

Android 组件库

JitPack 0.97.1

java 8

androidx

minSdkVersion 21

targetSdkVersion 32

Android原生组件库，为Html提供Android原生能力，增强H5

map: 5M左右

tim: 14M左右

Module目录及引用说明

依赖引入（需要添加jitpack仓库源）：

```
1 implementation 'com.github.TYKYTeam.swan-android-libray:module名:版本号'
```

例：

```
1 implementation 'com.github.TYKYTeam.swan-android-libray:0.2'
```

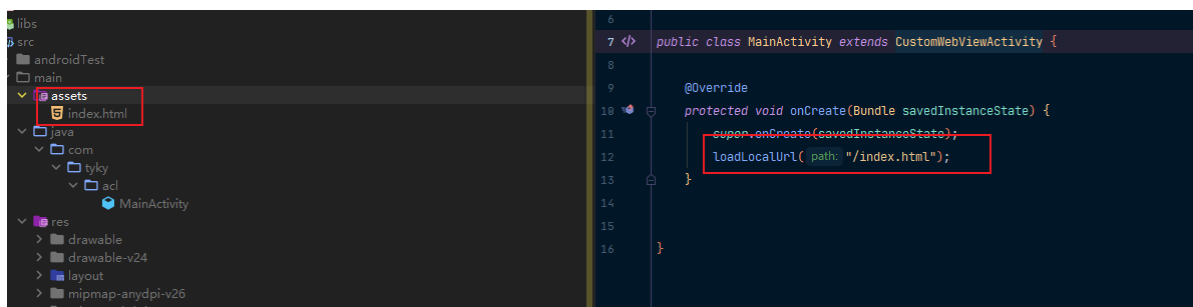
注：以下使用的传参都是需要Json的字符串，需要调用 `JSON.stringify()` 方法将对象转为字符串

webViewBase使用

提供一个CustomWebViewActivity，首页的MainActivity需要继承于此Activity，之后再onCreate方法中调用以下两种方法之一，来加载url

- `loadLocalUrl` 设置加载的url并加载
- `loadWebUrl` 加载url

PS: `loadLocalUrl`主要是自动拼接了assets里的本地资源html文件，如下图



由于文件选择的功能与Webview比较耦合，无法单独划分出一个module，所以内置在主module中包含拍照，选择相册图片，录制视频，选择本地视频文件，录音，选择本地音频文件，和选择其他文件

media使用

1.获取剪切板内容

`getTextFromClipboard()`

用来获取手机剪切板的文本内容

传参：无需传参

返回结果：

```
1 | {"code":200,"desc":"","result":"剪切板的内容"}
```

H5调用示例：

```
1 | if (window.android_media) {  
2 |     let result = window.android_media.getTextFromClipboard()  
3 |     let text = JSON.parse(result).result  
4 |     console.log(text)  
5 | }
```

2.复制文本

`copyTextToClipboard(paramStr)`

往手机的剪切板写入内容，实现H5复制功能

传参：

```
1 | {  
2 |     content:"hello world"  
3 | }
```

返回结果：

```
1 | //成功  
2 | {"code":200,"desc":"","result":""}
```

H5调用示例：

```
1 | if (window.android_media) {  
2 |     let content = {content:"hello world"}  
3 |     let result =  
4 |     window.android_media.copyTextToClipboard(JSON.stringify(content))  
5 |     console.log(result);  
6 | }
```

3.拨打电话

`callPhone(paramStr)`

拨打电话，会直接拨号

使用前需要声明电话权限，方法内已作了动态权限申请适配

```
1 | <uses-permission android:name="android.permission.CALL_PHONE" />
```

传参:

```
1  {  
2    phone: "10086"  
3  }
```

返回结果:

```
1  //成功  
2  {"code":200,"desc":"","result":""}
```

H5调用示例:

```
1  if (window.android_media) {  
2    let content = {  
3      phone: "10086"  
4    }  
5    let result = window.android_media.callPhone(JSON.stringify(content))  
6    console.log(result);  
7  }
```

4.跳转拨号页面

goToCall

会跳转到拨号页面，无需申请电话权限

传参:

```
1  {  
2    phone: "10086"  
3  }
```

返回结果:

```
1  //成功  
2  {"code":200,"desc":"","result":""}
```

H5调用示例:

```
1  if (window.android_media) {  
2    let content = {  
3      phone: "10086"  
4    }  
5    let result = window.android_media.goToCall(JSON.stringify(content))  
6    console.log(result);  
7  }
```

5.发送短信

sendSms

会跳转到短信发送页面，不会立即发送短信

传参：

```
1  {
2      phone:"10086", //手机号码
3      content:"111" //短信内容
4  }
```

返回结果：

```
1  //成功
2  {"code":200,"desc":"","result":""}
```

H5调用示例：

```
1  if (window.android_media) {
2      let content = {
3          phone: "10086",
4          content: "111"
5      }
6      let result = window.android_media.sendSms(JSON.stringify(content))
7      console.log(result)
8  }
```

6.图片预览

previewPicture

预览单张图片，之后点击图片会关闭

传参：

```
1  {
2      content: "",
3      type: 2
4  }
```

当 type 为1， content 传base64数据；

当 type 为2， content 传图片地址

返回结果：

```
1  //成功
2  {"code":200,"desc":"","result":""}
```

H5调用示例：

```

1  if (window.android_media) {
2      let content = {
3          content:
4      "https://gimg2.baidu.com/image_search/src=http%3A%2F%2Fimg.jj20.com%2Fup%2F1img%2Ftp01%2F1ZZQ214233446-0-1p.jpg&refer=http%3A%2F%2Fimg.jj20.com&app=2002&size=f9999,10000&q=a80&n=0&g=0n&fmt=jpeg?sec=1647143194&t=b67961882a5e7d01f6ecd8a503a9db47",
5      type: 2
6      }
7      let result = window.android_media.previewPicture(JSON.stringify(content))
8      console.log(result)
9  }

```

7.扫描二维码

qrScan 扫描二维码

传参:

```

1  {
2      callBackMethod: "" //扫描二维码成功后的回调js方法
3  }

```

返回结果:

```

1  //成功
2  {"code":200,"desc":"","result":""}

```

H5调用示例:

```

1  //object格式与返回结果一样，result即为二维码的文本数据
2  window.qrScanResult = function(object) {
3      console.log(object);
4  }
5
6  function qrScan() {
7      if (window.android_media) {
8          let content = {
9              callBackMethod: "qrScanResult"
10             }
11             window.android_media.qrScan(JSON.stringify(content))
12         }
13     }

```

8.语音播放文本

speakText 播放文本（如果没有tts引擎，会没法播放语音）

传参:

```

1  {
2      content: "这是一段测试语音的文本"
3  }

```

返回结果:

```
1 //成功
2 {"code":200,"desc":"","result":""}
```

H5调用示例:

```
1 if (window.android_media) {
2     let content = {
3         content: "这是一段测试语音的文本"
4     }
5     let result = window.android_media.speakText(JSON.stringify(content))
6     console.log(result)
7 }
```

9.获取手机通讯录

getContacts

传参:

```
1 {
2     callbackMethod: "myContacts" //回调方法
3 }
```

返回结果:

```
1 {"code":200,"desc":"","result":[{"name":"xing wan","number":"18778089853"},
  {"name":"one stars","number":"147663358935"}]}
```

H5调用示例:

```
1 window.myContacts = function(object) {
2     console.log(object);
3 }
4
5 if (window.android_media) {
6     let content = {
7         callbackMethod: "myContacts"
8     }
9     let result = window.android_media.getContacts(JSON.stringify(content))
10    alert(result);
11    console.log(result)
12 }
```

还未实现功能清单

8. 图片压缩 (H5传参)
9. 文件下载
10. 文件预览(文件先下载再预览)
11. 安装apk

notification通知

1. 发送通知

sendNotification 状态栏显示一条新通知

传参:

```
1  {
2      content: "测试内容11111",
3      title: "测试标题"
4  }
```

返回结果:

```
1  //成功
2  {"code":200,"desc":"","result":""}
```

H5调用示例:

```
1  if (window.notification) {
2      let content = {
3          content: "测试内容11111",
4          title: "测试标题"
5      }
6      let result =
window.notification.sendNotification(JSON.stringify(content))
7      console.log(result)
8  }
```

2. 设置桌面小红点

setBargeCount

传参:

```
1  {
2      content: "11" //未读数
3  }
```

返回结果:

```
1  //成功
2  {"code":200,"desc":"","result":""}
```

H5调用示例:

```
1  if (window.notification) {
2      let content = {
3          content: "11"
4      }
5      let result = window.notification.setBargeCount(JSON.stringify(content))
6      console.log(result)
7  }
```

share

默认分享会弹出分享方式（如分享给微信好友，复制文本，分享到QQ等选项）

1.分享文本

shareText 弹出分享的对话框，选择某一应用进行分享

传参：

```
1  {  
2    content: "这是分享的文本内容"  
3  }
```

返回结果：

```
1  //成功  
2  {"code":200,"desc":"","result":""}
```

H5调用示例：

```
1  if (window.share) {  
2    let content = {  
3      content: "这是分享的文本内容"  
4    }  
5    let result = window.share.shareText(JSON.stringify(content))  
6    console.log(result)  
7  }
```

