

ChatGPT 3.5



You
Create a program using zig language that opens a blank window



ChatGPT

Sure, I can provide you with a simple example using the Zig programming language to create a window using a graphical library. In this case, I'll use the `glfw` library for window creation. Before you run the code, make sure you have `glfw` installed on your system.

First, you need to install the Zig package manager, `zigmod`, if you haven't already. You can find instructions on the official Zig documentation: <https://ziglang.org/download/>

Once `zigmod` is installed, you can create a new project and add the `glfw` package dependency. Create a new file named `build.zig` with the following content:

Copy code

```
zig

const Builder = @import("std").build.Builder;

pub fn build(b: *Builder) void {
    const target = b.standardReleaseOptions();

    const exe = b.addExecutable("your_program_name_here", "src/main.zig");
    exe.setTarget(target);

    const glfw_dep = b.addPackage("gh:glfw/glfw", "3.3.4");
    glfw_dep.setTarget(target);
    exe.addPackage(glfw_dep);

    exe.install();

    const run_cmd = exe.run();

    b.default_step.dependOn(b.getInstallStep());
    b.installArtifact(run_cmd);
}
```

Replace `"your_program_name_here"` with the desired name for your program.

Now, create a `src` directory and within it, create a file named `main.zig` with the following code:

Copy code

```
zig

const glfw = @import("glfw");

pub fn main() void {
    if (glfw.init() != glfw.SUCCESS) {
        return;
    }

    defer glfw.terminate();

    const window = glfw.createWindow(640, 480, "Blank Window", null, null);
    if (window == null) {
        return;
    }

    glfw.makeContextCurrent(window);

    while (!glfw.windowShouldClose(window)) {
        glfw.pollEvents();
        // You can add your rendering code here
    }
}
```

This code initializes GLFW, creates a window, and enters a loop to keep the window open until the user closes it. Note that this example doesn't include any rendering code; you can add your own rendering logic within the loop.

Now, navigate to your project directory in the terminal and run:

Copy code

```
bash

zigmod fetch
zigmod build
./zig-cache/bin/your_program_name_here
```

This should open a blank window using GLFW. Make sure you have the necessary dependencies installed on your system, such as GLFW itself.

Please note that this example assumes you have a graphical environment and GLFW dependencies installed on your system. If you encounter any issues, refer to the GLFW documentation for installation instructions: <https://www.alfw.org/docs/latest/compile.html>

Message ChatGPT...



ChatGPT can make mistakes. Consider checking important information.