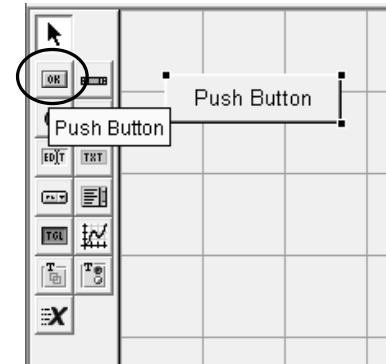


# Giao diện người dùng (GUI)

Viết lệnh cho các thành phần GUI

## Push Button (Nút nhấn)



## Hàm xử lý Callback của Push Button

- Khi nhấn chuột vào một nút nhấn, hàm sau được gọi  
`function pushbutton1_Callback(hObject, eventdata, handles)`  
... ..

Tên tag của button

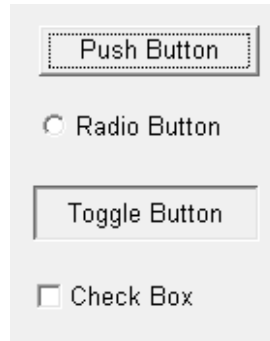
**VD:** nhấn nút thì thay tên của nút (thuộc tính 'String' của nút)

```
function pushbutton1_Callback(hObject,  
    eventdata, handles)  
set(gcbo, 'String', 'ten gi do');
```

## Một số thuộc tính quan trọng của Button

- **Enable:** cho phép nhấn (on/off)
- **String:** chuỗi hiển thị trên nút
- **Tag:** tên đối tượng nút phân biệt với các đối tượng khác trong chương trình
- **Value:** giá trị của nút.

## Toggle Button, Radio Button và Check Box (Nút bật tắt, Nút chọn và Ô check)



## Thuộc tính của Toggle Button

- Thuộc tính **Value** sẽ bằng thuộc tính **Max** nếu nút được nhấn hoặc được chọn, ngược lại sẽ bằng thuộc tính **Min**

```
function togglebutton1_Callback(hObject, eventdata, handles)
    button_state = get(hObject, 'Value');
    if button_state == get(hObject, 'Max')
        % Toggle button được nhấn
        ...
    elseif button_state == get(hObject, 'Min')
        % Toggle button không được nhấn
        ...
    end
```

## Toggle Button, Radio Button và Check Box (Nút bật tắt, Nút chọn và Ô check)

- Mặc định **Max=1**, **Min=0**
- Cũng có thể thay đổi trạng thái cho các Toggle Button, Radio Button và Check Box bằng cách đặt lại các thuộc tính **Value**, **Max**, **Min**

- Ví dụ:

```
set(handles.togglebutton1, 'Value',
    get(handles.togglebutton1, 'Max')) đặt nút
togglebutton1 sang trạng thái được nhấn
set(handles.radiobutton1, 'Value',
    get(handles.radiobutton1, 'Min')) đặt nút
radiobutton1 sang trạng thái bỏ chọn
```

## Edit Text (Ô nhập văn bản)



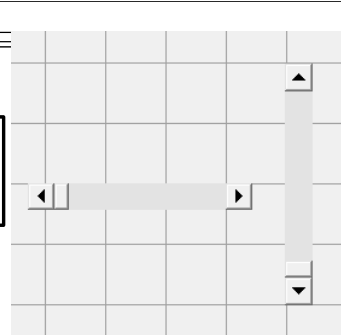
## Edit Text (Ô nhập văn bản)

- Thuộc tính **String** chứa chuỗi mà người dùng nhập vào  
**function edittext1\_Callback(hObject, eventdata, handles)**  
**user\_string = get(hObject, 'String');**  
**% các lệnh xử lý tiếp theo**
- Nếu thuộc tính **Max**, **Min** được đặt sao cho **Max-Min>1** thì người dùng có thể nhập nhiều hàng
- **Callback** được gọi khi đang thay đổi nội dung của Edit Text mà nhấp chuột ra bên ngoài (nhưng vẫn bên trong GUI), hoặc khi người dùng nhấn Enter (với Edit Text một hàng), Ctrl+Enter (với Edit Text nhiều hàng)

## Nhận dữ liệu số từ Edit Text

- Thuộc tính **String** trả về một chuỗi. Muốn nhận dữ liệu số thì sử dụng hàm **str2double**. Nếu người dùng không nhập số thì **str2double** trả về **NaN**  
**function edittext1\_Callback(hObject, eventdata, handles)**  
**user\_entry =**  
**str2double(get(hObject, 'string'));**  
**if isnan(user\_entry)**  
**errordlg('You must enter a numeric**  
**value', 'Bad Input', 'modal')**  
**return**  
**end**  
**% Proceed with callback...**

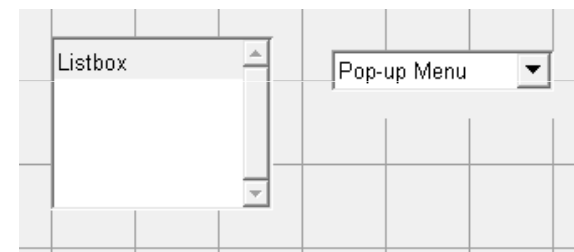
## Slider (thanh trượt)



