

Simple Dialog and Menu Function Guide (version 1.0)
BE-205, Winter '08-'09, Dr. C. S. Tritt

Modern computer users expect relatively sophisticated user interfaces. Matlab provides some built in functions that make creating interfaces having graphical controls (dialog boxes) relatively easy. A dialog box is a popup graphical window that accepts user input or displays a result. This handout introduces three functions (*inputdlg*, *msgbox* and *menu*) that can be used for user input and to display program results. These functions can have a number of arguments and take a number of forms, so see Matlab's online documentation for more information.

There are two ways in which graphical dialogs can operate. Modal operation involves the dialog being displayed and program execution pausing until the user supplies a response (usually in the form of using the mouse to click an on screen button). Non-modal operation involves the program displaying a dialog and continuing to execute.

The *inputdlg* function is used for modal user input. An example of the output produced by this function is shown in Figure 1. In its simplest form, *inputdlg* takes a single cell array argument containing one or more strings that are displayed as prompts. The user is provided with space in which to enter their response to each prompt. When the user clicks the "OK" button, *inputdlg* returns a cell array containing each of their responses in string form.

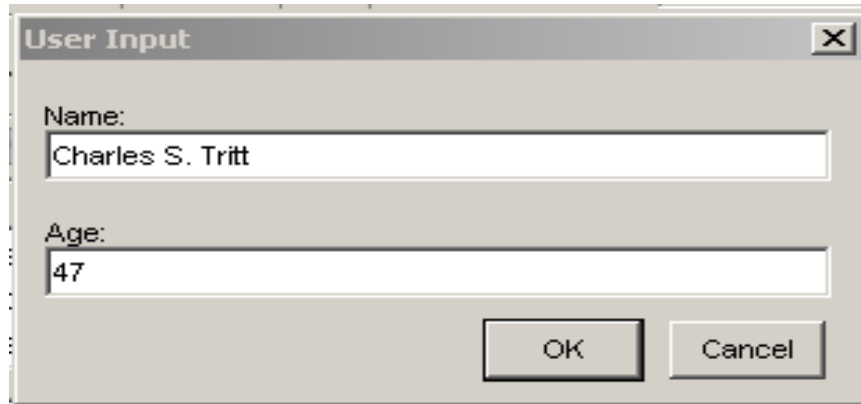


Figure 1: Display produced by the *inputdlg* function.

The *msgbox* function is used for modal or non-modal user output. An example of the output produced by this function is shown in Figure 2. In its simplest form, *msgbox* takes a single string or cell array argument and operates non-modally.

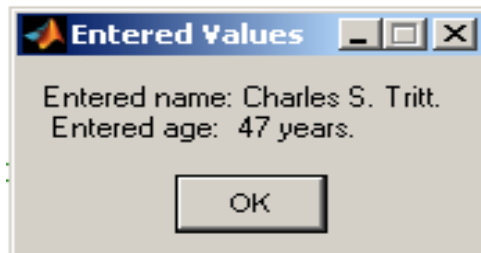


Figure 2: Display produced by the *msgbox* function.

The *menu* function is used for modal user input from a limited list of options. An example of the output displayed by this function is shown in Figure 3. It appears to be of a different generation and somewhat less sophisticated than *inputdlg* and *msgbox*. It accepts a series of strings as arguments. The first is a title that is displayed on the control and the remain strings are used to label the option buttons on the menu. When the user clicks one of these option buttons, the menu function returns the sequential number of the option selected.

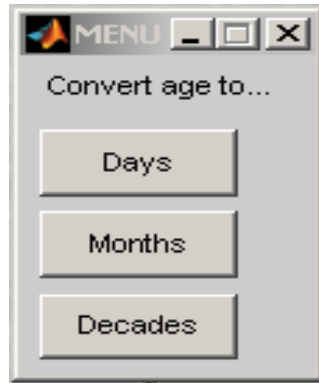


Figure 3: Display produced by the *menu* function.

As an example, a simple program code that demonstrates the use of each of these functions and was used to produce Figures 1 through 3 is attached.

I noticed a few unexpected behaviors while experimenting with these function on Windows XP. According to the Matlab documentation, *msgbox* controls can be made to behave modally, but I couldn't get this feature to work. Also the *menu* dialog appears on a different part of the screen than the *inputdlg* and *msgbox* dialogs. Finally, Matlab tended to behave strangely if I failed to close all open dialogs prior to returning to editing and rerunning the source code.

Sample Code

```
% Prompt for and get user input.

in_dlg_title = 'User Input';
prompt{1} = 'Name: ';
prompt{2} = 'Age: ';
answer = inputdlg(prompt, in_dlg_title);

% Extract name and age from user input.

name = answer{1};
age = str2double(answer{2});

% Display result.

out_dlg_title = 'Entered Values';
out_string = sprintf(...
    'Entered name: %s.\nEntered age: %3.0f years.', ...
    name, age);
msgbox(out_string, out_dlg_title, 'modal'); % Modal doesn't
work.

% Prompt for conversion type and convert age.

conv_type = menu('Convert age to...', 'Days', 'Months',
'Decades');

switch conv_type
    case 1
        conv_age = 365.25*age;
        conv_units = 'days';
    case 2
        conv_age = 12*age;
        conv_units = 'months';
    case 3
        conv_age = age/10;
        conv_units = 'decades';
end

% Display the results.

out_string = sprintf(...
    'Entered name: %s is %.1f %s old.', ...
    name, conv_age, conv_units);

msgbox(out_string, 'Converted Age');
```