

uniapp 安卓文件操作原生插件

目 录

使用说明	1
使用方法	1.1
更新日志	1.2
插件方法	2
从sd卡读取文件	2.1
保存文件到sd卡	2.2
获取文件大小	2.3
递归取得文件夹大小	2.4
格式化文件大小	2.5
递归求取目录文件个数	2.6
根据扩展名搜索sdcard文件	2.7
根据关键字搜索sdcard文件	2.8
获取目录下所有的文件	2.9
复制文件	2.10
移动文件	2.11
删除文件或目录	2.12
写文件	2.13
获取路径的文件名	2.14
读取文件(按行读取)	2.15
读取整个文件	2.16
远程下载文件到本地	2.17
获取U盘路径 v1.1.0	2.18
获取重定向真实url	2.19
批量下载文件(可下载重定向地址)	2.20
获取项目安装路径 v1.4.0	2.21
获取文件存储权限 v1.4.0	2.22
压缩文件夹 v1.5.0	2.23
压缩文件集合 v1.5.0	2.24
解压缩 v1.5.0	2.25
压缩包是否加密 v1.5.0	2.26
监听U盘权限 v1.6.0	2.27
取消监听U盘权限 v1.6.0	2.28
创建文件夹 v1.7.0	2.29
选择文件系统弹窗 v1.8.0	2.30

检查是否有“所有文件”访问权限 v1.8.0	2.31
跳转到所有文件访问权限页面 v1.8.0	2.32
判断文件或文件夹是否存在 v1.9.0	2.33
写入大文件 v1.10.0	2.34
base64保存成图片 v1.10.0	2.35
保存为二进制文件 v1.11.0	2.36

使用方法

介绍

安卓文件操作原生插件，读写文件，文件下载等

联系作者

关注微信公众号可联系作者



插件地址

<https://ext.dcloud.net.cn/plugin?id=14156>

示例文件

如果下载的文件不是zip文件，请将扩展名改为zip即可

权限

- android.permission.INTERNET
- android.permission.WRITE_EXTERNAL_STORAGE
- android.permission.READ_EXTERNAL_STORAGE

用法

在需要使用插件的页面加载以下代码

```
const module = uni.requireNativePlugin("leven-file-FileModule");
```

页面内容

```
<template>
  <view>
    <uni-card title="文件类">
      <button type="primary" @click="getFileFromSdcard">从sd卡取文件</button>
      <button type="primary" @click="saveFileToSdcard">保存文件到sd</button>
      <button type="primary" @click="getFileSizes">获取文件大小</button>
      <button type="primary" @click="getDirSize">递归取得文件夹大小</button>
      <button type="primary" @click="formatFileSize">格式化文件大小</button>
      <button type="primary" @click="getFileListCount">递归求取目录文件个数</button>
      <button type="primary" @click="searchSdCardExtFile">根据扩展名搜索sdcard文件</button>
      <button type="primary" @click="searchKeywordFile">根据关键字搜索sdcard文件</button>
      <button type="primary" @click="getDirAllFiles">获取目录下所有的文件</button>
      <button type="primary" @click="copyFile">复制文件</button>
      <button type="primary" @click="deleteFile">删除文件或目录</button>
      <button type="primary" @click="writeFile">写文件</button>
      <button type="primary" @click="getPathFileName">获取路径的文件名</button>
      <button type="primary" @click="readFileLine">读取文件(按行读取)</button>
      <button type="primary" @click="readFile">读取整个文件</button>
      <button type="primary" @click="downloadFile">下载文件</button>
    </uni-card>
  </view>
</template>

<script>
  import {
    requestAndroidPermission
  } from "@utils/permission.js"
  const module = uni.requireNativePlugin("leven-file-FileModule");
  export default {
    data() {
      return {}
    },
    mounted() {
      // 动态开启应用权限
      this.openAndroidPermission();
    },
    methods: {
      // 从sd卡取文件
      getFileFromSdcard() {
        module.getFileFromSdcard({
```

```

        filename: "leven_file/a/1.txt"
    }, res => {
        console.log(res)
    })
},
// 保存文件到sd
saveFileToSdcard() {
    module.saveFileToSdcard({
        filename: "leven_file/a/save1.txt",
        content: "这是一条测试内容"
    }, res => {
        console.log(res)
    })
},
// 获取文件大小
getFileSizes() {
    module.getFileSizes({
        filePath: "/storage/emulated/0/leven_file/a/1.txt",
    }, res => {
        console.log(res)
    })
},
// 递归取得文件夹大小
getDirSize() {
    module.getDirSize({
        dirPath: "/storage/emulated/0/qqmusic",
    }, res => {
        console.log(res)
    })
},
// 格式化文件大小
formatFileSize() {
    module.formatFileSize({
        size: 3456789,
    }, res => {
        console.log(res)
    })
},
// 递归求取目录文件个数
getFileListCount() {
    module.getFileListCount({
        dirPath: "/storage/emulated/0/qqmusic",
    }, res => {
        console.log(res)
    })
},
// 根据扩展名搜索sdcard文件
searchSdCardExtFile() {
    module.searchSdCardExtFile({
        dirPath: "/storage/emulated/0/qqmusic",

```

```

    ext: ['png', 'txt'],
  }, res => {
    console.log(res)
  })
},
// 根据关键字搜索sdcard文件
searchKeywordFile() {
  module.searchKeywordFile({
    dirPath: "/storage/emulated/0/qqmusic",
    keyword: "1",
  }, res => {
    console.log(res)
  })
},
// 获取目录下所有的文件
getDirAllFiles() {
  module.getDirAllFiles({
    dirPath: "/storage/emulated/0/qqmusic"
  }, res => {
    console.log(res)
  })
},
// 复制文件
copyFile() {
  module.copyFile({
    fromFilename: "/storage/emulated/0/qqmusic/easter_egg_config_res/easter_
    toFilename: "/storage/emulated/0/leven_file/b/egg_51.png"
  }, res => {
    console.log(res)
  })
},
// 删除文件或目录
deleteFile() {
  module.deleteFile({
    filePath: "/storage/emulated/0/leven_file/a/egg_51.png"
  }, res => {
    console.log(res)
  })
},
// 写文件
writeFile() {
  module.writeFile({
    filename: "/storage/emulated/0/leven_file/a/test_write.txt",
    content: "123456\n",
    append: true
  }, res => {
    console.log(res)
  })
},
// 获取路径的文件名

```

```

getPathFileName() {
  module.getPathFileName({
    filePath: "/storage/emulated/0/leven_file/a/test_write.txt"
  }, res => {
    console.log(res)
  })
},
// 读取文件(按行读取)
readFileLine() {
  module.readFileLine({
    filename: "/storage/emulated/0/leven_file/a/test_write.txt",
    startLine: 1,
    lineCount: 1
  }, res => {
    console.log(res)
  })
},
// 读取整个文件
readFile() {
  module.readFile({
    filename: "/storage/emulated/0/leven_file/a/test_write.txt",
    removeWrap: false
  }, res => {
    console.log(res)
  })
},
// 下载文件
downloadFile() {
  module.downloadFile({
    url: "http://www.yeyuboke.com/svgas/1.svga",
    saveDir: "/storage/emulated/0/leven_file/"
  }, res => {
    console.log(res)
  })
},
showToast(content) {
  uni.showToast({
    title: content,
    icon: "none"
  })
},
// 开启应用权限
openAndroidPermission() {
  let that = this;
  requestAndroidPermission("android.permission.READ_EXTERNAL_STORAGE").then(
    that.showToast(res)
  )
  requestAndroidPermission("android.permission.WRITE_EXTERNAL_STORAGE").then(
    that.showToast(res)
  )
}

