GLUONIX DESIGNER

Contents

[Root/Popup 6](#_Toc188086138)

[Nothing 7](#_Toc188086139)

[Maximize 7](#_Toc188086140)

[Restore 7](#_Toc188086141)

[Minimize 7](#_Toc188086142)

[Restart 7](#_Toc188086143)

[Close 7](#_Toc188086144)

[Hide 7](#_Toc188086145)

[Show 7](#_Toc188086146)

[Grab 8](#_Toc188086147)

[After 8](#_Toc188086148)

[Screen 8](#_Toc188086149)

[Bind 8](#_Toc188086150)

[Config 8](#_Toc188086151)

[Config\_Get 8](#_Toc188086152)

[Add\_Manu 8](#_Toc188086153)

[Add\_Sub\_Menu 8](#_Toc188086154)

[Add\_Separator 9](#_Toc188086155)

[Folder 9](#_Toc188086156)

[File 9](#_Toc188086157)

[Position 9](#_Toc188086158)

[Size 9](#_Toc188086159)

[Locate 9](#_Toc188086160)

[Locate\_Reverse 9](#_Toc188086161)

[Common 10](#_Toc188086162)

[Copy 10](#_Toc188086163)

[Delete 11](#_Toc188086164)

[Hide 11](#_Toc188086165)

[Show 11](#_Toc188086166)

[Focus 11](#_Toc188086167)

[Grab 11](#_Toc188086168)

[Bind 11](#_Toc188086169)

[Config 11](#_Toc188086170)

[Config\_Get 12](#_Toc188086171)

[Position 12](#_Toc188086172)

[Size 12](#_Toc188086173)

[Locate 12](#_Toc188086174)

[Locate\_Reverse 12](#_Toc188086175)

[Canvas/Scroll 13](#_Toc188086176)

[Clear 13](#_Toc188086177)

[Refresh 13](#_Toc188086178)

[Bind\_Item 13](#_Toc188086179)

[Hide\_Item 13](#_Toc188086180)

[Show\_Item 13](#_Toc188086181)

[Delete\_Item 13](#_Toc188086182)

[Delete\_All 14](#_Toc188086183)

[Find\_Near 14](#_Toc188086184)

[Find\_Overlap 14](#_Toc188086185)

[Line 14](#_Toc188086186)

[Polyline 14](#_Toc188086187)

[Add 14](#_Toc188086188)

[Remove 15](#_Toc188086189)

[Pie 15](#_Toc188086190)

[Arc 15](#_Toc188086191)

[Circle 15](#_Toc188086192)

[Rectangle 15](#_Toc188086193)

[Rectangle2 15](#_Toc188086194)

[Oval 16](#_Toc188086195)

[Polygon 16](#_Toc188086196)

[Add 16](#_Toc188086197)

[Remove 16](#_Toc188086198)

[Text 16](#_Toc188086199)

[Set 16](#_Toc188086200)

[Image 17](#_Toc188086201)

[Set 17](#_Toc188086202)

[Top 17](#_Toc188086203)

[Reset 17](#_Toc188086204)

[Update 17](#_Toc188086205)

[Update\_All 17](#_Toc188086206)

[Frame 18](#_Toc188086207)

[Clear 18](#_Toc188086208)

[Bar 18](#_Toc188086209)

[Set 18](#_Toc188086210)

[Get 18](#_Toc188086211)

[Button 19](#_Toc188086212)

[Set 19](#_Toc188086213)

[Check 19](#_Toc188086214)

[Set 19](#_Toc188086215)

[Get 19](#_Toc188086216)

[Compound 20](#_Toc188086217)

[Set 20](#_Toc188086218)

[Entry 20](#_Toc188086219)

[Set 20](#_Toc188086220)

[Get 20](#_Toc188086221)

[Image 21](#_Toc188086222)

[Set 21](#_Toc188086223)

[Rotate 21](#_Toc188086224)

[Label 21](#_Toc188086225)

[Set 21](#_Toc188086226)

[Roubel 22](#_Toc188086227)

[Set 22](#_Toc188086228)

[Line 22](#_Toc188086229)

[List 23](#_Toc188086230)

[Add 23](#_Toc188086231)

[Remove 23](#_Toc188086232)

[Set 23](#_Toc188086233)

[Get 23](#_Toc188086234)

[Clear 23](#_Toc188086235)

[Variable 24](#_Toc188086236)

[Create 24](#_Toc188086237)

[Get 24](#_Toc188086238)

[Radio 24](#_Toc188086239)

[Set 24](#_Toc188086240)

[Reset 24](#_Toc188086241)

[Scale 25](#_Toc188086242)

[Set 25](#_Toc188086243)

[Get 25](#_Toc188086244)

[Spinner 25](#_Toc188086245)

[Set 25](#_Toc188086246)

[Get 25](#_Toc188086247)

