



class GTK::V3::Gtk::GtkDialog

Table of Contents

- 1 [Synopsis](#)
- 2 [unit class GTK::V3::Gdk::GdkEventTypes;git also is GTK::V3::Gtk::GtkWindow;](#)
- 3 [Types](#)
 - 3.1 [enum GtkResponseType](#)
- 4 [Methods](#)
 - 4.1 [gtk_dialog_new](#)
 - 4.2 [gtk_dialog_run](#)
 - 4.3 [gtk_dialog_response](#)
 - 4.4 [new](#)

```
unit class GTK::V3::Gtk::GtkDialog;  
also is GTK::V3::Gtk::GtkWindow;
```

#=head2 GtkDialog — Create popup windows

Synopsis

```
my GTK::V3::Gtk::GtkDialog $dialog .= new(:build-id<simple-dialog>);  
  
# show the dialog  
my Int $response = $dialog.gtk-dialog-run;  
if $response == GTK_RESPONSE_ACCEPT {  
    ...  
}
```

unit class GTK::V3::Gdk::GdkEventTypes;git also is GTK::V3::Gtk::GtkWindow;

Types

enum GtkResponseType

Possible types of response

Response type	When
GTK_RESPONSE_NONE	Returned if an action widget has no response id, or if the dialog gets programmatically hidden or destroyed
GTK_RESPONSE_REJECT	Generic response id, not used by GTK+ dialogs
GTK_RESPONSE_ACCEPT	Generic response id, not used by GTK+ dialogs
GTK_RESPONSE_DELETE_EVENT	Returned if the dialog is deleted
GTK_RESPONSE_OK	Returned by OK buttons in GTK+ dialogs
GTK_RESPONSE_CANCEL	Returned by Cancel buttons in GTK+ dialogs
GTK_RESPONSE_CLOSE	Returned by Close buttons in GTK+ dialogs
GTK_RESPONSE_YES	Returned by Yes buttons in GTK+ dialogs
GTK_RESPONSE_NO	Returned by No buttons in GTK+ dialogs
GTK_RESPONSE_APPLY	Returned by Apply buttons in GTK+ dialogs
GTK_RESPONSE_HELP	Returned by Help buttons in GTK+ dialogs

Methods

gtk_dialog_new

Creates a new dialog box.

gtk_dialog_run

Blocks in a recursive main loop until the dialog either emits the “response” signal, or is destroyed. If the dialog is destroyed during the call to `gtk_dialog_run()`, `gtk_dialog_run()` returns `GTK_RESPONSE_NONE`

gtk_dialog_response

Emits the “response” signal with the given response ID. Used to indicate that the user has responded to the dialog in some way; typically either you or `gtk_dialog_run()` will be monitoring the `::response` signal and take appropriate action.

new

```
multi submethod BUILD ( Bool :$empty! )
```

Create an empty dialog

```
multi submethod BUILD ( :$widget! )
```

Create a dialog using a native object from elsewhere. See also `Gtk::V3::Glib::GObject`.

```
multi submethod BUILD ( Str :$build-id! )
```

Create a dialog using a native object from a builder. See also `Gtk::V3::Glib::GObject`.

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