

播放直播

1 简介

登录房间成功后，SDK 会自动拉流播放直播。用户无需手动触发拉流。

但如果想获取播放流状态，需要进行代理设置。

2 步骤

2.1 初始化 SDK

参考文档：[快速开始-初始化](#)

2.2 设置播放代理对象

调用如下 API 设置 ZegoAudioLivePlayerDelegate 代理对象。设置代理对象失败，会导致 App 收不到相关回调。

```
ZegoAudioRoomApi-Player.h
```

```
/**  
 设置 ZegoAudioLivePlayer 代理对象  
  
  @param playerDelegate 遵循 ZegoAudioLivePlayerDelegate 协议的对象  
  @attention 登录房间前，务必要设置代理对象  
  @note 设置代理对象失败，会导致 App 收不到相关回调  
 */  
- (void) setAudioPlayerDelegate: (id<ZegoAudioLivePlayerDelegate>)playerDelegate;
```

2.3 登录房间

调用如下 API 登录房间。无论是手动/自动发布直播模式，登录房间后，SDK 均会自动拉流，无需用户手动操作。

ZegoAudioRoomApi.h

/**

登录房间

@param roomID 房间 ID, App 需保证其唯一性, 长度不可超过 255 bytes

@param block 登录完成回调 block

@return true: 登录成功, false: 登录失败

*/

- (bool) loginRoom:(NSString *)roomID completionBlock:(ZegoAudioRoomBlock)block;

常见登录错误码参考如下:

错误码	说明
stateCode = 0	登录成功
stateCode = -1	没有配置为测试环境, 请调用 [setUseTestEnv:YES]
stateCode = -2	调用次数太频繁, 超出后台限制。请调整调用次数, 每分钟最多 30 次
stateCode = 1042	网络连不上, 请检查当前网络是否正常
stateCode = 4131	网络连接失败, 一般由于网络波动或其他异常导致, 请检查当前网络是否正常
stateCode = 1048677	测试环境或正式环境设置错误。如果使用测试环境, 请确保在初始化 SDK 之前调用 [setUseTestEnv:YES]
stateCode = 1048680	房间不存在, 请检查房间是否存在

请注意, 参数 roomID, 是由 App 业务端定义、实现并传入的。开发者在定义 roomID 时需要确保在同一时刻是唯一的。

2.4 处理播放状态回调

自动播放开始后, 开发者需在如下回调中处理播放状态和流信息。

ZegoAudioRoomApi-Player.h

```
/**
 * 播放状态更新回调
 *
 * @param stateCode 状态码。0 表示无错误
 * @param stream 流信息
 * @note 调用 [ZegoAudioRoomApi (Player) -setAudioPlayerDelegate] 设置代理对象成功后，客户端才能收到此通知
 */
- (void)onPlayStateUpdate:(int)stateCode stream:(ZegoAudioStream *)stream;
```

常见播放错误码参考如下：

错误码	说明
stateCode = 0	拉流成功
stateCode = 3	直播遇到严重错误。stateCode = 1/2/3 基本不会出现
stateCode = 5	获取流信息失败。请检查：1. 推流方是否推流成功。2. 延迟重试
stateCode = 6	流不存在。请检查：1. AppID 是否相同，要保证一致。2. 是否同时开启测试环境或同时在正式环境下
stateCode = 7	媒体服务器连接失败。请检查：1. 推流端是否正常推流。2. 正式环境和测试环境是否设置都是同一个。 3. 网络是否正常
stateCode = 8	DNS 解析失败。请检查网络是否正常
stateCode = 9	未 loginRoom 就直接 play/publish。请确认推流前已 loginRoom
stateCode = 10	逻辑服务器网络错误（网络断开约 3 分钟时会返回该错误）。请检查网络是否正常

2.5 退出房间

调用如下 API 退出房间。无论是手动/自动发布直播模式，退出房间后，会停止所有的拉流，**无需用户手动停止拉流**。

```
ZegoAudioRoomApi.h
```

```
/**
```

```
退出房间
```

```
@return true: 成功, false: 失败
```

```
@note 退出房间成功会停止所有的推拉流
```

```
*/
```

```
- (bool) logoutRoom;
```