```

使用方法

```
    }  
  }  
}  
</script>  
  
<style>  
  
</style>
```

更新日志

2025-04-23 v1.11.0

- [新增] 新增接口【[保存二进制文件](#)】

2025-04-21 v1.10.0

- [新增] 新增接口【[写入大文件](#)】
- [新增] 新增接口【[base64保存成图片](#)】

2025-03-06 v1.9.0

- [新增] 新增接口【[判断文件或文件夹是否存在](#)】

2024-10-28 v1.8.2

- [优化] 修复安卓10无法获取文件问题

2024-08-31 v1.8.1

- [优化] 优化下载文件因服务器压缩下载失败问题

2024-05-20 v1.8.0

- [新增]新增接口【[选择文件系统弹窗](#)】
- [新增]新增接口【[检查是否有“所有文件”访问权限](#)】
- [新增]新增接口【[跳转到所有文件访问权限页面](#)】

2024-02-22 v1.7.0

- [优化]压缩文件夹和压缩文件集合，如果目录不存则创建目录

- [新增] 新增接口【[创建文件夹](#)】

2024-01-19 v1.6.1

- [优化] 修复安卓高版本获取U盘路径报错的bug

2024-01-16 v1.6.0

- [优化] 修复获取u盘路径u盘拔出时未返回数据的bug
- [新增] 新增接口【[监听U盘权限](#)】
- [新增] 新增接口【[取消监听U盘权限](#)】

2024-01-15 v1.5.3

- [优化] 调用获取u盘路径，u盘拔出时没有返回数据的bug
- [优化] U盘插入时增加动态获取U盘权限弹窗

2024-01-10 v1.5.2

- [优化] 获取目录下所有的文件新增参数 `type` （获取的文件类型）【[详情](#)】
- [修复] 获取u盘路径接口u盘插拔时闪退的bug

2023-12-20 v1.5.1

- 【优化】获取项目的安装路径新增返回数据 `项目的static路径` 和 `项目的www路径` 【[查看详情](#)】

2023-11-22 v1.5.0

- 【新增】[压缩文件夹](#)
- 【新增】[压缩文件集合](#)
- 【新增】[解压缩](#)
- 【新增】[压缩包是否加密](#)

2023-11-07 v1.4.0

【新增】[获取项目安装路径](#)

【新增】[获取文件存储权限](#)

2023-10-11 v1.3.0

【新增】获取重定向真实url [【转到】](#)

【新增】批量下载文件(可下载重定向地址) [【转到】](#)

2023-09-12 v1.2.1

【修复】修复文件下载https地址判断错误的bug

2023-09-10 v1.2.0

【新增】文件移动 [【转到】](#)

2023-09-06 v1.1.2

【优化】优化文件复制

2023-09-04 v1.1.1

1. [优化]根据扩展名搜索文件新增入参：minSize(最小字节)、maxSize(最大字节)
2. [优化]根据关键字搜索文件新增入参：minSize(最小字节)、maxSize(最大字节)

2023-08-22 v1.1.0

1. 新增方法-获取U盘路径 [【转到】](#)
2. 递归求取目录文件个数新增参数 `child` , 是否递归子目录
3. 根据扩展名搜索sdcard文件新增参数 `child` , 是否递归子目录
4. 根据关键字搜索sdcard文件新增参数 `child` , 是否递归子目录
5. 获取目录下所有的文件新增参数 `child` , 是否递归子目录

更新日志

2023-08-19

v1.0.0

首次发布

从sd卡读取文件

方法名

getFileFromSdcard

用法

- 用法如下：

```
module.getFileFromSdcard({
  filename: "leven_file/a/1.txt"
}, res => {
  console.log(res)
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
filename	String	是	无	sd卡文件路径

回调

- 示例

```
{
  "data": {
    "content": "时间:2022-10-12---error:请将下面日志发送给合合信息技术支持：com.china"
  },
  "message": "",
  "code": 0
}
```

}
...

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

保存文件到sd卡

- 用法如下：

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
filename	String	是	无	sd卡文件路径
content	String	是	无	保存的内容

- 示例

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.filePath	String	保存文件的绝对路径
code	Integer	返回类型，0.成功，其他：失败

获取文件大小

方法名

getFileSizes

用法

- 用法如下：

```
module.getFileSizes({
  filePath: "/storage/emulated/0/leven_file/a/1.txt",
}, res => {
  console.log(res)
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
filePath	String	是	无	sd卡文件绝对路径

回调

- 示例

```
{
  "data": {
    "size": 534,
    "sizeFormat": "534.00B"
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示

获取文件大小

参数名	参数类型	参数描述
data	Object	数据对象
data.size	Integer	文件大小字节
data.sizeFormat	String	文件大小格式化数据
code	Integer	返回类型，0.成功，其他：失败

递归取得文件夹大小

方法名

getDirSize

用法

- 用法如下：

```
module.getDirSize({
  dirPath: "/storage/emulated/0/qqmusic",
}, res => {
  console.log(res)
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
dirPath	String	是	无	sd卡文件夹绝对路径

回调

- 示例

```
{
  "data": {
    "size": 102022309,
    "sizeFormat": "97.30M"
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示

递归取得文件夹大小

参数名	参数类型	参数描述
data	Object	数据对象
data.size	Integer	文件大小字节
data.sizeFormat	String	文件大小格式化数据
code	Integer	返回类型，0.成功，其他：失败

格式化文件大小

方法名

formatFileSize

用法

- 用法如下：

```
module.formatFileSize({
  size: 3456789,
}, res => {
  console.log(res)
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
size	Integer	是	无	字节数

回调

- 示例

```
{
  "data": {
    "sizeFormat": "3.30M"
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象

格式化文件大小

参数名	参数类型	参数描述
data.sizeFormat	String	格式化数据
code	Integer	返回类型，0.成功，其他：失败

递归求取目录文件个数

方法名

getFileListCount

用法

- 用法如下：

```
module.getFileListCount({
  dirPath: "/storage/emulated/0/qqmusic",
}, res => {
  console.log(res)
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
dirPath	String	是	无	文件夹绝对路径
child v1.1.0	Boolean	否	true	是否递归到子目录

回调

- 示例

```
{
  "data": {
    "count": 739
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象

递归求取目录文件个数

参数名	参数类型	参数描述
data.count	Integer	总数量
code	Integer	返回类型，0.成功，其他：失败

根据扩展名搜索sdcard文件

方法名

searchSdCardExtFile

用法

- 用法如下：

