# 趣看云连线SDK接口文档

趣看云连线SDK接口文档 1

一．QKMeetingCloud(云会议打通sdk) 2

二．QKRTCCloud(通用sdk) 5

三．RtcVideoView 7

scalingType 8

typeMatch 8

typeMisMatch 8

四．QkEnterBean 8

五．RTCVideoEncParam 8

六．QkMeetingListener 9

七．QkRTCListener 11

### 一．QKMeetingCloud(云会议打通sdk)

**1. getInstance(Context context)**

获取QKMeetingCloud实例，可以调用destroyInstance销毁单例对象  
 **2. destroyInstance()** 销毁QKMeetingCloud实例

**3. setDebugMode(boolean isDebug)**

设置调试模式

|  |  |
| --- | --- |
| isDebug | True 开启调试模式，输出日志  False 关闭调试模式，线上模式需要关闭 |

**4.enterRoom(QkEnterBean enterBean)**

加入房间，详见QkEnterBean中的定义  
 **5.exitRoom()**

退出房间  
 **6.setListener(QkMeetingListener listener)**

设置回调接口，用户获得各种sdk的状态通知，详见QkMeetingListener中的定义  
 **7.startLocalPreview(boolean isFront, RtcVideoView view)**

开启本地视频的预览

|  |  |
| --- | --- |
| Isfront | True 前置摄像头  False 后置摄像头 |
| View | 渲染视频的控件 |

**8.stopLocalPreview()**

关闭本地视频采集和预览  
 **9.startPush()**

开启推流 **10.stopPush()**

关闭推流  
 **11.startRemoteView(String id, String type, RtcVideoView view)**

开始显示远端的画面，在收到 SDK 的 onUserVideoAvailable(userId, true) 通知时，可以获知该远程用户开启了视频， 此后调用 startRemoteView(userId) 接口加载该用户的远程画面

|  |  |
| --- | --- |
| Id | 用户id |
| Type | 视频流类型  QKRTCDef.TYPE\_MAIN  摄像头画面  QKRTCDef.TYPE\_SCREEN  屏幕分享画面 |
| View | 显示视频的控件 |

**12.stopRemoteView(String id, String type)**

停止显示远端的画面，同时不拉取用户的对应流数据

|  |  |
| --- | --- |
| Id | 用户id |
| Type | 视频流类型  QKRTCDef.TYPE\_MAIN  摄像头画面  QKRTCDef.TYPE\_SCREEN  屏幕分享画面 |

**13.switchCamera(boolean isFront)**

切换摄像头

|  |  |
| --- | --- |
| Isfront | True 前置摄像头  False 后置摄像头 |

**14.enableAudio(boolean enable)**

开关音频

|  |  |
| --- | --- |
| Enable | True 开启  False 关闭 |

**15.enableBeauty(boolean enable)**

开关美颜

|  |  |
| --- | --- |
| Enable | True 开启  False 关闭 |

**16.setVideoEncoderParam(RTCVideoEncParam param)**

设置视频推流参数，详见RTCVideoEncParam  
 **17.hangupById(String id)**

挂断某个用户

|  |  |
| --- | --- |
| Id | 用户id |

**18.muteById(String id)**

禁言用户

|  |  |
| --- | --- |
| Id | 用户id |

**19.closeVideoById(String id)** 关闭用户的摄像头

|  |  |
| --- | --- |
| Id | 用户id |

**20.muteAll()** 禁言房间的所有用户

**21.getAllMember()**

返回所有推流用户的信息，会回调QkMeetingListener中的onGetAllmember **22.getMemberById(String id)**

获取某个推流用户的信息，不存在则放回null

|  |  |
| --- | --- |
| Id | 用户id |

### 二．QKRTCCloud(通用sdk)

**1.getInstance(Context context)**

获取QKRTCCloud实例，可以调用destroyInstance销毁单例对象  
 **2. destroyInstance()** 销毁QKRTCCloud实例

**3. setDebugMode(boolean isDebug)**

设置调试模式

|  |  |
| --- | --- |
| isDebug | True 开启调试模式，输出日志  False 关闭调试模式，线上模式需要关闭 |

**4.enterRoom(QkEnterBean enterBean)**

加入房间，详见QkEnterBean中的定义  
 **5.exitRoom()**

退出房间  
 **6.setListener(QkRTCListener listener)**

设置回调接口，用户获得各种sdk的状态通知，详见QkRTCListener中的定义  
 **7.startLocalPreview(boolean isFront, RtcVideoView view)**

