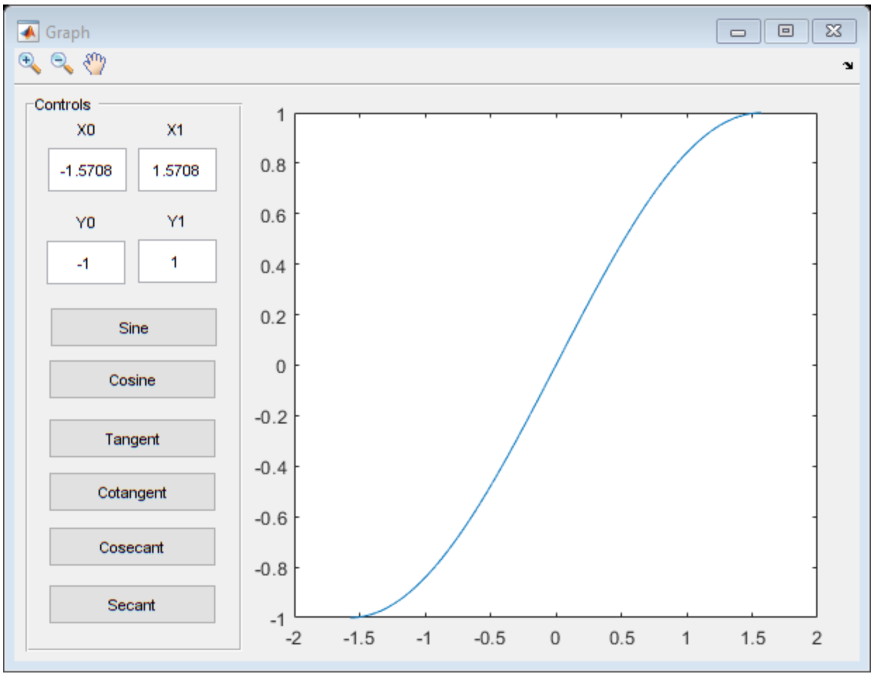
```matlab
% Assignment 7
% Author: Krunal Rank
% Adm No.: U18CO081
% IDE: MATLAB

function varargout = Assignment_7_U18CO081_Figure(varargin)
% ASSIGNMENT_7_U18CO081_FIGURE MATLAB code for Assignment_7_U18CO081_Figure.fig
%      ASSIGNMENT_7_U18CO081_FIGURE, by itself, creates a new ASSIGNMENT_7_U18CO081_FIGURE or raises the existing
%      singleton*.
%
%      H = ASSIGNMENT_7_U18CO081_FIGURE returns the handle to a new ASSIGNMENT_7_U18CO081_FIGURE or the handle to
%      the existing singleton*.
%
%      ASSIGNMENT_7_U18CO081_FIGURE('CALLBACK',hObject,eventData,handles,...) calls the local
%      function named CALLBACK in ASSIGNMENT_7_U18CO081_FIGURE.M with the given input arguments.
%
%      ASSIGNMENT_7_U18CO081_FIGURE('Property','Value',...) creates a new ASSIGNMENT_7_U18CO081_FIGURE or raises the
%      existing singleton*.  Starting from the left, property value pairs are
%      applied to the GUI before Assignment_7_U18CO081_Figure_OpeningFcn gets called.  An
%      unrecognized property name or invalid value makes property application
%      stop.  All inputs are passed to Assignment_7_U18CO081_Figure_OpeningFcn via varargin.
%
%      *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows only one
%      instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help Assignment_7_U18CO081_Figure

% Last Modified by GUIDE v2.5 02-May-2020 12:15:06

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @Assignment_7_U18CO081_Figure_OpeningFcn, ...
                   'gui_OutputFcn',  @Assignment_7_U18CO081_Figure_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 Assignment_7_U18CO081_Figure is made visible.
function Assignment_7_U18CO081_Figure_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 Assignment_7_U18CO081_Figure (see VARARGIN)

% Choose default command line output for Assignment_7_U18CO081_Figure
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);
axes(handles.graphAxes);

X0 = -pi/2;
X1 = pi/2;
Y0 = -1;
Y1 = 1;

set(handles.editTextX0,'String',num2str(X0));
set(handles.editTextX1,'String',num2str(X1));
set(handles.editTextY0,'String',num2str(Y0));
set(handles.editTextY1,'String',num2str(Y1));

x = X0:0.0001:X1;
f = sin(x);
plot(x,f);

% UIWAIT makes Assignment_7_U18CO081_Figure wait for user response (see UIRESUME)
% uiwait(handles.figure1);


% --- Outputs from this function are returned to the command line.