```
module.searchSdCardExtFile({
  dirPath: "/storage/emulated/0/qqmusic",
  ext: ['png', 'txt'],
}, res => {
  console.log(res)
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
dirPath	String	是	无	文件夹绝对路径
ext	Array	是	无	扩展名集合
child v1.1.0	Boolean	否	true	是否递归到子目录
minSize v1.1.1	int	否	0	最小字节值
maxSize v1.1.1	int	否	0	最大字节值

minSize和maxSize不传或全部为0则不通过文件大小筛选

回调

- 示例

```
{
  "data": {
```

根据扩展名搜索sdcard文件

[illegible]

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.list	Array	搜索的文件列表
code	Integer	返回类型，0.成功，其他：失败

根据关键字搜索sdcard文件

方法名

searchKeywordFile

用法

- 用法如下：

```
module.searchKeywordFile({
  dirPath: "/storage/emulated/0/qqmusic",
  keyword: "1",
}, res => {
  console.log(res)
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
dirPath	String	是	无	文件夹绝对路径
keyword	String	是	无	要搜索的关键字
child v1.1.0	Boolean	否	true	是否递归到子目录
minSize v1.1.1	int	否	0	最小字节值
maxSize v1.1.1	int	否	0	最大字节值

minSize和maxSize不传或全部为0则不通过文件大小筛选

回调

- 示例

```
{
  "data": {
```

```
"list": [  
  "/storage/emulated/0/qqmusic/eup/sys_log_1670762033268.txt",  
  "/storage/emulated/0/qqmusic/eup/jni_log_1670762033268.txt",  
  "/storage/emulated/0/qqmusic/eup/sys_log_1670762032778.txt",  
  "/storage/emulated/0/qqmusic/eup/jni_log_1670762032778.txt",  
  "/storage/emulated/0/qqmusic/firstPiece/QQMusicSource.A689A220871C",  
  "/storage/emulated/0/qqmusic/cache/258470226.mggc1",  
  "/storage/emulated/0/qqmusic/ubc/1588464000000.ubc",  
  "/storage/emulated/0/qqmusic/ubc/1590116400000.ubc",  
  "/storage/emulated/0/qqmusic/ubc/1590120000000.ubc",  
  "/storage/emulated/0/qqmusic/ubc/1590145200000.ubc",  
  "/storage/emulated/0/qqmusic/ubc/1590202800000.ubc",  
  "/storage/emulated/0/qqmusic/ubc/1590217200000.ubc",  
  "/storage/emulated/0/qqmusic/ubc/1590228000000.ubc",  
  "/storage/emulated/0/qqmusic/ubc/1590364800000.ubc",  
  "/storage/emulated/0/qqmusic/ubc/1590753600000.ubc",  
  "/storage/emulated/0/qqmusic/ubc/1590796800000.ubc",  
  "/storage/emulated/0/qqmusic/ubc/1590836400000.ubc",  
  "/storage/emulated/0/qqmusic/qrc/-443163595.producer",  
  "/storage/emulated/0/qqmusic/qrc/-443163595.qrc",  
  "/storage/emulated/0/qqmusic/qrc/-443163595.ex",  
  "/storage/emulated/0/qqmusic/qrc/-1240629447.producer",  
  "/storage/emulated/0/qqmusic/qrc/-1240629447.qrc",  
  "/storage/emulated/0/qqmusic/qrc/-1240629447.ex",  
  "/storage/emulated/0/qqmusic/qrc/-1220964185.producer",  
  "/storage/emulated/0/qqmusic/qrc/-1220964185.qrc",  
  "/storage/emulated/0/qqmusic/qrc/-1220964185.ex",  
  "/storage/emulated/0/qqmusic/qrc/1286534244.producer",  
  "/storage/emulated/0/qqmusic/qrc/1286534244.qrc",  
  "/storage/emulated/0/qqmusic/qrc/1286534244.ex",  
  "/storage/emulated/0/qqmusic/qrc/-1739244798.producer",  
  "/storage/emulated/0/qqmusic/qrc/-1739244798.qrc",  
  "/storage/emulated/0/qqmusic/qrc/-1739244798.ex",  
  "/storage/emulated/0/qqmusic/qrc/143040042.producer",  
  "/storage/emulated/0/qqmusic/qrc/143040042.qrc",  
  "/storage/emulated/0/qqmusic/qrc/143040042.ex",  
  "/storage/emulated/0/qqmusic/qrc/-1591745311.qrc",  
  "/storage/emulated/0/qqmusic/qrc/-1591745311.ex",  
  "/storage/emulated/0/qqmusic/qrc/-10780052.producer",  
  "/storage/emulated/0/qqmusic/qrc/-10780052.qrc",  
  "/storage/emulated/0/qqmusic/qrc/-10780052.ex",  
  "/storage/emulated/0/qqmusic/qrc/1205680110.qrc",  
  "/storage/emulated/0/qqmusic/qrc/1205680110.ex",  
  "/storage/emulated/0/qqmusic/qrc/-1186345238.producer",  
  "/storage/emulated/0/qqmusic/qrc/-1186345238.qrc",  
  "/storage/emulated/0/qqmusic/qrc/-1186345238.ex",  
  "/storage/emulated/0/qqmusic/qrc/-1363939784.producer",  
  "/storage/emulated/0/qqmusic/qrc/-1363939784.qrc",  
  "/storage/emulated/0/qqmusic/qrc/-1363939784.ex",  
  "/storage/emulated/0/qqmusic/qrc/1150628010.producer",  
]
```

```
"/storage/emulated/0/qqmusic/qrc/1150628010.qrc",
"/storage/emulated/0/qqmusic/qrc/1150628010.ex",
"/storage/emulated/0/qqmusic/qrc/-1084459363.producer",
"/storage/emulated/0/qqmusic/qrc/-1084459363.qrc",
"/storage/emulated/0/qqmusic/qrc/-1084459363.ex",
"/storage/emulated/0/qqmusic/qrc/1357645499.producer",
"/storage/emulated/0/qqmusic/qrc/1357645499.qrc",
"/storage/emulated/0/qqmusic/qrc/1357645499.ex",
"/storage/emulated/0/qqmusic/qrc/-1482197463.producer",
"/storage/emulated/0/qqmusic/qrc/-1482197463.qrc",
"/storage/emulated/0/qqmusic/qrc/-1482197463.ex",
"/storage/emulated/0/qqmusic/qrc/1163307333.producer",
"/storage/emulated/0/qqmusic/qrc/1163307333.qrc",
"/storage/emulated/0/qqmusic/qrc/1163307333.romaqr",
"/storage/emulated/0/qqmusic/qrc/1163307333.ex",
"/storage/emulated/0/qqmusic/qrc/1561232873.qrc",
"/storage/emulated/0/qqmusic/qrc/1561232873.romaqr",
"/storage/emulated/0/qqmusic/qrc/1561232873.ex",
"/storage/emulated/0/qqmusic/qrc/-623813994.producer",
"/storage/emulated/0/qqmusic/qrc/-623813994.qrc",
"/storage/emulated/0/qqmusic/qrc/-623813994.ex",
"/storage/emulated/0/qqmusic/qrc/1869459699.producer",
"/storage/emulated/0/qqmusic/qrc/1869459699.qrc",
"/storage/emulated/0/qqmusic/qrc/1869459699.ex",
"/storage/emulated/0/qqmusic/qrc/1193819098.producer",
"/storage/emulated/0/qqmusic/qrc/1193819098.qrc",
"/storage/emulated/0/qqmusic/qrc/1193819098.ex",
"/storage/emulated/0/qqmusic/qrc/-1407687359.producer",
"/storage/emulated/0/qqmusic/qrc/-1407687359.lrc",
"/storage/emulated/0/qqmusic/qrc/-1407687359.ex",
"/storage/emulated/0/qqmusic/qrc/-662524391.producer",
"/storage/emulated/0/qqmusic/qrc/-662524391.qrc",
"/storage/emulated/0/qqmusic/qrc/-662524391.ex",
"/storage/emulated/0/qqmusic/qrc/919436570.producer",
"/storage/emulated/0/qqmusic/qrc/919436570.qrc",
"/storage/emulated/0/qqmusic/qrc/919436570.ex",
"/storage/emulated/0/qqmusic/qrc/-1591745311.producer",
"/storage/emulated/0/qqmusic/qrc/-1381016494.qrc",
"/storage/emulated/0/qqmusic/qrc/-1381016494.ex",
"/storage/emulated/0/qqmusic/qrc/-313153247.qrc",
"/storage/emulated/0/qqmusic/qrc/-313153247.ex",
"/storage/emulated/0/qqmusic/qrc/1716522843.producer",
"/storage/emulated/0/qqmusic/qrc/1716522843.qrc",
"/storage/emulated/0/qqmusic/qrc/1716522843.ex",
"/storage/emulated/0/qqmusic/qrc/-1620632128.producer",
"/storage/emulated/0/qqmusic/qrc/-1620632128.qrc",
"/storage/emulated/0/qqmusic/qrc/-1620632128.ex",
"/storage/emulated/0/qqmusic/qrc/851840155.qrc",
"/storage/emulated/0/qqmusic/qrc/851840155.producer",
"/storage/emulated/0/qqmusic/qrc/851840155.ex",
```