2.分享图片

shareImage 弹出分享的对话框，选择某一应用进行分享

目前只支持传base64图片数据

传参：


```

1 {
2   content:
    "data:image/png;base64,iVBORw0KGgoAAAANSUHEugAAACKAAAAmCAIAAADWVCV0AAAAACXBIWX
    MAABJ0AAASdaHeZh94AAAAEXRFWHRTb2Z0d2FyZQBtbnlwYXN0ZV0Xzt0AAAiqSURBVFiFfVhbbx1
    XFF6+vfecq+1j05FGhqapQUVVo6QBtZXSl1AegpCiI1GJ0D60LxEFiRf+AmPTRMUP4CUP5QFHqHqo
    BQotJdA0QBSaEkJxjEOTFJxYqW/Hl+NzZmavxcOemTN2SJcsn5k9M/tb98vmUnudgALF/0CqIE1CV
    IvFYj1ckhrREeqld/qbFe+Hy+wrzZ7AEARIICxpAUioVFQNCFT/jGHYQBxiHwTYnYYkw8bh6w2kAF
    TUMGPeKfShioy7ngeiR9N8J4UckqIFhzTcDKKbFZPLo8jfupCiCoCKIUAQmzwlyPQcbkkAmzsfl2Y
    xKlpez5wbv6bFNiT7H7qc1eyDAC8f3qodP2m7vwTnSLxtw7x9BNta0XBDaxwtVXvk003eYSouWfCv
    icev6xtxrJ/ZXavVjEIZnnLQzBw5ajCTC4AsLRHwf7tsvfpfo2JvfuroI5rU2MQ4cvFSraRDazz04
    MDXPx/VKyRSwZ8urr78+6X1Lr5wYOArjw+1hqOywsvmyMUHoC63cjBGeAUDvz1jVABYFQWVgBLik0
    St97orXz99e6Mama89ZqyZnev98s3FhcXY07z2l/b6wvLcUzsaDVu43iZUZPYGgtwFUWSA9cs3Bqa
    vF/azKhDAQIO7KRwopHHy2p+7nxbpGh8/+e05W/OxNVZVxcubF1c9efTwtu0th0XEFqFVNOfr80UA
    kF46MvkrI5l1j579KqCEM1QCGRo1ivr3+gl+80zxxi3usjUTUwpAk+fbF9b1l/9TBoQP3N60oBjZhe
    rh1mxtl1jievzNrL/yZUVH8wrb5ve4w5BFRSyAhoHx0LZm/sDIw1tAGCRRQMnrpx9dGzu5ED/b9Y
    99tmUt0He9300IV8gcgje6MG2XOGBiYEWz8QfxwceTAqu3tqzz1kOutro9yh/9bklohn9wLmQgqi
    qJApYUACQqhrKxU3YqauUGIBXTVVEZN++fXv37jXGAHjnnXdmZmYCQLPZfOKJJ4aHhycnJ0wkje2f
    WBE1stLGD3+9MP1h71tPb8/yE/pwL8cysbjiprh8n1ERET279//zDPP00cAzM/PT09Ph6f1ev3Ik
    SMTExonTp3aiq3eqqqgKN0NnL7Q+eZX+ykFgIrs00QZGICaazc4v1h+w3t/9uzZc+f0kxz1lvcuXL
    jgcxIRa6lzzt9GFDWqRr1RTY2NGi6kfsB3exKAYvmFAnz3sokTBYRMaVIR7/3MzMzvq1df5Pz589e
    vxy9jhe//D7aKvBEqsXVdE9WqdkvaDxyEvJgZILr5Uac17DvdiqY9miSF955kmqYA0jT13he7iEi4
    LVMRYIg3kK6prtqa0FTrLjh5IXrm51oqm2vf+/bqB4vu8qxht0/0uia+tuHbvkhPIhKYCCsBL1xsw
    YZi3dUwqkMers1qJSS0TRvdVv2ucgJwNTd832j15t1k13/ixj294dIU27dvHxkZITk2Njy40Fiv1w
    8donTmzJ1du3YNDg7Oz8+naRp4KmJfVeJoIDVWAnbn1iQkVECT15KsABJAdPovZvq3ojaF86rey+H
    Dh5988knn3AsvvlCwsDA+Pv78888f03bMGNNqTSYNJ5Mk2SL3smtYEymtAkJTR0e5tpn7m6rCEV11
    U6h2V3Hx5XRTIUGUMEP97L1fx16+cuVKiLHV1dxxx399bGxsZ549AC5dunTy5Mk4jrGZUhgYozBCK
    1ivsJzVCyM6hZLMUVzCP/XqW4kyRiVB1EgvsZJXX3319ont4ev15evOp3P8+PHBwUEA7Xa71+vhNj
    JK0IqxIfu2a1SFIUQUpeLm8soEgr6zYuJegjifS+BSL0mSLC4uLi4u1rdokmr1Zev2yIIsODRko6D
    S1KqGoVUK4r0oJf1eQjHx6NqD3/CXfpLEctcd20ite9/+GIw7kQeRYwvZrCKP6aKVAEmX97RUqKk1
    Ko8e29j1kLw32bnc7qxvJM1so9GoVqu3AywtLd0J24FKq8YqKGCzggIXLIRWVaeF0AQur1YzfOBhm
    zqmcz8xfy1N00ceeeTAGQMsp2MAwIsvvngn7BQEjeQ6b1b6rS0yb1cArtS3Mze8Vkd2xX5CU3jvDx
    48ePT0VDHynTixIk7YXsg930jZLPqixht1VwBdFnXviD30AC9TtJb7qpPvfcvftSG2+8cbvcw3N
    Zww7mfg4K2ahKn1O1KOGqea8YkzwbGdpwuyX18RLmqZTU1NTU1N3grmD3AYwkueWRk1KRSyjiQ9p
    4f8Bu7Owlq7EX5744ufGDniogqFbVVCBDRct1ap/HxnvsNFF1fqsvVJSQBik7tibwiuk0jiliutDB
    qEDuaJ7DyNPWF+4skjjPtG6Z2frblGE9CRkb03VHY1b9dZCbajJoZRDCZo2pdJIBm/EGHF1YQsj94
    +kz+2T7QMMg+OWltkFNSvDyMNQ6BZmbolXNC6iEUAiZ+vVaGzn+P679n6yfuq9pbm2gQwCLWVTUgq
    NZIoxnmhGeu8Inn3AP31f2mpaa7JgklDSBD9xbhpVu+3u0vUFNGrVgVrzrsGhPaot3SMj94w072zQ
    AsDdd4++Ob328vt+eYue1hsT4nioIveO4tHdemi3f3g8aVRIGmxujfuDEsc15fvi9A6iL1xbnD0/O
    3rvtoGdA7XBstSoGEuyn4Kgmj9x5z//1n942zn17urquc+7R/fo3t3YkCDFavGQKwFRMVux15ur2
    fonbfk4gwAcS0dP1hxou0QEC9YLYt332rElznYeTivFurJqAFcpGP6D74gzMqRbxud3JJ95+84T
    +AUQxF/eH4iIcy7R1ovwry3oubkmYABxmrezVrJhnwCTzBJyN/WQxY5cG9q2DJ7buJ6JvymZ9hSmU
    nr99QKF51lFFdt6SQ/YTM4vDCfSn6JLQm5aZ20LDOBnOPJB5UPmIASwcr3wYoXn455FS6sECC5vgV
    SEi4bAhJNTiudjlF0H/cmTEkykAAAAAE1FTksuQmCC"
3 }

```

返回结果:

```

1 //成功
2 {"code":200,"desc":"","result":""}

```

H5调用示例:

```

1  if (window.share) {
2      let content = {
3          content:
            "data:image/png;base64,iVBORw0KGgoAAAANSuHEuGAAACkAAAAmCAIAAADWVCV0AAAAcXBIWX
            MAABJ0AAASdaHeZh94AAAAEXRFWHRTb2Z0d2FyZQBtbnlwYXN0ZV0Xzt0AAAIqSURBVFiFvHbbx1
            XFF6+vfecq+1j05fGhqapQUVVo6QBtZXSl1AeGpCiI1GJ0D60LxEFiRf+AmPTRMUP4CUP5QFHqHqo
            BQotJdAOQBsaEkJxjEOTFJxYQw/H1+NzZmavxcOemTN2Sjcsn5k9M/tb98vmUnudgALF/0CqIE1CV
            IvFYj1ckhrREqq1D/qbFe+Hy+wrZZ7AEARIICxpAUioVFQNCfT/jGHYQBxiHwTYnYYkw8bh6w2kAF
            TUMGPeKFShioy7nGEiR9N8J4UCKqIFhzTcdKKbFZPLo8jfUpCiCoCKIUAQmzwlyPQcbkkAmzSfL2Y
            xK1pez5wbv6bFNiT7H7qc1eyDAC8f3qodP2m7vwTnSLxtw7x9Bnta0XBDaxwtVXvk003eYSouWfCv
            iceV6xtxrJ/ZXavvjEIZnnLQzBw5ajCTC4AsLRHwf7tSvfpfo2JvfRoI5rU2MQ4cvFSraRDazz04
            MDXPx/VKyRSwZ8urr78+6X1Lr5wYOArjw+1hqOywsvmyMUHoC63cjBGeAUDvz1jvABYFQWvgBLik0
            St97oRxz99e6Mama89ZqyZnev98s3FhcXY07z21/b6wvLCuzsaDVu43iZUZPYGgtWfUwSA9cs3Bqa
            vF/azKhDAQIO7KRwopHHy2p+7nxpbGh8/+e05W/OxNVZVxcubF1c9efTwtu0th0xEfQVFNofr80UA
            kF46MvkrI51j579KqCEM1QCGRo1ivr3+g1+80zxxi3usjUTUwpAk+fbF9b11/9TBoQP3N60oBjZhe
            rh1mXt1JievzNrL/yZUVH8wrb5ve4w5BFRSyxAhoX0LZm/sDIw1tAGCRRQMnrpx9dGzu5ED/b9Y
            99tmUt0He9300IV8gcgje6MG2XogBiYEWz8QfxwceTaq3tqzz1kOutro9yh/9bk1oHn9wwLmQgqi
            qJAypUACqQhrkxu3YqauUGIBXTVVEZN++fxv37jXGAHjnnXdmZmYCQLPZFoKJJ4aHhycnJOwkjE2f
            WBE1stLGD3+9MP1h71tPb8/yE/pwL8cYsbjPrh8n1ERET279//zDPP00caZM/PT09Ph6f1ev3Ik
            SMTExonTp3aiq3eqqqgKN0NnL7Q+eZX+ykFgIrs00QZGICaazc4v1h+w3t/9uzZc+fOkXz11vcuXL
            jgcxIRA61zzt9GFDWqRr1RTY2NGi6kfSB3exKayVmFanz3sokTBYRMAVIR7/3MzMzVq1df5Pz589e
            vxy9jhe//D7akVbEqSXvdE9WqdkvaDxyEvJgZILr5Uac17DvdiqY9miSF955kmqYA0jT13he7iEi4
            LVMRYIg3kk6prtqa0FTrLjh5IXrm5loqm2vf+/bqB4vu8qxht0/Ouia+tuHbvkhPIhKYCCsBL1xsw
            Yzi3duWqkMers1qJSS0TRvdVv2ucgJwNTd832j15t1k13/ixj294dIU27dvHxkZITk2NjY40Fiv1w
            8dOnTmzJ1du3YNDg7Oz8+naRp4KmJFveJoIDVWAnbn1iQkVECT15KsABJAdPovZvq3ojaF86rey+H
            Dh5988knn3AsvVLCwsDA+Pv78888f03bMGNNqTSynJ5Mk2SL3smtYEymtAkJTr0e5tpn7m6rCEV11
            U6h2V3Hx5XRTiUGUMEP97L1fX16+cuVKiLHV1dxxx399bGxsZ549AC5dunTy5Mk4jrGZUhgyozBCK
            1ivsJzVCym6hZLMUVzCP/Xqw4kyRiVB1EgvsZJXX3319ont4ev15evOp3P8+PHBwUEA7Xa71+vhNj
            JK0IqxIfu2a1SFIUQupELm8soEgr6zyuJegiiFS+BSL0mSLC4uLi4ulrdOkmR1Zev2yIIsODRko6D
            S1KqGovUK4rooJf1eqjHx6NqD3/CXfpLECTcd20ite9/+GIw7kQeRYwvZrCKP6AKVAEmX97RUqkK1
            ko8e29j1kLw32bnc7qxvJM1so9Govqu3AywtLd0J24FKq8YqKGCzgqIXLIRWvaeF0AQURlyZfOBhM
            zqmcz8xfy1N00ceeeTAGQMsp2MAWIsVvngn7BQEjeQ6b1b6rs0yb1cArts3Mze8vkd2xx5CU3jvDx
            48ePT0VDHynTixIk7YXsg93OjZLPqixhT1VwBdFnXviD3OAC9TtJb7qpPvfcvftSG2+8cbvcw3N
            ZWW7mfg4K2ahKn1O1KOGqea8YykwBgDpwuyX18RLmqZTU1NTU1N3grmD3AYwkueWRk1KRSyjiQ9p
            4f8Bu7ow1q7EX5744ufGDniogqFbVVCBDRct1ap/HXnvsNFF1fqsVVJSQBik7tibwiuk0jiliutDB
            qEDuaJ7DyNPwF+4skjjPtG6Z2frb1GE9CRkb03VHY1b9dZCbajJoZRDCZo2pdJIBm/EGHF1YQsJ94
            +kz+2T7QMMg+OW1tkFNSvDyMNQ6BZmbo1XNC6iEUAiZ+vvaGzn+P679n6yfuq9pbm2gQwCLWVTUgq
            NZIOxnmhGeu8Inn3AP31f2mpaa7JgK1dSBD9XbhpVU+3u0vUFNGrvGvRzrsGhPaot3SMj94w072zQ
            AsDdd4++Ob328vt+eYue1hsT4nioIve04tHdemi3f3g8aVRIGmxujfuDEsc15fvi9A6iL1xbnD0/O
            3rvtoGdA7XBstSoGEuyn4Kgmj9x5z//1n942zn17urqUC+7R/fo3t3YkCDFavGQKWFRMVux15ur2
            fOnbfk4gWAcS20dP1hXou0QEC9LYt332rE1nznYeTiVFURJqAFcpGP6D74gZmQrBxud3JJ95+84T
            +AUQxF/eH4iIcy7R1ovwry3oubkmYABxmrezVrJhwnCTzBJyN/WQxY5cG9q2DJ7buJ6JvyMZ9hSmU
            nR99QKF511FFdt6SQ/YTM4VDCfSn6JLQm5aZ20LDOBnOPJB5UPmIASwcr3wYoXn455FS6sECC5vgV
            SEi4bAhJNTiUdjlf0H/cmTEkykKAAAAAE1FTksuQmCC"
4      }
5      let result = window.share.shareImage(JSON.stringify(content))
6      console.log(result)
7  }

