

GTK4 és Code::Blocks 25.03 telepítése Microsoft Windows 11 operációs rendszerre

The GIMP Toolkit

MSYS2

<https://www.msys2.org/>

https://github.com/msys2/msys2-installer/releases/download/2026-06-11/msys2-x86_64-20260611.exe

```
$ gcc --version
```

```
$ pacman -S mingw-w64-ucrt-x86_64-gtk4
```

```
$ pacman -S mingw-w64-ucrt-x86_64-pkg-config
```

```
$ pacman -S mingw-w64-ucrt-x86_64-toolchain base-devel
```

```
$ pacman -Suy
```

Path

Jobb kattintás a Start gombon -> Rendszer -> Speciális rendszerbeállítások -> Környezeti változók -> Rendszerváltozók -> Path -> Szerkesztés

C:\msys64\ucrt64\bin\

C:\msys64\ucrt64\lib\

C:\msys64\usr\bin\

CMD

```
pkg-config --cflags --libs gtk4
```

Window

```
md C:\Users\User\CodeBlocksProjects
```

```
md C:\Users\User\CodeBlocksProjects\window
```

```
cd C:\Users\User\CodeBlocksProjects\window
```

```
notepad window.c
```

```
--
```

```
#include <gtk/gtk.h>
```

```
static void activate (GtkApplication *app, gpointer user_data)
```

```
{
```

```
    GtkWidget *window;
```

```
    window = gtk_application_window_new (app);
```

```
    gtk_window_set_title (GTK_WINDOW (window), "Window");
```

```
    gtk_window_set_default_size (GTK_WINDOW (window), 400, 400);
```

```

    gtk_window_present (GTK_WINDOW (window));
}

int main (int argc, char **argv)
{
    GtkApplication *app;
    int status;
    app = gtk_application_new ("hu.nye.inf.window", G_APPLICATION_DEFAULT_FLAGS);
    g_signal_connect (app, "activate", G_CALLBACK (activate), NULL);
    status = g_application_run (G_APPLICATION (app), argc, argv);
    g_object_unref (app);
    return status;
}
--

```

notepad makefile

```

--
CC = gcc
CFLAGS = -Wall -O3 -std=c99
GTK_CFLAGS = `pkg-config --cflags gtk4`
GTK_LIBS = `pkg-config --libs gtk4`

all:
    @mkdir -p ./bin/Debug
    $(CC) $(GTK_CFLAGS) -o ./bin/Debug/window.exe window.c $(GTK_LIBS)
--

```

mingw32-make

bin\Debug\window.exe

Code::Blocks

<https://www.codeblocks.org/>

<https://sourceforge.net/projects/codeblocks/files/Binaries/25.03/Windows/codeblocks-25.03mingw-setup.exe>

Code::Blocks start

Settings -> Debugger

GDB/CDB debugger -> Create config -> pl.: MSYS2 UCRT 64 -> OK

GDB/CDB debugger -> MSYS2 UCRT 64 -> Executable path: ->

C:\msys64\ucrt64\bin\gdb.exe -> OK

Settings -> Compiler

Selected compiler -> Copy -> Add new compiler -> pl.: MSYS2 UCRT 64

Compiler Settings

Compiler Flags

General

Have gcc follow the 1999 C language standard [-std=c99]

Warnings

Enable all common compiler warnings (overrides many other settings) [-Wall]

Optimization

Optimize fully (for speed) [-O3]

Other Compiler Options -> `pkg-config --cflags gtk4`

Linker Settings -> Other Linker Options -> `pkg-config --libs gtk4`

Toolchain executables

Compiler's installation directory -> C:\msys64\ucrt64\

Debugger -> GDB/CDB debugger: -> MSYS2 UCRT 64

OK

Code::Blocks restart

Hello World

Create a new project -> Console application -> Go -> C -> Next

Project title: hello_world

Folder to create project in: C:\Users\User\CodeBlocksProjects\

Project filename: hello_world.cbp

Resulting filename: C:\Users\User\CodeBlocksProjects\hello_world\hello_world.cbp

Next

Compiler: MSYS2 UCRT 64

Finish

--

```
#include <gtk/gtk.h>
```

```
static void print_hello (GtkWidget *widget, gpointer data)
{
    g_print ("Hello World\n");
}
```

```

static void activate (GtkApplication *app, gpointer user_data)
{
    GtkWidget *window;
    GtkWidget *button;
    GtkWidget *box;
    window = gtk_application_window_new (app);
    gtk_window_set_title (GTK_WINDOW (window), "Window");
    gtk_window_set_default_size (GTK_WINDOW (window), 400, 400);
    box = gtk_box_new (GTK_ORIENTATION_VERTICAL, 0);
    gtk_widget_set_halign (box, GTK_ALIGN_CENTER);
    gtk_widget_set_valign (box, GTK_ALIGN_CENTER);
    gtk_window_set_child (GTK_WINDOW (window), box);
    button = gtk_button_new_with_label ("Hello World");
    g_signal_connect (button, "clicked", G_CALLBACK (print_hello), NULL);
    g_signal_connect_swapped (button, "clicked", G_CALLBACK (gtk_window_destroy),
window);
    gtk_box_append (GTK_BOX (box), button);
    gtk_window_present (GTK_WINDOW (window));
}

```

```

int main (int argc, char **argv)
{
    GtkApplication *app;
    int status;
    app = gtk_application_new ("hu.nye.inf.hello_world",
G_APPLICATION_DEFAULT_FLAGS);
    g_signal_connect (app, "activate", G_CALLBACK (activate), NULL);
    status = g_application_run (G_APPLICATION (app), argc, argv);
    g_object_unref (app);
    return status;
}

```

--

Rebuild

Run

Getting Started with GTK

https://docs.gtk.org/gtk4/getting_started.html