Autodesk MotionBuilder 2013

Programming in MotionBuilder || Focusing on Python

Naiqi Weng, Developer Consultant
Autodesk Developer Network

Module 5

Python UI

Module 5

Module’s Agenda

- UI Components
- Pre-Built UI Functionality
  - Message Box
  - Viewer Progress Bar
  - Additional Helper Class
- Building UI Functionality from Scratch
- Available Callback Classes
- Other possible UI toolkits and MotionBuilder
- Assignment

Creating UI components

- FBVisualComponents

ELEMENT	CLASS	DESCRIPTION
Arrow Button	FBArrowButton	Button with an Arrow
Button	FBButton	Button
List	FBList	List of items
Container	FBContainer	Container (similar to list)
Edit	FBEdit	Text edit box
Edit Number	FBEditNumber	Number edit box
Edit Color	FBEditColor	Color edit tool
Edit Vector	FBEditVector	Vector edit tool
Edit Property	FBEditProperty	Property Editing
Updated Edit Property	FBEditPropertyModern	Updated Property Editing
Image	FBImage	UI Image
Label	FBLabel	Text label
List	FBList	A List of Items
Memo	FBMemo	An editable file
Scroll Box	FBScrollBox	An area that scrolls
Slider	FBSlider	Slider
Spreadsheet	FBSpread	Spreadsheet
Tab Panel	FBTabPanel	Panel with tabs
Thermometer	FBThermometer	Graphical tool
Time Code	FBTimeCode	Time Display
Tree	FBTree	Hierarchal viewer

Pre-Built UI Functionality

- Message Box with up to Three Buttons
- Message Box with up to Three Buttons and a Check Box Option
- Message Box with up to Three Buttons and allows for user data

Viewer Progress Bar

- FBProgess
- Sits inside the Viewer
- Great for all types of status updates

Additional Helper Classes

- The ‘FBTabControl’ Class
- The ‘FBButtonGroup’ Class

- Located inside pyfbsdk_additions.py module

Building UI Functionality from Scratch

- We are going to look at building UI in a 3 stage process:
  - Creating and registering your tool
  - Creating UI components, buttons, lists, tabs, etc.
  - Attaching and playing your UI Components in your tool
    - Three Workflows

Creating and Registering your Tool

- Module pyfbsdk_additions.py
- Creating and registering your tool
- Setting the Size of your Tool Window
- Setting your Window to Show
- GetTools Helper Function

Creating Tool Issues

- Creating a Tool should not be confused with Showing a Tool
- Creating tools multiple times
  - CreateTool()
  - CreateUniqueTool()
- Python Tool Manager
  - C:\Program Files\Autodesk\MotionBuilder 2013\bin\config\PythonStartup
- Drag a script to show a tool
  - Scripts\UI\SafeToolCreationExample.py

Attaching and playing your UI Components in your Tool

- Workflow 1 – The Original Framework
- Workflow 2 – Box Layouts
- Workflow 3 – Grid Layouts