开启本地视频的预览

|  |  |
| --- | --- |
| Isfront | True 前置摄像头  False 后置摄像头 |
| View | 渲染视频的控件 |

**8.stopLocalPreview()**

关闭本地视频采集和预览  
 **9.startPush()**

开启推流 **10.stopPush()**

关闭推流  
 **11.startRemoteView(String id, String type, RtcVideoView view)**

开始显示远端的画面在收到 SDK 的 onUserVideoAvailable(userId, true) 通知时，可以获知该远程用户开启了视频， 此后调用 startRemoteView(userId) 接口加载该用户的远程画面

|  |  |
| --- | --- |
| Id | 用户id |
| Type | 视频流类型  QKRTCDef.TYPE\_MAIN  摄像头画面  QKRTCDef.TYPE\_SCREEN  屏幕分享画面 |
| View | 显示视频的控件 |

**12.stopRemoteView(String id, String type)**

停止显示远端的画面，同时不拉取用户的对应流数据

|  |  |
| --- | --- |
| Id | 用户id |
| Type | 视频流类型  QKRTCDef.TYPE\_MAIN  摄像头画面  QKRTCDef.TYPE\_SCREEN  屏幕分享画面 |

**13.switchCamera(boolean isFront)**

切换摄像头

|  |  |
| --- | --- |
| Isfront | True 前置摄像头  False 后置摄像头 |

**14.enableAudio(boolean enable)**

开关音频

|  |  |
| --- | --- |
| Enable | True 开启  False 关闭 |

**15.enableBeauty(boolean enable)**

开关美颜

|  |  |
| --- | --- |
| Enable | True 开启  False 关闭 |

**16.setVideoEncoderParam(RTCVideoEncParam param)**

设置视频推流参数，详见RTCVideoEncParam

**17. startScreenCapture(Intent screenData, RtcVideoView view);**

开启屏幕共享

|  |  |
| --- | --- |
| Screendata | 系统放回的intent |
| View | 显示视频的控件 |

**18. stopScreenCapture();**

关闭屏幕分享

**19. switchRole(String role);**

切换角色

|  |  |
| --- | --- |
| Role | QKRTCDef.RULE\_WATCHER  围观者，不能推流  QKRTCDef.RULE\_PARTNER  嘉宾，可以推流 |

### 三．RtcVideoView

**1. setMirror(boolean mirror)**

设置镜像显示

|  |  |
| --- | --- |
| mirror | True 镜像 false 不镜像 |

**2. setScalingType(ScalingType typeMatch， ScalingType typeMisMatch)**

设置画面填充方式

|  |  |
| --- | --- |
| scalingType | SCALE\_ASPECT\_FILL拉伸画面  SCALE\_ASPECT\_FIT 不拉伸  SCALE\_ASPECT\_BALANCED均衡 |
| typeMatch | 画面方向匹配 |
| typeMisMatch | 画面方向不匹配 |

### 四．QkEnterBean

|  |  |  |
| --- | --- | --- |
| appKey | Appkey | 都要传 |
| sig | 用户sign（最好通过服务器获取，本地加密会泄漏秘钥）  待签名内容content：session\_id\_role\_roomId  签名算法：sig = Base64(HmacSHA1(content, appSecret))  注：appkey和appSecret由趣看提供  示例代码  String sig = ""; String session = UUID.randomUUID().toString(); String role = QkRTCDef.RULE\_PARTNER; String id = "1"; String roomId = "111"; try {  String content = session + "\_" + id + "\_" + role + "\_" + roomId;  sig = new String(Base64.encode(HMACSHA1.HmacSHA1Encrypt(content, "appSecret"),  Base64.NO\_WRAP), StandardCharsets.UTF\_8); } catch (Exception e) {  Log.e("error", e.getMessage()); } | 都要传 |
| id | 用户id | 都要传 |
| roomId | 房间id | 都要传 |
| session | 唯一标识 | 都要传 |
| role | 角色 | 都要传 |
| clientType | 设备类型 | 云会议房间 |
| battery | 电量 | 云会议房间 |
| name | 用户名 | 云会议房间 |
| saveRecord | true 保存录像  false 不保存 | 普通房间 |
| isCloudMeeting | True 进入云会议房间  Fasle 普通房间 | 都传 |