```
"/storage/emulated/0/qqmusic/qrc/1444357536.qrc",
"/storage/emulated/0/qqmusic/qrc/1444357536.ex",
"/storage/emulated/0/qqmusic/qrc/1444357536.producer",
"/storage/emulated/0/qqmusic/qrc/-913271974.producer",
"/storage/emulated/0/qqmusic/qrc/-913271974.qrc",
"/storage/emulated/0/qqmusic/qrc/-913271974.ex",
"/storage/emulated/0/qqmusic/qrc/2068248319.qrc",
"/storage/emulated/0/qqmusic/qrc/2068248319.producer",
"/storage/emulated/0/qqmusic/qrc/2068248319.ex",
"/storage/emulated/0/qqmusic/qrc/-146713447.producer",
"/storage/emulated/0/qqmusic/qrc/-146713447.qrc",
"/storage/emulated/0/qqmusic/qrc/-146713447.ex",
"/storage/emulated/0/qqmusic/qrc/-709391483.producer",
"/storage/emulated/0/qqmusic/qrc/-709391483.qrc",
"/storage/emulated/0/qqmusic/qrc/-709391483.romaqr",
"/storage/emulated/0/qqmusic/qrc/-709391483.ex",
"/storage/emulated/0/qqmusic/qrc/-1629231329.producer",
"/storage/emulated/0/qqmusic/qrc/-1629231329.qrc",
"/storage/emulated/0/qqmusic/qrc/-1629231329.ex",
"/storage/emulated/0/qqmusic/qrc/-1466209437.qrc",
"/storage/emulated/0/qqmusic/qrc/-1466209437.translrc",
"/storage/emulated/0/qqmusic/qrc/-1466209437.ex",
"/storage/emulated/0/qqmusic/qrc/-1466209437.producer",
"/storage/emulated/0/qqmusic/qrc/1454686369.producer",
"/storage/emulated/0/qqmusic/qrc/1454686369.qrc",
"/storage/emulated/0/qqmusic/qrc/1454686369.ex",
"/storage/emulated/0/qqmusic/qrc/1501934993.producer",
"/storage/emulated/0/qqmusic/qrc/1501934993.qrc",
"/storage/emulated/0/qqmusic/qrc/1501934993.ex",
"/storage/emulated/0/qqmusic/qrc/1752180066.qrc",
"/storage/emulated/0/qqmusic/qrc/1752180066.producer",
"/storage/emulated/0/qqmusic/qrc/1752180066.ex",
"/storage/emulated/0/qqmusic/qrc/1122439825.qrc",
"/storage/emulated/0/qqmusic/qrc/1122439825.producer",
"/storage/emulated/0/qqmusic/qrc/1122439825.ex",
"/storage/emulated/0/qqmusic/qrc/1252567.qrc",
"/storage/emulated/0/qqmusic/qrc/1252567.producer",
"/storage/emulated/0/qqmusic/qrc/1252567.ex",
"/storage/emulated/0/qqmusic/qrc/403578751.producer",
"/storage/emulated/0/qqmusic/qrc/403578751.qrc",
"/storage/emulated/0/qqmusic/qrc/403578751.ex",
"/storage/emulated/0/qqmusic/qrc/-173668749.qrc",
"/storage/emulated/0/qqmusic/qrc/-173668749.ex",
"/storage/emulated/0/qqmusic/qrc/-173668749.producer",
"/storage/emulated/0/qqmusic/qrc/-1083098069.qrc",
"/storage/emulated/0/qqmusic/qrc/-1083098069.translrc",
"/storage/emulated/0/qqmusic/qrc/1301773512.producer",
"/storage/emulated/0/qqmusic/qrc/1301773512.qrc",
"/storage/emulated/0/qqmusic/qrc/1301773512.ex",
"/storage/emulated/0/qqmusic/qrc/-118179094.qrc",
```

