

02

OPEN ORIENTED

凹凸实验室

# 记移动端图片上传下载

—— 送礼项目 ——

# 送礼项目

02

在很久之前，做了一个**送礼项目的需求**

这个需求涉及到**用户上传图片和下载（保存）图片的功能**

简单介绍我这个项目使用到的一些 **API**



# FileList、file 对象

**FileList** 对象表示用户选择的文件列表，列表中的每一个文件就表示一个 **file** 对象

1、实现图片上传，首先要一个 input 标签，并且指定 type 类型为 file

然后，input 有一个 multiple 属性，表示允许用户传入多个文件

```
<input class="uploadBtn" type="file" multiple>
```

2、input 还有一个 accept 属性，限制上传的文件类型

```
<input class="uploadBtn" type="file" accept="image/*">
```

WebKit 浏览器，延迟弹框

```
<input class="uploadBtn" type="file" accept="image/gif, image/jpeg, image/jpg, image/png">
```

# file 对象

## file 对象的属性：

- name : 文件名，不包含路径
- type : 文件类型（图片类型都会以 image/ 开头）
- size : 文件大小
- lastModified : 文件最后修改的时间

```
uploadBtn.onChange(event => {  
  let files = event.target.files;  
  for (let item of files) {  
    // 文件类型为 image 类型，并且文件大小小于 200KB  
    if (item.type.includes('image/') && item.size < 204800) {  
      console.log(item.name);  
    }  
  }  
});
```

# FileReader 对象

FileReader 对象主要用来把文件读入内存，并且读取文件中的数据。

通过构造函数创建一个 FileReader 对象

```
let reader = new FileReader();
```

## FileReader 对象的方法：

- abort : 中断读取操作
- readAsArrayBuffer : 读取文件内容到 ArrayBuffer 对象中
- readAsBinaryString : 将文件读取为二进制数据
- **readAsDataURL** : 将文件读取为 **data: URL** 格式的字符串
- readAsText : 文件最后修改的时间

# 实现上传图片预览

```
<input class="uploadBtn" type="file" accept="image/gif, image/jpeg, image/jpg, image/png">


let uploadBtn = document.querySelector(".uploadBtn"),
    img = document.querySelector('.preview');

uploadBtn.onChange(() => {
    let files = uploadBtn.files,
        reader = new FileReader();

    if (files && files[0]) {
        reader.onload(ev => {
            img.src = ev.target.result;
        })
        reader.readAsDataURL(files[0]);
    }
});
```

# 存在的问题

iOS 和个别的 Android 手机**竖着拍照上传图片**后，将其绘制到 Canvas 后会发现图片旋转了 90 度，导致显示不正常。

exif.js 库可以获取图片的元数据

旋转角度	参数
0°	1
顺时针90°	6
逆时针90°	8
180°	3

```
EXIF.getData(file, function() {  
    let Orientation = EXIF.getTag(this, 'Orientation');  
});
```

# Blob 对象 - 下载文件

Blob 对象一个容器，用于存放二进制的的数据，有两个属性 size（字节长度） 和 type（MIME类型）  
创建：

1、Blob 对象可以使用 Blob() 构造函数来创建

```
let blob = new Blob(['test'], { type: 'text/plain' });
```

2、Canvas.toBlob() 来创建 Blob 对象，接受三个参数：回调函数、图片类型、图片质量

Blob 对象可以通过 window.URL 对象生成一个网络地址，结合 a 标签的 download 属性实现下载文件

```
let canvas = document.querySelector('.canvas'),
    link = document.querySelector('.link');

canvas.toBlob(blob => {
  // 使用 createObjectURL 生成地址，格式为 blob:null/fd95b806-db11-4f98-b2ce-5eb16b38ba36
  let url = URL.createObjectURL(blob);
  link.download = 'canvas';
  link.href = url;
}, 'image/jpeg', 0.5);

let blob = new Blob(['test'], { type: 'text/plain' });
```



HTML5 进阶系列：文件上传下载

移动端图片上传旋转、压缩的解决方案

THANKS  
FOR YOUR WATCHING