

- 1.聊天室
- 2.登录
- 3.回调
- 4.发言
- 5.提问

1.聊天室

聊天室管理器的对外核心类为 `PLVChatroomManager`，可以通过 SDK 对象的 `chatroomManager` 属性对其进行访问。

2.登录

聊天室的消息收发依赖于 socket，因此需要先进行 socket 的登录，后续才能正常使用聊天室。socket 管理器的对外核心类为 `PLVSocketManager`，可以通过 SDK 对象的 `socketManager` 属性对其进行访问：

```
// 监听socket连接状态
sdk.socketManager.onStatus()
// socket登录
sdk.socketManager.login()
```

详细使用代码可以参考 demo 项目的 `PLVLWatchLayout` 类。

3.回调

聊天室的 socket 通道消息、业务事件消息通过回调注册进行监听，包括：

- `onData`：socket 通道消息回调注册
- `eventNotify.on`：业务事件消息回调注册

以监听 socket 通道的公告/移除公告消息为例，可以通过以下方式监听：

```
sdk.chatroomManager.onData(PLVSocketOnEvent.MESSAGE, (data: string, event: string) => {  
    switch (event) {  
        // 公告  
        case PLVBulletinEvent.EVENT:  
            break;  
        // 移除公告  
        case PLVRemoveBulletinEvent.EVENT:  
            break;  
        default:  
            break;  
    }  
}, this)
```

详细使用代码可以参考 demo 项目的 `PLVLWatchLayout` 、 `PLVLChatListLayout` 类。

4.发言

可以使用聊天室的 `speak` 方法进行发言：

```
// message: 要发言的消息, replay: 携带的回复消息, 非回复时为undefined  
sdk.chatroomManager.speak(message, replay)
```

详细使用代码可以参考 demo 项目的 `PLVLChatInputView` 类。

5.提问

可以使用聊天室的 `quiz` 方法进行提问：

```
sdk.chatroomManager.quiz(message)
```

详细使用代码可以参考 demo 项目的 `PLVLChatInputView` 类。