**Libraries**

**Error-Prompt.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Error-Prompt |  | This VI shows a common error prompt dialog. |  | N |
| Display |  | Display method for an error prompt |  | N |

**Input (DBL).lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Input (DBL) Popup |  | The input (DBL) popup will display information to the user and wait for the operator to enter a value and click the button.  Note:</b> The function <b>Close Pop-Up (DBL).vi</b> can be used to send a value when closing this dialog. Otherwise, use Close Pop-Up.vi to close it early.  Inputs: -Graphic [typedef combobox] - Image (PNG) for Graphic for Popup -Message [String] - String Variable to populate Display -Theme [lvclass] - the theme for the style of the popup -Default Input [DBL] - default value for the prompt  Outputs -Input Value [DBL] - the user's entered data  <b> |  | N |
| Display |  | This vi will display the input pop up waiting for a type Double input from the user |  | N |

**Input (I32).lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Input (I32) Popup |  | The input (I32) popup will display information to the user and wait for the operator to enter a value and click the button.  Note:</b> The function <b>Close Pop-Up (I32).vi</b> can be used to send a value when closing this dialog. Otherwise, use Close Pop-Up.vi to close it early.  Inputs: -Graphic [typedef combobox] - Image (PNG) for Graphic for Popup -Message [String] - String Variable to populate Display -Theme [lvclass] - the theme for the style of the popup -Default Input [I32] - default value for the prompt  Outputs -Input Value [I32] - the user's entered data  <b> |  | N |
| Display |  | This vi will display the input pop up waiting for a type I32 input from the user |  | N |

**Input (String).lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Input (String) Popup |  | The input (String) popup will display information to the user and wait for the operator to enter a value and click the button.  Note:</b> The function <b>Close Pop-Up (String).vi</b> can be used to send a value when closing this dialog. Otherwise, use Close Pop-Up.vi to close it early.  Inputs: -Graphic [typedef combobox] - Image (PNG) for Graphic for Popup -Message [String] - String Variable to populate Display -Theme [lvclass] - the theme for the style of the popup -Default Input [string] - default value for the prompt  Outputs -Input Value [String] - the user's entered data  <b> |  | N |
| Display |  | This vi will display the input pop up waiting for a type String input from the user |  | N |

**One Button.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display One Button Popup |  | The one button popup will display information to the user and wait for the operator to click the button.  Note:</b> The function <b>Close Pop-Up.vi</b> can be used to close this dialog early.  Inputs: -Graphic [typedef combobox] - Image (PNG) for Graphic for Popup -Message [String] - String Variable to populate Display -Theme [lvclass] - the theme for the style of the popup  <b> |  | N |
| Display |  | This vi will display the one button popup with the supplied graphic and text label (contained in the object). It will wait for a button press. |  | N |

**Timeout One Button.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Timeout One Button Popup |  | The one button popup will display information to the user and wait for the operator to click the button or until the timeout expires.  Note:</b> The function <b>Close Pop-Up.vi</b> can be used to close this dialog early.  Inputs: Inputs: -Graphic [typedef combobox] - Image (PNG) for Graphic for Popup -Message [String] - String Variable to populate Display -Theme [lvclass] - the theme for the style of the popup -Timeout (s) [DBL] - the amount of seconds for the timeout -Button Text [String] - the boolean text of the button  <b> |  | N |
| Display |  | This vi displays the one button timeout popup. |  | N |

**Timeout Two Button.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Timeout Two Button Popup |  | The two button popup will display information to the user and wait for the operator to click the button or until time expires.  Note:</b> if the user does not make a selection before the timeout, the choice will output False  <b>Note:</b> The function <b>Close Pop-Up (Two Button Choice).vi</b> can be used to close this dialog early and send a value for the choice. Otherwise, use Close Pop-Up.vi to close this dialog early.  Inputs: -Graphic [typedef combobox] - Image (PNG) for Graphic for Popup -Message [String] - String Variable to populate Display -Theme [lvclass] - the theme for the style of the popup -Y-Text [string] - the string for the YES button -N-Text [string] - the string for the NO button -Timeout (s) [DBL] - the timeout duration  Outputs -Choice? [Boolean] - the user's button selection (YText = True, NText = False)  <b> |  | N |
| Display |  | This vi displays the two button timeout popup. |  | N |

