Welcome back to **CME 292**
Advanced MATLAB for Scientific Computing

WINTER 2023

Stanford University

Additional Topic: Interactive Figures and Images in MATLAB

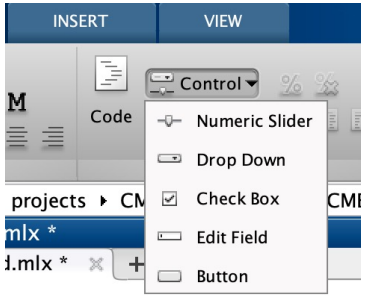
CME 292 LECTURE 2

2/2/2023

Graphical user interfaces (GUIs), also known as apps, provide point-and-click control of your software applications, eliminating the need for others to learn a language or type commands in order to run the application.

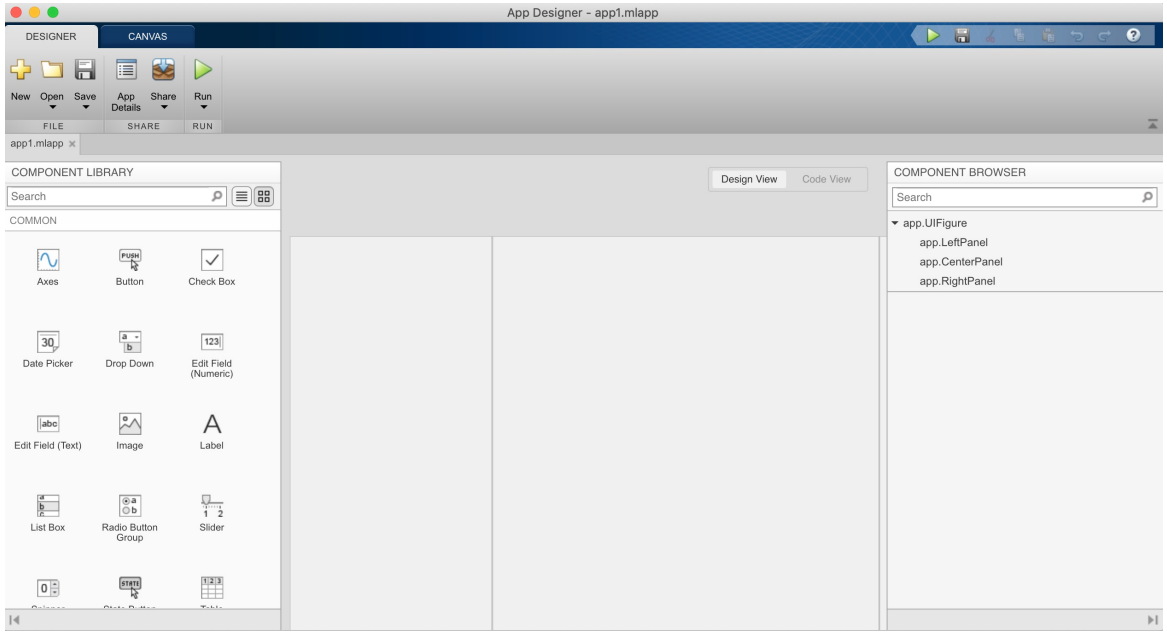
3 ways to create an app in MATLAB

1. Convert a script into a simple app
   - Use the Live Editor to convert a script into a simple app that has interactive controls allowing others to experiment with variables in your code.
   - Add sliders, dropdowns, edit fields, and buttons without writing any code.
   - Specify what parts of the script will run when a value is changed.
   - Hide the code to create simple apps and dashboards.



2. Create an app interactively

- Use App Designer, an interactive environment that integrates the two primary tasks of app building
- Lay out the visual components
- Program the app's behavior.



3. Create an app programmatically

- use MATLAB functions to define the layout and behavior of the app

GUI With Interactive Response-Plot Updates

`uicontrol` : create user interface control

Style of `UIControl` object:

- `pushbutton`
- `togglebutton`
- `checkbox`
- `radiobutton`
- `edit`
- `text`
- `slider`
- `listbox`
- `popupmenu`

- `uifigure` creates a figure for building a user interface and returns the `Figure` object.
- `uigridlayout` creates a grid layout manager for an app. It positions UI components along the rows and columns of an invisible grid that spans the entire figure or a container within the figure.
- `uipanel` creates a panel in the current figure and returns the `Panel` object.
- `uibutton` creates a push button in a new figure and returns the `Button` object.
- `uislider` creates a slider in a new figure window and returns the `Slider` object.