[Select 26](#_Toc188086248)

[Add 26](#_Toc188086249)

[Remove 26](#_Toc188086250)

[Set 26](#_Toc188086251)

[Get 26](#_Toc188086252)

[Clear 26](#_Toc188086253)

[Sort 26](#_Toc188086254)

[Separator 27](#_Toc188086255)

[Add 27](#_Toc188086256)

[Switch 27](#_Toc188086257)

[Set 27](#_Toc188086258)

[Get 27](#_Toc188086259)

[Text 28](#_Toc188086260)

[Tag 28](#_Toc188086261)

[Add 28](#_Toc188086262)

[Set 28](#_Toc188086263)

[Get 28](#_Toc188086264)

[Tree 29](#_Toc188086265)

[Add 29](#_Toc188086266)

[Edit 29](#_Toc188086267)

[Get 29](#_Toc188086268)

[Get\_All 29](#_Toc188086269)

[Remove 29](#_Toc188086270)

[Remove\_All 29](#_Toc188086271)

[Remove\_Selected 30](#_Toc188086272)

[Current 30](#_Toc188086273)

[Selected 30](#_Toc188086274)

[Child 30](#_Toc188086275)

[Parent 30](#_Toc188086276)

[Index 30](#_Toc188086277)

[Expand 30](#_Toc188086278)

[Select 31](#_Toc188086279)

[Export 31](#_Toc188086280)

[Bind / Bind\_Item 32](#_Toc188086281)

[Bind Specific 32](#_Toc188086282)

[For All 32](#_Toc188086283)

[Config 37](#_Toc188086284)

[Background 37](#_Toc188086285)

[Foreground 37](#_Toc188086286)

[Border\_Color 37](#_Toc188086287)

[Border\_Size 37](#_Toc188086288)

[Resize 37](#_Toc188086289)

[Resize\_Width 38](#_Toc188086290)

[Resize\_Height 38](#_Toc188086291)

[Move 38](#_Toc188086292)

[Move\_Top 38](#_Toc188086293)

[Move\_Left 38](#_Toc188086294)

[Top 38](#_Toc188086295)

[Left 38](#_Toc188086296)

[Width 39](#_Toc188086297)

[Height 39](#_Toc188086298)

[Font\_Size 39](#_Toc188086299)

[Font\_Weight 39](#_Toc188086300)

[Font\_Family 39](#_Toc188086301)

[Scrollbar 39](#_Toc188086302)

[Vertical 40](#_Toc188086303)

[Horizontal 40](#_Toc188086304)

[Last 40](#_Toc188086305)

[Value 40](#_Toc188086306)

[Ridge 40](#_Toc188086307)

[Disable 40](#_Toc188086308)

[Path 41](#_Toc188086309)

[URL 41](#_Toc188086310)

[Array 41](#_Toc188086311)

[Pil 41](#_Toc188086312)

[Photo 41](#_Toc188086313)

[Rotate 41](#_Toc188086314)

[Transparent 42](#_Toc188086315)

[Aspect\_Ratio 42](#_Toc188086316)

[Compound 42](#_Toc188086317)

[Align 42](#_Toc188086318)

[Secure 42](#_Toc188086319)

[Variable 43](#_Toc188086320)

[Minimum 43](#_Toc188086321)

[Maximum 43](#_Toc188086322)

[Increment 43](#_Toc188086323)

[Orient 43](#_Toc188086324)

[Height\_List 43](#_Toc188086325)

[Progress 44](#_Toc188086326)

[Zero 44](#_Toc188086327)

[Radius 44](#_Toc188086328)

[Shadow\_Size 44](#_Toc188086329)

[Shadow\_Color 44](#_Toc188086330)

[Background\_Selected 44](#_Toc188086331)

[Foreground\_Selected 45](#_Toc188086332)

[Translucent 45](#_Toc188086333)

# Root/Popup

## Nothing

Root.Nothing()

Return → False

## Maximize

Root.Maximize()

Return → Maximized Window

## Restore

Root.Restore()

Return → Restore Window Size

## Minimize

Root.Minimize()

Return → Minimize Window To Taskbar

## Restart

Root.Restart()

Return → Restart Application

## Close

Root.Close()

Return → Close Application

## Hide

Root.Hide()

Return → Hide Application

## Show

Root.Show()

Return → Show Application

## Grab

Root.Grab(Path=False)

Return → Returns and Saves screenshot of Application

## After

Root.After(Delay=1000, Function=lambda : Root.Nothing())

Return → Runs the provided function after milliseconds of delay

## Screen

Root.Screen()

Return → Returns screen size: {'Width':1920, 'Height': 1080}

## Bind

Root.Bind(On\_Click=lambda E: Root.Nothing())

Return → Runs on click function and list of other binds

## Config

Root.Config(Background=’#F5F5F5’)

Return → Sets background color and list of configs

## Config\_Get

Root.Config\_Get(‘Background’)

Return → Background color and list of configs

## Add\_Manu

Root.Add\_Manu(Main=False, Name, Command=False)

Return → Adds Menu Item. If Main is False, will Add to Top Bar, if command is given, No submenu can be added

## Add\_Sub\_Menu

Root.Add\_Sub\_Manu(Main, Name, Command=False)

Return → Adds Sub Menu Item. If command is given, No submenu can be added

## Add\_Separator

Root.Add\_Separator(Main)

Return → Adds horizontal separator in given step of menu

## Folder

Root.Folder(Initial='', Title='', Persistent=True)

Return → Request folder input from user.