**Timeout.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Timeout Popup |  | The timeout popup will display information to the user and wait the specified timeout depending on the number of seconds passed into this VI.  Note:</b> The function <b>Close Pop-Up.vi</b> can be used to close this dialog early.  Inputs: -Graphic [typedef combobox] - Image (PNG) for Graphic for Popup -Message [String] - String Variable to populate Display -Theme [lvclass] - the theme for the style of the popup -Timeout (s) [DBL] - the amount of seconds for the timeout  <b> |  | N |
| Display |  | This vi displays the timeout Popup. |  | N |

**Two Button.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Two Button Popup |  | The two button popup will display information to the user and wait for the operator to choose a button.  Note:</b> The function <b>Close Pop-Up (Two Button Choice).vi</b> can be used to close this dialog early and send a value for the choice of Boolean selection. Otherwise, to just close the dialog early, use Close Pop-Up.vi.  Inputs: -Graphic [typedef combobox] - Image (PNG) for Graphic for Popup -Message [String] - String Variable to populate Display -Theme [lvclass] - the theme for the style of the popup -YText [string] - the string for the YES button -NText [string] - the string for the NO button  Outputs -Choice [boolean] - the option that was chosen by the user  <b> |  | N |
| Display |  | This vi displays the two button popup. |  | N |

**Gif - Input (DBL).lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Input (DBL) Popup (Gif) |  | The input (DBL) popup will display information to the user and wait for the operator to enter a value and click the button.  Note:</b> This popup type allows the user to define a custom image for the display. The developer should test all settings prior to deployment to make sure that all images and text fit on the screen correctly.  <b>Note:</b> The function <b>Close Pop-Up (DBL).vi</b> can be used to send a value when closing this dialog. Otherwise, use Close Pop-Up.vi to close it early.  Inputs: -Graphic [path] - path to the image file to display -Message [String] - String Variable to populate Display with -Theme [lvclass] - the theme class for the pop up -Default Input [DBL] - default value for the prompt  Outputs -InputValue [DBL] - the user's input data  <b> |  | N |
| Display |  | This vi will display the input pop up waiting for a type Double input from the user  The graphic will be an input from the developer. |  | N |

**Gif - Input (I32).lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Input (I32) Popup (Gif) |  | The input (I32) popup will display information to the user and wait for the operator to enter a value and click the button.  Note:</b> This popup type allows the user to define a custom image for the display. The developer should test all settings prior to deployment to make sure that all images and text fit on the screen correctly.  <b>Note:</b> The function <b>Close Pop-Up (I32).vi</b> can be used to send a value when closing this dialog. Otherwise, use Close Pop-Up.vi to close it early.  Inputs: -Graphic [path] - path to the image file to display -Message [String] - String Variable to populate Display with -Theme [lvclass] - the theme for the style of the popup -Message Info [typedef enum] -Message Height [I32] - the height for the message text box (-1 = match height of image) -Message Width [I32] - the width for the message text box (-1 = match height of image) -Message Location [Ring] - the location of the image relative to the message text box (left, right,bottom, top) -Default Input [I32] - default value for the prompt  Outputs -InputValue [I32] - the user's input data  <b> |  | N |
| Display |  | This vi will display the input pop up waiting for a type I32 input from the user and will show a customized image. |  | N |

**Gif - Input (String).lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Input (String) Popup (Gif) |  | The input (String) popup will display information to the user and wait for the operator to enter a value and click the button.  Note:</b> This popup type allows the user to define a custom image for the display. The developer should test all settings prior to deployment to make sure that all images and text fit on the screen correctly.  <b>Note:</b> The function <b>Close Pop-Up (String).vi</b> can be used to send a value when closing this dialog. Otherwise, use Close Pop-Up.vi to close it early.  Inputs: -Graphic [path] - path to the image file to display -Message [String] - String Variable to populate Display with -Theme [lvclass] - the theme for the style of the popup -Message Info [typedef enum] -Message Height [I32] - the height for the message text box (-1 = match height of image) -Message Width [I32] - the width for the message text box (-1 = match height of image) -Message Location [Ring] - the location of the image relative to the message text box (left, right,bottom, top) -Default Input [String] - default value for the prompt  Outputs -Input Value [String] - the user's input data  <b> |  | N |
| Display |  | This vi will display the input pop up waiting for a type string input from the user while displaying a custom graphic to the user. |  | N |