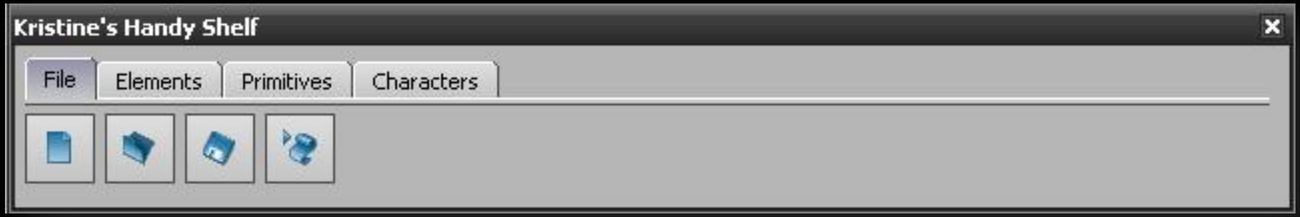
Available Callback Classes

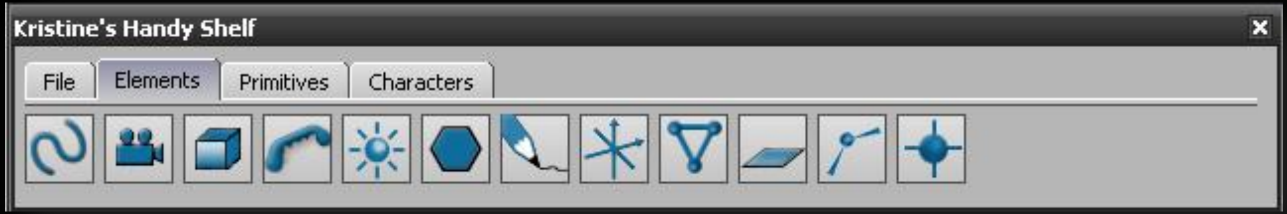
- FBEventActivate()
- FBEventDblClick()
- FBEventDragAndDrop()
- FBEventExpose()
- FBEventInput()
- FBEventMenu()
- FBEventResize()
- FBEventSceneChange()
- FBEventShow()
- FBEventSpread()
- FBEventTakeChange()
- FBEventTransaction()
- FBEventTree()
- FBEventTreeSelect()

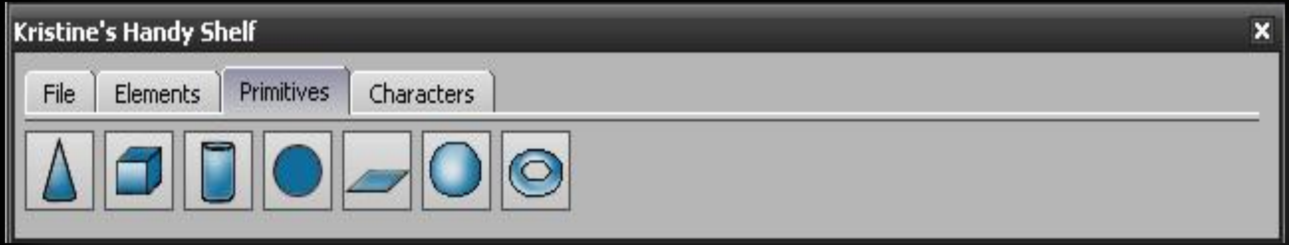
Other tool kits and MoBu

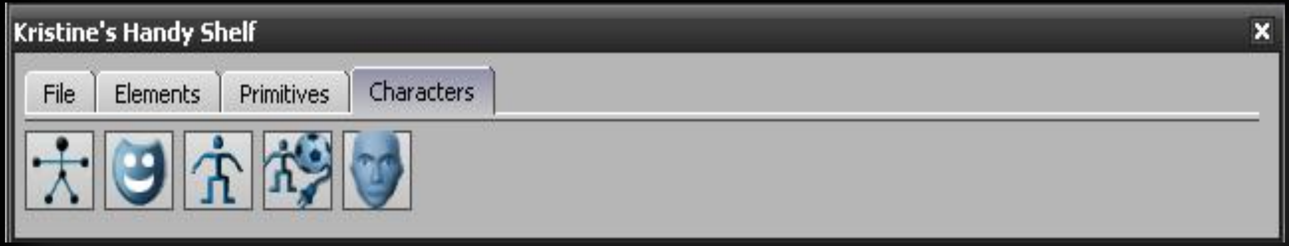
- Not supported
- Sit outside the event loop
- Will work, depending on what your would like to accomplish
- Example Took kits are:
  - Tkinter
  - wxPython
  - pyQT

Assignment


Kristine's Handy Shelf
File
Elements
Primitives
Characters


Kristine's Handy Shelf
File
Elements
Primitives
Characters


Kristine's Handy Shelf
File
Elements
Primitives
Characters


Kristine's Handy Shelf
File
Elements
Primitives
Characters

Assignment Tab Details

- Tab 1: graphical buttons for New File, Save File, Save File As, Open File
- Tab 2: graphical buttons for all our basic element creation
- Tab 3: graphical buttons FBX file primitives.
- Tab 4: a graphical buttons for character, character face, character extension, actor, and actor face

Next Agenda

- Takes and Layers
- Common Character Workflows
- Another automatic way to plot
- Miscellaneous Character Information
- Assignment