## File

Root.File( Initial='', Title='', Multiple=False, Default='.txt', Type=[["Text files", "\*.txt"], ["All files", "\*.\*"]])

Return → Request file input from user

## Position

Root.Position()

Return → Current position of Application [Left, Top]

## Size

Root.Size()

Return → Current size of Application [Width, Height]

## Locate

Root.Locate(Width, Height, Left, Top)

Return → Converts the values from percentage to pixels

## Locate\_Reverse

Root.Locate\_Reverse(Width, Height, Left, Top)

Return → Converts the values from pixels to percentage

# Common

## Copy

Widget.Copy(Name=False, Main=False)

Return → Creates a copy of widget and inside widgets with a new name, For Canvas items, no arg Name, only Main

## Delete

Widget.Delete()

Return → Delete widget and all inside widgets

## Hide

Widget.Hide()

Return → Hide Widget

## Show

Widget.Show()

Return → Show Widget

## Focus

Widget.Focus()

Return → Bring Widget To Focus {Works on Entry, Button, List & Select}

## Grab

Widget.Grab(Path=False)

Return → Returns and Saves screenshot of Widget

## Bind

Widget.Bind(On\_Click=lambda E: Root.Nothing())

Return → Runs on click function and list of other binds

## Config

Widget.Config(Background=’#F5F5F5’)

Return → Sets background color and list of configs

## Config\_Get

Widget.Config\_Get(‘Background’)

Return → Background color and list of configs

## Position

Widget.Position(Left=False, Top=False)

Return → Current position of Application [Left, Top]

## Size

Widget.Size(Width=False, Height=False)

Return → Current size of Application [Width, Height]

## Locate

Widget.Locate(Width, Height, Left, Top)

Return → Converts the values from percentage to pixels

## Locate\_Reverse

Widget.Locate\_Reverse(Width, Height, Left, Top)

Return → Converts the values from pixels to percentage

# Canvas/Scroll

Config List: 'Background', 'Border\_Color', 'Border\_Size', 'Resize', 'Resize\_Width', 'Resize\_Height', 'Move', 'Move\_Left', 'Move\_Top', 'Left', 'Top', 'Width', 'Height', 'Radius'

Canvas Config Addon: 'Shadow\_Size', ' Shadow\_Color'

Scroll Config Addon: 'Scrollbar', 'Vertical', 'Horizontal', 'Last'

## Clear

Canvas.Clear()

Return → Clears all widgets inside canvas or scroll

## Refresh

Canvas.Refresh()

Return → Update all idle tasks

## Bind\_Item

Canvas.Bind\_Item(Item, On\_Click=lambda E: Root.Nothing())

Return → Runs on click function and list of other binds on an item in canvas

## Hide\_Item

Canvas.Hide\_Item(Item)

Return → Hides an Item

## Show\_Item

Canvas.Show\_Item(Item)

Return → Show an Item

## Delete\_Item

Canvas.Delete\_Item(Item)

Return → Delete an Item

## Delete\_All

Canvas.Delete\_All()

Return → Delete all Items

## Find\_Near

Canvas.Find\_Near(X, Y)

Return → List of all items close to this pixel

## Find\_Overlap

Canvas.Find\_Overlap(X1, Y1, X2, Y2)

Return → List of all the items overlapping provided rectangle

## Line

Config List: 'Outline', 'Width', 'Height', 'Left', 'Top', 'Thickness', 'Resize'

Canvas.Line()

Return → Draw a line with provided parameters and return item

## Polyline

Config List: 'Outline', 'Thickness', 'Resize'

Canvas.Polyline()

Return → Draw a line with provided parameters and return item

### Add

Widget.Add(X, Y)

Return → Adds a new point to polyline

### Remove

Widget.Remove(Index)

Return → Removes a point from polyline

## Pie

Config List: 'Outline', 'Fill', 'Left', 'Top', 'Radius', 'Thickness', 'Resize', 'Start', 'Extent', 'Translucent'

Canvas.Pie()

Return → Draw a pie chart with provided parameters and return item

## Arc

Config List: 'Outline', 'Left', 'Top', 'Radius', 'Thickness', 'Resize', 'Start', 'Extent', 'Translucent'