**Gif - One Button.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display One Button Popup (Gif) |  | The custom one button popup will display information to the user and wait for the operator to acknowledge the message.  Note:</b> This popup type allows the user to define a custom image for the display. The developer should test all settings prior to deployment to make sure that all images and text fit on the screen correctly.  <b>Note:</b> The function <b>Close Pop-Up.vi</b> can be used to close this dialog early.  Inputs: -Graphic [path] - path to the image file to display -Message [String] - String Variable to populate Display with -Theme [lvclass] - the theme for the style of the popup -Message Info [typedef enum] -Message Height [I32] - the height for the message text box (-1 = match height of image) -Message Width [I32] - the width for the message text box (-1 = match height of image) -Message Location [Ring] - the location of the image relative to the message text box (left, right,bottom, top)  <b> |  | N |
| Display |  | This is the override method required for each child of the pop up class. It is the pop up that will display to the user.  To extend this function, create a child class of the pop up class. Then create an override method for this VI. Display.vi</b> is a protected function, therefore it cannot be called in a main application. This VI must be called by a caller function. See one of the standard pop-ups for a setup example.  <b> |  | N |

**Gif - Timeout One Button.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Timeout One Button Popup (Gif) |  | The custom timeout one button popup will display information to the user and wait for the operator to click the button or until the timeout expires.  Note:</b> This popup type allows the user to define a custom image for the display. The developer should test all settings prior to deployment to make sure that all images and text fit on the screen correctly.  <b>Note:</b> The function <b>Close Pop-Up.vi</b> can be used to close this dialog early.  Inputs: -Graphic [path] - path to the image file to display -Message [String] - String Variable to populate Display with -Theme [lvclass] - the theme for the style of the popup -Message Info [typedef enum] -Message Height [I32] - the height for the message text box (-1 = match height of image) -Message Width [I32] - the width for the message text box (-1 = match height of image) -Message Location [Ring] - the location of the image relative to the message text box (left, right,bottom, top) -Button Text [string] - the text for the button -Timeout (s) [DBL] - timeout duration  <b> |  | N |
| Display |  | This vi displays the custom image one button timeout popup. |  | N |

**Gif - Timeout Two Button.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Timeout Two Button Popup (Gif) |  | The custom two button popup will display information to the user and wait for the operator to click a button or until time expires.  Note:</b> This popup type allows the user to define a custom image for the display. The developer should test all settings prior to deployment to make sure that all images and text fit on the screen correctly.  <b>Note:</b> The function <b>Close Pop-Up (Two Button Choice).vi</b> can be used to send a value when closing this dialog. Otherwise, use Close Pop-Up.vi to close early.  Inputs: -Graphic [path] - path to the image file to display -Message [String] - String Variable to populate Display with -Theme [lvclass] - the theme for the style of the popup -Message Info [typedef enum] -Message Height [I32] - the height for the message text box (-1 = match height of image) -Message Width [I32] - the width for the message text box (-1 = match height of image) -Message Location [Ring] - the location of the image relative to the message text box (left, right,bottom, top) -YText [string] - the string for the YES button -NText [string] - the string for the NO button -Timeout (s) [DBL] - the timeout duration  Outputs -Choice? [Boolean] - the user select (YText = True, NText = False)  <b> |  | N |
| Display |  | This vi displays the custom image two button timeout popup. |  | N |

