note.md

# HTML5新特性 -- Unit05

# 1.Multer中间件

Multer中间件可以实现处理<input type="file">类型的表单数据。

在进行文件上传时，表单的要求如下：

A.表单的提交方式只能为POST

B.必须设置表单的enctype="multipart/form-data"属性

示例代码如下：

<form action="" method="post" enctype="multipart/form-data">
  
 ...
  
</form>

## 1.1 安装

npm install --save multer

## 1.2 使用Multer

//加载Multer模块
  
const multer = require('multer');
  
   
//设置Multer -- 上传位置 (destination)
  
const upload = multer({dest:'upload'});

### 1.2.1 请求参数

#### • single(filename)

当通过Multer中间件进行文件上传操作时，上传文件的信息会存储在req.file或req.files属性中；单文件上传时，可以通过single(filename)来进行操作，示例代码如下：

server.post('/single',upload.single('avatar'),(req,res)=>{
  
 res.send('文件上传成功');
  
});

在上述示例代码中，avatar 是指浏览框的名称 -- <input type="file" name="avatar">

#### • array(filename)

多文件上传时，可以通过array(filename)来进行操作，示例代码如下：

server.post('/single',upload.array('avatar'),(req,res)=>{
  
 res.send('文件上传成功');
  
});

### • 上传文件信息

在通过Multer中间件进行文件上传时，上传文件的信息存储在req.file(单文件上传)或req.files(多文件上传)属性中。

单文件上传时，req.file将返回对象类型的数据；多文件上传时，req.files将返回数组 ( 对象)类型的数据

每一个上传的文件信息包括：

• originalname，上传文件的原始名称（包含扩展名）

• filename，文件上传的名称

•destination ，上传文件存储的目标位置

• path，文件上传后的目标位置及名称，等价于destination + filename

• size，上传文件的尺寸(以字节为单位)

### • 自定义存储规则

通过Multer的diskStorage()方法实现自定义存储规则，其语法结构是：

//定义存储规则
  
var storage= multer.diskStorage({
  
 //用于定义上传文件时的目录规则
  
 destination:function(req,file,cb){
  
 ...
  
 },
  
 //用于定义上传文件时的文件名称规则
  
 filename:function(req,file,cb){
  
 ...
  
 }
  
});
  
   
//使用存储规则
  
multer({storage:storage})

destination属性用于定义上传文件时的目录规则

filename属性用于定义上传文件时的文件名称规则

# 2.UUID

UUID（Universally Unique Identifier）,通用唯一标识符，其目的是为了让分布式系统中所有的元素都能存在唯一的标识信息（主要指文件名称）。

UUID是一组由32位的16进制数字组成的字符串，以短横线进行分隔，形式为8-4-4-4-12

## 2.1 安装

npm install --save uuid

## 2.2 使用

const uuid = require('uuid');

### • v1()方法

v1()方法用于产生基于时间戳的UUID，语法结构是：

uuid.v1()

### • v4()方法

v4()方法用于产生基于随机数的UUID，语法结构是：

uuid.v4()

# 3.FormData对象

FormData对象提供用键/值对表示的表单数据，经过它的数据可以用AJAX提交

## • 创建FormData对象

variable = new FormData()

## • append()方法

append()方法用于实现添加一个新值到键名上，如果键名不存在，则自动创建，其语法结构是：

FormData.append(key,value)

# 4.DataTransfer对象

## • files属性

当页面中不存在<input type="file">时，可通过DataTransfer对象的files 属性来获取被拖放的文件对象，其返回FileList的对象

FileList DataTransfer.files

## • FileList对象

FileList对象通过来源于HTML 表单的<input type="file">元素，也可能来源于用户的拖放操作。

• length属性

length属性用于获取文件的数量， 其语法结构是：

variable = FileList.length

• item()方法

item()方法用于获取指定索引值位置的文件对象，其语法结构是：

File FileList.item(index)

也可以通过FileList[index]来获取指定索引值的文件对象