Button Group Demo (Version 1.1)
Created by Dr. C. S. Tritt
Last revised: January 24, 2007

Background and Equations

None required.

Operations Description

This program demonstrates the use of button groups, radio buttons, grouped push buttons and panels. The GUI consists of two grouped push buttons (labeled On and Off) and three grouped radio buttons (Red, Green and Blue) and a panel. When the Off button is selected, the panel is gray (regardless of the radio button states) and the radio buttons are enabled. When the On button is selected, the panel changes to the specified color and the radio buttons are disabled.

User Interface Description

An unlabeled panel (statusPanel) that changes colors in response to user selections.

A button group (powerGroup) labeled “Power” containing “On” and “Off” toggle buttons.

Two toggle buttons (onToggle and offToggle) labeled “On” and “Off,” respectively.

A button group (colorGroup) labeled “Color” containing “Red,” “Green” and “Blue” radio buttons.

Three radio buttons (redRadio, greenRadio and blueRadio) labeled “Red,” “Green” and “Blue,” respectively.

Algorithms

Note: It is probably not necessary in this example to maintain the state of this GUI in application specific handle properties. However, it is often necessary to do so in “real” programs and therefore I am choosing to do so here.

OpeningFcn

Preset the buttons to the desired state (offToggle and greenRadio) selected.

Create application handle properties (PowerStr and ColorStr) and store the initial state in them. Note: Allowed PowerStr values are “off” and “on” and ColorStr values are “red,” “green” and “blue.”

Update the central GUI data structure.

colorGroup_SelectionChangeFcn

```
Retrieve tag of selected button.

Switch based on hObject tag

    Case redRadio:
        Change stored state (ColorStr) to ‘red.’
    Case blueRadio:
        Change stored state (ColorStr) to ‘blue.’
    Case greenRadio:
.       Change stored state (ColorStr) to ‘green.’

Update central GUI data.
```

powerGroup_SelectionChangeFcn

```
Retrieve the current state of the powerGroup (tag of selected option).

Switch based on hObject tag

    Case onToggle

        Save new power state (PowerStr)
        Disable color buttons

        If  stored color state (handles.ColorStr) is red
            Set status panel (handles.statusPanel) background color to red.
        Else if stored color state (handles.ColorStr) is blue
            Set status panel (handles.statusPanel) background color to blue.
        Else if stored color state (handles.ColorStr) is green
            Set status panel (handles.statusPanel) background color to green.

    Case offToggle

        Save new power state  (PowerStr)
        Enable color buttons.
        Reset status panel (handles.statusPanel) background color to initial color
          (handles.InitColor).
```

Source Code

```
function varargout = groupDemo(varargin)
% GROUPDEMO M-file for groupDemo.fig
%
% See ButtonGroupDemo.doc file for more information.
% The status panel part of this program is not yet working.
%
% Created by Dr. C. S. Tritt
% Last revised 1/19/07
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Copyright 2002-2003 The MathWorks, Inc.

% Edit the above text to modify the response to help groupDemo

% Last Modified by GUIDE v2.5 18-Jan-2007 09:38:57

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @groupDemo_OpeningFcn, ...
                   'gui_OutputFcn',  @groupDemo_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT


% --- Executes just before groupDemo is made visible.
function groupDemo_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to groupDemo (see VARARGIN)

% Choose default command line output for groupDemo
handles.output = hObject;

% Store initial status panel background color.
handles.InitColor = get(handles.statusPanel, 'backgroundColor');

% Initialize the GUI state.
set(handles.offToggle, 'Value', 1); % Preset this option. Assume Max is 1.
```

```matlab
set(handles.greenRadio, 'Value', 1); % Preset this option. Assume Max is 1.

% Create and initialize the handles state properties.
handles.PowerStr = 'off'; % Allowed values: off, on.
handles.ColorStr = 'green';% Allowed values: red, green, blue.

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes groupDemo wait for user response (see UIRESUME)
% uiwait(handles.figure1);


% --- Outputs from this function are returned to the command line.
function varargout = groupDemo_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;


% --------------------------------------------------------------------
function colorGroup_SelectionChangeFcn(hObject, eventdata, handles)
% hObject    handle to colorGroup (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Dr. Tritt's Comments...
% The group tag is used to generate the name of the ChangeFcn callback.
% The hObject passed to this callback is the newly selected button.

switch get(hObject,'Tag')
    case 'redRadio'
        handles.ColorStr = 'red';
    case 'blueRadio'
        handles.ColorStr = 'blue';
    case 'greenRadio'
        handles.ColorStr = 'green';
end

% Update handles structure
guidata(hObject, handles);

% --------------------------------------------------------------------
function powerGroup_SelectionChangeFcn(hObject, eventdata, handles)
% hObject    handle to powerGroup (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

switch get(hObject,'Tag')
    case 'onToggle'
        handles.PowerStr = 'on'; % Save the new state.

        % Disable the color radio buttons.
        set(handles.redRadio, 'Enable', 'off');
        set(handles.greenRadio, 'Enable', 'off');
        set(handles.blueRadio, 'Enable', 'off');

        % Change the statusPanel color.
        if strcmp(handles.ColorStr, 'red')
           set(handles.statusPanel, 'backgroundColor', [1 0 0])
        elseif strcmp(handles.ColorStr, 'green')
           set(handles.statusPanel, 'backgroundColor', [0 0.7 0])
        elseif strcmp(handles.ColorStr, 'blue')
           set(handles.statusPanel, 'backgroundColor', [0 0 1])
        end
    case 'offToggle'

        handles.PowerStr = 'off'; % Save the new state.

        % Enable the color radio buttons.
        set(handles.redRadio, 'Enable', 'on');
        set(handles.greenRadio, 'Enable', 'on');
        set(handles.blueRadio, 'Enable', 'on');

        % Restore initial status panel background color.
        set(handles.statusPanel, 'BackgroundColor', handles.InitColor);
end
```