

The Simple Zenity Guide

About

Zenity is a cross-platform application which allows the execution of GTK+ widgets from the command line or scripts. It provides simple, minimal syntax to allow the user to enter and receive information graphically.

The program does not intend to provide a full-featured interface covering every possible use case. It does however provide a common widget set which can easily be integrated into applications.

Message Dialogs

Message Dialogs are used to display a range of messages to the user. There are four variants; information, warning, error and question.

```
zenity --info --text="This is an information dialog"
```

```
zenity --error --text="This is an error dialog"
```

```
zenity --warning --text="This is a warning dialog"
```

```
zenity --question --text="This is a question dialog"
```

The `--question` option has one major difference to the other three options. It provides two buttons; Yes and No. Using these buttons, the response from the user can be collected and handled by your script.

Entry

An Entry provides a dialog which can be used to receive textual input from the user. In most cases, this widget will be used to retrieve a small amount of textual input.

```
zenity --entry --text="Enter some text"
```

The Entry dialog displays two buttons, an OK and Cancel. When Cancel is pressed, the Entry disappears. When the OK button is pressed, the text which has been entered is returned to the terminal. From there, it can be manipulated, stored in a variable, etc.

In some cases, an Entry may need to retrieve a password from the user. In this circumstance, it is advisable to hide the password to ensure that an unauthorised person can not see it.

```
zenity --entry --hide-text
```

Text Info

A Text Info widget provides a large dialog which is used to display small, medium or large amounts of text to the user. To display the text, it must be loaded from a text file.

```
zenity --text-info --filename="/home/user/textfile.txt"
```

Scale

A Scale widget provides a way to enter numbers into the application using a sliding scale.

```
zenity --scale --value=50
```

By default, the minimum number that can be entered is 0, and the maximum is 100. The step value (the number the scale adjusts by), which has a default of 1 can also be manipulated. To change these values, the following options can be used.

```
zenity --scale --min-value=10 --max-value=20 --step-value=2
```

The Scale dialog which appears has two buttons, OK and Cancel. When the OK button is pressed, the value selected is returned to the script where it can then be handled.

Calendar

The Calendar dialog is used to allow a user to select a date. The dialog has two buttons, an OK and Cancel and when OK is selected, the selected data is returned to the program for handling.

```
zenity --calendar
```

By default, the calendar displays the current date when it is loaded. A specific day, month and year can be set.

```
zenity --calendar --day=15 --month=5 --year=1996
```

When the selected date is returned to the script, it will be in the format dd/mm/yy. This can be changed to allow a variety of customisations.

```
zenity --calendar --date-format=%d/%m/%Y
```

The above command uses % to indicate the date type. Using these, Zenity can distinguish between two and four numbers year values, text and numeric month values, etc.

Type	Value	Example
Year	%Y	1996
Year	%y	96
Month	%h	Dec
Month	%B	December
Month	%m	12
Day	%a	Wed
Day	%A	Wednesday
Day	%d	15
Date	%D (default)	15/05/96

Progress

A Progress widget can be used to display the percentage completion of a particular job.

```
zenity --progress --percentage=0
```

The percentage value must be a number between 0 and 100, to indicate the percentage completed. An alternative to displaying a percentage completion is to pulsate the progress bar. This indicates activity but does not indicate the amount completed.

```
zenity --progress --pulsate
```

In some cases, it may be useful to close the dialog when 100% completion occurs. This can be done using.

```
zenity --progress --auto-close
```

List

The List widget is the most complex widget, however is also the most powerful. It provides a way to display content in a table-based layout, with columns identifying the data held in rows. It can also be used to retrieve information from the user in the form of check buttons and radio buttons, along with text input.

A single column example would look as follows.

```
zenity --list --column=Pet Dog Cat Hamster
```

A multiple column example would look as follows.

```
zenity --list --column=Pet --column=Name Dog Spot Cat Simba Hamster Nibble
```

Note that if the values to be inserted into the list contain more than one word, apostrophes are required, otherwise zenity will treat each word as a separate item.

To change the text displayed on the dialog.

```
zenity --list --text='List of animals' --column=Pet Dog Cat Hamster
```

As mentioned previously, check buttons and radio buttons can also be added to allow for user input.

```
zenity --list --checklist --column=Selected --column=Pet True Dog False Cat  
False Hamster
```

```
zenity --list --radiolist --column=Selected --column=Pet False Dog False Cat  
True Hamster
```

The True and False values indicate whether the check button for the appropriate item is selected or not.

The major difference between check buttons and radio buttons is that multiple check buttons can be selected at a time, whereas only one radio button can be selected at a time.

To enable multiple selections within the List.

```
zenity --list --multiple --column=Pet Dog Cat Hamster
```

If the List only contains one column, it might be useful to hide the headers as they are usually not required in this circumstance.

```
zenity --list --hide-header --column=Pet Dog Cat Hamster
```

Notification

A Notification displays an icon in the Notification Area / System Tray of the desktop. This can then be used to indicate the completion of a job, or an error occurring.

```
zenity --notification --text="This is a notification"
```

When the notification is clicked, it is removed from the Notification Area / System Tray.

File Selection

The File Selection dialog allows for the selection of a single or multiple files and directories.

```
zenity --file-selection
```

The File Selection has two buttons, an OK and a Cancel button. When OK is pressed, any files or directories which are currently selected are returned to your program for handling.

To enable the selection of multiples files or directories.

```
zenity --file-selection --multiple
```

To enable the ability to select directories from the File Selection.

```
zenity --file-selection --directory
```

Another useful function of the File Selection dialog is the ability to use it to save files if necessary.

```
zenity --file-selection --save
```

By default, when multiple items have been selected using the File Selection dialog, the paths to those files are separated by the '|' character. This can be changed to another character, such as a comma.

```
zenity --file-selection --separator=", "
```

General Parameters

There are a range of general parameters which can be specified on any of the widgets above.

<code>--title="Title text"</code>	displays a title on the dialog
<code>--width=100</code>	sets the width of the dialog box
<code>--height=100</code>	sets the height of the dialog box
<code>--icon-path="/home/user/icon.png"</code>	displays an icon in the title bar of the window