Demo

Share Data Among Callbacks

Graphics callback functions must accept at least two input arguments:

- The handle of the object whose callback is executing
  - used within the callback function to refer to the callback object
- The event data structure
  - can be empty for some callbacks or contain specific information that is described in the property description for that object

UserData is a property that all graphics objects in MATLAB possess, and can be used to store any single, user-defined array with a particular object.

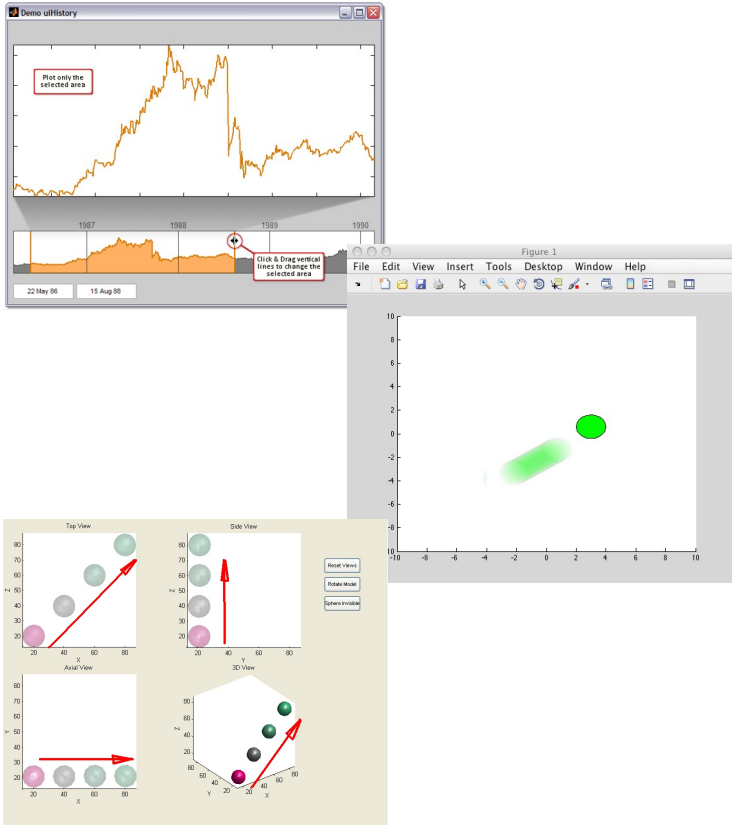
Demo

Some interesting examples of GUI on File Exchange

- Interactive Data Navigation Pane Widget
  https://www.mathworks.com/matlabcentral/fileexchange/23423-interactive-data-navigation-pane-widget

- Moveit2 - Move a graphical object with the mouse
  https://www.mathworks.com/matlabcentral/fileexchange/23304-moveit2-move-a-graphical-object-with-the-mouse

- Move a 3D object with mouse in a traditional 4-view window
  https://www.mathworks.com/matlabcentral/fileexchange/29140-move-a-3d-object-with-mouse-in-a-traditional-4-view-window



Identify Coordinates Using Cursor

```
[x,y] = ginput(n)
```

This allows us to identify the coordinates of n points within Cartesian, polar, or geographic axes.

- To choose a point, move the cursor to the desired location and press either a mouse button or a key on the keyboard.
- Press the **Return** key to stop before all n points are selected.
- The coordinates of your selected points are returned in `x,y`.

Demo

Create ROI (Region of Interest) in Images

A **region of interest (ROI)** is a portion of an image that you want to filter or operate on in some way.

We can represent an ROI as a binary mask image.

In the mask image, pixels that belong to the ROI are set to 1 and pixels outside the ROI are set to `0`.

[Image Processing Toolbox]

- masking out region by selection
- freehand ROI

Fun with MATLAB

LET'S PLAY THE GAME!

`xpbombs`