

# Como Criar Programas Gráficos e Interativos com Shell Script

No mundo do **Shell Script** há possibilidade de criarmos também programas interativos pelo próprio terminal e programas gráficos. Para isso temos muitas ferramentas de comandos, onde um ou mais delas já veem previamente instaladas nas distribuições **Linux**. Aqui vamos falar de duas ferramentas, mas citaremos uma série de outras pra você pesquisar e aprender.

- **Dialog**

(Xdialog, Kdialog, Gdialog, Whiptail, PythonDialog, Udpn,...)

- **Yad**

(evolução do Zenity)

# Dialog

**Dialog** é um aplicativo usado em **shell scripts** que cria widgets (*menus, avisos, barras de progresso, etc*) em modo texto (**CLI**). Ele usa a biblioteca *curses* ou a *ncurses*. Versões posteriores proveem aos usuários a possibilidade de usar o mouse, isto é, em um **xterm**. Foi criado por Savio Lam (primeira versão reportada [0.3] em 1994). Ele foi posteriormente modificado por várias pessoas. Desde 1999 tem sido mantido (e re-escrito) por Thomas Dickey. Existe somente um fork para **FreeBSD**.



Meu Primeiro Programa com Dialog Shell Script

<Aceitar>



# Comandos do Dialog

Para criar uma tela simples basta rodar esse **comando** na **Shell**

*dialog --msgbox 'Meu Primeiro Programa com Dialog Shell Script' 5 50*

COMANDO



MENSAGEM



LARGURA



**dialog** --msgbox 'Meu Primeiro Programa com Dialog Shell Script' 5 50



TIPO DE DIÁLOGO



ALTURA

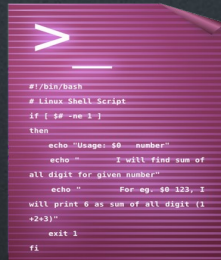
# Tipos de Caixa do Dialog

Tipo da caixa	Desenha uma caixa onde o usuário...
calendar	Vê um calendário e escolhe uma data
checkboxlist	Vê uma lista de opções e escolhe várias
fselect	Digita ou escolhe um arquivo
gauge	Vê uma barra de progresso (porcentagem)
infobox	Vê uma mensagem, sem botões
inputbox	Digita um texto qualquer
menu	Vê um menu e escolhe um item
msgbox	Vê uma mensagem e aperta o botão OK
passwordbox	Digita uma senha
radiolist	Vê uma lista de opções e escolhe uma
tailbox	Vê a saída do comando tail -f
tailboxbg	Vê a saída do comando tail -f (em segundo plano)
textbox	Vê o conteúdo de um arquivo
timebox	Escolhe um horário
yesno	Vê uma pergunta e aperta o botão YES ou o NO



Vamos ver um mini-aplicativo que dá pra aprender várias possibilidades de uma só vez e explicando cada parte do mesmo.

distros.sh



# Existem diversos programas clones/estilo/baseados em Dialog

- **Xdialog** - É o Dialog para a interface gráfica, que usa a biblioteca Gtk+.  
<http://xdialog.dyns.net/>
- **Kdialog** - O Dialog do time do KDE, que usa a biblioteca Qt.  
<http://developer.kde.org/documentation/tutorials/kdialog/t1.html>
- **Whiptail** - Escrito pela Red Hat, é um clone modo texto, que usa a biblioteca newt ao invés da ncurses.  
<http://linux.math.tifr.res.in/manuals/man/whiptail.html>
- **Udpm** - O "User Dialog Perl Module" não é um clone do Dialog, mas uma interface Perl para trabalhar de maneira padronizada com os seguintes "sabores": dialog, cdialog, whiptail, gdialog e Kdialog.  
<http://udpm.sourceforge.net/>
- **PythonDialog** - Módulo em Python para usar trabalhar de maneira padronizada com os seguintes "sabores": dialog, whiptail e Xdialog.  
<http://people.via.ecp.fr/~flo/2002/pythondialog/pythondialog.html>
- **Gdialog** — Se tornou Zenity, que por sua vez foi evoluído e se tornou o Yad veremos a seguir - O antigo Dialog do time do Gnome, que usa a biblioteca Gtk. Aparentemente o Zenity é o novo "Dialog oficial" do Gnome e o gdialog vai ser aposentado. O interessante desse clone é que carrega o nome do ilustre Alan Cox como desenvolvedor participante.  
<http://freshmeat.net/projects/zenity>

**Dica:** É possível fazer um Shell Script que escolha usar o Dialog texto ou o gráfico, dependendo se o usuário está no X ou não. Basta checar a existência da variável \$DISPLAY, que só é definida quando o X está sendo executado.



# YAD

**Yad** significa para o **Shell Script** "**Y**et **A**nother **D**ialog" em *português* "Ainda outro diálogo". É um fork do **Zenity**, melhorado. É uma ferramenta para criar interface gráfica para o usuário (GUI) em **Shell Script**.

Para instalar o **YAD** no seu sistema, veja o passo a passo nesse link

## Como Compilar o YAD no GNU/Linux

<http://terminalroot.com.br/2017/03/como-compilar-o-yad-no-gnu-linux.html>

A sinopse do programa é : **yad** [--DIALOG] [--OPTIONS]

MANUAL: **man yad**

Existem várias opções pro **YAD** (--entry, --calendar, --color,...)

Você pode consultar exemplos em: <https://sourceforge.net/p/yad-dialog/wiki/Examples/>

Vamos criar um **Formulário de Contato com Yad**, que inclui diversas opções de uma só vez e dá pra entender como funciona.