

# 抢答功能

---

## 功能概述

- 本功能只支持讲师，且不支持多角色实时同步操作。
- 支持设置倒计时时间。观众需要在倒计时结束前进行抢答。
- 支持提前结束。
- 可查询抢答排行榜。

## 代码示例

注意，在开始以下步骤前需要先进行全局初始化设置。可[参考文档](#)。

## 示例代码

```
const app = new RushAnswer();
// 抢答状态。
const appData = {
  // 抢答题 id
  rushAnswerId: '',
  // 剩余时间。单位：秒。
  overTime: 0,
  // 提交人数。
  answerNum: 0,
};
// 定时器。
let timer = 0;
// 清理定时器。
function clearTimer() {
  if (!timer) return;
  clearTimeout(timer);
  timer = 0;
}
// 判断倒计时是否结束，未结束情况下每秒获取最新抢答状态。
async function updateStatus() {
  clearTimer();
  // 获取当前抢答状态
  const { ongoingStatus, overTime, answerNum, rushAnswerId } = await app.getStatus();
  // 是否正在进行抢答
  if (ongoingStatus) {
    appData = Object.assign(appData, { overTime, answerNum, rushAnswerId });
    timer = setTimeout(updateStatus, 1000);
  } else {
    appData = Object.assign(appData, { overTime: 0 });
  }
}

// 开始抢答。
const startResult = await app.start(30);

// 获取抢答状态。
updateStatus();

// 结束抢答。
app.stop(appData.rushAnswerId);
// 获取已回答的观众列表。
const result = await app.getAnswerList(appData.rushAnswerId);
```

## 流程图