```
"/storage/emulated/0/qqmusic/qrc/1495717187.qrc",
"/storage/emulated/0/qqmusic/qrc/-813120802.qrc",
"/storage/emulated/0/qqmusic/qrc/-813120802.ex",
"/storage/emulated/0/qqmusic/qrc/-1346537094.producer",
"/storage/emulated/0/qqmusic/qrc/1880551055.translrc",
"/storage/emulated/0/qqmusic/qrc/1880551055.ex",
"/storage/emulated/0/qqmusic/qrc/-1910762404.producer",
"/storage/emulated/0/qqmusic/qrc/-1910762404.ex",
"/storage/emulated/0/qqmusic/qrc/1851345085.ex",
"/storage/emulated/0/qqmusic/qrc/1851345085.producer",
"/storage/emulated/0/qqmusic/qrc/-1913778505.producer",
"/storage/emulated/0/qqmusic/qrc/-1913778505.qrc",
"/storage/emulated/0/qqmusic/qrc/-1913778505.translrc",
"/storage/emulated/0/qqmusic/qrc/-1913778505.ex",
"/storage/emulated/0/qqmusic/qrc/929947231.producer",
"/storage/emulated/0/qqmusic/qrc/453414570.qrc",
"/storage/emulated/0/qqmusic/qrc/453414570.translrc",
"/storage/emulated/0/qqmusic/qrc/453414570.ex",
"/storage/emulated/0/qqmusic/qrc/453414570.producer",
"/storage/emulated/0/qqmusic/qrc/-1799386311.qrc",
"/storage/emulated/0/qqmusic/qrc/-1799386311.ex",
"/storage/emulated/0/qqmusic/qrc/-1799386311.producer",
"/storage/emulated/0/qqmusic/qrc/1860946097.producer",
"/storage/emulated/0/qqmusic/qrc/-840070281.qrc",
"/storage/emulated/0/qqmusic/qrc/-840070281.ex",
"/storage/emulated/0/qqmusic/qrc/-2125738814.qrc",
"/storage/emulated/0/qqmusic/qrc/-2125738814.ex",
"/storage/emulated/0/qqmusic/qrc/2107964463.qrc",
"/storage/emulated/0/qqmusic/qrc/2107964463.romaqr",
"/storage/emulated/0/qqmusic/qrc/2107964463.ex",
"/storage/emulated/0/qqmusic/qrc/333490106.qrc",
"/storage/emulated/0/qqmusic/qrc/1565231009.qrc",
"/storage/emulated/0/qqmusic/qrc/-581495485.qrc",
"/storage/emulated/0/qqmusic/qrc/-581495485.producer",
"/storage/emulated/0/qqmusic/qrc/-1509578113.qrc",
"/storage/emulated/0/qqmusic/qrc/-1509578113.translrc",
"/storage/emulated/0/qqmusic/qrc/-1509578113.ex",
"/storage/emulated/0/qqmusic/qrc/-886849916.ex",
"/storage/emulated/0/qqmusic/qrc/-211127315.ex",
"/storage/emulated/0/qqmusic/qrc/-1261335359.qrc",
"/storage/emulated/0/qqmusic/qrc/-1261335359.ex",
"/storage/emulated/0/qqmusic/qrc/-1261335359.producer",
"/storage/emulated/0/qqmusic/qrc/816713018.qrc",
"/storage/emulated/0/qqmusic/qrc/-413340894.producer",
"/storage/emulated/0/qqmusic/qrc/1641164731.qrc",
"/storage/emulated/0/qqmusic/qrc/1641164731.ex",
"/storage/emulated/0/qqmusic/qrc/-1466945852.ex",
"/storage/emulated/0/qqmusic/qrc/-1315749965.qrc",
"/storage/emulated/0/qqmusic/qrc/-1080392956.producer",
"/storage/emulated/0/qqmusic/qrc/-1080392956.ex",
```

```
"/storage/emulated/0/qqmusic/qrc/-1545569796.producer",
"/storage/emulated/0/qqmusic/qrc/-1924153637.producer",
"/storage/emulated/0/qqmusic/qrc/-1375015985.qrc",
"/storage/emulated/0/qqmusic/qrc/-1375015985.ex",
"/storage/emulated/0/qqmusic/qrc/-1375015985.producer",
"/storage/emulated/0/qqmusic/qrc/1377521614.producer",
"/storage/emulated/0/qqmusic/qrc/1315909736.producer",
"/storage/emulated/0/qqmusic/qrc/595244175.producer",
"/storage/emulated/0/qqmusic/qrc/595244175.qrc",
"/storage/emulated/0/qqmusic/qrc/595244175.ex",
"/storage/emulated/0/qqmusic/qrc/1612644463.producer",
"/storage/emulated/0/qqmusic/qrc/1612644463.qrc",
"/storage/emulated/0/qqmusic/qrc/1577269239.qrc",
"/storage/emulated/0/qqmusic/qrc/1557808398.qrc",
"/storage/emulated/0/qqmusic/qrc/265452160.romaqr",
"/storage/emulated/0/qqmusic/qrc/265452160.ex",
"/storage/emulated/0/qqmusic/qrc/-1976223131.qrc",
"/storage/emulated/0/qqmusic/qrc/-1976223131.ex",
"/storage/emulated/0/qqmusic/qrc/-1076792942.producer",
"/storage/emulated/0/qqmusic/qrc/-1076792942.qrc",
"/storage/emulated/0/qqmusic/qrc/-1076792942.ex",
"/storage/emulated/0/qqmusic/qrc/-1189120660.producer",
"/storage/emulated/0/qqmusic/qrc/563142729.producer",
"/storage/emulated/0/qqmusic/qrc/563142729.qrc",
"/storage/emulated/0/qqmusic/qrc/-1818292625.producer",
"/storage/emulated/0/qqmusic/qrc/-1818292625.ex",
"/storage/emulated/0/qqmusic/qrc/2133454120.qrc",
"/storage/emulated/0/qqmusic/qrc/2133454120.ex",
"/storage/emulated/0/qqmusic/qrc/785201891.ex",
"/storage/emulated/0/qqmusic/qrc/-2020090117.producer",
"/storage/emulated/0/qqmusic/qrc/-655885014.qrc",
"/storage/emulated/0/qqmusic/qrc/2012701354.producer",
"/storage/emulated/0/qqmusic/qrc/2012701354.qrc",
"/storage/emulated/0/qqmusic/qrc/2012701354.ex",
"/storage/emulated/0/qqmusic/qrc/1253047543.producer",
"/storage/emulated/0/qqmusic/qrc/1253047543.qrc",
"/storage/emulated/0/qqmusic/qrc/1253047543.ex",
"/storage/emulated/0/qqmusic/qrc/-1507317522.qrc",
"/storage/emulated/0/qqmusic/qrc/-1507317522.ex",
"/storage/emulated/0/qqmusic/qrc/-1536199596.producer",
"/storage/emulated/0/qqmusic/qrc/-1536199596.qrc",
"/storage/emulated/0/qqmusic/qrc/-1536199596.ex",
"/storage/emulated/0/qqmusic/qrc/1294676680.qrc",
"/storage/emulated/0/qqmusic/qrc/1294676680.ex",
"/storage/emulated/0/qqmusic/qrc/1160886602.producer",
"/storage/emulated/0/qqmusic/qrc/1160886602.qrc",
"/storage/emulated/0/qqmusic/qrc/1160886602.ex",
"/storage/emulated/0/qqmusic/qrc/1628982744.producer",
"/storage/emulated/0/qqmusic/qrc/1628982744.qrc",
"/storage/emulated/0/qqmusic/qrc/1801152420.producer",
```

根据关键字搜索sdcard文件

```
        "/storage/emulated/0/qqmusic/qrc/1801152420.qrc",
        "/storage/emulated/0/qqmusic/qrc/1117111022.qrc",
        "/storage/emulated/0/qqmusic/qrc/1117111022.translrc",
        "/storage/emulated/0/qqmusic/qrc/1117111022.ex",
        "/storage/emulated/0/qqmusic/qrc/1033094693.producer",
        "/storage/emulated/0/qqmusic/qrc/791558813.qrc",
        "/storage/emulated/0/qqmusic/qrc/1819665138.qrc",
        "/storage/emulated/0/qqmusic/qrc/1819665138.ex",
        "/storage/emulated/0/qqmusic/qrc/1561232873.producer",
        "/storage/emulated/0/qqmusic/qrc/-1575273992.producer",
        "/storage/emulated/0/qqmusic/qrc/-1575273992.ex",
        "/storage/emulated/0/qqmusic/qrc/-75972116.producer",
        "/storage/emulated/0/qqmusic/qrc/-75972116.lrc",
        "/storage/emulated/0/qqmusic/qrc/-415346838.qrc",
        "/storage/emulated/0/qqmusic/qrc/-415346838.ex",
        "/storage/emulated/0/qqmusic/qrc/-1554379218.producer",
        "/storage/emulated/0/qqmusic/qrc/-1554379218.qrc",
        "/storage/emulated/0/qqmusic/qrc/-295771321.qrc",
        "/storage/emulated/0/qqmusic/qrc/1949272812.qrc",
        "/storage/emulated/0/qqmusic/qrc/-1279495095.ex",
    ]
},
"message": "",
"code": 0
}
```

回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.list	Array	搜索的文件列表
code	Integer	返回类型，0.成功，其他：失败

获取目录下所有的文件

方法名

getDirAllFiles

用法

- 用法如下：