```

3.分享图文

shareTextImage 弹出分享的对话框，选择某一应用进行分享

目前只支持传base64图片数据,测试发现文字没法获取，和分享图片效果一样

传参：

```

1 {
2     title:"分享的标题",
3     content:
        "data:image/png;base64,iVBORw0KGgoAAAANSuHEuGAAACkAAAAmCAIAAADWVCV0AAAAcXBIWX
        MAABJ0AAASdaHeZh94AAAAEXRFWHRTb2Z0d2FyZQBtbnlwYXN0ZV0Xzt0AAAiqSURBVFiFfVhbbx1
        XFF6+vfecq+1j05fGhqapQUVVo6QBtZXSl1AeGpCiI1GJ0D60LxEFiRf+AMPTRMUP4CUP5QFHqHqo
        BQotJdA0QBsaEkJxjEOTFJxYQw/H1+NzZmavxcOemTN2Sjcsn5k9M/tb98vmUnudgALF/0CqIE1CV
        IvFYj1ckhRREqq1D/qbFe+Hy+wrZZ7AEARIICxpAUioVfQNCfT/jGHYQBxiHwTYnYYkw8bh6w2kAF
        TUMGPeKFShioy7nGEiR9N8J4UCKqIFhzTcdKkbFZPLo8jfUpCiCoCKIUAQmzwlyPQcbkkAmzSfL2Y
        xK1pez5wbv6bFNiT7H7qc1eyDAC8f3qodP2m7vwTnSLxtw7x9Bnta0XBDaxwtVXvk003eYSouWfCv
        iceV6xtxrJ/ZXavvjEIZnnLQzBw5ajCTC4AsLRHwf7tSvfpfo2JvfuoI5rU2MQ4cvFSraRDazz04
        MDXPx/VKyRSwZ8urr78+6X1Lr5wYOArjw+1hqOywsvmyMUHoC63cjBGeAUDvz1jvABYFQWvgBLik0
        St97orXz99e6Mama89ZqyZnev98s3FhcXY07z21/b6wvLCuzsaDVu43iZUZPYGgtWfUwSA9cs3Bqa
        vF/azKhDAQIO7KRwopHHy2p+7nxpbGh8/+e05W/OxNVZVxcubF1c9efTwtu0th0xEFqFVNofr80UA
        kF46MvkrI51j579KqCEM1QCGRo1ivr3+g1+80zxxi3usjUTUwpAk+fbF9b11/9TBoQP3N60oBjZhe
        rh1mXt1JievzNrL/yZUVH8wrb5ve4W5BFRSyxAhoX0LZm/sDIw1tAGCRRQMNRxp9dGzu5ED/b9Y
        99tmUt0He9300IV8gcgje6MG2XogBiyEWz8QfXwceTAqu3tqzz1kOutro9yh/9bk1oHn9wwLmQgqi
        qJAypUACqQhrkxu3YqauUGIBXTVVEZN++fxv37jXGAHjnnXdmZmYCQLPZfOKJJ4aHhycnJ0WkjE2f
        WBE1stLGD3+9MP1h71tPb8/yE/pwL8cYsbjPrh8n1ERET279//zDPP00caZM/PT09Ph6f1ev3Ik
        SMTExonTp3aiq3eqqqgKN0NnL7Q+eZX+ykFgIrs00QZGICaazc4v1h+w3t/9uzZc+fOkXz11vcuXL
        jgcxIRA61zzt9GFDWqRr1RTY2NGi6kfSB3exKayVmFanz3sokTBYRMAVIR7/3MzMzVq1df5Pz589e
        vxy9jhe//D7akVbEqSXvdE9wqdkvaDxyEvJgZILr5Uac17DvdiqY9miSF955kmqYA0jT13he7iEi4
        LVMRYIg3kK6prtqa0FTrLjh5IXrm51oqm2vf+/bqB4vu8qxht0/Ouia+tuHbvkhPIhKYCCsBL1xsw
        Yzi3duwqkMers1jJSS0TRvdVv2ucgJwNTd832j15t1k13/ixj294dIU27dvHxkZITk2NjY40Fiv1w
        8dOnTmzJ1du3YNDg7Oz8+naRp4KmJFVeJoIDVWAnbn1iQkVECT15KsABJAdPovZvq3ojaF86rey+H
        Dh5988knn3AsvVLCwsDA+Pv78888f03bMGNNqTSynJ5Mk2SL3smtYEymtAkJTr0e5tpn7m6rCEV11
        U6h2V3Hx5XRtIUGUMep97L1fx16+cuVKiLHV1dxxx399bGxsZ549AC5dunTy5Mk4jrGZUhgyozBCK
        1ivsJzVCym6hZLMUVzCP/Xqw4kyRiVB1EgvSZJXX3319onT4ev15evOp3P8+PHBwUEA7Xa71+vhNj
        JK0IqxIfu2a1SFIUQupELm8soEgr6zyuJegiiFS+BSL0mSLC4uLi4ulrdOkmR1Zev2yIIsODRko6D
        S1KqGovUK4rooJf1eQjHx6NqD3/CXfpLECTcd20ite9/+GIw7kQeRYwvZrCKP6AKVAEmX97RUqkK1
        ko8e29j1kLw32bnc7qxvJM1so9Govqu3AywtLd0J24FKq8YqKGCzgqIXLIRWvaeF0AQURlyZfOBhM
        zqmcz8xfy1N00ceeeTAGQMsp2MAWIsVvngn7BQEjeQ6b1b6rs0yb1cArts3Mze8vkd2xX5CU3jvDx
        48ePT0VDHynTixIk7YXsg930jZLPqixhT1VwBdFnXviD3OAC9TtJb7qpPvfcvftSG2+8cbvcw3N
        ZWW7mfg4K2ahKn1o1KOGqea8YykwBgDpwuyX18RLmqZTU1NTU1N3grmD3AYwkueWRk1KRSyjiQ9p
        4f8Bu7ow1q7EX5744ufGDniogqFbVVCBDRct1ap/HxnvsNFF1fqsVVJSQBik7tibwiuk0jiliutDB
        qEDuaJ7DyNPwF+4skjjPtG6Z2frb1GE9CRkb03VHY1b9dZCbajJoZRDCZo2pdJIBm/EGHF1YQsj94
        +kz+2T7QMMg+OW1tkFNSvDyMNQ6BZmbo1XNC6iEUAiZ+vvaGzn+P679n6yfuq9pbm2gQwCLWVTUgq
        NZIOxnmhGeu8Inn3AP31f2mpaa7JgK1dSBD9Xbhpvu+3u0vUFNGrvGvRzrsGhPaot3SMj94w072zQ
        AsDdd4++Ob328vt+eYue1hsT4nioIve04tHdemi3f3g8aVRIGmxujfuDEsc15fvi9A6iL1xbnD0/O
        3rvtoGdA7XBstSoGEuyn4Kgmj9x5z//1n942zn7urquc+7R/fo3t3YkCDFavGQKwFRmvux15ur2
        fonbfk4gWAcS20dP1hXou0QEC9YLYt332re1nznYeTiVFurJqAFcpGP6D74gZmQrBxud3JJ95+84T
        +AUQxF/eH4iicy7R1ovwry3oubkmYABxmrezVrJhwnCTzBJyN/WQxY5cG9q2DJ7buJ6JvyMZ9hSmU
        nR99QKF511FFdt6SQ/YTM4VDCfSn6JLQm5aZ20LDOBnOPJB5UPmIASwcr3wYoXn455FS6sECC5vgV
        SEi4bAhJNTiUdjlf0H/cmTEkykKAAAAAE1FTksuQmCC"
4 }