Canvas.Arc()

Return → Draw an arc with provided parameters and return item

## Circle

Config List: 'Outline', 'Fill', 'Left', 'Top', 'Radius', 'Thickness', 'Resize', 'Translucent'

Canvas.Circle()

Return → Draw a circle with provided parameters and return item

## Rectangle

Config List: 'Outline', 'Fill', 'Width', 'Height', 'Left', 'Top', 'Thickness', 'Resize', 'Translucent'

Canvas.Rectangle()

Return → Draw a rectangle with provided parameters and return item

## Rectangle2

Config List: 'Outline', 'Fill', 'Width', 'Height', 'Left', 'Top', 'Angle', 'Thickness', 'Resize', 'Translucent'

Canvas.Rectangle2()

Return → Draw a rotated rectangle with provided parameters and return item

## Oval

Config List: 'Outline', 'Fill', 'Width', 'Height', 'Left', 'Top', 'Thickness', 'Resize', 'Translucent'

Canvas.Oval()

Return → Draw an oval with provided parameters and return item

## Polygon

Config List: 'Outline', 'Fill', 'Thickness', 'Resize', 'Translucent'

Canvas. Polygon()

Return → List of points to create polygon, Draw a polygon with provided parameters and return item

### Add

Widget.Add(X, Y)

Return → Adds a new point to polygon

### Remove

Widget.Remove(Index)

Return → Removes a point from polygon

## Text

Config List: 'Width', 'Height', 'Left', 'Top', 'Color', 'Size', 'Value', 'Weight', 'Font', 'Anchor', 'Justify', 'Resize'

Canvas.Text()

Return → Draws a text and return the item

### Set

Widget.Set(Vlaue)

Return → Sets a new text to item

## Image

Config List: 'Width', 'Height', 'Left', 'Top', 'Anchor', 'Url', 'Array', 'Pil', 'Photo', 'Resize'

Canvas.Image()

Return → Draws an image and return item

### Set

Widget.Set(Path)

Return → Sets a new image to item

## Top

Scroll.Top()

Return → Moves scrollbar to top

## Reset

Scroll.Reset()

Return → Resets the size of scroll frame and moves or hides scrollbars

## Update

Scroll.Update(Widget)

Return → Updates the size of Scroll region based on provided widget. It is recommended to provide the widget that id most to right and bottom of scroll. You can add an hidden widget so the size given properly to scroll the frame

## Update\_All

Scroll.Update\_All()

Return → Will update the size of scroll frame based on all the widgets in scroll.

# Frame

Config List: 'Background', 'Border\_Color', 'Border\_Size', 'Resize', 'Resize\_Width', 'Resize\_Height', 'Move', 'Move\_Left', 'Move\_Top', 'Left', 'Top', 'Width', 'Height'

## Clear

Frame.Clear()

Return → Clears all widgets inside frame

# Bar

Config List: 'Background', 'Foreground', 'Border\_Color', 'Border\_Size', 'Resize', 'Resize\_Width', 'Resize\_Height', 'Move', 'Move\_Left', 'Move\_Top', 'Left', 'Top', 'Width', 'Height', 'Progress', 'Zero'

## Set

Bar.Set(Value=10)

Return → Sets the bar to specific position

## Get

Bar.Set()

Return → The current position of bar

# Button

Config List: 'Background', 'Foreground', 'Border\_Color', 'Border\_Size', 'Resize\_Font', 'Resize', 'Resize\_Width', 'Resize\_Height', 'Move', 'Move\_Left', 'Move\_Top', 'Left', 'Top', 'Width', 'Height', 'Font\_Size', 'Font\_Weight', 'Font\_Family', 'Value', 'Ridge', 'Disable'

## Set

Button.Set(Value=’Click Me’)

Return → Sets the name of the button

# Check

Config List: 'Background', 'Foreground', 'Border\_Color', 'Border\_Size', 'Resize', 'Resize\_Width', 'Resize\_Height', 'Move', 'Move\_Left', 'Move\_Top', 'Left', 'Top', 'Width', 'Height'

## Set

Check.Set(Check=True)

Return → Sets the current state of check box

## Get

Check.Set()

Return → The current state of check button

# Compound

Config List: 'Background', 'Foreground', 'Border\_Color', 'Border\_Size', 'Resize\_Font', 'Resize', 'Resize\_Width', 'Resize\_Height', 'Move', 'Move\_Left', 'Move\_Top', 'Left', 'Top', 'Width', 'Height', 'Font\_Size', 'Font\_Weight', 'Font\_Family', 'Value', 'Path', 'Url', 'Array', 'Pil', 'Rotate', 'Transparent', 'Compound', 'Aspect\_Ratio'

## Set

Compound.Set(Path, Value)