```
module.getDirAllFiles({
  dirPath: "/storage/emulated/0/qqmusic"
}, res => {
  console.log(res)
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
dirPath	String	是	无	文件夹绝对路径
child v1.1.0	Boolean	否	true	是否递归到子目录
type v1.5.2	Integer	否	1	获取文件类型， 1.所有文件，2.只获取文件夹，3.只获取文件

回调

- 示例

```
{
  "data": {
    "list": [
      "/storage/emulated/0/qqmusic/safemode.slog",
      "/storage/emulated/0/qqmusic/config/storage.cfg",
      "/storage/emulated/0/qqmusic/eup/sys_log_1670762033268.txt",
      "/storage/emulated/0/qqmusic/eup/jni_log_1670762033268.txt",
      "/storage/emulated/0/qqmusic/eup/sys_log_1670762032778.txt",
    ]
  }
}
```

获取目录下所有的文件

```
        "/storage/emulated/0/qqmusic/eup/jni_log_1670762032778.txt",
        "/storage/emulated/0/qqmusic/vip_center/.nomedia",
        "/storage/emulated/0/qqmusic/firstPiece/journal",
        "/storage/emulated/0/qqmusic/firstPiece/QQMusicSource.A689A220871C4F5A94",
        "/storage/emulated/0/qqmusic/moments/.nomedia",
        "/storage/emulated/0/qqmusic/externalDB/extra_player_process_db",
        "/storage/emulated/0/qqmusic/externalDB/extra_player_process_db-journal",
        "/storage/emulated/0/qqmusic/externalDB/extra_main_process_db",
        "/storage/emulated/0/qqmusic/externalDB/extra_main_process_db-journal",
        "/storage/emulated/0/qqmusic/fonts/hkljh.ttf.zip",
        "/storage/emulated/0/qqmusic/fonts/SHARE_PEAK_CRAD_FONT_KEY.otf",
        "/storage/emulated/0/qqmusic/fonts/BIG_FZLTDHK_FONT_KEY.ttf",
        "/storage/emulated/0/qqmusic/song/.qqmusic.mp3.bgm",
        "/storage/emulated/0/qqmusic/splash/.nomedia",
    ]
},
"message": "",
"code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.list	Array	文件列表
code	Integer	返回类型，0.成功，其他：失败

复制文件

方法名

copyFile

用法

- 用法如下：

```
module.copyFile({
  fromFilename: "/storage/emulated/0/qqmusic/easter_egg_config_res/easter_egg_
  toFilename: "/storage/emulated/0/leven_file/a/egg_51.png"
}, res => {
  console.log(res)
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
fromFilename	String	是	无	要复制的文件路径
toFilename	String	是	无	目的文件路径

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

移动文件

方法名 **v1.2.0**

`moveFile`

此方法会移动成功后会删除移动前的文件

用法

- 用法如下：

```
module.moveFile({
  fromFilename: "/storage/emulated/0/qqmusic/easter_egg_config_res/easter_egg_
  toFilename: "/storage/emulated/0/leven_file/a/egg_51.png"
}, res => {
  console.log(res)
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
fromFilename	String	是	无	要复制的文件路径
toFilename	String	是	无	目的文件路径

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

删除文件或目录

方法名

deleteFile

用法

- 用法如下：

```
module.deleteFile({
  filePath: "/storage/emulated/0/leven_file/a/egg_51.png"
}, res => {
  console.log(res)
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
filePath	String	是	无	文件或目录的路径

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象

参数名	参数类型	参数描述
code	Integer	返回类型，0.成功，其他：失败

写文件

方法名

writeFile

用法

- 用法如下：

```
module.writeFile({
  filename: "/storage/emulated/0/leven_file/a/test_write.txt",
  content: "123456\n",
  append: true
}, res => {
  console.log(res)
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
filename	String	是	无	要写入的文件路径
content	String	是	无	文件内容
append	Boolean	否	false	是否追加，不追加的话会覆盖文件

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0
}
```

写文件

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

获取路径的文件名

- 用法如下：

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
filePath	String	是	无	文件路径

- 示例

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.filename	String	文件名
code	Integer	返回类型，0.成功，其他：失败

读取文件(按行读取)

方法名

readFileLine

用法

- 用法如下：

```
module.readFileLine({
  filename: "/storage/emulated/0/leven_file/a/test_write.txt",
  startLine: 1,
  lineCount: 1
}, res => {
  console.log(res)
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
filename	String	是	无	文件路径
startLine	String	是	无	开始行
lineCount	String	是	无	读取的行数

回调

- 示例

```
{
  "data": {
    "list": [
      "123456123456"
    ]
  },
  "message": "",
  "code": 0
}
```

读取文件(按行读取)

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.list	Array	每行的内容
code	Integer	返回类型，0.成功，其他：失败

读取整个文件

方法名

readFile

用法

- 用法如下：

```
module.readFile({
  filename: "/storage/emulated/0/leven_file/a/test_write.txt",
  removeWrap: false
}, res => {
  console.log(res)
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
filename	String	是	无	文件路径
removeWrap	String	否	false	是否删除换行符

回调

- 示例

[illegible]

- 回调说明：

读取整个文件

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.content	String	文件内容
code	Integer	返回类型，0.成功，其他：失败

远程下载文件到本地

方法名

downloadFile

用法

- 用法如下：

```
module.downloadFile({
  url: "http://www.yeyuboke.com/svgga/1.svga",
  saveDir: "/storage/emulated/0/leven_file/"
}, res => {
  console.log(res)
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
url	String	是	无	url地址
saveDir	String	否	无	保存的文件地址 (绝对路径)

回调

- 示例

```
{
  "data": {
    "type": "start"
  },
  "message": "开始下载",
  "code": 0
}

{
  "data": {
    "percent": 0.15394142772234098,
    "type": "downloading",
```

```
        "size": 250197,
        "totalSize": 1625274
    },
    "message": "下载中",
    "code": 0
}

