

jQuery- 第1天

认识 jQuery 是针对 DOM 的封装的本质，能够基于 jQuery 完成节点查找、事件监听、样式处理等基础操作。

- 知道如何在网页中应用 jQuery
- 理解 jQuery 实例与原生 DOM 对象的区别
- 知道何时使用选择器获得 jQuery 实例
- 能够基于网页元素的结构关系获得 jQuery 实例
- 能够使用 jQuery 实例方法处理事件监听及样式

了解 jQuery 的本质是针对 DOM 进行的一系列封装，知道如何下载 jQuery 并在网页中应用。

介绍

[jQuery](#) 是一个 Javascript 专注于 DOM 操作的类库，通过一系列的封装，不仅简化了 DOM 操作，还处理了不同浏览器之间的兼容，极大的提升了编码的效率，其口号是 Write Less Do More!

下载

jQuery 从形式上来看只是一个独立的 .js 文件，可以在其[官网](#)免费下载使用，在其官网上提供了多个不同的版本供开发者下载，一般推荐下载最新版本（目前为 v3.5.1），值得注意的是 jQuery 自 v2.0 开始不再兼容 IE6/7/8。

通常下载 jQuery 时需要有 3 个文件，分别为：

jquery-版本号.js

jquery-版本号.min.js

jquery-版本号.min.map

其中 jquery-版本号.js 与 jquery-版本号.min.js 内部分代码是一致的，区别仅仅是 jquery-版本号.min.js 中不包含注释、换行、缩进等，甚至变量名也被处理成单个字母形式，这样做的目的是压缩代码量，使文件体积变小。

jquery-版本号.min.map 是官方在压缩代码时自动生成的一个文件，该文件中记录了 jquery 压缩前后的对应关系，浏览器调试时能够快速定位到出错代码的位置。

首先将下载好的 jQuery 文件使用 script 标签引入到网页标签中

由于 jquery-版本号.min.js 的文件体积较小，所以推荐使用。

假如基于原生 DOM 实现上面的交互效果不得不编写大量的代码，然而基于 jQuery 的实现只用了很少的代码，这便是开发中推荐使用 jQuery 的最主要原因！！！！

了解 jQuery 的基础概念及工作机制，知道原生 DOM 对象与 jQuery 对象的区别及联系。

jQuery 是基于 Javascript 构造函数的原型对象实现的，通过为原型对象添加属性或方法的方式实现对 DOM 操作的封装

- 总结：
- 网页中引入 jQuery 后会得到一个全局的函数 jQuery 或 \$
- \$ 是 jQuery 函数的别名，使用 \$ 比使用 jQuery 更方便
- 每次调用 jQuery 或 \$ 函数都会得到一个**新的实例**
- 原型对象中存在许多的方法，调用这些方法实现 DOM 的各种操作

jQuery 中的选择器是用来获取 DOM 节点的，其功能类似于原生的 `querySelectorAll` 方法，jQuery 中所支持的选择器与 CSS 的选择器几乎一致。

- **选择器：jQuery通过元素的选择器获取元素、注意jQuery获取的是伪数组形式的，称为jQuery的对象**
- 标签选择器
- 类选择器
- id 选择器
- 后代选择器

- 总结：
- 将 CSS 的选择器以参数形式传给 jQuery 函数后便可以获取相应的 DOM 节点
- 通过 jQuery 选择器获取 DOM 节点的同时也得到了一个 jQuery 实例
- 调用实例方法完成相应的 DOM 操作
- 注：几乎所有 CSS 选择器都可以被 jQuery 支持，课堂上不必列举所有的情形！

虽然在 jQuery 中利用选择器能获取到相应的 DOM 节点，但此时的 DOM 节点并非是原生的 DOM 对象，因此也就无法直接调用原生的 DOM 方法

- 在 jQuery 中利用选择器获取的 DOM 节点 jQuery 构造函数的实例对象，只能调用实例的方法才能完成 DOM 操作
- 我们将 jQuery 获取的 DOM 节点对象称为 jQuery 对象，css 方法是 jQuery 对象的方法，而 style 是原生 DOM 对象中的一个属性。
- 在 jQuery 中并非只能通过选择器获取 jQuery 对象，它还可以将任意原生 DOM 对象转换成 jQuery 对象，

注：由于 jQuery 实例中包含了许多关于 DOM 操作的方法，因此基于 jQuery 开发时会将原生 DOM 对象转换成 jQuery 对象后再使用。

学习 jQuery 中实例方法的基本用法，借助 jQuery 封装的实例方法查找、筛选网页元素，并能处理事件监听及样式处理。

事件

快捷方法

基础方法

自定义事件

在 jQuery 中以原生事件类型的名称为依据封装了相对应的事件处理方法

- 我们将以原生事件类型名称做为方法名为 DOM 添加事件的方式称为快捷方法，
- 总结：
- 语法简洁，接收回调函数做为参数，事件触发时该回调函数被执行
- 事件触发后回调函数中的 this 指向添加事件的原生 DOM
- \$(this) 是将原生 DOM 对象转换为 jQuery 对象

