**竞争抢答：**

设计一个抢答程序，三个线程，抢答成功的显示成功抢答，失败的返回抢答失败的信息；

|  |
| --- |
| package com.company;  import java.util.concurrent.Callable;  import java.util.concurrent.ExecutionException;  import java.util.concurrent.FutureTask;  /\*\*  \* @author JackWen  \*/  public class Competition {  public static void main(String[] args) throws ExecutionException, InterruptedException {  RushToAnswer rushToAnswer = new RushToAnswer() ;  FutureTask<String> task = new FutureTask<String>(rushToAnswer) ;  new Thread(task,"竞争者A").start();  new Thread(task,"竞争者B").start();  new Thread(task,"竞争者C").start();  System.out.println(task.get());  }  }  /\*\*  \* 抢答线程  \*/  class RushToAnswer implements Callable<String>{  //抢答处理  private boolean flag = false ;  @Override  public String call() throws Exception {  synchronized (this){  if (this.flag == false){  //改变抢答状态  this.flag = true ;  //抢答成功  return Thread.currentThread().getName()+ "抢答成功！" ;  }else{  return Thread.currentThread().getName()+"抢答失败！ ";  }  }  }  } |

Runnable 无法返回参数，因此需要用到Callable 接口实现多线程；