

aaudio 范例: 线程句柄

```
//线程句柄
import console;

//thread.create_suspended = true 创建线程以前可设置该值,指定线程创建后会暂停,否则直接运行

/*
下面创建了一个线程,并返回了线程句柄。

注意如果线程启动函数的代码有语法错误,会抛出异常,
不应当在创建线程时使用try语句忽略异常,而应当在开发时排除这种语法错误。

一般建议的是单个进程中每CPU核创建25线程,通常情况下建议大家可以保持在50线程以下。
有些用户创建上千个线程,这是完全不必要的,CPU的能力是有限的,太多的线程在排队不会更快只会更慢。
当创建的线程太多例如数千线程、这可能会耗尽资源导致创建线程失败,这种原因导致无法创建线程,thread.create就会返回空值。
一般不建议去判断thread.create是否返回空值,没有任何必要、也不应该同时创建这么多线程。
*/
var thrdHandle = thread.create(
    function() {
        import console;

        /*
        thread.lock创建一个临界区,
        在多线程中同时只能有一个线程进入相同名字的临界区,
        这是为了避免同时访问同一资源,例如大家都争着去写控制台,你写几个字符,他写几个字符,结果写出来谁也看不懂了。
        aaudio中需要用临界区的时候并不多,因为每个线程都有自己的变量环境不会相互冲突,很少需要去加锁考虑同步这些麻烦的问题。
        */
        thread.lock("写控制台",function(){
            console.log("线程在执行",thread.getId() )
        })

        /*
        thread.get用于访问线程共享变量。
        每个线程的全局变量环境都是独立的,而 thread.get() 则用于获取所有线程共享的全局变量。
        */
        while(!thread.get("红灯停")){ //这里如果高级点可以用 thread.event来实现信号判断,参考本目录下的定时器范例
            console.log("线程在执行",time() )
            sleep(1000)
        }
        console.log("停了!")
    }
)

//使用线程句柄就可以控制线程,例如下面的函数暂停线程
thread.suspend(thrdHandle)

//线程恢复运行
thread.resume(thrdHandle)

//线程是否在运行
var act = thread.stillActive(thrdHandle)

//下面的函数让线程休眠3000毫秒,也就是3秒,休眠就是啥都不干把CPU让给别的线程
sleep(3000)

//每个线程的全局变量环境都是独立的,而 thread.set() 则用于设置所有线程共享的全局变量。
thread.set("红灯停", true)

/*
等待所有线程完成,下面的参数可以是一个或多个线程句柄,也可以是包含多个线程句柄的数组。
这个函数是阻塞并等待所有线程完成,如果是只等待某个线程返回,则需要使用 thread.waitOne()
thread.waitOne() 如果在界面线程中他不会阻塞界面的消息循环(就是还能响应用户对界面的操作,而不是卡住没响应)
*/
thread.waitAll(thrdHandle)

//句柄不用的时候一定要用下面的函数关闭(切记:句柄是绝对不会自动释放的),关闭句柄并不会关闭线程
raw.closehandle(thrdHandle)

/*
即然句柄不会自动释放,那么如果仅仅是创建一个线程,也不需要通过句柄来管理
您可以使用 thread.invoke() 函数,除了不会返回句柄,其他用法与 thread.create() 一样。
*/
thread.invoke(
    function() {
        import console;
        console.log("线程在执行",thread.getId() )
```

```
    }  
)  
sleep(100)  
  
/*  
您可以使用 thread.command 在线程间交互通信, 请参考《多线程开发入门》  
http://bbs.aardio.com/forum.php?mod=viewthread&tid=13625&from=portal  
  
您还可以使用 thread.event 来实现线程间的同步, 请参考《多线程中的交通信号灯: thread.event》  
http://bbs.aardio.com/forum.php?mod=viewthread&tid=11531  
  
或者使用 thread.works, thread.manage 这些线程管理器来批量的管理线程句柄,  
请参考此目录中的其他范例。  
*/  
  
console.pause(true);
```

[Markdown 格式](#)