```

返回结果:

```

1 //成功
2 {"code":200,"desc":"","result":""}

```

H5调用示例:

```

1  if (window.share) {
2      let content = {
3          title:"分享的标题",
4          content:
            "data:image/png;base64,iVBORw0KGgoAAAANSUheUgAAACkAAAAmCAIAAADWVCV0AAAACXBIWX
            MAABJ0AAASdaHeZh94AAAAEXRFWHRTb2Z0d2FyZQBtbnlwYXN0ZV0Xzt0AAAIqSURBVFiFfvhbbx1
            XFF6+vfecq+1j05fGhqapQUVVo6QBtZXSl1AeGpCi1lGJ0D60LxEFiRf+AmPTRMUP4CUP5QFHqHqo
            BQotJda0QBSaEkJxjEOTFJxYqW/Hl+NzZmavxcOemTN2Sjcsn5k9M/tb98vmUnudgALF/0CqIElCV
            IvFYj1ckhRREqqld/qbFe+Hy+wrzZ7AEARIICxpAUioVFQNCfT/jGHYQBxiHwTYnYYkw8bh6w2kAF
            TUMGPEkFShioy7nGEiR9N8J4UCKqIFhzTCDKkbFZPLo8jfupCiCoCKIUAQmzwlyPQcbkkAmzSfL2Y
            XKlpez5wbv6bFNiT7H7qc1eyDAC8f3qodP2m7vWtNSLxtw7x9BNta0XBDaxwtVXvk003eYSouWFCv
            icev6xtxrJ/ZXavVjEiZnnLQzBw5ajCTC4AsLRHwf7tSvfpfo2JvfuRoI5rU2MQ4cvFSraRDazz04
            MDXPx/VKYRSwZ8urr78+6X1Lr5wYOArjw+1hqOywsvmyMUHoC63cjBGeAUDvzljVABYFQWvgBLik0
            St97orxz99e6Mama89ZqyZnev98s3FhcXY07z2l/b6wvLCuzsaDVu43iZUZPYGgtwFUwSA9cs3Bqa
            vF/azKhDAQIO7KRwopHHy2p+7nxpbGh8/+e05W/OxNVZVxcubF1c9efTwtu0th0xEFqFVNofr80UA
            kF46MvkrI5lj579KqCEM1QCGR0livr3+g1+80zxxi3usjUTUwpAk+fbF9b1l/9TBoQP3N60oBjZhe
            rh1mxtlJieVzNrL/yZUVH8Wrb5ve4W5BFRSyXAhohX0LZm/sDIwltAGCRRQMNRxpX9dGzu5ED/b9Y
            99tmUt0He9300IV8gcgje6MG2XogBiYEWz8QfXwceTAqu3tqzz1kOutro9yh/9bkloHn9wwLmGgqi
            qJAypUACqQhrkXu3YqauUGIBXTVVEZN++fxv37jXGAHjnnxdmZmYCQLPZF0KJJ4aHhycnJ0WkjE2f
            WBE1stLGD3+9MP1h71tPb8/yE/pWL8cYsbjiPrhh8n1ERET279//zDPP0oCAZM/PT09Ph6f1ev3Ik
            SMTExonTp3aiq3eqggqKN0NnL7Q+eZX+ykFgIRs00QZGICaazc4v1h+w3t/9uzZc+f0kXzllvcuXL
            jgcxIRA6lzzt9GFDWqRr1RTY2NGi6kFSB3exKAYvmFAnz3sokTBYRMAVIR7/3MzMzVq1dF5Pz589e
            vxy9jhe//D7aKvBEqsXvDE9wqdkvaDxyEvJgZILr5Uac17DvdiqY9miSF955kmqYA0jT13he7iEi4
            LVMRYIg3Kk6prtqa0FTrLjh5IXrm51oqm2vf+/bqB4vu8qxht0/Ouia+tuHbvkhPIhKYCCSBL1xsw
            YZi3dUwqkMerslqJSS0TRvdVv2ucgJwNTd832jl5t1k13/ixj294dIU27dvHxkZITk2NjY40Fiv1w
            8donTmzJldu3YNDg7Oz8+naRp4KmjFveJoIDVWAnbNliQkVECT15KsABJAdPOVzVq3ojaF86rey+H
            Dh5988knn3AsvvLCwsDA+Pv78888f03bMGNNqTSynJ5Mk2SL3smtYEymtAkJTr0e5tpn7m6rCEVl1
            U6h2V3Hx5XrTIUGUMep97L1fXl6+cuVKiLHV1dXXX399bGxsZ549AC5dunTy5Mk4jrGZUhgyozBCK
            livsJzVCym6hZLMUVzCP/XqW4kyRiVB1EgvsZJXX3319Ont4evl5evOp3P8+PHBWUEA7Xa71+vhNj
            JK0IqxIfu2a1SFIUQupELm8soEgr6zYuJegiiFS+BSL0mSLC4uLi4ulrdOkmRlZev2yIIsODRko6D
            S1kqGovUK4RoOjF1eqjHx6NqD3/CXfpLECTcd20ite9/+GIw7kQeRYwvZrCKP6AKVAEmX97RUqkK1
            ko8e29j1kLw32bnc7qxvJMlso9Govqu3AywtLd0J24FKq8YqKGCzgqIXLIRWvaeF0AQURlyZFoBhm
            zqmcz8Xfy1N00ceeeTAGQMsp2MAWIsVvngn7BQEjeQ6b1b6rs0yb1cArts3Mze8Vkd2xx5CU3jvDx
            48ePT0VDHynTixIk7YXsg930jZLPqixhT1VwBdFnxViD3OAC9TtJb7qpPvfcvvtSG2+8cbvcw3N
            ZWW7mfg4K2ahKn1o1KOGqea8YykwzBgDpwuyX18RLmqZTU1NTU1N3grmD3AYwkueWRk1KRSyjiQ9p
            4f8Bu7Owlq7EX5744ufGDniogqFbVVCBDRct1ap/HxnvsNFF1fqsVVJSQBik7tibwiuk0jLiutDB
            qEDuaJ7DYNPWF+4skjjPtG6Z2frblGE9CRkb03VHY1b9dZCbajJoZRDCZo2pdJIBm/EGHF1YQsj94
            +kz+2T7QMMg+OWltkFNSvDyMNQ6BZmbolXNC6iEUAiZ+vvaGzn+P679n6yfuq9pbm2gQwCLWVTUgq
            NZIOxnmhGeu8Inn3AP31f2mpaa7JgKldSBD9XbhpVu+3u0vUFNGrVgVrzrsGhPaot3SMj94w072zQ
            AsDdd4++Ob328vt+eYUe1hst4nioIveO4tHdemi3f3g8aVRIGmxujfuDESc15fvi9A6iL1xbnD0/O
            3rvtoGdA7XBstSoGEuyn4kgmnj9x5z//ln942zn7urquc+7R/fo3t3YkCDFavGQKWFRMVux15ur2
            fonbfk4gwAcS0dP1hXou0QEC9LYt332rElznYeTiVFURJqAFcPGP6D74gZmQrBxud3JJ95+84T
            +AUQXF/eH4iIcy7Rlovwry3oubkmYABxmrezVrJhNWCTzBJyN/WQxY5cG9q2DJ7buJ6JvymZ9hSmU
            nR99QKF5l1FFdt6SQ/YTM4vDCfSn6JLQm5aZ20LDOBnOPJB5UPmIASwcr3wYoXn455FS6sECC5vgv
            SEi4bAhJNTiudjlf0H/cmTEkykAAAAAE1FTksuQmCC"
5      }
6      let result = window.share.shareTextImage(JSON.stringify(content))
7      alert(result);
8      console.log(result)
9  }

```

4.分享文本给微信好友

shareToWechatFriend

分享文本给微信好友，会直接弹出选择好友的列表页面

传参：

```
1  {  
2      content: "分享的文本内容"  
3  }
```

返回结果：

```
1  //成功  
2  {"code":200,"desc":"","result":""}
```

H5调用示例：

```
1  if (window.share) {  
2      let content = {  
3          content: "分享的文本内容"  
4      }  
5      let result = window.share.shareToWechatFriend(JSON.stringify(content))  
6      console.log(result)  
7  }
```

device设备

1.获取设备mac地址

getMacAddress

获取设备mac地址

传参：无需

返回结果：

```
1  //成功 result中即为mac地址数据  
2  {"code":200,"desc":"","result":""}
```

H5调用示例：

```
1  if (window.device) {  
2      let result = window.device.getMacAddress()  
3      alert(result);  
4  }
```

2.获取唯一设备id

getUniqueDeviceId

获取唯一设备id

传参：无需

返回结果：

```
1 //成功
2 {"code":200,"desc":"","result":"2196aa50b16cd36d39df62298561809a1"}
```

H5调用示例：

```
1 if (window.device) {
2     let result = window.device.getUniqueDeviceId()
3     alert(result);
4 }
```

3.判断当前设备是否为平板

isTablet

当前设备是否为平板

传参：无需

返回结果：

```
1 //成功
2 {"code":200,"desc":"","result":true}
```

H5调用示例：

```
1 if (window.device) {
2     let result = window.device.isTablet()
3     alert(result);
4 }
```

4.判断当前设备是否为模拟器

isEmulator

判断当前设备是否为模拟器

传参：无需

返回结果：

```
1 //成功
2 {"code":200,"desc":"","result":false}
```

H5调用示例：

```
1 | if (window.device) {
2 |     let result = window.device.isEmulator()
3 |     alert(result);
4 | }
```

5.获取当前设备系统信息

getDeviceInfo

获取当前设备系统信息

传参：无需

返回结果：

```
1 | //成功
2 | {
3 |     "code":200,
4 |     "desc": "",
5 |     "result":{
6 |         "model":"ZUKZ2121",//设备型号
7 |         "manufacturer":"ZUK" //设备厂商
8 |         "versionName":"11",//系统版本名
9 |         "versionCode":30,//系统版本号
10 |     }
11 | }
```

H5调用示例：

```
1 | if (window.device) {
2 |     let result = window.device.getDeviceInfo()
3 |     alert(result);
4 | }
```

6.判断是否为某个机型品牌

isBrand

判断当前机型是否为某个品牌

传参：

```
1 | {
2 |     "content": "1"//型号类型，详见下表
3 | }
```


类型	品牌
1	360
2	酷派
3	金立
4	谷歌
5	HTC
6	华为
7	乐视
8	联想
9	LG
10	魅族
11	摩托罗拉
12	努比亚
13	一加
14	Oppo
15	三星
16	锤子
17	索尼
18	Vivo
19	小米
20	中兴

返回结果：

```
1 | ("code":200,"desc":"","result":false)
```

H5调用示例：

```
1 | //判断是否为360品牌手机
2 | if (window.device) {
3 |     let content = {
4 |         "content": "1"
5 |     }
6 |     let result = window.device.isBrand(JSON.stringify(content))
7 |     console.log(result);
8 |     alert(result);
9 | }
```


7.截图

takeScreenshot

传参:

```
1  {
2      callBackMethod: "getScreenshot"
3  }
```

返回结果:

```
1  //getScreenshot中回调的result为图片本地路径
2  {"code":200,"desc":"","result":"/storage/emulated/0/DCIM/com.tyky.ac1/1653012630717_100.JPG"}
```

H5调用示例:

```
1  window.getScreenshot = function(res){
2      console.log(res);
3  }
4
5  if (window.device) {
6      let content = {
7          callBackMethod: "getScreenshot"
8      }
9      let result = window.device.takeScreenshot(JSON.stringify(content))
10     console.log(result);
11     alert(result);
12 }
```

8.获取当前屏幕方向状态

getScreenOrientation

传参: 无需

返回结果:

```
1  //0:竖屏 1: 横屏
2  {"code":200,"desc":"","result":0}
```

H5调用示例:

```
1  if (window.device) {
2      let result = window.device.getScreenOrientation()
3      console.log(result);
4      alert(result);
5  }
```

9.获取屏幕分辨率

getScreenWH

传参: 无需

返回结果:

```
1 | '{"code":200,"desc":"","result":{"width":1080,"height":1920}}'
```

H5调用示例:

```
1 | if (window.device) {
2 |     let result = window.device.getScreenWH()
3 |     console.log(result);
4 |     alert(result);
5 | }
```

storage使用

1.保存数据

saveStorage

保存数据到Android内置的SharePerfence

传参:

```
1 | {
2 |     key: "test",
3 |     value: "hello world" //value可以是布尔值，字符串，整数，不能是一个JsonObject对象
4 | }
```

返回结果:

```
1 | //成功
2 | {"code":200,"desc":"","result":false}
```

H5调用示例:

```
1 | if (window.storage) {
2 |     let content = {
3 |         key: "test",
4 |         value: "hello world"
5 |     }
6 |     let result = window.storage.saveStorage(JSON.stringify(content))
7 |     alert(result);
8 |     console.log(result)
9 | }
```

2.读取数据

getStorage

根据key值去获取保存的数据

传参:

```
1  {  
2      key: "test"  
3  }
```

返回结果:

```
1  //成功 若result为空,则表示该key没有对应的数据  
2  {"code":200,"desc":"","result":""}
```

H5调用示例:

```
1  if (window.storage) {  
2      let content = {  
3          key: "test"  
4      }  
5      let result = window.storage.getStorage(JSON.stringify(content))  
6      alert(result);  
7      console.log(result)  
8  }
```

listener使用

1. 注册网络断开连接监听

registerNetworkDisconnectionListener

注册网络断开连接的监听回调

传参:

```
1  //传两个js回调的方法名,之后Android原生会执行对应的JS方法  
2  {  
3      "callBackMethod": "networkDisconnect"  
4  }
```

返回结果:

```
1  //成功  
2  {"code":200,"desc":"","result":""}
```

H5调用示例:

```

1 window.networkDisconnect = function() {
2     alert("网络已断开")
3 }
4
5 if (window.listener) {
6     let content = {
7         "callbackMethod": "networkDisconnect"
8     }
9     let result =
window.listener.registerNetworkDisconnectionListener(JSON.stringify(content)
10 )
11     console.log(result)
12 }

```

2. 注册网络成功连接监听

registerNetworkConnectionListener

注册网络成功连接监听回调

传参：

```

1 //传两个js回调的方法名，之后Android原生会执行对应的JS方法
2 {
3     "callbackMethod": "networkConnect"
4 }

```

返回结果：

```

1 //成功
2 {"code":200,"desc":"","result":""}

```

H5调用示例：

```

1 window.networkConnect = function() {
2     alert("网络已连接")
3 }
4
5 if (window.listener) {
6     let content = {
7         "callbackMethod": "networkConnect"
8     }
9     let result =
window.listener.registerNetworkConnectionListener(JSON.stringify(content))
10     console.log(result)
11 }

