

A basic GTK+ program:

```
1 #include <gtk/gtk.h>
2
3 static void
4 close_window (GtkWidget *widget)
5 {
6     gtk_main_quit ();
7 }
8
9 int
10 main (void)
11 {
12     GtkWidget *window;
13     gtk_init (0, NULL);
14     window = gtk_window_new (GTK_WINDOW_TOPLEVEL);
15     g_signal_connect (window,
16                       "destroy",
17                       G_CALLBACK (close_window),
18                       NULL);
19     gtk_widget_show_all (window);
20     gtk_main ();
21     return 0;
22 }
```

Compile it using:

```
$ gcc `pkg-config --cflags gtk+-3.0` -o hello
  hello.c `pkg-config --libs gtk+-3.0` -Wall
```

For debugging:

```
$ gcc ... -g -Og # for compiling
$ gdb ./hello # start debugger
# Find out where warnings come from:
$ G_DEBUG="fatal-warnings" gdb ./hello
# Find memory errors:
$ G_SLICE="debug-blocks" gdb ./hello
$ G_SLICE="always-malloc" valgrind --leak-check=full > out 2>&1
# Record program flow:
$ rr record ./hello
$ rr replay
```

Version 2.0 with a button:

```
1 #include <gtk/gtk.h>
2
3 static void
4 close_window (GtkWidget *widget)
5 {
6     gtk_main_quit ();
7 }
8
9 int
10 main (void)
11 {
12     GtkWidget *window, *grid, *hello_label, *button;
13     gtk_init (0, NULL);
14     window = gtk_window_new (GTK_WINDOW_TOPLEVEL);
15     grid = gtk_grid_new ();
16     gtk_orientable_set_orientation (GTK_ORIENTABLE (grid),
17                                     GTK_ORIENTATION_VERTICAL);
18     hello_label = gtk_label_new ("Hello World!");
19     g_object_set (G_OBJECT (hello_label),
20                 "expand", TRUE,
21                 NULL);
22     button = gtk_button_new_with_mnemonic ("_Quit");
23     gtk_container_add (GTK_CONTAINER (window),
24                       grid);
25     gtk_container_add (GTK_CONTAINER (grid),
26                       hello_label);
27     gtk_container_add (GTK_CONTAINER (grid),
28                       button);
29     g_signal_connect (button,
30                     "clicked",
31                     G_CALLBACK (close_window),
32                     NULL);
33     g_signal_connect (window,
34                     "destroy",
35                     G_CALLBACK (close_window),
36                     NULL);
37     gtk_widget_show_all (window);
38     gtk_main ();
39     return 0;
40 }
```