroundColor", "yellow");  
});
```

层次选择器:

#div p *div中所有的p标签元素*

#div > p *直接的子元素*

//获取的是div中所有的p标签元素

```
//$("#dv p").css("backgroundColor", "red");
```

//获取的是div中直接的子元素

```
//$("#dv>p").css("backgroundColor", "red");
```

//获取的是div后面的第一个p标签元素

```
//$("#dv+p").css("backgroundColor", "red");
```

//获取的是div后面所有直接的兄弟元素p标签元素

```
//$("#dv~p").css("backgroundColor", "red");
```

5: .html() 方法, 设置标签中间显示其他标签及内容, 类似于innerHTML

.text() 方法, 设置标签中间显示的文本内容, 类似于innerText

.val() 方法. 设置input标签中value的值, 类似于value

.css() 方法, . 设置元素的样式, 类似于style

//.text(); 如果小括号中写内容, 就是设置文字内容

//如果.text();小括号什么也不写,表示的是获取这个元素中的文字内容

//.html();小括号中可以直接写标签的字符串内容,就是设置div中的元素内容

//.html();小括号中什么也没有,表示的时候获取div中的元素内容

val()方法,小括号中写内容就是设置元素的value属性

*.val()方法,小括号中什么也不写,获取元素的value属性的值

*就是元素的value属性

/*.css();方法,该方法如果只是设置一个样式的属性和值,那么这个方法写两个参数,第一个参数是属性,第二个参数是值

```
$("#dv").css("backgroundColor","yellow");
```

/*.css();方法,里面可以直接写键值对的方式

```
$("#dv").css({"width":"300px","height":"200px","backgroundColor":"yellow"});
```