Return → Sets the image path or name of compound button

# Entry

Config List: 'Background', 'Foreground', 'Border\_Color', 'Border\_Size', 'Resize\_Font', 'Resize', 'Resize\_Width', 'Resize\_Height', 'Move', 'Move\_Left', 'Move\_Top', 'Left', 'Top', 'Width', 'Height', 'Font\_Size', 'Font\_Weight', 'Font\_Family', 'Align', 'Disable', 'Secure'

## Set

Entry.Set(Value)

Return → Sets the value to entry

## Get

Entry.Get()

Return → The value to entry

# Image

Config List: 'Background', 'Border\_Color', 'Border\_Size', 'Resize', 'Resize\_Width', 'Resize\_Height', 'Move', 'Move\_Left', 'Move\_Top', 'Left', 'Top', 'Width', 'Height', 'Path', 'Url', 'Array', 'Pil', 'Rotate', 'Transparent', 'Aspect\_Ratio'

## Set

Image.Set(Path)

Return → Sets the image path and reloads image. If Url is True, Path is Url & if Array is True, Path is cv2 Frame Array (RGB)

## Rotate

Image.Rotate(Value=10)

Return → Rotates image in given angle degree values

# Label

Config List: 'Background', 'Foreground', 'Resize\_Font', 'Resize', 'Resize\_Width', 'Resize\_Height', 'Move', 'Move\_Left', 'Move\_Top', 'Left', 'Top', 'Width', 'Height', 'Font\_Size', 'Font\_Weight', 'Font\_Family', 'Align', 'Value'

## Set

Label.Set(Value)

Return → Sets text value of label

# Roubel

Config List: 'Background', 'Foreground', 'Resize\_Font', 'Resize', 'Resize\_Width', 'Resize\_Height', 'Move', 'Move\_Left', 'Move\_Top', 'Left', 'Top', 'Width', 'Height', 'Font\_Size', 'Font\_Weight', 'Font\_Family', 'Value', 'Radius', 'Shadow\_Size', 'Shadow\_Color'

## Set

Label.Set(Value)

Return → Sets text value of label

# Line

Config List: 'Background', 'Border\_Color', 'Border\_Size', 'Resize', 'Resize\_Width', 'Resize\_Height', 'Move', 'Move\_Left', 'Move\_Top', 'Left', 'Top', 'Width', 'Height'

# List

Config List: 'Background', 'Foreground', 'Border\_Color', 'Border\_Size', 'Resize\_Font', 'Resize', 'Resize\_Width', 'Resize\_Height', 'Move', 'Move\_Left', 'Move\_Top', 'Left', 'Top', 'Width', 'Height', 'Font\_Size', 'Font\_Weight', 'Font\_Family', 'Disable'

## Add

List.Add(Value)

Return → Adds the value to list

## Remove

List.Remove(Value)

Return → Removes the value from list

## Set

List.Set(Value)

Return → Sets the value to list

## Get

List.Get()

Return → The current value to list

## Clear

List.Clear()

Return → Delete all the values in list

# Variable

## Create

Variable = Gluonix.Variable()

Return → Creates a variable object for Radio Button

## Get

Variable.Get()

Return → Returns current values of variable set by radio buttons

# Radio

Config List: 'Background', 'Foreground', 'Border\_Color', 'Border\_Size', 'Resize', 'Resize\_Width', 'Resize\_Height', 'Move', 'Move\_Left', 'Move\_Top', 'Left', 'Top', 'Width', 'Height', 'Value', 'Variable'

## Set

Radio.Set()

Return → Activates current radio and unset all the other radios for same variable

## Reset

Radio.Reset()

Return → Unset radio

# Scale

Config List: 'Background', 'Border\_Color', 'Border\_Size', 'Resize', 'Resize\_Width', 'Resize\_Height', 'Move', 'Move\_Left', 'Move\_Top', 'Left', 'Top', 'Width', 'Height', 'Increment', 'Minimum', 'Maximum', 'Orient', 'Disable'

## Set

Scale.Set(Value)

Return → Set scale to given value

## Get

Scale.Get()

Return → Current values of scale

# Spinner

Config List: 'Background', 'Foreground', 'Border\_Color', 'Border\_Size', 'Resize\_Font', 'Resize', 'Resize\_Width', 'Resize\_Height', 'Move', 'Move\_Left', 'Move\_Top', 'Left', 'Top', 'Width', 'Height', 'Font\_Size', 'Font\_Weight', 'Font\_Family', 'Align', 'Increment', 'Minimum', 'Maximum', 'Disable'

## Set

Spinner.Set(Value)

Return → Set spinner to given value

## Get

Spinner.Get()

Return → Current values of spinner

# Select

Config List: 'Background', 'Foreground', 'Border\_Color', 'Border\_Size', 'Resize\_Font', 'Resize', 'Resize\_Width', 'Resize\_Height', 'Move', 'Move\_Left', 'Move\_Top', 'Left', 'Top', 'Width', 'Height', 'Font\_Size', 'Font\_Weight', 'Font\_Family', 'Disable'