{
    "data": {
        "type": "success",
        "path": "/storage/emulated/0/leven_file/1.svga"
    },
    "message": "下载完成",
    "code": 0
}
```

回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.type	String	下载类型，start：开始下载， downloading：下载中， success：下载完成，error：下 载出错
data.percent	Float	当前下载的进度(<=1)
data.size	Integer	已下载的大小，单位：字节
data.totalSize	Integer	文件总大小，单位：字节
data.path	String	下载完成后的文件路径
code	Integer	返回类型，0.成功，其他：失败

获取U盘路径 v.1.1.0

方法名

getUpanPath

该方法会监听USB的插拔，会在callback中返回，具体返回信息请参考下方说明

7.0以上系统会返回USB路径列表，7.0以下系统需要插拔一次才能获取到USB路径，而且只能在插拔监听中获取

用法

- 用法如下：

```
module.getUpanPath(res => {  
  console.log(res)  
})
```

- 参数说明
无

回调

- 示例

```
//7.0以上系统会返回，7.0以下系统不会返回此数据  
{  
  "data": {  
    "pathList": [  
      "/mnt/USBSD0/"  
    ]  
  },  
  "message": "",  
  "code": 0  
}  
  
//USB插入返回  
{  
  "data": {  
    "type": "mounted",  
    "path": "/mnt/USBSD0/"  
  },  
}
```

```
        "message": "",
        "code": 0
    }

    //USB拔出返回
    {
        "data": {
            "type": "unmounted",
            "path": "/mnt/USBSD0/"
        },
        "message": "",
        "code": 0
    }
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.pathList	Array	USB路径列表，7.0以上系统才会有次数据
data.type	String	监听类型，mounted：USB插入，unmounted：USB拔出
data.path	String	监听的USB路径
code	Integer	返回类型，0.成功，其他：失败

获取重定向真实url

方法名

getRedirectUrl

用法

- 用法如下：

```
module.getRedirectUrl({
  url: "http://mhcg.392.me/adiuc.php?d=59930"
}, res => {
  console.log(res)
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
url	String	是	无	url地址

回调

- 示例

```
{
  "data": {
    "location": "https://mp3.haoge500.com/hot/2004/09-11/59930.mp3?ltd=584",
    "url": "http://mhcg.392.me/adiuc.php?d=59930",
    "isRedirect": true
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示

参数名	参数类型	参数描述
data	Object	数据对象
data.location	String	重定向后的地址
data.url	String	原始地址
data.isRedirect	Boolean	是否重定向，true：重定向，false：未重定向
code	Integer	返回类型，0.成功，其他：失败

批量下载文件(可下载重定向地址)

方法名

batchDownload

用法

- 用法如下：

```
module.batchDownload({
  list: [
    "http://mhcg.392.me/adiuc.php?d=59930",
    "http://www.yeyuboke.com/svgga/1.svgga"
  ],
  saveDir: "/storage/emulated/0/leven_file/"
}, res => {
  console.log(res)
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
list	Array	是	无	下载地址列表
saveDir	String	否	无	保存的文件地址 (绝对路径)

回调

- 示例

```
{
  "data": {
    "url": "http://mhcg.392.me/adiuc.php?d=59930",
    "type": "start"
  },
  "message": "开始下载",
  "code": 0
}
```

批量下载文件(可下载重定向地址)

```
{
  "data": {
    "percent": 0.00029610403430180174,
    "type": "downloading",
    "url": "http://mhcg.392.me/adiuc.php?d=59930",
    "size": 1024,
    "totalSize": 3458244
  },
  "message": "下载中",
  "code": 0
}

{
  "data": {
    "url": "http://www.yeyuboke.com/svgas/1.svga",
    "type": "success",
    "path": "/storage/emulated/0/leven_file/1.svga"
  },
  "message": "下载完成",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.url	String	当前下载的地址
data.type	String	下载类型，start：开始下载， downloading：下载中， success：下载完成，error：下 载出错
data.percent	Float	当前下载的进度(<=1)
data.size	Integer	已下载的大小，单位：字节
data.totalSize	Integer	文件总大小，单位：字节
data.path	String	下载完成后的文件路径
code	Integer	返回类型，0.成功，其他：失败

获取项目安装路径 v1.4.0

方法名

getPackagePath

用法

- 用法如下：

```
module.getPackagePath(res => {  
  console.log(res)  
})
```

- 参数说明
无

回调

- 示例

```
{  
  "message": "",  
  "code": 0,  
  "data": {  
    "path": "/storage/emulated/0/Android/data/test.file.com/",  
    "wwwPath": "/storage/emulated/0/Android/data/test.file.com/apps/__UNI__",  
    "staticPath": "/storage/emulated/0/Android/data/test.file.com/apps/__U"  
  }  
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.path	String	安装路径
data.staticPath	v1.5.1String	项目的static路径

参数名	参数类型	参数描述
data.wwwPath v1.5.1	String	项目的www路径
code	Integer	返回类型，0.成功，其他：失败

获取文件存储权限 v1.4.0

方法名

filePermission

用法

- 用法如下：

```
module.filePermission(res => {
  console.log(res)
})
```

- 参数说明
无

回调

- 示例

```
{
  "message": "",
  "code": 0,
  "data": {}
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

压缩文件夹 v1.5.0

方法名

zipFolder v1.5.0

用法

- 用法如下：

```
module.zipFolder({
  //压缩的文件夹路径
  path: "/storage/emulated/0/leven_file",
  //压缩后的文件路径
  zipPath: "/storage/emulated/0/leven_files/1.zip",
  //压缩密码
  password: ""
}, res => {
  console.log(res)
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
path	String	是	无	要压缩的文件夹地址
zipPath	String	否	自动生成	压缩后的文件路径
password	String	否	无	压缩密码

回调

- 示例

```
{
  "data": {
    "status": "start"
  },
  "message": "",
}
```

```

    "code": 0
  }

  {
    "data": {
      "status": "running",
      "progress": 12
    },
    "message": "",
    "code": 0
  }

  {
    "data": {
      "type": true,
      "status": "finish",
      "zipPath": "/storage/emulated/0/leven_files/1.zip"
    },
    "message": "",
    "code": 0
  }
}

```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.status	String	压缩状态，start：开始压缩， running：压缩中，finish：压缩完成
data.progress	Integer	压缩进度(百分比)，在压缩中时 返回
data.type	Boolean	压缩是否完成
data.zipPath	Boolean	压缩后的文件路径，压缩成功时 返回
code	Integer	返回类型，0.成功，其他：失败

压缩文件集合 v1.5.0

- 用法如下：

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
list	Array	是	无	要压缩的文件路径集合
zipPath	String	否	自动生成	压缩后的文件路径
password	String	否	无	压缩密码

- 示例

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.status	String	压缩状态，start：开始压缩，running：压缩中，finish：压缩完成
data.progress	Integer	压缩进度(百分比)，在压缩中时返回
data.type	Boolean	压缩是否完成
data.zipPath	Boolean	压缩后的文件路径，压缩成功时返回
code	Integer	返回类型，0.成功，其他：失败

解压缩 v1.5.0

方法名

unZip

v1.5.0

用法

- 用法如下：

```
module.unZip({
  //压缩文件路径
  path: "/storage/emulated/0/leven_files/2.zip",
  //解压后的文件夹
  unzipPath: "/storage/emulated/0/leven_files/2",
  //压缩密码
  password: "123456"
}, res => {
  console.log(res)
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
path	String	是	无	要解压的文件路径
unzipPath	String	是	无	解压后的文件夹路径
password	String	否	无	解压密码

