

用户信息

本文档主要提供 用户模块(user) 提供的用户信息 Api 说明。

一、观众详细信息

Interface 接口： `UserInfoDetail`

属性名	说明	类型
<code>userId</code>	用户 id	<code>string</code>
<code>nick</code>	观众昵称	<code>string</code>
<code>pic</code>	观众头像	<code>string</code>
<code>actor</code>	观众头衔	<code>string</code>
<code>openId</code>	微信 openid	<code>string</code>
<code>unionId</code>	微信 unionId	<code>string</code>
<code>authType</code>	该用户的授权方式	<code>AuthType</code>
<code>label</code>	标签	<code>Object</code>

二、获取当前用户信息

用于获取当前用户的详细信息。

未进行观看条件授权或创建 SDK 没有传入用户信息时返回 `undefined`。

Api 方法： `getUserInfo(): undefined | UserInfoDetail`

返回值说明： 用户信息， `undefined | UserInfoDetail` 类型

示例：

```
const userInfo = watchCore.user.getUserInfo();
console.log('用户信息', userInfo);
```

三、判断是否为特殊的用户身份

用户模块提供 Api 用于判断用户身份是否为内置的特殊用户身份，如讲师、管理员、助教等。

Api 方法： `isSpecialUserType(userType: unknown, actor?: string): boolean`

参数说明：

- `userType`：用户身份，`unknown` 类型，必传
- `actor`：用户头衔，`string` 类型，选传

返回值说明： 是否特殊身份

示例：

```
import { ChatUserType } from '@polyv/live-watch-sdk';
console.log(watchCore.user.isSpecialUserType(ChatUserType.Teacher)); // true
console.log(watchCore.user.isSpecialUserType(ChatUserType.Manager)); // true
console.log(watchCore.user.isSpecialUserType(ChatUserType.Guest)); // true
console.log(watchCore.user.isSpecialUserType(ChatUserType.Student)); // false
console.log(watchCore.user.isSpecialUserType(ChatUserType.Viewer)); // false
console.log(watchCore.user.isSpecialUserType(ChatUserType.Dummy)); // false
console.log(watchCore.user.isSpecialUserType(ChatUserType.Dummy, '观众')); // false
```

四、获取当前用户 userId

用于获取当前用户的 `userId`，默认以时间戳字符串作为用户 `userId`，关于 `userId` 说明如下：

- 微信环境下进行微信授权后，以用户的微信 `openid` 作为 `userId`；
- 白名单授权后，以填写的会员码作为 `userId`（优先级高于微信授权）；

未进行观看条件授权或创建 SDK 没有传入用户信息时返回 `undefined`。

Api 方法： `getUserId(): undefined | string`

返回值说明： 用户 `userId`，`undefined | string` 类型

示例：

```
const userId = watchCore.user.getUserId();
console.log('用户 userId', userId);
```

五、获取当前用户昵称

用于获取当前用户的昵称。

未进行观看条件授权或创建 SDK 没有传入用户信息时返回 `undefined`。

Api 方法： `getUserNick(): undefined | string`

返回值说明： 用户昵称，`undefined | string` 类型

示例：

```
const nickname = watchCore.user.getUserNick();
console.log('用户昵称', nickname);
```

六、修改当前用户昵称

用于观众二次修改昵称。

- 通过 [UserEvents.CurrentUserSetNick](#) 事件监听修改昵称完成。
- 通过 [UserEvents.SetNickError](#) 事件监听设置昵称失败。

Api 方法： `updateUserNick(nickname: string): void`

参数说明：

- `nickname`：新的昵称，`string` 类型，必传

示例：

```
watchCore.user.eventEmitter.on(UserEvents.CurrentUserSetNick, () => {
  toast.success('修改昵称成功');
});

watchCore.user.eventEmitter.on(UserEvents.SetNickError, (data) => {
  toast.error('修改昵称失败: ' + data.message);
});

watchCore.user.updateUserNick('新的用户昵称');
```