**Gif - Timeout.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Timeout Popup (Gif) |  | The timeout popup will display information to the user and wait the specified timeout depending on the number of seconds passed into this VI.  Note:</b> This popup type allows the user to define a custom image for the display. The developer should test all settings prior to deployment to make sure that all images and text fit on the screen correctly.  <b>Note:</b> The function <b>Close Pop-Up.vi</b> can be used to close this dialog early.  Inputs: -Graphic [path] - path to the image file to display -Message [String] - String Variable to populate Display with -Theme [lvclass] - the theme for the style of the popup -Message Info [typedef enum] -Message Height [I32] - the height for the message text box (-1 = match height of image) -Message Width [I32] - the width for the message text box (-1 = match height of image) -Message Location [Ring] - the location of the image relative to the message text box (left, right,bottom, top) -Timeout (s) [DBL] - the amount of time to show the popup  <b> |  | N |
| Display |  | This vi displays the custom timeout Popup. |  | N |

**Gif - Two Button.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Two Button Popup (Gif) |  | The custom image two button popup will show a dialog to the user and wait for their input on a positive or negative selection.  Note:</b> This popup type allows the user to define a custom image for the display. The developer should test all settings prior to deployment to make sure that all images and text fit on the screen correctly.  <b>Note:</b> The function <b>Close Pop-Up (Two Button Choice).vi</b> can be used to send a value when closing this dialog. Otherwise, use Close Pop-Up.vi to close the dialog early.  Inputs: -Graphic [path] - path to the image file to display -Message [String] - String Variable to populate Display with -Theme [lvclass] - the theme for the style of the popup -Message Info [typedef enum] -Message Height [I32] - the height for the message text box (-1 = match height of image) -Message Width [I32] - the width for the message text box (-1 = match height of image) -Message Location [Ring] - the location of the image relative to the message text box (left, right,bottom, top) -YText [String] - the text for the affirmative choice button -NText [String] - the text for the negative chioce button  Outputs -Choice [Boolean] - the user's selection of choice (YText = True, NText = False)  <b> |  | N |
| Display |  | This VI will display a custom image to the user along with their message and buttons for pos/neg choice |  | N |

**Gif Parent.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Read Delays |  | read the delay array for the gif frames |  | N |
| Read Gif Images |  | read the image array for the gif file |  | N |
| Display |  | This is the override method required for each child of the pop up class. It is the pop up that will display to the user.  To extend this function, create a child class of the pop up class. Then create an override method for this VI. Display.vi</b> is a protected function, therefore it cannot be called in a main application. This VI must be called by a caller function. See one of the standard pop-ups for a setup example.  <b> |  | N |
| Gif Load and Size Reader |  | get the information for the size of the custom image and the background info |  | N |
| Gif Picture Grab |  | read the information for the custom image |  | N |
| Template Pattern - Caller |  | Template Pattern to be used when loading a Display Pop Up |  | N |

**SVG - Input (DBL).lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Input (DBL) Popup (SVG) |  | The input (DBL) popup will display information to the user and wait for the operator to enter a value and click the button.  Note:</b> This popup type allows the user to define a custom image for the display. The developer should test all settings prior to deployment to make sure that all images and text fit on the screen correctly.  <b>Note:</b> The function <b>Close Pop-Up (DBL).vi</b> can be used to send a value when closing this dialog. Otherwise, use Close Pop-Up.vi to close it early.  Inputs: -Graphic [path] - path to the image file to display -Message [String] - String Variable to populate Display with -Theme [lvclass] - the theme class for the pop up -Default Input [DBL] - default value for the prompt  Outputs -InputValue [DBL] - the user's input data  <b> |  | N |
| Display |  | This vi will display the input pop up waiting for a type Double input from the user  The graphic will be an input from the developer. |  | N |

**SVG - Input (I32).lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Input (I32) Popup (SVG) |  | The input (I32) popup will display information to the user and wait for the operator to enter a value and click the button.  Note:</b> This popup type allows the user to define a custom image for the display. The developer should test all settings prior to deployment to make sure that all images and text fit on the screen correctly.  <b>Note:</b> The function <b>Close Pop-Up (I32).vi</b> can be used to send a value when closing this dialog. Otherwise, use Close Pop-Up.vi to close it early.  Inputs: -Graphic [path] - path to the image file to display -Message [String] - String Variable to populate Display with -Theme [lvclass] - the theme for the style of the popup -Message Info [typedef enum] -Message Height [I32] - the height for the message text box (-1 = match height of image) -Message Width [I32] - the width for the message text box (-1 = match height of image) -Message Location [Ring] - the location of the image relative to the message text box (left, right,bottom, top) -Default Input [I32] - default value for the prompt  Outputs -InputValue [I32] - the user's input data  <b> |  | N |
| Display |  | This vi will display the input pop up waiting for a type I32 input from the user and will show a customized image. |  | N |