## Add

Select.Add(Value)

Return → Adds the value to select

## Remove

Select.Remove(Value)

Return → Removes the value from select

## Set

Select.Set(Value)

Return → Sets the value to select

## Get

Select.Get()

Return → The current value to select

## Clear

Select.Clear()

Return → Delete all the values in select

## Sort

Select.Sort()

Return → Sorts list values A - Z

# Separator

Config List: 'Background', 'Border\_Color', 'Border\_Size', 'Resize', 'Resize\_Width', 'Resize\_Height', 'Move', 'Move\_Left', 'Move\_Top', 'Left', 'Top', 'Width', 'Height'

## Add

Separator.Add(Frame)

Return → Adds frame to control the movement and size, Need two frames and open provided, frames can’t be changed.

# Switch

Config List: 'Background', 'Border\_Color', 'Border\_Size', 'Resize', 'Resize\_Width', 'Resize\_Height', 'Move', 'Move\_Left', 'Move\_Top', 'Left', 'Top', 'Width', 'Height'

## Set

Check.Set(Check=True)

Return → Sets the current state of check box

## Get

Check.Get()

Return → The current state of check button

# Text

Config List: 'Background', 'Foreground', 'Border\_Color', 'Border\_Size', 'Resize', 'Resize\_Width', 'Resize\_Height', 'Move', 'Move\_Left', 'Move\_Top', 'Left', 'Top', 'Width', 'Height', 'Font\_Size', 'Font\_Weight', 'Disable'

## Tag

Text.Tag(Name, Font\_Size=12, Font\_Weight='normal', Font='Times New Roman', Foreground="#000000", Background="#FFFFFF")

Return → Creates a custom tag to use in text

## Add

Text.Add(Value, Tag=’Default’)

Return → Sets the current state of check box

## Set

Text.Set(Value)

Return → Sets values to text area

## Get

Text.Get()

Return → The values of text area

# Tree

Config List: 'Background', 'Foreground', 'Border\_Color', 'Border\_Size', 'Resize', 'Resize\_Width', 'Resize\_Height', 'Move', 'Move\_Left', 'Move\_Top', 'Left', 'Top', 'Width', 'Height', 'Font\_Size', 'Font\_Weight', 'Background\_Selected', 'Foreground\_Selected '

## Add

Tree.Add(Name, Parent=’’, Index=’end’, Value=[], ID=False)

Return → Create a tree item with specific value

## Edit

Tree.Edit(Name=False, Value=False, Tag=False, ID=False)

Return → Edits parameters of on focus tree item or by providing the id of item

## Get

Tree.Get(ID=False)

Return → The values of on focus tree item or by providing the id of item

## Get\_All

Tree.Get\_All(ID=False)

Return → The everything of on focus tree item or by providing the id of item

## Remove

Tree.Remove(ID)

Return → Deletes given item

## Remove\_All

Tree.Remove\_All()

Return → Deletes all items in the tree

## Remove\_Selected

Tree.Remove\_Selected()

Return → Deletes all selected items in the tree

## Current

Tree.Current()

Return → The item in focus

## Selected

Tree.Selected()

Return → List of selected items

## Child

Tree.Child(ID)

Return → List of all children of item

## Parent

Tree.Parent(ID)

Return → Parent of given item

## Index

Tree.Index(ID)

Return → Index of an item

## Expand

Tree.Expand(ID)

Return → Expand the current tree item

## Select

Tree.Select(ID)

Return → Makes specific item in focus

## Export

Tree.Export(Path)

Return → Exports tree structure to text file ‘.txt’

# Bind / Bind\_Item

## Bind Specific

Cursor\_Hand

Cursor\_Loading

Cursor\_Resize\_Vertical

Cursor\_Resize\_Horizontal

Cursor\_Arrow

Cursor

On\_Show (**No Event**)

On\_Hide (**No Event**)

On\_Close (Root & Popup) (**No Event**)

On\_Resize (Root, Popup, Frame, Canvas & Scroll) (**No Event**)

On\_Change (List, Select, Check, Switch & Radio) (**No Event**)