```

3. 来电监听

registerPhoneListener

来电时候的监听回调

传参：

```
1 //传递个js回调的方法名，之后Android原生会执行对应的JS方法
2 {
3     "callbackMethod": "phoneCall"
4 }
```

返回结果：

```
1 //成功
2 {"code":200,"desc":"","result":""}
```

H5调用示例：

```
1 window.phoneCall = function() {
2     alert("来电了。。")
3 }
4
5 if (window.listener) {
6     let content = {
7         "callbackMethod": "phoneCall"
8     }
9     let result =
window.listener.registerPhoneListener(JSON.stringify(content))
10     console.log(result)
11 }
```

map

1. 获取当前地址

getLocation

获取当前地址

传参：

```
1 //传递个js回调的方法名，之后Android原生会执行对应的JS方法
2 {
3     "callbackMethod": "getLocation"
4 }
```

返回结果：

```
1 //成功
2 {"code":200,"desc":"","result":""}
```

H5调用示例：

```
1 window.getLocation = function(object) {
2     //object内容为{"addrStr":"中国广东省深圳市福田区梅林街道梅坳八路","city":"深圳
市","district":"福田区","province":"广东省","street":"梅坳八路"}
3     console.log(object);
4     alert(JSON.stringify(object))
5 }
6
```

```

7  function startLocation() {
8      if (window.map) {
9          let content = {
10             "callbackMethod": "getLocation"
11         }
12         let result = window.map.getLocation(JSON.stringify(content))
13         alert(result)
14         console.log(result)
15     }
16 }

```

2.地图检索

poiSearch

打开地图页面，并展示当前的位置

传参：

```

1  {
2      "keyword": "小吃",
3      "city": "深圳市",
4      "tags": "火锅",
5      "callbackMethod": "getLocation"
6  }

```

返回结果：

```

1  //成功
2  {
3      "code": 200,
4      "desc": "",
5      "result": [
6          {
7              "address": "广东深圳市罗湖区人民路二横街12号负1层",
8              "area": "罗湖区",
9              "city": "深圳市",
10             "name": "东门小吃街",
11             "latitude": 22.552882,
12             "longitude": 114.127109,
13             "phoneNum": "(0755)82221797",
14             "province": "广东省"
15         }
16     ]
17 }

```

H5调用示例：

```

1  window.getLocation = function(object) {
2      console.log(object);
3      alert(JSON.stringify(object))
4  }
5
6  let content = {
7      "keyword": "小吃",
8      "city": "深圳市",

```

```

9     "tags": "火锅",
10    "callbackMethod": "getLocation",
11  }
12  if (window.map) {
13    window.map.poiSearch(JSON.stringify(content))
14  }

```

3.判断当前是否在某地点附近

ptInCircle

传入经纬度坐标，转为具体地址信息

传参：

```

1  {
2    "latitude": 22.57871696606192,
3    "longitude": 114.06792497377637,
4    "distance": 70,
5    "callbackMethod": "getLocation"
6  }

```

- `latitude` 目标点纬度
- `longitude` 目标点经度
- `distance` 目标点范围半径（单位为m）

返回结果：

```

1  //成功
2  {
3    "code": 200,
4    "desc": "",
5    "result": {
6      "address": "广东省深圳市福田区梅坳三路",
7      "area": "福田区",
8      "city": "深圳市",
9      "latitude": 22.5790620003812,
10     "longitude": 114.06740399999997,
11     "name": null,
12     "phoneNum": null,
13     "province": "广东省",
14     "street": "梅坳三路"
15   }
16 }

```

H5调用示例：

```

1  window.getLocation = function(object) {
2    console.log(object);
3    alert(JSON.stringify(object))
4  }
5
6  let content = {
7    "latitude": 22.57871696606192,
8    "longitude": 114.06792497377637,
9    "distance": 70,

```

```

10     "callbackMethod": "getLocation"
11 }
12
13 if (window.map) {
14     let result = window.map.ptInCircle(JSON.stringify(content))
15     alert(result)
16     console.log(result)
17 }

```

4.地址转坐标

geocode

传入地址，转为经纬度坐标

传参：

```

1 {
2     "endName": "广东省深圳市福田区梅林街道梅都社区中康路136号深圳新一代产业园4栋M层沿街商
    铺(手扶梯旁)",
3     "city": "深圳",
4     "callbackMethod": "getLocation"
5 }

```

返回结果：

```

1 //成功
2 {
3     "code": 200,
4     "desc": "",
5     "result": {
6         "address": null,
7         "area": null,
8         "city": null,
9         "latitude": 22.57871696606192,
10        "longitude": 114.06792497377637,
11        "name": null,
12        "phoneNum": null,
13        "province": null,
14        "street": ""
15    }
16 }

```

H5调用示例：

```

1 window.getLocation = function(object) {
2     console.log(object);
3     alert(JSON.stringify(object))
4 }
5
6 let content = {
7     "endName": "广东省深圳市福田区梅林街道梅都社区中康路136号深圳新一代产业园4栋M层沿街
    商铺(手扶梯旁)",
8     "city": "深圳",
9     "callbackMethod": "getLocation"
10 }

```



```

11
12   if (window.map) {
13       let result = window.map.geocode(JSON.stringify(content))
14       alert(result)
15       console.log(result)
16   }

```

5.坐标转地址

reverseGeoCode

传入经纬度坐标，转为具体地址信息

传参：

```

1  {
2      "latitude": 22.579062,
3      "longitude": 114.067404,
4      "callbackMethod": "getLocation"
5  }

```

返回结果：

```

1  //成功
2  {
3      "code": 200,
4      "desc": "",
5      "result": {
6          "address": "广东省深圳市福田区梅坳三路",
7          "area": "福田区",
8          "city": "深圳市",
9          "latitude": 22.5790620003812,
10         "longitude": 114.06740399999997,
11         "name": null,
12         "phoneNum": null,
13         "province": "广东省",
14         "street": "梅坳三路"
15     }
16 }

```

H5调用示例：

```

1  window.getLocation = function(object) {
2      console.log(object);
3      alert(JSON.stringify(object))
4  }
5
6  let content = {
7      "latitude": 22.579062,
8      "longitude": 114.067404,
9      "callbackMethod": "getLocation"
10 }
11
12 if (window.map) {
13     let result = window.map.reverseGeoCode(JSON.stringify(content))
14     alert(result)

```

```
15 | console.log(result)
16 | }
```

6.计算两点距离

getDistance

传入两个点的经纬度，计算两点的距离

传参：

```
1 | {
2 |   "startLatitude": 22.579054,
3 |   "startLongitude": 114.067389,
4 |   "endLatitude": 22.57871696606192,
5 |   "endLongitude": 114.06792497377637
6 | }
```

- startLatitude 起点纬度
- startLongitude 起点经度
- endLatitude 终点纬度
- endLongitude 终点经度

返回结果：

```
1 | //成功 单位为m
2 | {"code":200,"desc":"","result":67.15}"
```

H5调用示例：

```
1 | //公司位置和荣耀体验店的距离
2 | let content = {
3 |   "startLatitude": 22.579054,
4 |   "startLongitude": 114.067389,
5 |   "endLatitude": 22.57871696606192,
6 |   "endLongitude": 114.06792497377637
7 | }
8 | if (window.map) {
9 |   let result = window.map.getDistance(JSON.stringify(content))
10 |   alert(result)
11 |   console.log(result)
12 | }
```

待完成功能

mapNav 地图导航

1.地图显示当前位置

showLocationInMap

打开地图页面，并展示当前的位置

传参：无需

返回结果：

```
1 //成功
2 {"code":200,"desc":"","result":""}
```

H5调用示例：

```
1 if (window.mapNav) {
2     let result = window.mapNav.showLocationInMap()
3     alert(result)
4     console.log(result)
5 }
```

2.步行规划

walkingRouteSearch

传参：

```
1 {
2     "startName": "腾讯大厦",
3     "startCityName": "深圳",
4     "endName": "新一代产业园",
5     "endCityName": "深圳",
6 }
```

- `startName` 起点名称
- `endName` 终点名称
- `startCityName` 起点所在城市名
- `endCityName` 终点所在城市名

返回结果：

```
1 //成功
2 {"code":200,"desc":"","result":""}
```

H5调用示例：

```
1  if (window.mapNav) {
2      let content = {
3          "startName": "腾讯大厦",
4          "startCityName": "深圳",
5          "endName": "新一代产业园",
6          "endCityName": "深圳",
7      }
8      let result = window.mapNav.walkingRouteSearch(JSON.stringify(content))
9      alert(result)
10     console.log(result)
11 }
```

3.骑行规划

ridingRouteSearch

传参:

```
1  {
2      "startName": "腾讯大厦",
3      "startCityName": "深圳",
4      "endName": "新一代产业园",
5      "endCityName": "深圳",
6  }
```

- `startName` 起点名称
- `endName` 终点名称
- `startCityName` 起点所在城市名
- `endCityName` 终点所在城市名

返回结果:

```
1  //成功
2  {"code":200,"desc":"","result":""}
```

H5调用示例:

```
1  if (window.mapNav) {
2      let content = {
3          "startName": "腾讯大厦",
4          "startCityName": "深圳",
5          "endName": "新一代产业园",
6          "endCityName": "深圳",
7      }
8      let result = window.mapNav.ridingRouteSearch(JSON.stringify(content))
9      alert(result)
10     console.log(result)
11 }
```

4.驾车规划

drivingRouteSearch

传参:

```
1  {
2    "startName": "腾讯大厦",
3    "startCityName": "深圳",
4    "endName": "新一代产业园",
5    "endCityName": "深圳",
6  }
```

- `startName` 起点名称
- `endName` 终点名称
- `startCityName` 起点所在城市名
- `endCityName` 终点所在城市名

返回结果:

```
1  //成功
2  {"code":200,"desc":"","result":""}
```

H5调用示例:

```
1  if (window.mapNav) {
2    let content = {
3      "startName": "腾讯大厦",
4      "startCityName": "深圳",
5      "endName": "新一代产业园",
6      "endCityName": "深圳",
7    }
8    let result = window.mapNav.drivingRouteSearch(JSON.stringify(content))
9    alert(result)
10   console.log(result)
11 }
```

5.步行导航

walkNavigation

传参:

```
1  {
2    "endName": "新一代产业园",
3  }
```

- `endName` 终点地址名称（限制为当前位置的城市）

返回结果:

```
1  //成功
2  {"code":200,"desc":"","result":""}
```

H5调用示例:

```
1  if (window.mapNav) {
2      let content = {
3          "endName": "三九花园",
4      }
5      let result = window.mapNav.walkNavigation(JSON.stringify(content))
6      alert(result)
7      console.log(result)
8  }
```

6.骑行导航

cycleNavigation

传参:

```
1  {
2      "endName": "新一代产业园",
3  }
```

- `endName` 终点地址名称（限制为当前位置的城市）

返回结果:

```
1  //成功
2  {"code":200,"desc":"","result":""}
```

H5调用示例:

```
1  if (window.mapNav) {
2      let content = {
3          "endName": "三九花园",
4      }
5      let result = window.mapNav.cycleNavigation(JSON.stringify(content))
6      alert(result)
7      console.log(result)
8  }
```

待完成功能

1. 地图选点

hardware

1.获取当前电量数值

getBattery

获取当前电量数值

传参: 无需

返回结果:

```
1 //成功
2 {"code":200,"desc":"","result":"98%"}
```

H5调用示例:

```
1 if (window.hardware) {
2     let result = window.hardware.getBattery()
3     alert(result)
4     console.log(result)
5 }
```