**SVG - Input (String).lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Input (String) Popup (SVG) |  | The input (String) popup will display information to the user and wait for the operator to enter a value and click the button.  Note:</b> This popup type allows the user to define a custom image for the display. The developer should test all settings prior to deployment to make sure that all images and text fit on the screen correctly.  <b>Note:</b> The function <b>Close Pop-Up (String).vi</b> can be used to send a value when closing this dialog. Otherwise, use Close Pop-Up.vi to close it early.  Inputs: -Graphic [path] - path to the image file to display -Message [String] - String Variable to populate Display with -Theme [lvclass] - the theme for the style of the popup -Message Info [typedef enum] -Message Height [I32] - the height for the message text box (-1 = match height of image) -Message Width [I32] - the width for the message text box (-1 = match height of image) -Message Location [Ring] - the location of the image relative to the message text box (left, right,bottom, top) -Default Input [String] - default value for the prompt  Outputs -Input Value [String] - the user's input data  <b> |  | N |
| Display |  | This vi will display the input pop up waiting for a type string input from the user while displaying a custom graphic to the user. |  | N |

**SVG - One Button.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display One Button Popup (SVG) |  | The custom one button popup will display information to the user and wait for the operator to acknowledge the message.  Note:</b> This popup type allows the user to define a custom image for the display. The developer should test all settings prior to deployment to make sure that all images and text fit on the screen correctly.  <b>Note:</b> The function <b>Close Pop-Up.vi</b> can be used to close this dialog early.  Inputs: -Graphic [path] - path to the image file to display -Message [String] - String Variable to populate Display with -Theme [lvclass] - the theme for the style of the popup -Message Info [typedef enum] -Message Height [I32] - the height for the message text box (-1 = match height of image) -Message Width [I32] - the width for the message text box (-1 = match height of image) -Message Location [Ring] - the location of the image relative to the message text box (left, right,bottom, top)  <b> |  | N |
| Display |  | This is the override method required for each child of the pop up class. It is the pop up that will display to the user.  To extend this function, create a child class of the pop up class. Then create an override method for this VI. Display.vi</b> is a protected function, therefore it cannot be called in a main application. This VI must be called by a caller function. See one of the standard pop-ups for a setup example.  <b> |  | N |

**SVG - Timeout One Button.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Timeout One Button Popup (SVG) |  | The custom timeout one button popup will display information to the user and wait for the operator to click the button or until the timeout expires.  Note:</b> This popup type allows the user to define a custom image for the display. The developer should test all settings prior to deployment to make sure that all images and text fit on the screen correctly.  <b>Note:</b> The function <b>Close Pop-Up.vi</b> can be used to close this dialog early.  Inputs: -Graphic [path] - path to the image file to display -Message [String] - String Variable to populate Display with -Theme [lvclass] - the theme for the style of the popup -Message Info [typedef enum] -Message Height [I32] - the height for the message text box (-1 = match height of image) -Message Width [I32] - the width for the message text box (-1 = match height of image) -Message Location [Ring] - the location of the image relative to the message text box (left, right,bottom, top) -Button Text [string] - the text for the button -Timeout (s) [DBL] - timeout duration  <b> |  | N |
| Display |  | This vi displays the custom image one button timeout popup. |  | N |