## For All

On\_Configure

On\_Destroy

On\_Expose

On\_Visibility

On\_Motion

On\_Click

On\_Release

On\_Double\_Click

On\_Triple\_Click

On\_Middle\_Click

On\_Middle\_Release

On\_Middle\_Double\_Click

On\_Middle\_Triple\_Click

On\_Right\_Click

On\_Right\_Release

On\_Right\_Double\_Click

On\_Right\_Triple\_Click

On\_Drag

On\_Middle\_Drag

On\_Right\_Drag

On\_Mouse\_Wheel

On\_Hover\_In

On\_Hover\_Out

On\_Key

On\_Key\_Release

On\_Focus\_Out

On\_Map

On\_Unmap

On\_Copy

On\_Cut

On\_Paste

On\_Undo

On\_Redo

On\_Control\_Click

On\_Control\_Release

On\_Control\_Double\_Click

On\_Control\_Triple\_Click

On\_Control\_Middle\_Click

On\_Control\_Middle\_Release

On\_Control\_Middle\_Double\_Click

On\_Control\_Middle\_Triple\_Click

On\_Control\_Right\_Click

On\_Control\_Right\_Release

On\_Control\_Right\_Double\_Click

On\_Control\_Right\_Triple\_Click

On\_Control\_Drag

On\_Control\_Middle\_Drag

On\_Control\_Right\_Drag

On\_Control\_Mouse\_Wheel

On\_Control\_Hover\_In

On\_Control\_Hover\_Out

On\_Alt\_Click

On\_Alt\_Release

On\_Alt\_Double\_Click

On\_Alt\_Triple\_Click

On\_Alt\_Middle\_Click

On\_Alt\_Middle\_Release

On\_Alt\_Middle\_Double\_Click

On\_Alt\_Middle\_Triple\_Click

On\_Alt\_Right\_Click

On\_Alt\_Right\_Release

On\_Alt\_Right\_Double\_Click

On\_Alt\_Right\_Triple\_Click

On\_Alt\_Drag

On\_Alt\_Middle\_Drag

On\_Alt\_Right\_Drag

On\_Alt\_Mouse\_Wheel

On\_Alt\_Hover\_In

On\_Alt\_Hover\_Out

On\_Shift\_Click

On\_Shift\_Release

On\_Shift\_Double\_Click

On\_Shift\_Triple\_Click

On\_Shift\_Middle\_Click

On\_Shift\_Middle\_Release

On\_Shift\_Middle\_Double\_Click

On\_Shift\_Middle\_Triple\_Click

On\_Shift\_Right\_Click

On\_Shift\_Right\_Release

On\_Shift\_Right\_Double\_Click

On\_Shift\_Right\_Triple\_Click

On\_Shift\_Drag

On\_Shift\_Middle\_Drag

On\_Shift\_Right\_Drag

On\_Shift\_Mouse\_Wheel

On\_Shift\_Hover\_In

On\_Shift\_Hover\_Out

# Config

Multiple values in config can be set and get at same time

Widget.Config(Background=’#F5F5F5’, Foreground=’red’, …..)

Widget.Config\_Get(‘Background’, ‘Foreground’, …..)

Will return a dictionary of requested configurations

## Background

Sets the background of widget. Can be simple HTML values like ‘red’, ‘green’, etc. or any Hex color value ‘#F5F5F5’. False, it will take the background of the Main Frame of the Widget

Widget.Config(Background=’#F5F5F5’)

## Foreground

Sets the foreground of widget. Can be simple HTML values like ‘red’, ‘green’, etc. or any Hex color value ‘#F5F5F5’

Widget.Config(Foreground=’#F5F5F5’)

## Border\_Color

Sets the border color of widget. Can be simple HTML values like ‘red’, ‘green’, etc. or any Hex color value ‘#F5F5F5’. This feature does not work on LITE objects.

Widget.Config(Border\_Color=’#000000’)

## Border\_Size

Sets the border size of widget. In pixel values. This feature does not work on LITE objects.

Widget.Config(Border\_Size=2)

## Resize

Defines if the widget can be resized when main root window resizes.

Widget.Config(Resize=True)

## Resize\_Width

Defines if the widget can be resized only width wise main root window resizes.

Widget.Config(Resize\_Width=True)

## Resize\_Height

Defines if the widget can be resized only height wise main root window resizes.

Widget.Config(Resize\_Height=True)

## Move

Defines if the widget can be moved when main root window resizes.

Widget.Config(Move=True)

## Move\_Top

Defines if the widget can be moved only up and down main root window resizes.

Widget.Config(Move\_Top=True)

## Move\_Left

Defines if the widget can be resized only left and right main root window resizes.

Widget.Config(Move\_Left=True)

## Top

Changes the top position of widget. Only accepted in pixels.

Widget.Config(Top=100)

## Left

Changes the left position of widget. Only accepted in pixels.

Widget.Config(Left=150)

## Width

Changes the width of widget. Only accepted in pixels.

Widget.Config(Width=100)

\* If the inside widget is provided in percentage, changed the main widgets will affect the inside widget after the application restarts

## Height

Changes the height of widget. Only accepted in pixels.

Widget.Config(Height=100)

\* If the inside widget is provided in percentage, changed the main widgets will affect the inside widget after the application restarts

## Font\_Size

Font size for the widgets which support text. Only accepted in pixels.

Widget.Config(Font\_Size=20)

## Font\_Weight

Font weight for the widgets which support text. ‘normal’, ‘bold’

Widget.Config(Font\_Weight=’normal’)

## Font\_Family

Font family for the widgets which support text. Only tKinter specific.