2.获取当前网络状态

getNetworkState

获取当前网络状态

传参: 无需

返回结果:

```
1 //成功
2 {"code":200,"desc":"","result":"NETWORK_WIFI"}
```

result字段的数值如下:

- NETWORK_ETHERNET 以太网
- NETWORK_WIFI Wifi
- NETWORK_4G
- NETWORK_3G
- NETWORK_2G
- NETWORK_UNKNOWN 未知网络
- NETWORK_NO 无网络

H5调用示例:

```
1 if (window.hardware) {
2     let result = window.hardware.getNetworkState()
3     alert(result)
4     console.log(result)
5 }
```

3.获取媒体最大音量

getMaxVolume

获取当前网络状态

传参: 无需

返回结果:

```
1 //成功
2 {"code":200,"desc":"","result":"25"}
```

PS:不同手机，返回的最大音量可能会不同

H5调用示例:

```
1  if (window.hardware) {  
2      let result = window.hardware.getMaxVolume()  
3      alert(result)  
4      console.log(result)  
5  }
```

3.获取媒体音量

getVolume

获取当前媒体音量

传参: 无需

返回结果:

```
1  //成功  
2  {"code":200,"desc":"","result":"11"}
```

H5调用示例:

```
1  if (window.hardware) {  
2      let result = window.hardware.getVolume()  
3      alert(result)  
4      console.log(result)  
5  }
```

4.设置媒体音量

setVolume

设置当前的媒体音量

传参:

```
1  {  
2      content: "11"  
3  }
```

返回结果:

```
1  //成功  
2  {"code":200,"desc":"","result":""}
```

H5调用示例:


```

1  if (window.hardware) {
2      let content = {
3          content: "11"
4      }
5      let result = window.hardware.setVolume(JSON.stringify(content))
6      alert(result)
7      console.log(result)
8  }

```

5.获取屏幕亮度

getBrightness

获取当前屏幕亮度

传参：无需

返回结果：

```

1  //成功
2  {"code":200,"desc":"","result":"11"}

```

H5调用示例：

```

1  if (window.hardware) {
2      let result = window.hardware.getBrightness()
3      alert(result)
4      console.log(result)
5  }

```

6.设置屏幕亮度

setBrightness

设置当前的屏幕亮度

传参：

```

1  {
2      content: "11"
3  }

```

注：亮度范围为0-255

返回结果：

```

1  //成功
2  {"code":200,"desc":"","result":""}

```

由于设置屏幕亮度需要权限的申请，所以第一次设置屏幕亮度会出现不成功（因为是申请权限操作），返回的结果为

```

1  {"code":202,"desc":"未授予权限，设置亮度失败","result":""}

```

需要注意判断一下

H5调用示例:

```
1  if (window.hardware) {
2      let content = {
3          content: "11"
4      }
5      let result = window.hardware.setBrightness(JSON.stringify(content))
6      alert(result)
7      console.log(result)
8  }
```

未完成

- 0. NFC
- 1. 蓝牙配对
- 2. 指纹识别
- 3. 人脸识别 (包含活体检测)

debugger

注意：由于配置页面中需要扫一扫的功能，目前是复用了media的Module中的扫一扫页面，导致使用此模块需要前提引入media的模块

1.跳转到debugger页面

goSettingPage

跳转到debugger页面，可输入一个在线地址用来测试页面功能（也含有扫一扫功能）

传参：无需

返回结果：

```
1  //成功
2  {"code":200,"desc":"","result":""}
```

H5调用示例:

```
1  if (window.debugger) {
2      let result = window.debugger.goSettingPage()
3      console.log(result)
4  }
```

2.Bugly配置

目前引入了Bugly对接，注意公司的网络环境对Bugly有限制，无法测试，建议使用移动网络

使用步骤：

在AndroidManifest文件中，声明application，并修改Bugly的配置

```
<application
  android:name="com.tyky.webviewBase.BaseApplication"
  android:allowBackup="true"

  android:icon="@mipmap/ic_launcher"
  android:label="AndroidComponentLibrary"
  android:roundIcon="@mipmap/ic_launcher_round"
  android:supportsRtl="true"
  android:networkSecurityConfig="@xml/network_security_config"
  android:theme="@style/Theme.AndroidComponentLibrary"
  tools:targetApi="n">

  <activity android:name=".MainActivity">
    <intent-filter>
      <action android:name="android.intent.action.MAIN" />

      <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
  </activity>

  <!-- Bugly配置 -->
  <!-- 配置APP ID -->
  <meta-data
    tools:node="replace"
    android:name="BUGLY_APPID"
    android:value="4ec93574ce" />
  <!-- 配置APP版本号 -->
  <meta-data
    tools:node="replace"
    android:name="BUGLY_APP_VERSION"
    android:value="test-version" />
  <!-- 配置APP渠道号 -->
  <meta-data
    tools:node="replace"
    android:name="BUGLY_APP_CHANNEL"
    android:value="m-debug" />
  <!-- 配置Bugly调试模式 (true或者false) -->
  <meta-data
    tools:node="replace"
    android:name="BUGLY_ENABLE_DEBUG"
    android:value="true" />

</application>
```

注: debugger默认有个appId, 下面这个是在APP的AndroidManifest.xml文件, 会将debugger模块中的 AndroidManifest.xml 覆盖掉

```
1 <!-- Bugly配置 -->
2 <!-- 配置APP ID -->
3 <meta-data
4   tools:node="replace"
5   android:name="BUGLY_APPID"
6   android:value="4ec93574ce" />
```

```

7  <!-- 配置APP版本号 -->
8  <meta-data
9      tools:node="replace"
10     android:name="BUGLY_APP_VERSION"
11     android:value="test-version" />
12 <!-- 配置APP渠道号 -->
13 <meta-data
14     tools:node="replace"
15     android:name="BUGLY_APP_CHANNEL"
16     android:value="m-debug" />
17 <!-- 配置Bugly调试模式（true或者false） -->
18 <meta-data
19     tools:node="replace"
20     android:name="BUGLY_ENABLE_DEBUG"
21     android:value="true" />

```

如果有需求需要改动，只需要在主Module中，修改上述数值即可

注意：BaseApplication中是使用了**反射+注解**的方式实现了Module自动初始化（假如Module中使用的库需要在Application中进行初始化）

剩余功能

- h5是否也要做成可配置的方式，来实现更换接口地址（优先读storege里的数据），入口设计思路为依次按下音量键+和-次
- 360加固快捷打包

update使用

更新后台的部署操作，及更新

考虑几套更新策略

page

1.跳转其他APP页面

gotoApp

跳转其他APP页面

传参：

```

1  {
2      "packageName": "com.android.settings", //app应用应用包名
3      "activityName": "com.android.settings.Settings" //具体的某个页面Activity（全
    包名）
4  }
5

```

返回结果：

```

1  //成功
2  {"code":200,"desc":"","result":""}

```

H5调用示例:

```
1  if (window.page) {
2      //下面这里是进入设置页面
3      let content = {
4          "packageName": "com.android.settings",
5          "activityName": "com.android.settings.Settings"
6      }
7      let result = window.page.gotoApp(JSON.stringify(content))
8      console.log(result)
9  }
```

2.使用手机外部浏览器打开链接

openUrlByBrowser

使用手机的浏览器，打开链接

传参:

```
1  {
2      "content": "https://www.baidu.com",
3  }
4
```

返回结果:

```
1  //成功
2  {"code":200,"desc":"","result":""}
```

H5调用示例:

```
1  if (window.page) {
2      //下面这里是进入设置页面
3      let content = {
4          "content": "https://www.baidu.com/",
5      }
6      let result = window.page.openUrlByBrowser(JSON.stringify(content))
7      console.log(result)
8  }
```

tim

腾讯云IM模块

目前使用此模块，会弹出有个错误提示（应用频繁停止），怀疑是存储权限没有给到导致，但之后授权重新打开APP，就不会出现此问题

依赖之后，需要配置主module的Androidmanifest文件

```
1  android:name="com.tencent.qcloud.tim.demo.DemoApplication"
2  tools:replace="android:allowBackup"
```

1.进入登录页

gotoImLogin 进入IM内置的登录页

传参：无

返回结果：

```
1 //成功
2 {"code":200,"desc":"","result":""}
```

H5调用示例：

```
1 if (window.tim) {
2     window.tim.gotoImLogin()
3 }
```

2.IM登录

login

传参：

```
1 {
2     "userId": "5913635",//im的userId
3     "userSig": "",//im的签名，不传会使用内置的sdkid计算
4     "isGotoPage": true //是否登录后跳转到IM的会话列表页面
5 }
```

返回结果：

```
1 //成功
2 {"code":200,"desc":"","result":""}
```

H5调用示例：

```
1 if (window.tim) {
2     let content = {
3         "userId": "5913635",
4         "userSig": "",
5         "isGotoPage": true
6     }
7     let result = window.tim.login(JSON.stringify(content))
8     alert(result)
9 }
```

3.IM退出

logout

传参：无需

返回结果：

```
1 //成功
2 {"code":200,"desc":"","result":""}
```

H5调用示例:

```
1 if (window.tim) {
2     let result = window.tim.logout()
3     alert(result)
4 }
```