**SVG - Timeout Two Button.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Timeout Two Button Popup (SVG) |  | The custom two button popup will display information to the user and wait for the operator to click a button or until time expires.  Note:</b> This popup type allows the user to define a custom image for the display. The developer should test all settings prior to deployment to make sure that all images and text fit on the screen correctly.  <b>Note:</b> The function <b>Close Pop-Up (Two Button Choice).vi</b> can be used to send a value when closing this dialog. Otherwise, use Close Pop-Up.vi to close early.  Inputs: -Graphic [path] - path to the image file to display -Message [String] - String Variable to populate Display with -Theme [lvclass] - the theme for the style of the popup -Message Info [typedef enum] -Message Height [I32] - the height for the message text box (-1 = match height of image) -Message Width [I32] - the width for the message text box (-1 = match height of image) -Message Location [Ring] - the location of the image relative to the message text box (left, right,bottom, top) -YText [string] - the string for the YES button -NText [string] - the string for the NO button -Timeout (s) [DBL] - the timeout duration  Outputs -Choice? [Boolean] - the user select (YText = True, NText = False)  <b> |  | N |
| Display |  | This vi displays the custom image two button timeout popup. |  | N |

**SVG - Timeout.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Timeout Popup (SVG) |  | The timeout popup will display information to the user and wait the specified timeout depending on the number of seconds passed into this VI.  Note:</b> This popup type allows the user to define a custom image for the display. The developer should test all settings prior to deployment to make sure that all images and text fit on the screen correctly.  <b>Note:</b> The function <b>Close Pop-Up.vi</b> can be used to close this dialog early.  Inputs: -Graphic [path] - path to the image file to display -Message [String] - String Variable to populate Display with -Theme [lvclass] - the theme for the style of the popup -Message Info [typedef enum] -Message Height [I32] - the height for the message text box (-1 = match height of image) -Message Width [I32] - the width for the message text box (-1 = match height of image) -Message Location [Ring] - the location of the image relative to the message text box (left, right,bottom, top) -Timeout (s) [DBL] - the amount of time to show the popup  <b> |  | N |
| Display |  | This vi displays the custom timeout Popup. |  | N |

**SVG - Two Button.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Two Button Popup (SVG) |  | The custom image two button popup will show a dialog to the user and wait for their input on a positive or negative selection.  Note:</b> This popup type allows the user to define a custom image for the display. The developer should test all settings prior to deployment to make sure that all images and text fit on the screen correctly.  <b>Note:</b> The function <b>Close Pop-Up (Two Button Choice).vi</b> can be used to send a value when closing this dialog. Otherwise, use Close Pop-Up.vi to close the dialog early.  Inputs: -Graphic [path] - path to the image file to display -Message [String] - String Variable to populate Display with -Theme [lvclass] - the theme for the style of the popup -Message Info [typedef enum] -Message Height [I32] - the height for the message text box (-1 = match height of image) -Message Width [I32] - the width for the message text box (-1 = match height of image) -Message Location [Ring] - the location of the image relative to the message text box (left, right,bottom, top) -YText [String] - the text for the affirmative choice button -NText [String] - the text for the negative chioce button  Outputs -Choice [Boolean] - the user's selection of choice (YText = True, NText = False)  <b> |  | N |
| Display |  | This VI will display a custom image to the user along with their message and buttons for pos/neg choice |  | N |

**SVG Parent.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display |  | This is the override method required for each child of the pop up class. It is the pop up that will display to the user.  To extend this function, create a child class of the pop up class. Then create an override method for this VI. Display.vi</b> is a protected function, therefore it cannot be called in a main application. This VI must be called by a caller function. See one of the standard pop-ups for a setup example.  <b> |  | N |
| Template Pattern - Caller |  |  |  | N |

**Custom - Input (DBL).lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Input (DBL) Popup (Custom) |  | The input (DBL) popup will display information to the user and wait for the operator to enter a value and click the button.  Note:</b> This popup type allows the user to define a custom image for the display. The developer should test all settings prior to deployment to make sure that all images and text fit on the screen correctly.  <b>Note:</b> The function <b>Close Pop-Up (DBL).vi</b> can be used to send a value when closing this dialog. Otherwise, use Close Pop-Up.vi to close it early.  Inputs: -Graphic [path] - path to the image file to display -Message [String] - String Variable to populate Display with -Theme [lvclass] - the theme class for the pop up -Default Input [DBL] - default value for the prompt  Outputs -InputValue [DBL] - the user's input data  <b> |  | N |
| Display |  | This vi will display the input pop up waiting for a type Double input from the user  The graphic will be an input from the developer. |  | N |

