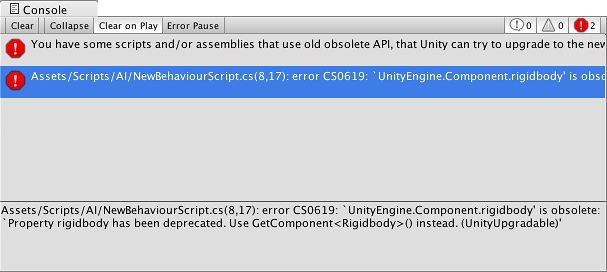
**脚本工具**

本部分介绍 Unity Editor 中的工具以及 Unity 提供的可帮助您开发脚本的工具。

# Console 窗口

**Console** 窗口（菜单：\_\_Window > Console\_\_）显示 Unity 生成的错误、警告和其他消息。为了方便调试，还可以使用 [Debug.Log](file:///E:\UnityDocumentation\ScriptReference\Debug.Log.html)、[Debug.LogWarning](file:///E:\UnityDocumentation\ScriptReference\Debug.LogWarning.html) 和 [Debug.LogError](file:///E:\UnityDocumentation\ScriptReference\Debug.LogError.html) 函数，在控制台中显示自定义的消息。



Console 窗口的工具栏有许多选项可影响消息显示方式。

Clear 按钮可移除从代码中生成的所有消息，但会保留编译器错误。可通过启用 Clear On Play 选项来设置在运行游戏时自动清除控制台。

还可以更改控制台中消息的显示和更新方式。Collapse 选项仅显示重复出现的错误消息的第一个实例。有时在每次帧更新时会重复生成的运行时错误（例如 null 引用），此选项在这种情况下非常有用。使用 Error Pause 选项时，只要从脚本调用 Debug.LogError，就会使回放暂停（但是，请注意 Debug.Log 不会这样暂停）。需要在执行中冻结特定点的回放并检查场景时，这会很方便。

最后，有两个选项可用于查看有关错误的其他信息。控制台选项卡菜单上的 Open Player Log 和 Open Editor Log 菜单项可访问 Unity 的日志文件，这些文件将记录可能未在控制台上显示的详细信息。请参阅关于[日志文件](file:///E:\UnityDocumentation\Manual\LogFiles.html)的页面以了解更多信息。

## 过时 API 警告和自动更新

除了其他消息，Unity 还显示有关在代码中使用过时 API 调用的警告。例如，Unity 曾在 MonoBehaviour 和其他类中使用“快捷方式”来访问常见组件类型。因此，比如说，可使用如下代码来访问对象上的[刚体](file:///E:\UnityDocumentation\Manual\class-Rigidbody.html)：

//"rigidbody"变量是类的一部分，未在用户脚本中声明。

Vector3 v = rigidbody.velocity;

这些快捷方式已被弃用，因此现在应该使用如下代码：

// 使用 GetComponent 来访问组件。

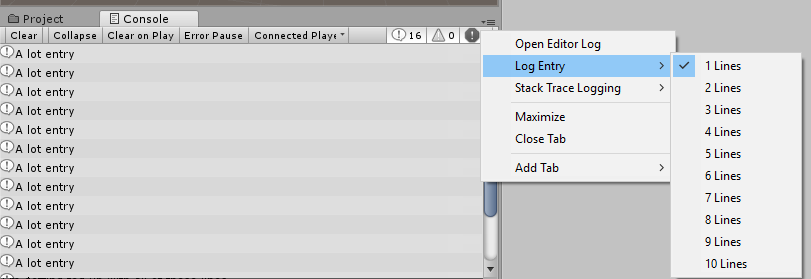
Rigidbody rb = GetComponent<Rigidbody>();

Vector3 v = rb.velocity;

检测到过时 API 调用时，Unity 将显示相关的警告消息。双击此消息时，Unity 将自动尝试将已弃用的用法升级到建议的等效用法。

## 调整行计数

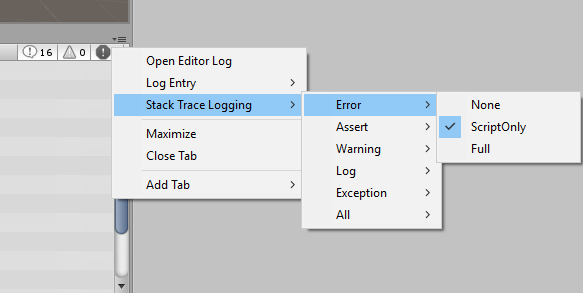
要调整日志条目在列表中显示的行数，请单击惊叹号按钮，选择 \_\_Log Entry\_\_，然后选择行数。



