Model

数据的表现形式。可以创建、校验、销毁、保存到服务端。

当UI操作（比如点击按钮）导致model变更，model会“触发”一个“change”事件，使用这个Model展示的所有View都会接收到该事件并做出响应（重新渲染）。

View

数据展现形式，如表格、文本框等。

backbone语法 参数是非严格形式的json，key可以不带双引号

Backbone.Events 事件，可以混入任何Object，使该Object拥有绑定和触发自定义方法的能力。

on object.on(event,callback,[context])

示例：object.on(“alert”,function(msg){alert(msg);})

给object绑定alert事件。event参数可以使用“:”配置命名空间，如“change:select”，也可以同时绑定多个事件，用空格符分开，如“change:select change:author”。

context上下文，如 object.on(“change”,o.render,o)，o是另一个对象。

off object.off([event],[callback],[context]) 跟on相反，事件解绑

如果没有指定context，就解除所有event和callback相同的事件绑定；如果没有指定callback，就解除所有event相同的事件绑定；如果没有指定event，就解除所有事件绑定。

trigger 触发事件 object.trigger(event,[\*args])

Backbone.Model 模型，提供一系列基本方法。我们可以通过继承模型来获得这些能力。

非常重要：Model的属性，是通过get和set方法设置的，内部用一个hash来维护。model.get(attr)和model.attr是两码事。

extend Backbone.Model.extend(properties,[classProperties])

自动设置原型链，以便子类继承。

调用父类方法(super)：

var Note = Backbone.Model.extend({

set:function(attributes,options){

Backbone.Model.prototype.set.call(this,attributes,options);

…

}

});

initialize 相当于Constructor 构造方法，new一个实例的时候会调用该方法。

new Book({“name”:”三国演义”})，这里的参数是通过调用set方法设置的，但不会触发事件。

get get(attribute) 略

set set(attributes,[options])

get和set明显是一对。set的第一个参数和jQuery里的css方法一样，可同时设置多个参数。第二个参数用于控制某些行为，比如传入{silent:true}，则不会触发相应的change事件。另外，如果该model拥有validate方法，则会先执行该方法，如不通过则不执行set方法。

escape 和get类似，不过获取的attribute值经过了转义处理，以避免xss攻击。

has model.has(attr) 如果attr是一个非null并且非undefined的值，返回true，否则返回false

unset model.unset(attr,[options])

从属性hash表里删除掉attr属性，并根据options设置决定是否触发相应change事件。

clear model.clear([options])

删除model的所有属性，并根据options设置决定是否触发相应change事件。

id model.id

id是一个特别属性，可以是字符串、整数或者uuid。如果使用set方法设置了id，则会将该值覆盖到model.id。Model可以通过id从collection中获取，并且id默认用于生成Model的URL。

idAttribute model.idAttribute

对于某些后端（比如MongoDB），id的形式是“\_id”，此时可以设置model.idAttribute属性，将id映射到“\_id”上。

cid

Model的内部特殊属性（property，不是attribute），当Model第一次被初始化时自动生成。当Model没有保存到服务端或者Model的id没有最终确定时，cid很有用。

attributes model.attributes

attributes是持有Model状态的内部hash，请通过set和get来操作。如果确实想得到attributes的一份拷贝，请使用toJSON方法。

changed model.changed

内部维护的一个hash，保存那些修改过的attributes。同样请不要直接访问，而使用set和change来操作。可以通过model.changedAttributes得到一个副本。

defaults model.defaults or model.defaults()

为attributes设置默认值，当新建一个Model时，没有设置值的attribute就会被设为默认值。注意：默认值在所有实例间共享。

toJSON model.toJSON()

将model.attributes转换为一个json对象。如果想得到json字符串，请使用JSON.stringify(model)。

fetch model.fetch([options])

代理给Backbone.sync，从服务端获取数据重设Model状态，返回一个jqXHR（jQuery中的XMLHttpRequest对象）。如果返回的数据和现有Model不一致，则会触发一个change事件。（该change事件跟单独的attribute变动引发的change有何不同？）接受success和error回调，这两个函数的入参都是(model,response)。

save model.save([attributes],[options])

代理给Backbone.sync，将model保存到数据库，返回一个jqXHR对象。如果model拥有validate方法并且校验失败，则不会保存。如果model是isNew状态，保存将会以create方式进行，否则就是update。

调用save时，（如果attributes有变动），会立即触发change事件，如果想延迟该事件，可以设置{wait:true}。

save也接受success和error回调，入参也是(model,response)。validate方法校验不通过或者服务端validate不通过都会触发error，其中服务端错误会返回200状态码。

destroy model.destroy([options])

相当与delete，方法定义与fetch和save差不多。如果Model的状态是isNew，返回false。否则会立即触发destroy事件，如果想延迟该事件，可以设置{wait:true}。

validate model.validate(attributes)

在set和save之前执行，如果校验失败，则值不会发生改变。

isvalid model.isvalid()

检查Model当前状态是否正确，调用validate实现。

url model.url()

如果Model加入了Collection，则返回“/[collection.url]/[id]”，否则返回“[urlRoot]/[id]”。这两者必须要设置一个，否则返回不了url。（如何设置collection.url？）

urlRoot model.urlRoot or model.urlRoot()

参见url

parse model.parse(response)

当服务端返回模型数据（fetch或者save）时会调用该方法。

clone model.clone()

isNew model.isNew() 没有id返回true

change model.change()

手动触发change事件和change:attritue事件。适用于这种场景：为了快速设置attributes，设置了{silent:true}，设置完所有attributes后，手动触发。

hasChanged model.hasChanged([attributes])

判断某个attribute自从上次被改变后，是否被改变过。在change事件里比较有用。

changedAttributes model.changedAttributes([attributes])

获取Model中修改过的attributes。也可以传入一个外部的attributes，返回相异的attributes。用于判断View中的哪些部分需要update。

previous model.previous(attribute)

在change事件中，该方法可以用来获取一个被修改的attribute的旧值。

previousAttributes model.previousAttrutes()

用于版本比较。

Backbone.Collection 一组模型的集合

extend Backbone.Collection.extend(properties,[classProperties])

语法跟Model的extend一致。

model collection.model 不明

constructor/initialize new Collection([models],[options])

models collection.models

偶尔会需要直接使用。

toJSON collection.toJSON()

将collection格式化为json对象数组。

Underscore.Methods 28个，操作数组的一些通用方法。