4.获取消息未读数

getUnreadCount

使用前，需要调用上述2中的IM登录

传参:

```
1 {
2     "callbackMethod": "loadUnreadCount",
3 }
```

返回结果:

```
1 //成功
2 {"code":200,"desc":"","result":""}
```

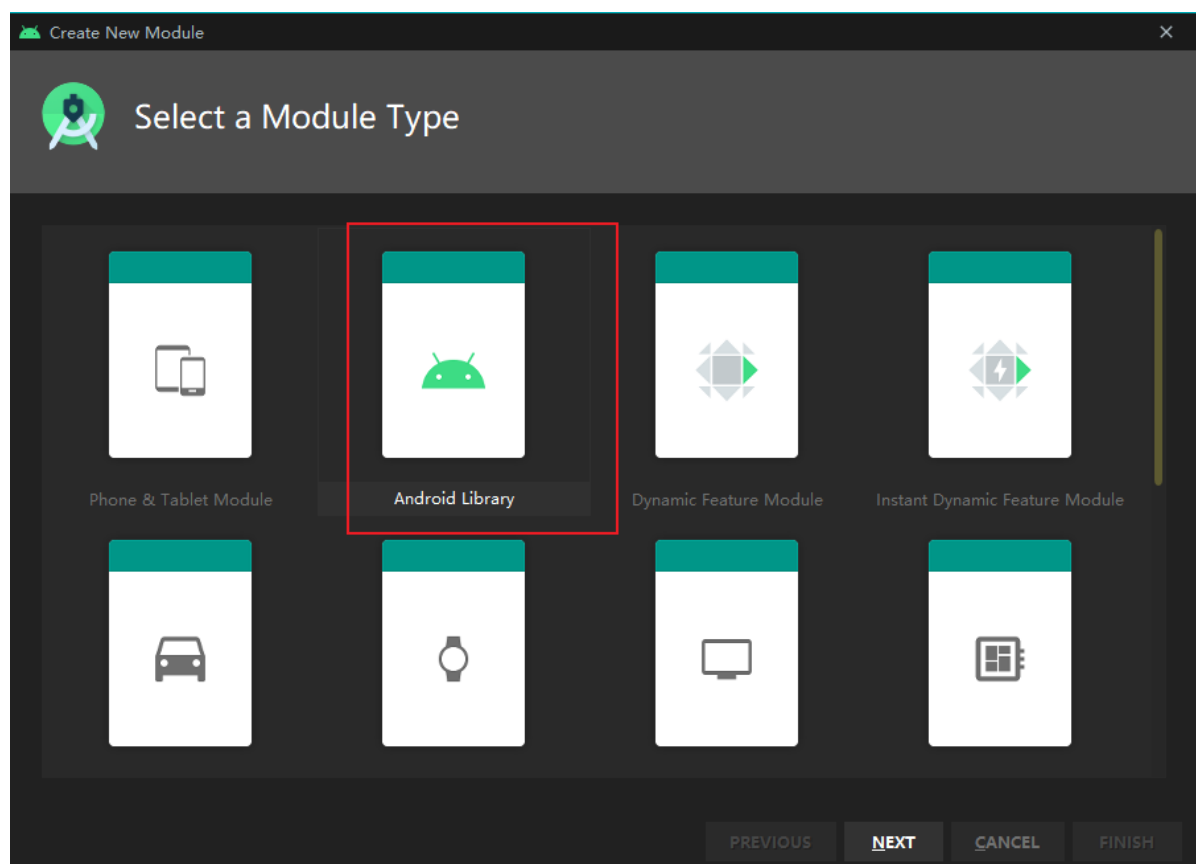
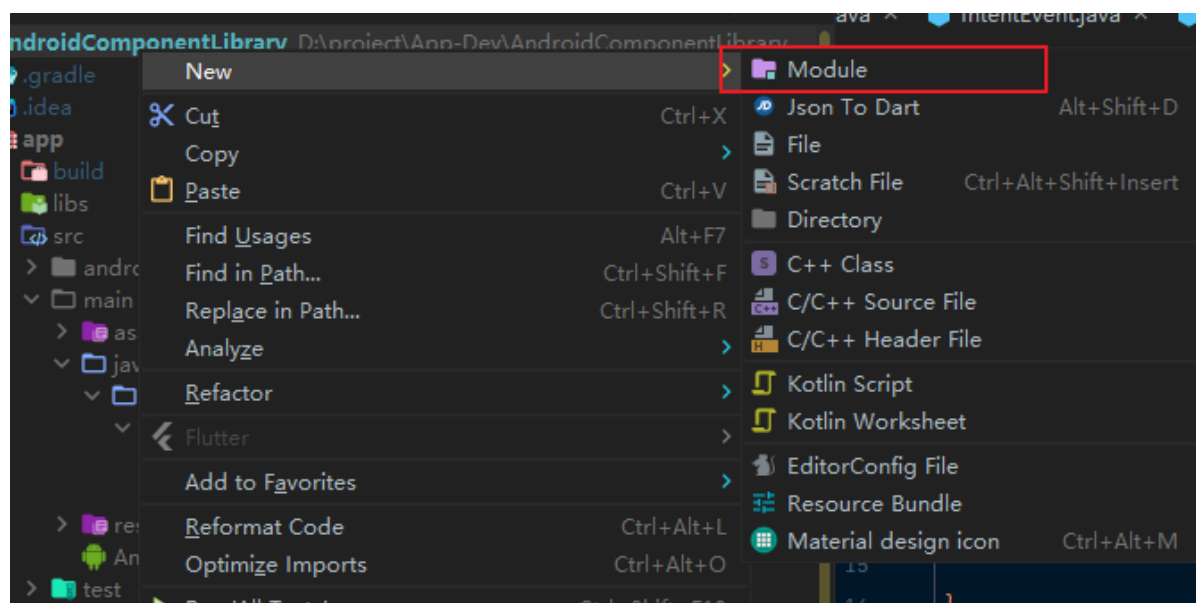
H5调用示例:

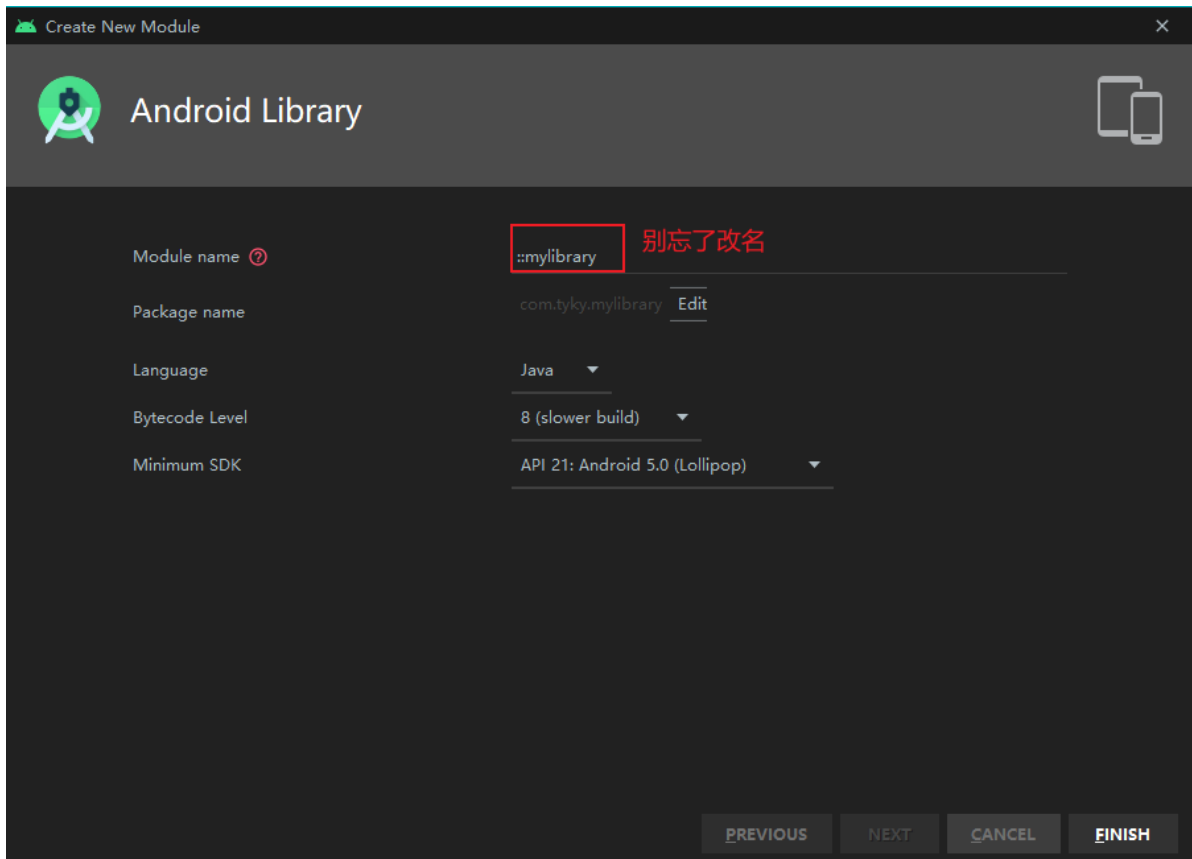
```
1 //返回数据以回调方式的回传给h5
2 window.loadUnreadCount = function(object){
3     alert(JSON.stringify(object))
4 }
5
6 if (window.tim) {
7     let content = {
8         "callbackMethod": "loadUnreadCount",
9     }
10    let result = window.tim.getUnreadCount(JSON.stringify(content))
11    console.log(result)
12 }
```