function varargout = Assignment_7_U18CO081_Figure_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 editTextX0_Callback(hObject, eventdata, handles)
% hObject    handle to editTextX0 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of editTextX0 as text
%        str2double(get(hObject,'String')) returns contents of editTextX0 as a double


% --- Executes during object creation, after setting all properties.
function editTextX0_CreateFcn(hObject, eventdata, handles)
% hObject    handle to editTextX0 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end



function editTextX1_Callback(hObject, eventdata, handles)
% hObject    handle to editTextX1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of editTextX1 as text
%        str2double(get(hObject,'String')) returns contents of editTextX1 as a double


% --- Executes during object creation, after setting all properties.
function editTextX1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to editTextX1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end



function editTextY0_Callback(hObject, eventdata, handles)
% hObject    handle to editTextY0 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of editTextY0 as text
%        str2double(get(hObject,'String')) returns contents of editTextY0 as a double


% --- Executes during object creation, after setting all properties.
function editTextY0_CreateFcn(hObject, eventdata, handles)
% hObject    handle to editTextY0 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end



function editTextY1_Callback(hObject, eventdata, handles)
% hObject    handle to editTextY1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of editTextY1 as text
%        str2double(get(hObject,'String')) returns contents of editTextY1 as a double


% --- Executes during object creation, after setting all properties.
function editTextY1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to editTextY1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
```

```
end


% --- Executes on button press in sineButton.
function sineButton_Callback(hObject, eventdata, handles)
% hObject    handle to sineButton (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
strX0 = get(handles.editTextX0,'String');
strX1 = get(handles.editTextX1,'String');
strY0 = get(handles.editTextY0,'String');
strY1 = get(handles.editTextY1,'String');

X0 = sscanf(strX0,'%lf',1);
X1 = sscanf(strX1,'%lf',1);
Y0 = sscanf(strY0,'%lf',1);
Y1 = sscanf(strY1,'%lf',1);
if ~isempty(X0) && ~isnan(X0) && ~isempty(X1) && ~isnan(X1) && ~isempty(Y0) && ~isnan(Y0) && ~isempty(Y1) && ~isnan(Y1)
    if X0 < X1 && Y0 < Y1
        x = X0:0.0001:X1;
        f = sin(x);
        plot(x,f);
        xlim([X0 X1]);
        ylim([Y0 Y1]);
    else
        f = msgbox({'Invalid Value:';'Please check whether X0 < X1 and Y0 < Y1.'},'Invalid Value Error','error');
    end
else
    f = msgbox({'Invalid Value:';'Please check values for X0, X1, Y0 and Y1.';'Note the constraints X0<X1 and Y0<Y1';},'Invalid Value Error','error');
end

% --- Executes on button press in cosButton.
function cosButton_Callback(hObject, eventdata, handles)
% hObject    handle to cosButton (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
strX0 = get(handles.editTextX0,'String');
strX1 = get(handles.editTextX1,'String');
strY0 = get(handles.editTextY0,'String');
strY1 = get(handles.editTextY1,'String');

X0 = sscanf(strX0,'%lf',1);
X1 = sscanf(strX1,'%lf',1);
Y0 = sscanf(strY0,'%lf',1);
Y1 = sscanf(strY1,'%lf',1);
if ~isempty(X0) && ~isnan(X0) && ~isempty(X1) && ~isnan(X1) && ~isempty(Y0) && ~isnan(Y0) && ~isempty(Y1) && ~isnan(Y1)
    if X0 < X1 && Y0 < Y1
        x = X0:0.0001:X1;
        f = cos(x);
        plot(x,f);
        xlim([X0 X1]);
        ylim([Y0 Y1]);
    else
        f = msgbox({'Invalid Value:';'Please check whether X0 < X1 and Y0 < Y1.'},'Invalid Value Error','error');
    end
else
    f = msgbox({'Invalid Value:';'Please check values for X0, X1, Y0 and Y1.';'Note the constraints X0<X1 and Y0<Y1';},'Invalid Value Error','error');
end

% --- Executes on button press in tanButton.
