

讲师通过 HTTP 回复学员提问

接口描述

由业务服务端调用，在指定频道内以讲师身份回复某位观众的「提问私聊」记录；行为与聊天室 WebSocket 事件 T_ANSWER 对齐（写入提问记录、更新列表数据、向仍在线的相关用户推送 message）。接口支持 HTTPS 协议。

接口URL

```
https://api.polyv.net/live/v5/chat/redirect/channel/teacher-answer/post
```

请求方式

POST

接口约束

- 1、接口同时支持 HTTP、HTTPS，建议使用 HTTPS 确保接口安全。
- 2、** viewerUserId ** 为被回复的观众在业务里的用户 ID。若该观众当前正以观众身份在目标频道内在线（已连上聊天室），系统可识别其连接并优先用于实时推送；若未在线，仍会落库，但实时推送可能无法送达该观众。
- 3、**讲师展示信息**：若频道内能识别到**已在线的讲师身份**（与频道约定规则一致），将使用该讲师的昵称、头像等；若当前没有在线讲师信息可识别，须在请求里传 `teacherNick`、`teacherPic`，否则展示字段可能为空。
- 4、`msgType` 可选；不传或空则按文本处理。传 `image` 表示图片回复，此时 `content` 须为 JSON 字符串，对象字段为：`width` (number)、`height` (number)、`url` (string)、`id` (string, 可选)，与 WebSocket T_ANSWER 及站内图片提问格式一致；服务端会走图片校验逻辑。

请求体参数描述

参数名	必选	类型	说明
appId	true	String	账号appId【详见 获取密钥 】
timestamp	true	Long	当前13位毫秒级时间戳，3分钟内有效
sign	true	String	签名，为32位大写的MD5值，生成签名的appSecret密钥作为通信数据安全的关键信息，严禁保存在客户端直接使用，所有API都必须通过客户自己服务器中转调用POLYV服务器获取响应数据【详见 签名生成规则 】
roomId	true	String(1,100)	频道号（与观众/讲师在聊天室登录时使用的channelId / roomId 一致）
content	true	String(1,10000)	文本回复时为正文；当msgType 为 image 时须为JSON 字符串，序列化对象格式为 { "width": number, "height": number, "url": string, "id"?: string }
viewerUserId	true	String(1,2000)	被回复的观众 userId（提问侧学员的业务 ID）
teacherNick	false	String	讲师昵称；当频道内没有可识别的在线讲师信息时建议传入，否则展示可能为空
teacherPic	false	String	讲师头像 URL；当频道内没有可识别的在线讲师信息时建议传入

参数名	必选	类型	说明
msgType	false	String	可选。传 <code>image</code> 表示图片回复，与 WebSocket <code>T_ANSWER</code> 一致；不传则按文本回复处理

示例

请求体 JSON：

```
{
  "roomId": "412738",
  "content": "同学你好，这个问题我们在第二节课会讲到。",
  "viewerUserId": "user_abc_001",
  "teacherNick": "王老师",
  "teacherPic": "https://liveimages.videocc.net/defaultImg/avatar/viewer.png"
}
```

图片回复示例（`msgType` 为 `image`，`content` 为 JSON 字符串）：

```
{
  "roomId": "412738",
  "viewerUserId": "user_abc_001",
  "msgType": "image",
  "content": "{\"width\":640,\"height\":360,\"url\":\"https://example.com/img.png\",\"id\":\"img_001\"}",
  "teacherNick": "王老师",
  "teacherPic": "https://liveimages.videocc.net/defaultImg/avatar/viewer.png"
}
```

响应体 JSON（成功）：

```
{
  "code": 200,
  "status": "success",
  "message": "发送成功",
  "data": {
    "id": 123456
  }
}
```

其中 `data.id` 为该条回复写入提问流水后的记录 ID（与 socket 回调中 `T_ANSWER` 的 `id` 含义一致）。

WebSocket 推送说明

调用成功后，服务端会向聊天室推送 `message` 事件，内容为 JSON 字符串，解析后 `**EVENT` 为 `T_ANSWER` `**`，主要字段包括：`roomId`、`content`、`user`（讲师展示信息）、`s_userId`（被回复的观众 `userId`）、`id`（本条回复 `id`）、`msgType` 等。

- **观众**：若该观众当前仍在该频道聊天室中在线，一般能收到上述推送；若已离线或未连接，则无法收到实时消息，但历史提问列表中仍可出现本条回复（以各端列表接口为准）。
- **讲师**：若讲师当前正以讲师身份在该频道内在线，一般能收到推送；若讲师未在线或未以可识别身份连接，则可能收不到实时推送，但记录仍会写入。

通过讲师端 WebSocket 自行发送 `T_ANSWER` 时，还可能向整个频道房间广播；本 HTTP 接口以服务端判定在线连接并定向推送为主，具体与实现保持一致。

响应参数描述

参数名	类型	说明
code	Integer	200 表示成功；400 多为参数校验失败；500 多为业务错误或服务端异常
status	String	<code>success</code> / <code>fail</code> / <code>error</code>
message	String	提示文案，如「发送成功」或具体错误说明
data	Object	成功时包含 <code>id</code> （回复记录 ID）；失败时可为空或附带校验信息

异常示例

参数校验失败（与 `checkDataType` 返回一致）：

```
{
  "code": 400,
  "status": "fail",
  "message": "缺少参数roomId"
}
```

频道与账号不匹配（`x-auth-user-id` 与 `roomId` 归属不一致）：

```
{
  "code": 400,
  "status": "fail",
  "message": "roomId非法"
}
```

未传 `x-auth-user-id`（中间件直接返回，HTTP 状态码多为 500，body 仍为 JSON）：

```
{
  "code": 400,
  "status": "fail",
  "message": "accountId is required"
}
```

服务端异常：

```
{
  "code": 500,
  "status": "error",
  "message": "unknown error"
}
```

（具体 message 以实际错误信息为准。）