



# Gnome::Gtk3::Main

## Main loop and Events — Library initialization, main event loop, and events

### Table of Contents

- 1 [Synopsis](#)
- 2 [Methods](#)
- 2.1 [new](#)
- 2.2 [gtk\\_events\\_pending](#) Checks if any events are pending.
- 2.3 [gtk\\_main](#)
- 2.4 [gtk\\_main\\_level](#)
- 2.5 [gtk\\_main\\_quit](#)
- 2.6 [gtk\\_main\\_iteration](#)
- 2.7 [gtk\\_main\\_iteration\\_do](#)

```
unit class Gnome::Gtk3::Main;
```

## Synopsis

```
my Gnome::Gtk3::Main $main .= new;

# Setup user interface
...
# Start main loop
$main.gtk_main;

# Elsewhere in some exit handler
method exit ( ) {
    Gnome::Gtk3::Main.new.gtk_main_quit;
}
```

## Methods

### new

Create a GtkMain object. Initialization of GTK is automatically executed if not already done. Arguments from the command line are provided to this process.

```
submethod BUILD ( Bool :$check = False )
```

- `$check`; Use checked initialization. Program will not fail when commandline arguments do not parse properly.

## **gtk\_events\_pending** Checks if any events are pending.

This can be used to update the UI and invoke timeouts etc. while doing some time intensive computation.

```
method gtk_events_pending ( )
```

Example

```
# Computation going on ...  
  
$main.gtk_main_iteration() while ( $main.gtk_events_pending() );  
  
# Computation continued ...
```

## **gtk\_main**

Runs the main loop until `gtk_main_quit()` is called. You can nest calls to `gtk_main()`. In that case `gtk_main_quit()` will make the innermost invocation of the main loop return.

```
method gtk_main ( )
```

## **gtk\_main\_level**

Returns the current nesting level of the main loop.

```
method gtk_main_level ( --> Int )
```

## **gtk\_main\_quit**

Makes the innermost invocation of the main loop return when it regains control.

```
method gtk_main_quit ( )
```

## **gtk\_main\_iteration**

Runs a single iteration of the mainloop.

If no events are waiting to be processed GTK+ will block until the next event is noticed. If you don't want to block look at `gtk_main_iteration_do()` or check if any events are pending with `gtk_events_pending()` first.

```
method gtk_main_iteration ( --> Int )
```

Returns 1 if `gtk_main_quit()` has been called for the innermost mainloop

## **gtk\_main\_iteration\_do**

Runs a single iteration of the mainloop. If no events are available either return or block depending on the value of blocking.

```
method gtk_main_iteration_do ( Int $blocking --> Int )
```

- `$blocking`; 1 if you want GTK+ to block if no events are pending.

Returns 1 if `gtk_main_quit()` has been called for the innermost mainloop

Generated using Pod::Render, Pod::To::HTML, Camelia™ (butterfly) is © 2009 by Larry Wall