function tanButton_Callback(hObject, eventdata, handles)
% hObject    handle to tanButton (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
strX0 = get(handles.editTextX0,'String');
strX1 = get(handles.editTextX1,'String');
strY0 = get(handles.editTextY0,'String');
strY1 = get(handles.editTextY1,'String');

X0 = sscanf(strX0,'%lf',1);
X1 = sscanf(strX1,'%lf',1);
Y0 = sscanf(strY0,'%lf',1);
Y1 = sscanf(strY1,'%lf',1);
if ~isempty(X0) && ~isnan(X0) && ~isempty(X1) && ~isnan(X1) && ~isempty(Y0) && ~isnan(Y0) && ~isempty(Y1) && ~isnan(Y1)
    if X0 < X1 && Y0 < Y1
        x = X0:0.0001:X1;
        f = tan(x);
        plot(x,f);
        xlim([X0 X1]);
        ylim([Y0 Y1]);
    else
        f = msgbox({'Invalid Value:';'Please check whether X0 < X1 and Y0 < Y1.'},'Invalid Value Error','error');
    end
else
    f = msgbox({'Invalid Value:';'Please check values for X0, X1, Y0 and Y1.';'Note the constraints X0<X1 and Y0<Y1';},'Invalid Value Error','error');
end

% --- Executes on button press in cotButton.
function cotButton_Callback(hObject, eventdata, handles)
% hObject    handle to cotButton (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
strX0 = get(handles.editTextX0,'String');
```

```
strX1 = get(handles.editTextX1,'String');
strY0 = get(handles.editTextY0,'String');
strY1 = get(handles.editTextY1,'String');

X0 = sscanf(strX0,'%lf',1);
X1 = sscanf(strX1,'%lf',1);
Y0 = sscanf(strY0,'%lf',1);
Y1 = sscanf(strY1,'%lf',1);
if ~isempty(X0) && ~isnan(X0) && ~isempty(X1) && ~isnan(X1) && ~isempty(Y0) && ~isnan(Y0) && ~isempty(Y1) && ~isnan(Y1)
    if X0 < X1 && Y0 < Y1
        x = X0:0.0001:X1;
        f = cot(x);
        plot(x,f);
        xlim([X0 X1]);
        ylim([Y0 Y1]);
    else
        f = msgbox({'Invalid Value:';'Please check whether X0 < X1 and Y0 < Y1.'},'Invalid Value Error','error');
    end
else
    f = msgbox({'Invalid Value:';'Please check values for X0, X1, Y0 and Y1.';'Note the constraints X0<X1 and Y0<Y1';},'Invalid Value Error','error');
end

% --- Executes on button press in cosecButton.
function cosecButton_Callback(hObject, eventdata, handles)
% hObject    handle to cosecButton (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
strX0 = get(handles.editTextX0,'String');
strX1 = get(handles.editTextX1,'String');
strY0 = get(handles.editTextY0,'String');
strY1 = get(handles.editTextY1,'String');

X0 = sscanf(strX0,'%lf',1);
X1 = sscanf(strX1,'%lf',1);
Y0 = sscanf(strY0,'%lf',1);
Y1 = sscanf(strY1,'%lf',1);
if ~isempty(X0) && ~isnan(X0) && ~isempty(X1) && ~isnan(X1) && ~isempty(Y0) && ~isnan(Y0) && ~isempty(Y1) && ~isnan(Y1)
    if X0 < X1 && Y0 < Y1
        x = X0:0.0001:X1;
        f = csc(x);
        plot(x,f);
        xlim([X0 X1]);
        ylim([Y0 Y1]);
    else
        f = msgbox({'Invalid Value:';'Please check whether X0 < X1 and Y0 < Y1.'},'Invalid Value Error','error');
    end
else
    f = msgbox({'Invalid Value:';'Please check values for X0, X1, Y0 and Y1.';'Note the constraints X0<X1 and Y0<Y1';},'Invalid Value Error','error');
end

% --- Executes on button press in secButton.
function secButton_Callback(hObject, eventdata, handles)
% hObject    handle to secButton (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
strX0 = get(handles.editTextX0,'String');
strX1 = get(handles.editTextX1,'String');
strY0 = get(handles.editTextY0,'String');
strY1 = get(handles.editTextY1,'String');

X0 = sscanf(strX0,'%lf',1);
X1 = sscanf(strX1,'%lf',1);
Y0 = sscanf(strY0,'%lf',1);
Y1 = sscanf(strY1,'%lf',1);
if ~isempty(X0) && ~isnan(X0) && ~isempty(X1) && ~isnan(X1) && ~isempty(Y0) && ~isnan(Y0) && ~isempty(Y1) && ~isnan(Y1)
    if X0 < X1 && Y0 < Y1
        x = X0:0.0001:X1;
        f = sec(x);
        plot(x,f);
        xlim([X0 X1]);
        ylim([Y0 Y1]);
    else
        f = msgbox({'Invalid Value:';'Please check whether X0 < X1 and Y0 < Y1.'},'Invalid Value Error','error');
    end
else
    f = msgbox({'Invalid Value:';'Please check values for X0, X1, Y0 and Y1.';'Note the constraints X0<X1 and Y0<Y1';},'Invalid Value Error','error');
end
```



Published with MATLAB® R2018b