- Thuộc tính **Value** xác định trạng thái của Slider
- Thuộc tính **Max** và **Min** xác định cực đại và cực tiểu cho thuộc tính **Value** của Slider

```
function slider1_Callback(hObject, eventdata, handles)  
slider_value = get(hObject, 'Value');  
% Proceed with callback...
```

## List Box và Pop-up Menu



## List Box và Pop-up Menu

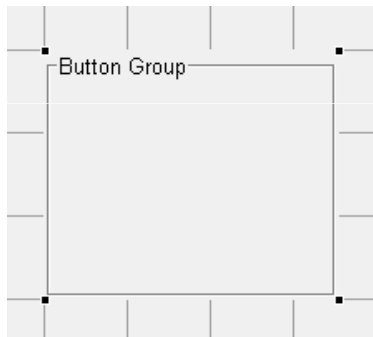
- Thuộc tính **Value** chứa chỉ số của mục được chọn. Chỉ số của mục đầu tiên là 1
- Thuộc tính **String** là mảng cell chứa danh sách các mục chọn của List Box và Pop-up Menu
- Ví dụ dưới đây cho phép lấy giá trị của chuỗi được chọn trong List Box

```
function listbox1_Callback(hObject,
    eventdata, handles)
index_selected = get(hObject,'Value');
list = get(hObject,'String');
item_selected = list{index_selected};
% Convert from cell array to string
```

## Gọi callback của List Box

- Callback của List Box được gọi sau sự kiện nhả chuột hoặc nhấn một số phím đặc biệt
- Khi các phím mũi tên thay đổi thuộc tính **Value** của List Box, callback **Callback** được gọi đồng thời đặt thuộc tính **SelectionType** của figure thành **normal**
- Phím Enter và Space không làm thay đổi thuộc tính **Value** của List Box, nhưng **Callback** vẫn được gọi, đồng thời đặt thuộc tính **SelectionType** của figure thành **open**
- Nếu người dùng nhấp đúp chuột thì **Callback** được gọi sau mỗi lần nhấp. MatLab đặt thuộc tính **SelectionType** của figure thành **normal** ở lần thứ nhất và thành **open** ở lần thứ hai
- Do đó ta có thể biết được người dùng nhấp đúp chuột hay không dựa vào việc truy vấn thuộc tính **SelectionType** của figure

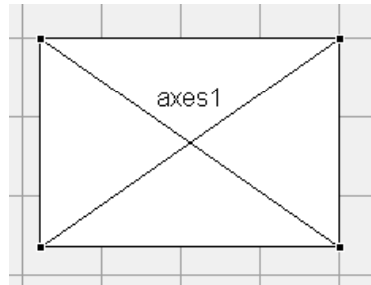
## Button Group



## Button Group

- Button Group chứa các thành phần GUI, đồng thời giúp ta quản lý việc lựa chọn của Radio Button và Toggle Button
- Nếu Button Group chứa Radio Button, Toggle Button hoặc cả hai thì Button Group chỉ cho phép một trong số chúng được chọn. Nếu một cái được chọn thì những cái khác sẽ được bỏ chọn
- Callback **SelectionChangeFcn** của Button Group được gọi khi có sự thay đổi chọn
- Tham số đầu vào **hObject** trỏ tới Radio Button hoặc Toggle Button được chọn
- Các callback của các thành phần khác trong GUI có thể truy xuất tới đối tượng được chọn thông qua thuộc tính **SelectedObject** của Button Group

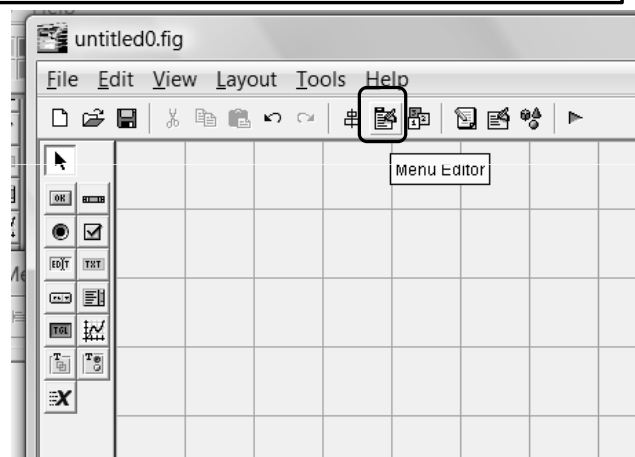
## Axes (Hệ trục tọa độ)



## Axes (Hệ trục tọa độ)

- Hệ trục tọa độ dùng để vẽ đồ thị hoặc hiển thị hình ảnh
  - Muốn vẽ đồ thị lên một hệ trục tọa độ, ta thêm handles tới hệ trục tọa độ đó vào hàm muốn sử dụng
  - Ví dụ sau vẽ đồ thị lên hệ trục **axes1**
- ```
[X,Y] = meshgrid(-3:.125:3);
Z = peaks(X,Y);
meshc(handles.axes1,X,Y,Z);
axis([-3 3 -3 3 -10 5]);
```

## Menu



## Menu

- Để soạn thảo menu chọn biểu tượng