七、获取当前用户头像地址

用于获取当前用户的头像地址，观看页 SDK 提供 `DEFAULT_VIEWER_AVATAR` 默认观众头像的常量。

未进行观看条件授权或创建 SDK 没有传入用户信息时返回 `undefined`。

Api 方法： `getUserAvatar(): undefined | string`

返回值说明： 用户头像地址， `undefined | string` 类型

示例：

```
import { DEFAULT_VIEWER_AVATAR } from '@polyv/live-watch-sdk';

const userAvatar = watchCore.user.getUserAvatar();
console.log('用户头像', userAvatar);
console.log('默认头像', DEFAULT_VIEWER_AVATAR);
```

八、获取当前用户的微信 openId

用于获取当前用户的微信 openId，仅在微信(非)静默授权后才返回

Api 方法： `getUserOpenId(): undefined | string`

返回值说明： 微信 openId， `undefined | string` 类型

示例：

```
const openId = watchCore.user.getUserOpenId();
console.log('openId', openId);
```

九、获取当前用户的微信 unionId

用于获取当前用户的微信 unionId，仅在微信(非)静默授权后才返回

Api 方法： `getUserUnionId(): undefined | string`

返回值说明： 微信 unionId, `undefined | string` 类型

示例：

```
const unionId = watchCore.user.getUserUnionId();
console.log('unionId', unionId);
```

十、获取当前用户的头衔

Api 方法： `getUserActor(): undefined | string`

从 v0.11.0 版本开始支持

返回值说明： 用户头衔, `undefined | string` 类型

示例：

```
const actor = watchCore.user.getUserActor();
console.log('actor', actor);
```

十一、获取当前用户相关的标签

Api 方法： `getUserLabels(): undefined | Object`

从 v2.18.0 版本开始支持

返回值说明： 用户头衔

```
const labels = watchCore.user.getUserLabels();
console.log('labels', labels);
```，`undefined | Object` 类型


十二、是否需要隐藏用户头衔

Api 方法: `needHideUserActor(): boolean`

> 从 v1.2.0 版本开始支持

十三、获取当前用户的授权方式

用户获取当前用户的授权方式，未授权时返回 `AuthType.None`

Api 方法: `getUserAuthType(): AuthType`

返回值说明: 授权方式, `AuthType` 类型

示例:

```js
const authType = watchCore.user.getUserAuthType();
console.log('当前用户的授权方式', authType);
```

十四、当前是否需要手机号实名认证

Api 方法: `needRealNameAuth(): Promise<boolean>`

从 v0.8.0 版本开始支持

返回值说明: 是否需要认证, `Promise<boolean>` 类型

示例:

```
const needAuth = watchCore.user.needRealNameAuth();
if (needAuth) {
  console.log('需要认证, 显示认证弹层');
}
```

十五、保存实名认证信息

Api 方法: `saveRealNameData(option: SaveRealNameOptions): Promise<RealNameResult>`

从 v0.8.0 版本开始支持

参数说明:

- option: 选项参数, `SaveRealNameOptions` 类型, 必传, 详细类型说明如下

参数名	说明	类型	必须	默认值
<code>areaCode</code>	手机区号	<code>string</code>	是	-
<code>phoneNumber</code>	手机号	<code>string</code>	是	-
<code>smsCode</code>	短信验证码	<code>string</code>	是	-

返回值说明: `Promise<RealNameResult>` 类型

十六、获取当前观众的云席信息

Api 方法: `getUserSeatInfo(): undefined | ViewerSeatInfo`

从 v0.10.0 版本开始支持

返回值说明: `undefined | ViewerSeatInfo` 类型

十七、获取当前观众的跟进人信息

Api 方法: `getUserSaleInfo(): ViewerSaleInfo`

从 v1.1.0 版本开始支持

返回值说明: 用户跟进人信息, `ViewerSaleInfo` 类型, 详细类型说明如下

属性名	说明	类型
saleId	跟进人 id	undefined string

十八、获取邀请观看邀请员/分销员Id

Api 方法: `getInviteWatchSaleId(): null | string`

从 v2.0.0 版本开始支持

返回值说明: `null | string` 类型

十九、获取邀请观看上级邀请员的 inviteWatchSaleId

Api 方法: `getInviteWatchInviteSalesId(): null | string`

从 v2.2.0 版本开始支持

返回值说明: `null | string` 类型