Raylib: A C nyelvű játékfejlesztési könyvtár

A **raylib** egy egyszerű és könnyen használható C nyelvű könyvtár, amit játékfejlesztéshez és multimédiás alkalmazásokhoz készítettek.

Fő jellemzői

- **Platformfüggetlen**: Windows, Linux, MacOS és több más rendszeren is működik.
- **Könnyű Integrálni**: Nincs szükség külső függőségekre vagy bonyolult telepítési folyamatokra.
- **2D és 3D támogatás**: Támogatja mind a 2D, mind a 3D grafikai és audio funkcionalitást.

Alapvető funkciók

- **Ablakkezelés**: Könnyű létrehozni és kezelni az ablakokat, valamint kezelni az eseményeket.
- **Grafika**: Támogatja a 2D-s és 3D-s rajzolást, textúrák és sprite-ok kezelését.
- **Hangkezelés**: Képes audio fájlok lejátszására és hang effektek kezelésére.
- **Bemenet kezelése**: Egér, billentyűzet és más bemeneti eszközök támogatása.

Források

- **Hivatalos Weboldal**: raylib.com
- **Dokumentáció**: Teljes körű API dokumentáció elérhető a hivatalos weboldalon.
- **Példák és Tutorialok**: Sok példa és oktatóanyag található az oldalon

Egyszerű telepítés

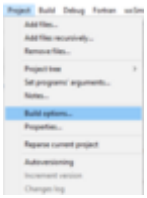
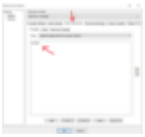
A raylib függőséggekkel, példakóddal telepíthető fejlesztőkörnyezet:

Hordozható codeblocks

Telepítés és konfiguráció

Ha már korábban letöltött CodeBlocks rendszeren szeretnénk használni a raylib-et, akkor az alábbi leírás alapján használjuk:

1. Van saját telepítője is, de egyszerűbb ha letöltjük az **include** és **lib** fájlokat tartalmazó raylib.zip -et.
2. Hozzunk létre egy konzol alkalmazást a codeblocks-ban
3. A létrejött **main.c** fájl könyvtárába tömörítsük ki a letöltött zip-et

4. Nyissuk meg a **Projekt/Build options** menüt: 
5. A linker settings fülön állítsuk be **/lib/libraylibdll.a**: 
6. A search directories fülön adjuk hozzá az **include** könyvtárat: 
7. Ugyanitt váltsunk a **linker** fülre és adjuk hozzá a **lib** könyvtárat: 
8. A zip-ből másoljuk ki a raylib.dll-t a bin/RELEASE vagy bin/DEBUG könyvtárakba, ha elindításkor nem találja rendszer.

Írjuk felül a jelenlegi main.c tartalmát az alábbi kóddal és futtassuk a programot.

```
#include <raylib.h>

int main() {
    // Window creation
    const int screenWidth = 640;
    const int screenHeight = 480;
    InitWindow(screenWidth, screenHeight, "raylib example");

    // Initialization
    SetTargetFPS(60); // Set our app to run at 60 frames-per-second

    // Main game loop
    while (!WindowShouldClose()) { // Detect window close button or ESC key
        // Update

        // Draw
        BeginDrawing();

            ClearBackground(LIGHTGRAY); // Clear the background with a color

            int MaxX = GetScreenWidth();
            int MaxY = GetScreenHeight();

            for (int i = 1; i <= 10; i++) {
                Color lineColor = GetColor(i + 5); // Choose color
                DrawLine(0, 0, (MaxX / 10) * i, MaxY, lineColor); // Draw
line

                int R = (MaxY - 10) / (2 * i); // Circle radius
```

```
                DrawCircle(MaxX - R, MaxY / 2, R, lineColor); // Draw circle

                // Draw ellipse
                DrawEllipse(MaxX / 2, MaxY / 8, MaxX / (4 * i), MaxY / 8,
lineColor);
            }

        EndDrawing();
    }

    // De-Initialization
    CloseWindow(); // Close window and OpenGL context

    return 0;
}
```

From:
https://edu.iit.uni-miskolc.hu/ - **Institute of Information Science - University of Miskolc**

Permanent link:
https://edu.iit.uni-miskolc.hu/muszaki_informatika:raylib

Last update: **2025/03/12 18:17**