Tutoriales

Esta sección esta dedicada a los tutoriales básicos y avanzados sobre orx, un, código abierto, portable, ligero, manejador de datos & motor de juegos orientado al 2D.

Configuración

Este tutorial muestra como configurar diferentes entornos de programación (IDE) para trabajar con orx. [1)]

- Microsoft Visual Studio (C++) para Windows: VS2008 Tutorial / Descargar Visual Studio 2008 (Express version)
- CodeLite para Windows, Linux y Mac OS X: Linux Tutorial / Descargar CodeLite
- XCode para Mac OS X: XCode Tutorial / Descargar XCode / Instalando Xcode 3.2.6 en Lion
- iPhone/iPad: iPhone Tutorial
- Android: Android Tutorial
- Android Native: Android Native Tutorial

Comunidad

Estos tutoriales son creados por la comunidad como una manera de dar rápidas respuestas a cuestiones simples.
Ellos son grandes recursos para descubrir como hacer cosas con orx!

- Tutoriales de Grey - Más tutoriales de configuración de entornos de programación aquí.
- Tutoriales de Iarwain
- Tutoriales de Sausage
- Tutoriales de tdomhan

Básico

Esta sección te introducirá a los tutóriales básicos de Orx.

Puedes descargar todos los ejecutables para Windows (mingw, msvs2005 & msvs2008), Linux y MacOS X desde aqui.

Los primeros nueves tutoriales (#1- #9) usan por defecto el lanzador de orx como capa principal. Esto tiene como ventaja hacer pruebas y prototipos de forma rápida [2)]

Estan compilados como bibliotecas dinámicas que son cargados en tiempo de ejecución (especificando el nombre en la linea de comandos, o a través del fichero de configuración).
A continuación explicamos cuales son los comportamientos por defecto del lanzador `orx.exe/orx`.

```
Este es un tutorial básico en C
Como estamos usando el executable por defecto en este tutorial, este código
```

será cargado y ejecutado como plugin en tiempo de ejecución.

Algunos conceptos son manipulados por nosotros a través del ejecutable principal.
Primero que nada, serán cargados todos los plugins y módulos disponibles. En caso que necesites solo algunos de ellos, entonces es mejor que escribas tu propio ejecutable en lugar de usar los plugins. Esto podremos verlo en el tutorial Aplicación estándar.

El ejecutable principal también manipula algunas teclas como:

* F11 como interruptor para la sincronía vertical
* Escape para salir
* F12 para hacer una captura de pantalla
* Backspace para recargar todos los ficheros de configuracion (provistos por la activación del historial de configuración)

El programa también se cierra si la señal orxSYSTEM_EVENT_CLOSE es enviada.

Sin embargo, usando orx tradicionalmente como biblioteca y construyendo to propio ejecutable es totalmente posible y fácil de hacer.
Esto podemos verlo en el tutorial #10 Aplicación estándar en C++ o en el #11 Spawner en C. Este tutorial muestra además como escribir código C++ usando orx el cual ha sido escrito en C. De la misma manera, puedes hacer tu programa usando cualquier lenguage que tenga interfaces con C.

Más adelante tendremos wrappers para algunos lenguajes comunes como c++, C# entre otros. Si quieres contribuir con ORX escribiendo cualquier tipo de wrapper por favor contáctanos a través del foro forum.

Aquí está la lista de los tutoriales básicos existentes hasta el momento:

1. [C] objeto
2. [C] reloj
3. [C] fotograma
4. [C] animación
5. [C] vista & cámara
6. [C] música & sonido
7. [C] efectos
8. [C] física
9. [C] desplazamiento
10. [C++] aplicación estándar & localización
11. [C] reproductor & sombreador
12. [C] sombras & luces]

[1)]
Todos esos IDEs con libres y pueden ser descargados desde internet.
[2)]
una sola línea para inicializar, sin necesidad de escribir una función, ni ciclos que manipular

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

Permanent link:
https://www.orx-project.org/wiki/es/orx/tutorials/main?rev=1496130646

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