

计算机二级DELPHI控件:捕捉delphi中thread异常计算机二级  
考试 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/565/2021\\_2022\\_\\_E8\\_AE\\_A1\\_](https://www.100test.com/kao_ti2020/565/2021_2022__E8_AE_A1_)

[E7\\_AE\\_97\\_E6\\_9C\\_BA\\_E4\\_c97\\_565702.htm](https://www.100test.com/kao_ti2020/565/2021_2022__E8_AE_A1_E7_AE_97_E6_9C_BA_E4_c97_565702.htm) 2009年下半年全国  
计算机等级考试你准备好了没?考计算机等级考试的朋友,2009  
年下半年全国计算机等级考试时间是2009年9月19日至23日。

更多优质资料尽在百考试题论坛 百考试题在线题库 Threads是  
我们进行应用和设计时不可缺少的利器，然而它却不是轻易  
就可以掌握的。作为一个不可视系统组件，它封装在TThread  
类中，由于一个子线程可以与主线程同时运行，因此，来自  
子Thread的异常在主程序里未必能捕捉到，这样，来自子线  
程的异常就会导致Application的错误甚至是崩溃，也可能造成  
主程序都结束了，某个Thread还因等待同步对象的信号还在  
那儿自己运行着。所以，对于有必要进行异常控制的Thread  
就必须进行异常处理，这个异常处理块最好独立于主程序的  
异常处理模块。我们都知晓对通常异常的捕获都用一个

try..finally块来处理，而对来Thread的异常也不例外：

```
procedure TMyThread.Execute. begin try // 在安全区应该做的工作
```

```
except // 处理所有的异常 end. end. 通常，这样的处理可以
```

```
正常的工作，但却不是恰当的解决方法。我们希望不仅把异常
```

```
信息传递给用户，而且要求在不影响Thread继续工作的前提
```

```
下，由Application或系统单元(致命异常)来进一步处理异常
```

```
。要做这样处理，首先，我们在自己的 Thread 类里定义一个
```

```
异常对象，由这个对象承载各种要处理的异常类实例。其次
```

```
，建立响应异常的同步事件。对EAbort消息加以抑制，对来
```

```
自程序本身的异常由Application处理，对系统级异常，一般交
```

与操作系统来完成。以下是一个简单的异常捕捉应用框架。

```
unit Unit1. interface uses Windows, Messages, SysUtils, Classes,
Graphics, Controls, Forms, Dialogs, StdCtrls. type TForm1 =
class(TForm) Button1: TButton. procedure Button1Click(Sender:
TObject). private { Private declarations } Procedure RunThread.
public { Public declarations } end. TBaseThread = class(TThread)
private FException: Exception. procedure DoHandleException.
protected procedure Execute. override. //父类函数为虚，在子类
再重载其而处理具体事宜 procedure HandleException. virtual.
public end. TMyThread = class(TBaseThread) private ... protected
procedure Exec . override. procedure HandleException. override. ...
public ... end. var Form1: TForm1. implementation {$R *.DFM}
procedure TBaseThread.DoHandleException. begin // 关闭当前主
窗体对鼠标的响应 if GetCapture $#@60.$#@62. 0 then
SendMessage(GetCapture, WM_CANCELMODE, 0, 0). // 判断异
常的范围并做相应处理 if FException is Exception then
Application.ShowException(FException) else
SysUtils.ShowException(FException, nil). ... end. procedure
TBaseThread.Execute. begin FException := nil. try ... //处理一些事
情 except //如果发生了异常 HandleException. end. end.
procedure TBaseThread.HandleException. begin //得到当前异常
对象 FException := Exception(ExceptObject). try //避免因 EAbort
消息使程序推出 if not (FException is EAbort) then
Synchronize(DoHandleException). finally FException := nil. end.
end. procedure TMyThread.Execute. begin ... end. procedure
TMyThread.HandleException. begin ... end. procedure
```

TForm1.RunThread. begin //为 TMyThread 类创建实例 with  
TMyThread.Create(True) do begin FreeOnTerminate := True.  
Resume. end. end. ... 2009年上半年全国计算机等级考试参考答  
案请进入计算机考试论坛 2009年全国计算机等级考试报名信  
息汇总 2009年NCRE考试有新变化 2009年全国计算机等级考  
试大纲 2009年上半年全国计算机二级考试试题及答案 2009年  
上半年全国计算机等级考试试题答案汇总 100Test 下载频道开  
通，各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)