**Custom - Input (I32).lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Input (I32) Popup (Custom) |  | The input (I32) popup will display information to the user and wait for the operator to enter a value and click the button.  Note:</b> This popup type allows the user to define a custom image for the display. The developer should test all settings prior to deployment to make sure that all images and text fit on the screen correctly.  <b>Note:</b> The function <b>Close Pop-Up (I32).vi</b> can be used to send a value when closing this dialog. Otherwise, use Close Pop-Up.vi to close it early.  Inputs: -Graphic [path] - path to the image file to display -Message [String] - String Variable to populate Display with -Theme [lvclass] - the theme for the style of the popup -Message Info [typedef enum] -Message Height [I32] - the height for the message text box (-1 = match height of image) -Message Width [I32] - the width for the message text box (-1 = match height of image) -Message Location [Ring] - the location of the image relative to the message text box (left, right,bottom, top) -Default Input [I32] - default value for the prompt  Outputs -InputValue [I32] - the user's input data  <b> |  | N |
| Display |  | This vi will display the input pop up waiting for a type I32 input from the user and will show a customized image. |  | N |

**Custom - Input (String).lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Input (String) Popup (Custom) |  | The input (String) popup will display information to the user and wait for the operator to enter a value and click the button.  Note:</b> This popup type allows the user to define a custom image for the display. The developer should test all settings prior to deployment to make sure that all images and text fit on the screen correctly.  <b>Note:</b> The function <b>Close Pop-Up (String).vi</b> can be used to send a value when closing this dialog. Otherwise, use Close Pop-Up.vi to close it early.  Inputs: -Graphic [path] - path to the image file to display -Message [String] - String Variable to populate Display with -Theme [lvclass] - the theme for the style of the popup -Message Info [typedef enum] -Message Height [I32] - the height for the message text box (-1 = match height of image) -Message Width [I32] - the width for the message text box (-1 = match height of image) -Message Location [Ring] - the location of the image relative to the message text box (left, right,bottom, top) -Default Input [String] - default value for the prompt  Outputs -Input Value [String] - the user's input data  <b> |  | N |
| Display |  | This vi will display the input pop up waiting for a type string input from the user while displaying a custom graphic to the user. |  | N |

**Custom - One Button.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display One Button Popup (Custom) |  | The custom one button popup will display information to the user and wait for the operator to acknowledge the message.  Note:</b> This popup type allows the user to define a custom image for the display. The developer should test all settings prior to deployment to make sure that all images and text fit on the screen correctly.  <b>Note:</b> The function <b>Close Pop-Up.vi</b> can be used to close this dialog early.  Inputs: -Graphic [path] - path to the image file to display -Message [String] - String Variable to populate Display with -Theme [lvclass] - the theme for the style of the popup -Message Info [typedef enum] -Message Height [I32] - the height for the message text box (-1 = match height of image) -Message Width [I32] - the width for the message text box (-1 = match height of image) -Message Location [Ring] - the location of the image relative to the message text box (left, right,bottom, top)  <b> |  | N |
| Display |  | This VI will display the custom image and message to the user |  | N |

**Custom - Timeout One Button.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Timeout One Button Popup (Custom) |  | The custom timeout one button popup will display information to the user and wait for the operator to click the button or until the timeout expires.  Note:</b> This popup type allows the user to define a custom image for the display. The developer should test all settings prior to deployment to make sure that all images and text fit on the screen correctly.  <b>Note:</b> The function <b>Close Pop-Up.vi</b> can be used to close this dialog early.  Inputs: -Graphic [path] - path to the image file to display -Message [String] - String Variable to populate Display with -Theme [lvclass] - the theme for the style of the popup -Message Info [typedef enum] -Message Height [I32] - the height for the message text box (-1 = match height of image) -Message Width [I32] - the width for the message text box (-1 = match height of image) -Message Location [Ring] - the location of the image relative to the message text box (left, right,bottom, top) -Button Text [string] - the text for the button -Timeout (s) [DBL] - timeout duration  <b> |  | N |
| Display |  | This vi displays the custom image one button timeout popup. |  | N |