关于Module创建

若是想要创建一个module，需要进行以下步骤

1.新建module

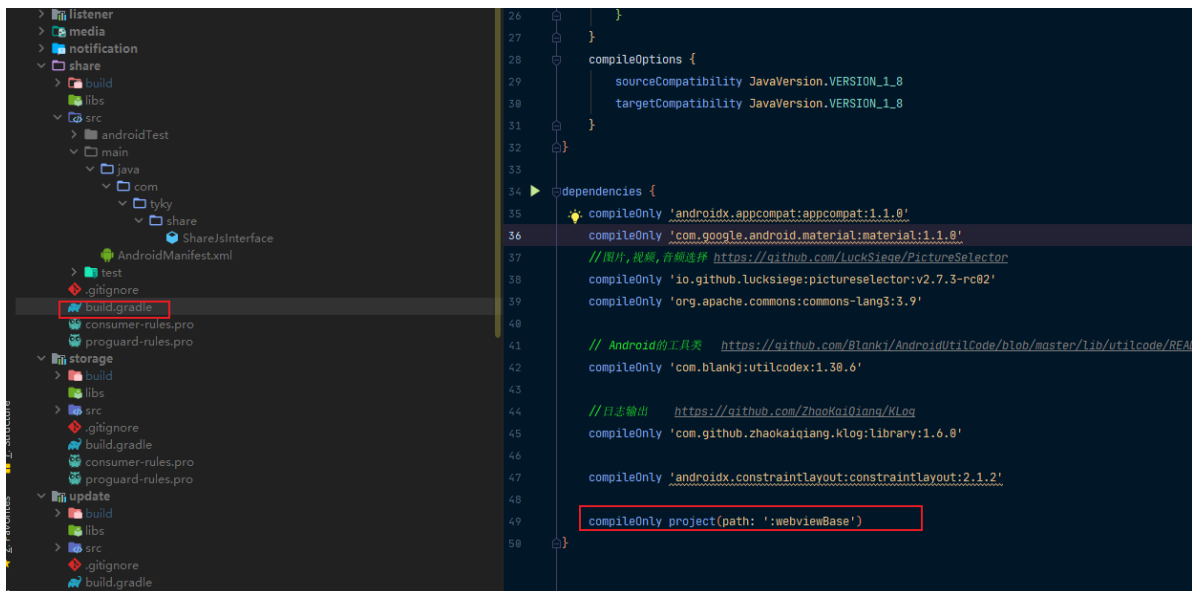




注意：由于封装了自动扫描包实现了Js注入功能，规定包名一定要为 `com.tyky`

2.module依赖修改

由于新建的module，没有核心库 `webViewBase` 的依赖，所以，需要导入依赖

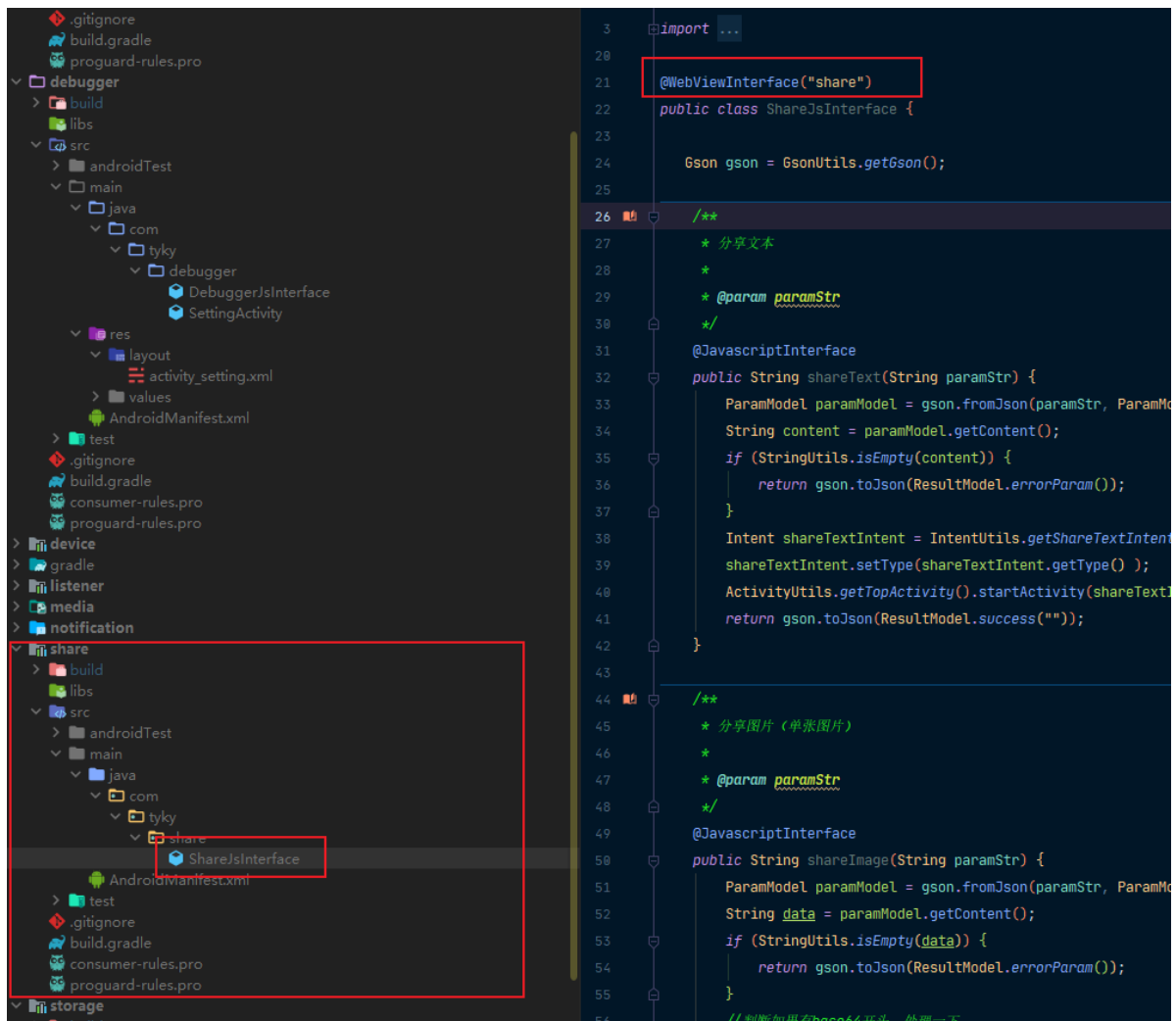


上面的截图，由于module都是与webViewBase在同一项目中，所以可以这样写

实际上，这部分源码在对应的项目中是不存在的，所以，需要导入jitpack上的那个 `webViewBase` 依赖

3.创建Js注入类

使用Gradle进行sync，之后在新建的module中创建一个类，并标上注解 `WebViewInterface`



@WebViewInterface("share")

此注解是个人自定义的一个注解，主要是实现了自动注入Js方法的功能，而无需去设置webview，可以十分方便的实现功能扩展

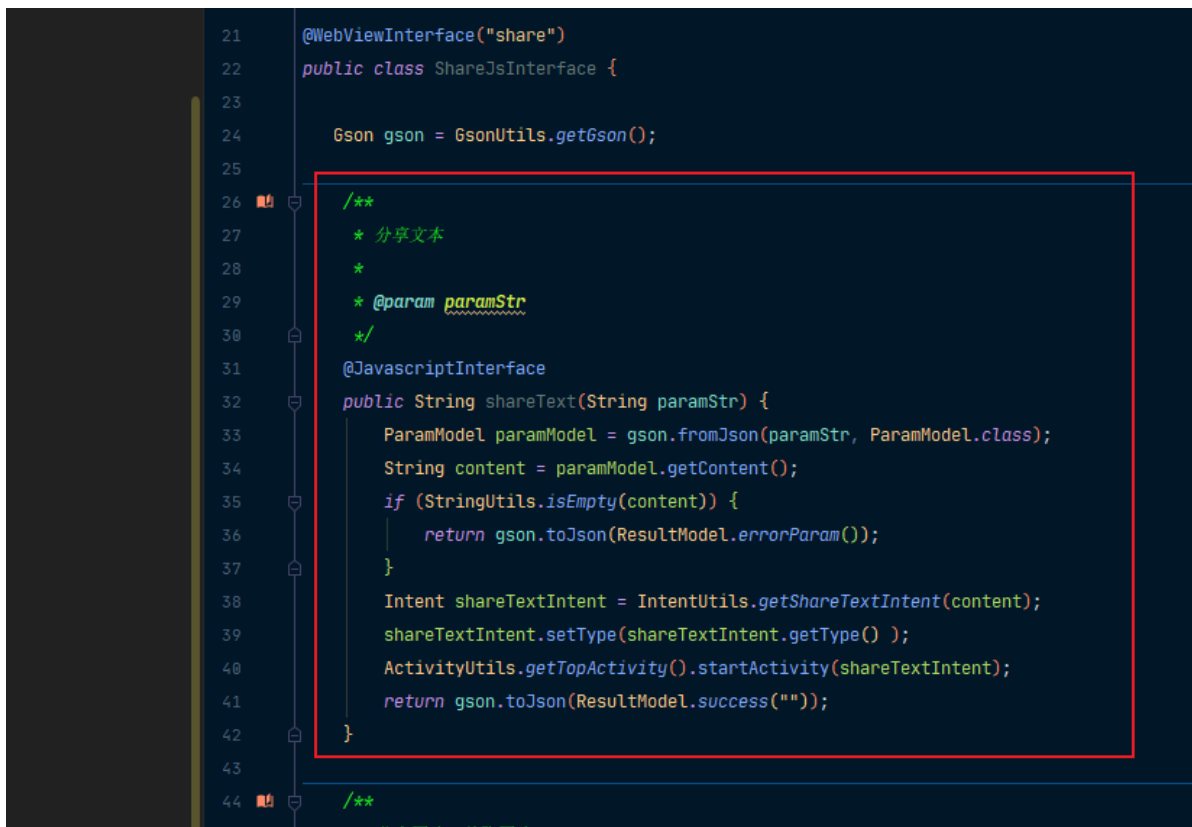
后面的参数为之后注入到js中的使用方式

以上面为例：

之后注入到js代码中，就可以在js层通过 `window.share.xx()` 去调用我们原生Android的方法

4.创建js注入方法

这里不再过多赘述，就写个注解即可



如果想要保持返回结果规范，可以参考上述代码

PS: 之后使用不要忘记了依赖自己新建的module哦!!

补充

需要Application进行初始化操作

目前是使用了注解+反射方式，实现了Application的初始化，下面介绍一下步骤：

- 1.新建类，标明注解 `@ApplicationInit`
- 2.实现接口 `ModuleInit`

```
1 @ApplicationInit
2 public class UpdateModuleInit implements ModuleInit {
3
4     @Override
5     public void init(Application application) {
6         //写你的初始化操作
7     }
8 }
```

PS: 由于采用的包名扫描方式，**需要注意包名得在 `com.tyky` 下**

清单文件怎么写？

如果需求是要在module新建页面或者其他四大组件，可以在AndroidManifest文件中这样写对应的声明：

下面给出的Activity的声明，其他组件类似，这里不再赘述

```
1 <application>
2     <activity android:name=".SettingActivity"></activity>
3 </application>
```

如果是需要权限的，也是在此声明对应的权限，同时也要考虑到动态权限申请的情况

比较好的方式就是卸载对应js方法里,用到就声明

ParamModel实体类字段：

```
1  /**
2   * 电话号码
3   */
4  private String phone;
5
6  /**
7   * 标题
8   */
9  private String title;
10 /**
11  * 内容
12  */
13 private String content;
14 /**
15  * 类型
16  */
17 private Integer type;
18 /**
19  * 回调方法，多个用逗号隔开
20  */
21 private String callBackMethod;
22
23 /**
24  * shareperference存储需要用到的key和value
25  */
26 private String key;
27 private String value;
```

功能清单

地图

考虑使用哪种地图SDK？ 目前只对接过百度地图SDK，各款地图SDK使用的坐标系有所差异

1. 获取当前位置地址
2. 地图选择
3. 地图显示当前位置
4. 路径规划
5. 路径导航

分享

1. 分享到微信好友
2. 分享给朋友圈
3. 分享到QQ

4. 使用浏览器打开
5. 单纯复制链接

监听回调

1. 网络状态监听
2. 来电监听
3. 返回键监听

媒体

1. 拍照
2. 录制视频
3. 录音
4. 图片选择
5. 图片预览
6. 二维码扫描
7. 拨打电话
8. 发送短信
9. 图片压缩
10. 安装apk
11. 文件下载
12. 获取剪切板内容
13. 写入剪切板内容

页面

1. 设置页面标题
2. 设置页面全屏
3. 设置隐藏导航条及还原
4. 退出APP

通知

1. 提供一个固定格式可发通知，后端服务或者是前端也可以调用实现弹出通知
2. 预先对接友盟SDK推送通知功能

登录（如果有原生实现登录）

1. 提供登录用户信息
2. 判断用户是否登录

页面跳转

1. 跳转登录页面（如果有原生登录功能）
2. 跳转微信小程序

APP存储

1. 提供读写SharePerfences的功能，信息可永久保存（不卸载APP的情况下）
2. 提供判断是否第一次进入APP

硬件

1. NFC
2. 蓝牙配对
3. 指纹识别
4. 人脸识别（包含活体检测）
5. 获取当前电量数值
6. 设置屏幕亮度
7. 获取当前网络状态（数据网络还是Wifi）

开发辅助配置功能（不是提供给H5的）

1. bugly SDK接入
2. 提供配置webview加载的地址（本地资源或是网络url地址）
3. h5是否也要做成可配置的方式，来实现更换接口地址（优先读storege里的数据），入口设计思路为依次按下音量键+和-次
4. 360加固快捷打包

推进流程疑惑点

1.文件预览（pdf，doc，ppt，xls），

目前网上的资料只有实现了pdf和doc的预览，其他的格式没有找到有方案，

pdf可以使用webview进行预览（使用pdf.js），doc则需要解压doc文件，然后实现渲染加载

自己实现比较困难，目前3个思路：

- 1.由后台+前端实现在线预览文档页面，app则可以直接通过访问url来预览文件
- 2.app内置实现，只能实现pdf或doc预览（需要先下载在本地），其他格式不行
- 3.与分享类似，打开一个第三方应用来查看文件（也是需要将文件下载在本地）

2.分享功能，目前是实现的打开一个分享的对话框，然后有用户选择对应的方式，这样是否满足需求？

目前支持文本，图片，图文分享

图片需要有图片的uri地址，但是h5只能传过来数据，是否是要讲图片数据先下载在本机的缓存目录中？需要考虑传多张图片（可以是图片地址或者是base64）

3.下载

由于存储权限的变更，下载实现需要对多不同Android版本进行测试验证，预估可能会花费较多时间，

4.无法解析安装包

360加固导致的某些Android设备无法成功解析安装包，出问题的大多是 Android11或Android12，没有设备，无法复现，且没有解决思路

目前确认apk重新签名是无误的，包含了v1+v2+v3版本的签名

4.通知栏点击的pendintent如何封装

通知有时候需要点击跳转到具体页面，这个pendingintent该如何实现封装，接收前端传的参数？

困难点和疑惑点

1.人脸识别（活体检测）

百度人脸识别

需要对接具体SDK，且SDK无法固定（客户可能会有所要求），无对接经验，需要参考具体SDK文档和demo进行编写，封装

集成SDK导致APP体积过大，是否考虑分离多模块进行打包？

单独库，项目配置文件，自定义gradle脚本，组件化module开发

2.地图导航和路径规划

各个地图采用的定位坐标系不一致，需要应该明确使用哪个平台的地图？还是多套都有所对接，然后按需使用？

地图导航和路径规划是否有必要做？

3.指纹识别

功能是否有需要？无相关开发经验

4.打包问题

是固定使用一个包名来打包H5？固定包名是否会对相关的SDK使用有所影响？

如何动态变更包名，保证是不同的applicationId？

关于动态打包

5.后续更新

拆分module，依赖，后续如何更新？

6.腾讯x5内核

集成之后要从网络中下载渲染引擎，约莫30M

建设目标

1.支持任意的前端框架，提供原生的能力

2.学习成本低，轻量

3.