# 1.Cocos的定时器函数

|  |
| --- |
|  |

# 2.把一个节点从场景里面移除掉

|  |
| --- |
|  |

# 3.删除一个组件实例

|  |
| --- |
|  |

# 4.节点的其他运行重要属

|  |
| --- |
|  |

# 5.演练

## 1.新建一个项目，起名classTwenty-eight-timer-and-remove-node,然后在assets文件夹里面新建3个文件夹：AssetsPackage，Scenes和Scripts，然后把主场景作为main保存到Scenes文件夹中

|  |
| --- |
|  |

## 2.在Scripts文件夹里面新建一个GameApp脚本

|  |
| --- |
|  |

## 3.然后在场景在新建一个空节点,起名GameApp,然后把脚本添加到这个节点上面

|  |
| --- |
|  |

## 4.双击脚本用vscode打开,然后把多余的内容删除

|  |
| --- |
|  |

## 5.定时器其实是组件的方法.下面我们来添加一个触发一次的定时器

|  |
| --- |
|  |

### 运行项目,一开始没有任何输出,5秒后输出我们设置的内容

|  |
| --- |
|  |
|  |

## 6.如果你需要一直触发的定时器,你需要使用schedule接口,这个接口非常灵活,有很多参数可调,他会触发你设定的次数+1次

|  |
| --- |
|  |

### 运行项目,结果如下

|  |
| --- |
|  |

## 7.任何我们还可以取消一个定时器,为了方便观察,我们把上面的10次改为macro.REPREAT\_FOREVER

|  |
| --- |
|  |

### 运行程序,效果如下

|  |
| --- |
|  |

## 8.如果你想取消特点定的定时器,这个时候,你在定义定时器的时候,就不能够使用lambda函数来做第一个参数,而是需要使用命名函数来作为第一个参数,我们先把上面的定时器改造一下代码如下

|  |
| --- |
|  |

### 运行程序,发现他是可以正常触发的

|  |
| --- |
|  |

## 9,然后我们需要这个命名函数来取消这个定时器

|  |
| --- |
|  |

### 运行项目,效果如下

|  |
| --- |
|  |

# 下面我们来学习节点的移除

## 1.在场景中创建一个Canvas节点,然后在Canvas下面创建一个Sprite Splash节点,起名: Item

|  |
| --- |
|  |

## 2.然后我们在GameApp脚本的定时器里面移除节点,先做移除但是不销毁

|  |
| --- |
|  |

### 运行程序,刚开始是可以看到整个节点的,5秒后它就看不见了

|  |
| --- |
|  |
|  |

## 3.然后我们来移除并且销毁节点

|  |
| --- |
|  |

### 运行程序,效果如下

|  |
| --- |
|  |
|  |

### 从外观说看,效果是一样的.

# 下面我们来学习用代码删除组件实例

## 1.把移除节点的代码注释了,然后我们来删除精灵节点的精灵组件,代码如下

|  |
| --- |
|  |

### 运行项目,刚开始是可以看到精灵的,5秒后也看不见了

|  |
| --- |
|  |
|  |

# 下面我们来学习节点的一些比较重要的属性

## 1.节点的layer,也就是节点的类型,我们可以使用内置的类型,也可以自己添加类型,如这里我们自己添加一个Test类型,方法是点击layer一栏的edit按钮,在0-19中任意一个输入一个名字,如我们在0一栏填写Test,然后可以关闭窗口

|  |
| --- |
|  |

## 2,点击GameApp节点,在右边的layer中选择我们刚刚创建的Test

|  |
| --- |
|  |

## 3.然后我们可以在代码在获取layer的值

|  |
| --- |
|  |

### 得到的是数字1

|  |
| --- |
|  |