

# 系统学习前端之FormData详解

xmlhttprequest   formData   javascript   **zhangguixu**   2016年08月25日发布

## FormData

### 1. 概述

FormData类型其实是在XMLHttpRequest 2级定义的，它是为序列化表以及创建与表单格式相同的数据（当然是用于XHR传输）提供便利。

### 2. 构造函数

创建一个formData对象实例有几种方式

1、创建一个 **空对象** 实例

```
var formData = new FormData();
```

此时可以调用append()方法来添加数据

2、使用已有的表单来初始化一个对象实例

假如现在页面已经有一个表单

```
<form id="myForm" action="" method="post">
  <input type="text" name="name">名字
  <input type="password" name="psw">密码
  <input type="submit" value="提交">
</form>
```

我们可以使用这个表单元素作为初始化参数，来实例化一个formData对象

```
// 获取页面已有的一个form表单
var form = document.getElementById("myForm");
// 用表单来初始化
var formData = new FormData(form);
// 我们可以根据name来访问表单中的字段
var name = formData.get("name"); // 获取名字
var psw = formData.get("psw"); // 获取密码
// 当然也可以在此基础上，添加其他数据
formData.append("token", "kshdfiwi3rh");
```

### 3. 操作方法

首先，我们要明确formData里面存储的数据形式，一对key/value组成一条数据，key是唯一的，一个key可能对应多个value。如果是使用表单初始化，每一个表单字段对应一条数据，它们的HTML name属性即为key值，它们value属性对应value值。

key	value
k1	[v1,v2,v3]
k2	v4

#### 3.1 获取值

我们可以通过get(key)/getAll(key)来获取对应的value，

```
formData.get("name"); // 获取key为name的第一个值
formData.get("name"); // 返回一个数组，获取key为name的所有值
```

#### 3.2 添加数据

我们可以通过append(key, value)来添加数据，如果指定的key不存在则会新增一条数据，如果key存在，则添加到数据的末尾

```
formData.append("k1", "v1");
formData.append("k1", "v2");
formData.append("k1", "v1");
```

### 3.3 设置修改数据

我们可以通过set(key, value)来设置修改数据，如果指定的key不存在则会新增一条，如果存在，则会修改对应的value值。

```
formData.append("k1", "v1");
formData.set("k1", "1");
formData.getAll("k1"); // ["1"]
```

### 3.4 判断是否该数据

我们可以通过has(key)来判断是否对应的key值

```
formData.append("k1", "v1");
formData.append("k2", null);

formData.has("k1"); // true
formData.has("k2"); // true
formData.has("k3"); // false
```

### 3.5 删除数据

通过delete(key)，来删除数据

```
formData.append("k1", "v1");
formData.append("k1", "v2");
formData.append("k1", "v1");
formData.delete("k1");

formData.getAll("k1"); // []
```

### 3.6 遍历

我们可以通过entries()来获取一个迭代器，然后遍历所有的数据，

```
formData.append("k1", "v1");
formData.append("k1", "v2");
formData.append("k2", "v1");

var i = formData.entries();

i.next(); // {done:false, value:["k1", "v1"]}
i.next(); // {done:fase, value:["k1", "v2"]}
i.next(); // {done:fase, value:["k2", "v1"]}
i.next(); // {done:true, value:undefined}
```

可以看到返回迭代器的规则

- 1. 每调用一次next()返回一条数据，数据的顺序由添加的顺序决定
- 2. 返回的是一个对象，当其done属性为true时，说明已经遍历完所有的数据，这个也可以作为判断的依据
- 3. 返回的对象的value属性以数组形式存储了一对key/value，数组下标0为key，下标1为value，如果一个key值对应多个value，会变成多对key/value返回

我们也可以通过values()方法只获取value值

```
formData.append("k1", "v1");
formData.append("k1", "v2");
formData.append("k2", "v1");

var i = formData.entries();

i.next(); // {done:false, value:"v1"}
i.next(); // {done:fase, value:"v2"}
i.next(); // {done:fase, value:"v1"}
i.next(); // {done:true, value:undefined}
```

## 4. 发送数据

我们可以通过xhr来发送数据

```
var xhr = new XMLHttpRequest();
xhr.open("post", "login");
xhr.setRequestHeader("Content-Type", "application/x-www-form-urlencoded");
xhr.send(formData);
```

这种方式可以来实现文件的异步上传。

1. [兼容性查询](#)

2. [MDN](#)

3. 《JavaScript高级程序设计》

2016年08月25日发布

...

赞

|

8

收藏

|

63


### 你可能感兴趣的文章

- [XMLHttpRequest FormData](#) 7 收藏，2.3k 浏览
- [XMLHttpRequest Level 2 FormData 提交二进制文件](#) 3 收藏，3k 浏览
- [【译】使用FormData对象](#) 1 收藏，796 浏览

### 3 条评论


默认排序

时间排序




李留白 · 2017年05月19日

3.1 获取值 下面的 formData.get("name"); // 返回一个数组，获取key为name的所有值 应该是getAll吧？

 赞


+1

回复



6666 · 2017年04月14日

想问 如何接受这些不同的数据


 赞

回复

文件用 Request.Files["name"],文本用reques.from["name"]注意大小写


— [rao\\_581b0cc697bde](#) · 2017年04月17日

添加回复



文明社会，理性评论

发布评论



zhangguixu

538 声望

关注作者

发布于专栏

[前端与生活](#)

前端基础笔记

13 人关注

关注专栏



首页



问答



头条



专栏



讲堂



更多