通过此设置，可根据上下文量与窗口中能够容纳的条目数来确定窗口所需的粒度。

## 堆栈跟踪日志记录

可以指定在将日志消息输出到控制台或日志文件时应如何准确捕获堆栈跟踪。

日志条目行计数

错误消息不是很明确时，这通常很有帮助。通过查看堆栈跟踪，可以推导错误来自哪个引擎区域。 堆栈跟踪日志记录有三个选项：

* **None** - 不会输出堆栈跟踪
* **ScriptOnly** - 只输出托管堆栈跟踪
* **Full** - 同时输出本机和托管堆栈跟踪，注意：解析完整堆栈跟踪是一项高成本的操作，应仅用于调试目的

还可以通过脚本 API 来控制堆栈跟踪日志记录。请参阅关于 [Application.stackTraceLogType](file:///E:\UnityDocumentation\ScriptReference\Application-stackTraceLogType.html) 的 API 参考文档以了解更多详细信息。

# 日志文件

在开发过程中，有时可能需要从构建的独立平台播放器、目标设备或 Editor 的日志中获取信息。通常，在遇到问题时需要查看这些文件，以具体了解问题的发生位置。

在 macOS 上，可通过标准 **Console.app** 实用程序统一访问播放器和 Editor 的日志。

在 Windows 上，Editor 日志放置在默认情况下未在 Windows 资源管理器中显示的文件夹中。请参阅下文。

## Editor

要查看 Editor 日志，请在 Unity 的 Console 窗口中选择 **Open Editor Log**。

| 操作系统 | 日志文件 |
| --- | --- |
| **macOS** | ~/Library/Logs/Unity/Editor.log |
| **Windows** | C:\Users\username\AppData\Local\Unity\Editor\Editor.log |

在 macOS 上，可通过标准 **Console.app** 实用程序统一访问所有日志。

在 Windows 上，Editor 日志文件存储在本地应用程序数据文件夹 <LOCALAPPDATA>\Unity\Editor\Editor.log，其中的 <LOCALAPPDATA> 由 [CSIDL\_LOCAL\_APPDATA](https://msdn.microsoft.com/en-us/library/windows/desktop/bb762494%28v=vs.85%29.aspx) 定义。

## 播放器

| 操作系统 | 日志文件 |
| --- | --- |
| **macOS** | ~/Library/Logs/Unity/Player.log |
| **Windows** | C:\Users\username\AppData\LocalLow\CompanyName\ProductName\output\_log.txt |
| **Linux** | ~/.config/unity3d/CompanyName/ProductName/Player.log |

请注意，在 Windows 和 Linux 独立平台上，可以更改日志文件的位置（或禁止日志记录）。请参阅有关[命令行参数](file:///E:\UnityDocumentation\Manual\CommandLineArguments.html)的文档以了解更多详细信息。

### iOS

通过 GDB 控制台或 Organizer Console 访问 XCode 中的设备日志。当应用程序未通过 XCode 调试器运行时，后一种控制台对于获取崩溃日志非常有用。

[故障排除](file:///E:\UnityDocumentation\Manual\TroubleShootingIPhone.html)和[报告崩溃错误](file:///E:\UnityDocumentation\Manual\iphone-bugreporting.html)指南可能对您有用。

### Android

使用 [logcat 控制台](http://developer.android.com/guide/developing/tools/adb.html#logcat)访问设备日志。为此，请使用 **Android SDK/platform-tools** 目录中的 **adb** 应用程序以及尾随的 **logcat** 参数：

$ adb logcat

检查 LogCat 的另一种方法是使用 [Dalvik Debug Monitor Server (DDMS)](http://developer.android.com/guide/developing/debugging/ddms.html)。可从 **Eclipse** 启动或从 **Android SDK/tools** 内部启动 DDMS。DDMS 还提供了许多其他与调试相关的工具。

### 通用 Windows 平台

| 设备 | 日志文件 |
| --- | --- |
| **桌面端** | %USERPROFILE%\AppData\Local\Packages<productname>\TempState\UnityPlayer.log |
| **Windows Phone** | 可使用 [Windows Phone Power Tools](https://wptools.codeplex.com/) 进行检索。也可使用 [Windows Phone IsoStoreSpy](https://isostorespy.codeplex.com/)。 |

### WebGL

在 WebGL 上，日志输出将写入浏览器的 [JavaScript 控制台](file:///E:\UnityDocumentation\Manual\webgl-debugging.html)。

## 在 Windows 上访问日志文件

On Windows, the log files are stored in locations that are hidden by default.

在 Windows Vista/7 的 Windows 资源管理器中，请使用 **Tools** > **Folder Options…** > **View（选项卡）** 来显示 AppData 文件夹。默认情况下会隐藏 Tools 菜单；按一次 Alt 键即可显示。