在 jQuery 中使用快捷方法能够快速为 DOM 节点添加事件监听，然而灵活性方面略显不足，因此 jQuery 又提供了较为灵活的 on/off、bind/unbind、one 方法处理 DOM 事件

- 总结：
- on、bind、one 语法结构与 addEventListener 类似，分别接受事件类型和回调函数做为参数
- on 和 bind 含义基本一致推荐使用 on 添加事件监听，one 添加的事件监听只生效 1 次
- off 和 unbind 用来移除 DOM 的事件，无论以何种方式添加的事件均可被移除，推荐使用 off 方法
- 事件回调函数中的 this 仍然指向的是添加事件监听的原生 DOM

在 jQuery 中除了支持原生 DOM 事件类型外，还允许开发者自定义事件类型

- 总结：
- 自定义事件类型语法与原生事件类型一致
- 自定义事件只能通过 trigger 方法才能被触发
- trigger 方法也能用来触发原生事件类型
- 注：自定义事件是一种高级的用法，不必深入理解只需要了解其基本用法即可。

借助 DOM 动态更改网页元素样式是开发中最常用到的，为此 jQuery 中对样式的操作进行封装，具体分为以下两种情形：

- 行内样式
- 类名操作

通过 css 方法动态修改 DOM 的行内样式

- 总结：
- 接受两个参数（分别为样式属性和值）时，逐一改变网页元素单个的样式属性
- 接受一个参数时（对象类型），同时修改网页元素的多个样式属性
- 样式属性值为长度时，省略长度单位 px
- 接受单个样式属性做为参数时，能够获取该属性对应的值
- 注：css 方法对应原生 DOM 操作中的 style 属性

jQuery 中封装了为网页元素添加、移除、检测、切换类名的方法。

- 总结：
- addClass 方法为网页元素添加一个或多个类名，多个类名用空格进行分隔
- removeClass 方法移除网页元素的一个或多个类名，多个类名用空格进行分隔
- hasClass 方法检测网页元素是否包含某个特定的类名
- toggleClass 方法切换（添加/移除）一个或多个类名，多个类名用空格进行分隔
- 注：以上 4 个方法的所实现的功能分别对应原生 DOM 中的 classList 的 add、remove、contains 和 toggle 4 个方法。

链式也叫链式操作，它是 jQuery 中提供的一种简便的语法结构，可以在一定程度上精简代码、增强代码的可读性

- 总结：
- 链式操作只是一种简便的语法结构
- 链式操作时实例方法自左向右依次被执行

在 jQuery 中不仅可以使用选择器查找网页元素，还可以基于网页元素的结构关系查找新的网页元素，分为以下几种情形：

- 父子关系
- 兄弟关系

Find、children、parent、parents

- find 方法，参照某父元素查找其**后代元素**，该方法以选择器做为参数，在其后代元素查找符合选择器条件的网页元素
- children 方法，参照某父元素查找其**子元素**、注: children 方法还允许传入选择器获取指定的子元素。
- parent 方法，参照某个子元素查找其**父元素**
- parents 方法，参照某个子元素查找其**祖先元素**、注: parents 方法还允许传入选择器获取指定的祖先元素。

siblings、prev、next

- siblings 方法，查找所有同级结构关系的元素、注: siblings 方法还允许传入选择器获取指定的兄弟元素。
- prev 方法，查找当前元素之前的同级结构元素
- next 方法，查找当前元素之后的同级结构元素

大多数情况下 jQuery 获取的网页元素多以集合（伪数组）形式存在，同时 jQuery 还允许基于这个集合进一步筛选出更精确的网页元素

- first 方法，筛选出集合中的第一个元素
- last 方法，筛选出集合中的最后一个元素
- eq 方法，以索引为依据筛选集合中的元素

jQuery 中封装了设置和读取网页元素属性的方法

- 总结：
- attr 接受属性名为参数时用于获取该属性对应的值
- attr 接受属性和值两个参数时用于设置该属性且赋值
- 注：针对自定义属性可以使用 data 方法。

jQuery 中封装了设置和读取网页元素文本内容的方法

- 总结：
- text 相当于原生的 innerText，它无法解析 html 标签
- html 相当于原生的 innerHTML，能够解析文本中的 html 标签
- html 和 text 方法不传入参数时用于读取文本内容



传智教育旗下高端IT教育品牌