### 五．RTCVideoEncParam

|  |  |
| --- | --- |
| videoResolution | 分辨率  QkRTCDef.QK\_RTC\_RESOLUTION\_1080X1920  QkRTCDef.QK\_RTC\_RESOLUTION\_720X1280  QkRTCDef.QK\_RTC\_RESOLUTION\_540X960 |
| videoFps | 帧率 |
| videoBitrate | 码率 |

### 六．QkMeetingListener

**1.onEnterRoom()**

加入房间成功  
 **2.onExitRoom()**

离开房间  
 **3.onError(int errCode, String errMsg)**

错误信息

|  |  |
| --- | --- |
| errCode | 错误码  ERR\_NULL = 0;  ERR\_ROOM\_ENTER\_FAIL = -1101;  加入房间失败 ERR\_ROOM\_LEAVE\_FAIL = -1111;  离开房间失败 ERR\_SOCKET\_EVENT\_ERROR = -1103;  socket错误 ERR\_SOCKET\_NOT\_CONNECT = -1105;  socket未连接 ERR\_SCREEN\_CANCEL = -1104;  屏幕分享取消 ERR\_SCREEN\_FAIL = -1106;  屏幕分享失败 ERR\_PUSH\_FAIL = -1107;  推流失败 |
| errMsg | 错误信息 |

**4.onUserVideoAvailable(String id, boolean available)**

远端用户是否存在可播放的视频流

|  |  |
| --- | --- |
| Id | 用户id |
| available | 存在（用户开始推流）  不存在（用户停止推流） |

**5.onUserScreenAvailable(String id, boolean available)**

远端用户是否存在可播放的屏幕分享流

|  |  |
| --- | --- |
| Id | 用户id |
| available | 存在（用户开始屏幕分享）  不存在（用户停止屏幕分享） |

**6.onOffLine()** 异地登录 **7.onStatsReport(StatsReport[] statsReports)** 推流信息回调  
 **8.onKickOut()** 被踢出房间  
 **9.onRoomClose(String msg)**

房间关闭 **10.onMute()**

被禁言  
 **11.onRoleUpdate(String role)**

角色被改变 **12.onCameraClose()**

摄像头被关闭  
  
**13.onRemoteMsgUpdate(QkMemberBean memberBean)**

远端用户信息改变，远端用户的摄像头麦克风信息改变时触发  
 **14.onGetAllMember(List<QkMemberBean> list)**

获取到的所有用户信息getAllMember()的回调

### 七．QkRTCListener

**1.onEnterRoom()**

加入房间成功  
 **2.onExitRoom()**

离开房间  
 **3.onError(int errCode, String errMsg)**

错误信息

|  |  |
| --- | --- |
| errCode | 错误码  ERR\_NULL = 0;  ERR\_ROOM\_ENTER\_FAIL = -1101;  加入房间失败 ERR\_ROOM\_LEAVE\_FAIL = -1111;  离开房间失败 ERR\_SOCKET\_EVENT\_ERROR = -1103;  socket错误 ERR\_SOCKET\_NOT\_CONNECT = -1105;  socket未连接 ERR\_SCREEN\_CANCEL = -1104;  屏幕分享取消 ERR\_SCREEN\_FAIL = -1106;  屏幕分享失败 ERR\_PUSH\_FAIL = -1107;  推流失败 |
| errMsg | 错误信息 |

**4.onUserVideoAvailable(String id, boolean available)**

远端用户是否存在可播放的视频流

|  |  |
| --- | --- |
| Id | 用户id |
| available | 存在（用户开始推流）  不存在（用户停止推流） |

**5.onUserScreenAvailable(String id, boolean available)**

远端用户是否存在可播放的屏幕分享流

|  |  |
| --- | --- |
| Id | 用户id |
| available | 存在（用户开始屏幕分享）  不存在（用户停止屏幕分享） |

**6.onOffLine()** 异地登录 **7.onStatsReport(StatsReport[] statsReports)** 推流信息回调