'Times New Roman', 'Helvetica', etc.

Widget.Config(Font\_Family= 'Times New Roman')

## Scrollbar

Width of scrollbar. Only accepted in pixels.

Widget.Config(Scrollbar=20)

\* Exclusive to scroll frame

## Vertical

If vertical scroll bar should be displayed at initial state of frame.

Widget.Config(Vertical=True)

\* Exclusive to scroll frame

## Horizontal

If horizontal scroll bar should be displayed at initial state of frame.

Widget.Config(Horizontal=True)

\* Exclusive to scroll frame

## Last

Define the last widget of scrolls frame, the widget which is most right and bottom of frame.

Widget.Config(Last=Widget)

\* Exclusive to scroll frame

## Value

Sets text value for widgets that support text.

Widget.Config(Value=’Click Here’)

## Ridge

If the button should have ridged border

Widget.Config(Ridge=True)

\* Exclusive to button

## Disable

Disbales the function of widget

Widget.Config(Disbale=True)

\* Exclusive to button, list, entry, select, spinner, scale, text

## Path

Gives the image path for widget

Widget.Config(Path=’./Image.png’)

\* Exclusive to image, compound, canvas image

## URL

Defines if image path given is a http url

Widget.Config(Url=True)

\* Exclusive to image, compound, canvas image

## Array

Defines if image path is cv2 array

Widget.Config(Array=True)

\* Exclusive to image, compound, canvas image

## Pil

Defines if image path is pillow image

Widget.Config(Pil=True)

\* Exclusive to image, compound, canvas image

## Photo

Defines if image path is tk photo image

Widget.Config(Photo=True)

\* Exclusive to canvas image

## Rotate

Rotates the image to specific degree of angle

Widget.Config(Rotate=90)

\* Exclusive to image, compound, canvas image

## Transparent

Defines if the provided image support to be transparent in nature.

Widget.Config(Transparent=True)

\* Exclusive to image, compound, canvas image

## Aspect\_Ratio

If the given image should keep aspect ration when fitting to current widget.

Widget.Config(Aspect\_Ratio=True)

\* Exclusive to image, compound, canvas image

## Compound

Location of text on the image. ‘left’, ‘right’, ‘top’, ‘bottom’, ‘center’

Widget.Config(Compound=’center’)

\* Exclusive to compound

## Align

Location of text in widget

Widget.Config(Align=’center’)

\* Entry: ‘left’, ‘right’, ‘center’

\* Spinner: ‘left’, ‘right’, ‘center’

\* Label: ‘n’, ‘e’, ‘w’, ‘s’, ‘ne’, ‘nw’, ‘se’, ‘sw’, ‘center’

## Secure

Hides the entry text with \*

Widget.Config(Secure=True)

\* Exclusive to Entry

## Variable

Provides the variable to multiple radio buttons to store the radio value.

Widget.Config(Variable=Temp\_Variable)

\* Exclusive to Radio

## Minimum

Minimum value for widget

Widget.Config(Minimum=0)

\* Exclusive to Scale, Spinner

## Maximum

Maximum value for widget

Widget.Config(Maximum=100)

\* Exclusive to Scale, Spinner

## Increment

Increment step for widget

Widget.Config(Increment=1)

\* Exclusive to Scale, Spinner

## Orient

Orientation of the widget

Widget.Config(Orient=’Horizontal’)

\* Exclusive to Scale, Seperator

## Height\_List

Height of the list when on click widget

Widget.Config(Height\_List=500)

\* Exclusive to Select

## Progress

Initial value for the widget

Widget.Config(Progress=10)

\* Exclusive to Bar

## Zero

Position of widget. ‘Left’, ‘Right’, ‘Top’, ‘Bottom’

Widget.Config(Zero=’Left’)

\* Exclusive to Bar

## Radius

Radius for rounded corners.

Widget.Config(Radius=20)

\* Exclusive to Canvas, Roubel

## Shadow\_Size

Applies shadow size to all sides.

Widget.Config(Shadow\_Size=10)

\* Exclusive to Canvas, Roubel

## Shadow\_Color

Applies shadow color to all sides.

Widget.Config(Shadow\_Color=’#FF0000’)

\* Exclusive to Canvas, Roubel

## Background\_Selected

Background color of current selected item.

Widget.Config(Background\_Selected=’#FFFFFF’)

\* Exclusive to Tree

## Foreground\_Selected

Foreground color of current selected item.

Widget.Config(Foreground\_Selected=’#000000’)

\* Exclusive to Tree

## Translucent

Makes Fill Color Translucent.

Widget.Config(Translucent=True)

\* Exclusive to Canvas Item Rectangle, Rectangle2, Circle, Oval, Arc, Pie, & Polygon