本页由killman.liu翻译自原英文教程 main页

# 教程首页

本教程主要包含Orx的基础和高级教程Orx是一个开源、跨平台、轻量级、数据驱动的2D游戏引擎。

## 安装

这些教程演示了如何设置不同的编程环境(IDE)来运行orx[1)]

- Microsoft Visual Studio (C++) for Windows: Tutorial (英文版） / Download (Express version)
- CodeLite for Windows, Linux and Mac OS X: Tutorial (英文版） / Download
- XCode for Mac OS X: Tutorial / Download

## 基础

本节将要介绍orx的基础知识。

你可以从这里下载Windows(mingw, msvs2005 & msvs2008). Linux 和 MacOS X下的可执行文件（包括项目文件，数据和源码）。

前九个基础教程（#1 – #9）使用默认的orx启动程序加载，这样易于快速测试/制作原型。[2)]

它们被编译成运行时加载（在命令行 [3)] 上 或配置文件中指定它们的名字）的动态连接库。

此外，下面的内容[4)]解释了哪些行为是由默认的orx.exe/orx 启动程序提供的。

这是一个基础的C教程。 由于我们在本教程中使用默认的可执行文件，下面的代码将以插件的方式加载和执行。

另外，一些基础设施有主执行文件为我们处理。 首先，它会加载所有可用的插件和模块。如果你只需要其中的一些，最好编写你自己的可执行文件而不是插件。这部分包含在后面的教程中。

主执行文件还处理下面这些键盘输入：

- F11 是纵向对齐切换
- Escape 退出
- F12 截屏
- 退格键(Backspace) 重新载入全部配置文件

如果有 orxSYSTEM_EVENT_CLOSE事件发生，程序也会退出。

不过，如果使用orx作为传统库构建你自己的可执行文件当然也是可以的（也很容易做到）。这在教程 #10(使用C++编写) 和 教程 #11 (使用 C编写)会讲到。教程 #10 还演示了如何使用orx编写C++代码[5)]。同样地，你可以用任何可与C连接的语言编写程序。

在将来的发布中将会为某些常见语言提供封装。如果你想编写这种封装库，为orx做贡献，请通过论坛联系我们。

当前提供的基础教程列表：

1. [C] 对象(Object)
2. [C] 时钟（Clock） [6)]
3. [C] 帧的创建(Frame)
4. [C] 动画(animation)
5. [C] 视口与摄像头（viewport & camera）
6. [C] 声音与音乐(sound & music)
7. [C] 特效(fx)
8. [C] 物理特性（physics)
9. [C] 卷轴效果(scrolling)
10. [C++] 独立程序与本地化教程（stand alone & localization）
11. [C] spawner & shader

# 原官方教程

官方教程主页（英文版）

# 社区内容

有一些Orx爱好者写了一些教程，可以作为官方的教程的补充材料

- Grey's 教程（英文版）
- 九天(JTianLing)的教程

1) 所有的IDE都是免费可以从英特网上下载的

2) 用一行代码完成初始化，不需要写main函数，没有循环要处理

3) 另提供了 .bat/.sh 脚本方便启动所有示例

4) 你将在在所有教程的源文件的开始处看到这些内容

5) Orx本身由C编写

6) 译者注：计时器(timer)

From:
https://www.orx-project.org/wiki/ - **Orx Learning**

Permanent link:
**https://www.orx-project.org/wiki/cn/orx/tutorials/main?rev=1529588633**

Last update: **2025/09/30 17:26 (14 months ago)**