回调

- 示例

```
{
  "data": {
    "status": "start"
  },
  "message": "",
}
```

```
    "code": 0
  }

  {
    "data": {
      "status": "running",
      "progress": 12
    },
    "message": "",
    "code": 0
  }

  {
    "data": {
      "type": true,
      "status": "finish"
    },
    "message": "",
    "code": 0
  }
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.status	String	解压状态，start：开始解压， running：解压中，finish：解 压完成
data.progress	Integer	解压进度(百分比)，在解压中时 返回
data.type	Boolean	解压是否完成
code	Integer	返回类型，0.成功，-104：密码 错误，其他：失败

压缩包是否加密 v1.5.0

方法名

zipIsEncrypted v1.5.0

用法

- 用法如下：

```
module.zipIsEncrypted({
  //压缩文件路径
  path: "/storage/emulated/0/leaven_files/1.zip"
}, res => {
  console.log(res)
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
path	String	是	无	压缩文件路径

回调

- 示例

```
{
  "data": {
    "isEncrypted": false
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象

参数名	参数类型	参数描述
data.isEncrypted	Boolean	true：已加密，false：未加密
code	Integer	返回类型，0.成功，其他：失败

监听U盘权限 v1.6.0

方法名

getUpanPermission v1.6.0

USB插入时会有USB权限弹窗提示，如果取消权限有可能导致U盘的文件获取不到

用法

- 用法如下：

```
module.getUpanPermission(res => {  
  console.log(res)  
})
```

- 参数说明
无

回调

- 示例

```
//USB插入返回  
{  
  "data": {  
    "type": "mounted"  
  },  
  "message": "",  
  "code": 0  
}  
  
//USB拔出返回  
{  
  "data": {  
    "type": "unmounted"  
  },  
  "message": "",  
  "code": 0  
}  
  
//权限结果返回
```

```
{
  "data": {
    "type": "permission"
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.type	String	监听类型，mounted：USB插入，unmounted：USB拔出，permission：usb权限获取结果
code	Integer	返回类型，0.成功，其他：失败

取消监听U盘权限 v1.6.0

方法名

cancelUpanPermission

v1.6.0

用法

- 用法如下：

```
module.cancelUpanPermission(res => {
  console.log(res)
})
```

- 参数说明
无

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

创建文件夹 v1.7.0

方法名

mkdirs v1.7.0

用法

- 用法如下：

```
module.mkdirs({
  //文件夹路径
  dir: "/storage/emulated/0/leven_files"
}, res => {
  console.log(res)
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
dir	String	是	无	要创建的文件夹路径

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象

参数名	参数类型	参数描述
code	Integer	返回类型，0.成功，其他：失败

选择文件系统弹窗 v1.8.0

方法名

selectFiles v1.8.0

用法

- 用法如下：

```
module.selectFiles({
  //文件类型
  type: "*/*"
}, res => {
  this.writeLog(JSON.stringify(res))
});
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
type	String	否	*/*	需要展示的文件格式

回调

- 示例

```
{
  "data": {
    "uriString": "content://com.android.providers.media.documents/document/doc",
    "path": "/storage/emulated/0/test.txt"
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.uriString	String	Uri的原生数据
data.path	String	选择的文件路径
code	Integer	返回类型，0.成功，其他：失败

检查是否有“所有文件”访问权限 v1.8.0

方法名

checkAllFilesPermission

v1.8.0

用法

- 用法如下：

```
module.checkAllFilesPermission(res => {
  this.writeLog(JSON.stringify(res))
});
```

- 参数说明
无

回调

- 示例

```
{
  "data": {
    "status": true
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.status	Boolean	开启状态，true：已开启，false：未开启
code	Integer	返回类型，0.成功，其他：失败

跳转到所有文件访问权限页面 v1.8.0

方法名

toAllFilesPermissionPage v1.8.0

用法

- 用法如下：

```
module.toAllFilesPermissionPage(res => {
  this.writeLog(JSON.stringify(res))
});
```

- 参数说明
无

回调

- 示例

```
{
  "data": {
    "status": true
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.status	Boolean	开启状态，true：已开启，false：未开启
code	Integer	返回类型，0.成功，其他：失败

判断文件或文件夹是否存在 v1.9.0

方法名

fileExists

v1.9.0

用法

- 用法如下：

```
module.fileExists({
  //文件或文件夹夹路径
  path: "/storage/emulated/0/leven_files/1.txt"
}, res => {
  this.writeLog(JSON.stringify(res))
});
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
path	String	是	无	文件或文件夹夹路径

回调

- 示例

```
{
  "data": {
    "isExists": true
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示

参数名	参数类型	参数描述
data	Object	数据对象
data.isExists	Boolean	true：文件存在，false：文件不存在
code	Integer	返回类型，0.成功，其他：失败

写入大文件 v1.10.0

方法名

writeLargerFile

v1.10.0

用法

- 用法如下：

```
module.writeLargerFile({
  //文件保存路径
  path: "/storage/emulated/0/leven_file/1.txt",
  //文件内容
  content: ""
}, res => {
  console.log(JSON.stringify(res))
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
path	String	否	无	文件路径
content	String	是	无	文件内容

回调

- 示例

```
{
  "data": {
    "status": "onStart"
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.status	String	写入状态，onStart：开始写入，onProgress：写入进度，onComplete：写入完成，onFail：写入失败
data.path	String	写入的文件路径
data.total	Integer	写入的文件总大小，单位：字节
data.position	Integer	当前写入的大小，单位：字节
data.progress	float	写入的进度，百分比需要*100
data.length	Integer	写入完成后文件大小，onComplete时返回
data.message	Integer	错误信息，onFail时返回
code	Integer	返回类型，0.成功，其他：失败

base64保存图片 v1.10.0

方法名

base64ToImage

v1.10.0

用法

- 用法如下：

```
module.base64ToImage({
  //文件保存路径
  path: "/storage/emulated/0/leven_file/1.jpg",
  //base64数据
  content: ""
}, res => {
  console.log(JSON.stringify(res))
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
path	String	否	无	文件或文件夹夹路径
content	String	是	无	base64数据

回调

- 示例

```
{
  "data": {
    "path": ""
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.path	String	保存的文件路径
code	Integer	返回类型，0.成功，其他：失败

保存为二进制文件 v1.11.0

方法名

saveToBinary v1.10.0

用法

- 用法如下：

```
module.saveToBinary({
  //文件保存路径
  path: "/storage/emulated/0/leven_file/1.bin",
  //文件内容
  content: ""
}, res => {
  console.log(JSON.stringify(res))
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
path	String	否	无	文件路径
content	String	是	无	文件内容

回调

- 示例

```
{
  "data": {
    "status": "onStart"
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.status	String	写入状态，onStart：开始写入，onProgress：写入进度，onComplete：写入完成，onFail：写入失败
data.path	String	写入的文件路径
data.total	Integer	写入的文件总大小，单位：字节
data.position	Integer	当前写入的大小，单位：字节
data.progress	float	写入的进度，百分比需要*100
data.length	Integer	写入完成后文件大小，onComplete时返回
data.message	Integer	错误信息，onFail时返回
code	Integer	返回类型，0.成功，其他：失败