**Custom - Timeout Two Button.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Timeout Two Button Popup (Custom) |  | The custom two button popup will display information to the user and wait for the operator to click a button or until time expires.  Note:</b> This popup type allows the user to define a custom image for the display. The developer should test all settings prior to deployment to make sure that all images and text fit on the screen correctly.  <b>Note:</b> The function <b>Close Pop-Up (Two Button Choice).vi</b> can be used to send a value when closing this dialog. Otherwise, use Close Pop-Up.vi to close early.  Inputs: -Graphic [path] - path to the image file to display -Message [String] - String Variable to populate Display with -Theme [lvclass] - the theme for the style of the popup -Message Info [typedef enum] -Message Height [I32] - the height for the message text box (-1 = match height of image) -Message Width [I32] - the width for the message text box (-1 = match height of image) -Message Location [Ring] - the location of the image relative to the message text box (left, right,bottom, top) -YText [string] - the string for the YES button -NText [string] - the string for the NO button -Timeout (s) [DBL] - the timeout duration  Outputs -Choice? [Boolean] - the user select (YText = True, NText = False)  <b> |  | N |
| Display |  | This vi displays the custom image two button timeout popup. |  | N |

**Custom - Timeout.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Timeout Popup (Custom) |  | The timeout popup will display information to the user and wait the specified timeout depending on the number of seconds passed into this VI.  Note:</b> This popup type allows the user to define a custom image for the display. The developer should test all settings prior to deployment to make sure that all images and text fit on the screen correctly.  <b>Note:</b> The function <b>Close Pop-Up.vi</b> can be used to close this dialog early.  Inputs: -Graphic [path] - path to the image file to display -Message [String] - String Variable to populate Display with -Theme [lvclass] - the theme for the style of the popup -Message Info [typedef enum] -Message Height [I32] - the height for the message text box (-1 = match height of image) -Message Width [I32] - the width for the message text box (-1 = match height of image) -Message Location [Ring] - the location of the image relative to the message text box (left, right,bottom, top) -Timeout (s) [DBL] - the amount of time to show the popup  <b> |  | N |
| Display |  | This vi displays the custom timeout Popup. |  | N |

**Custom - Two Button.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Two Button Popup (Custom) |  | The custom image two button popup will show a dialog to the user and wait for their input on a positive or negative selection.  Note:</b> This popup type allows the user to define a custom image for the display. The developer should test all settings prior to deployment to make sure that all images and text fit on the screen correctly.  <b>Note:</b> The function <b>Close Pop-Up (Two Button Choice).vi</b> can be used to send a value when closing this dialog. Otherwise, use Close Pop-Up.vi to close the dialog early.  Inputs: -Graphic [path] - path to the image file to display -Message [String] - String Variable to populate Display with -Theme [lvclass] - the theme for the style of the popup -Message Info [typedef enum] -Message Height [I32] - the height for the message text box (-1 = match height of image) -Message Width [I32] - the width for the message text box (-1 = match height of image) -Message Location [Ring] - the location of the image relative to the message text box (left, right,bottom, top) -YText [String] - the text for the affirmative choice button -NText [String] - the text for the negative chioce button  Outputs -Choice [Boolean] - the user's selection of choice (YText = True, NText = False)  <b> |  | N |
| Display |  | This VI will display a custom image to the user along with their message and buttons for pos/neg choice |  | N |

**PNG Parent.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display |  | This is the override method required for each child of the pop up class. It is the pop up that will display to the user.  To extend this function, create a child class of the pop up class. Then create an override method for this VI. Display.vi</b> is a protected function, therefore it cannot be called in a main application. This VI must be called by a caller function. See one of the standard pop-ups for a setup example.  <b> |  | N |
| PNG Picture Setup |  | read the information for the custom image |  | N |
| PNG Picture Size Reader |  | get the information for the size of the custom image and the background info |  | N |
| Template Pattern - Caller |  |  |  | N |

**Plain - One Button.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display One Button Popup (Plain) |  | The one button popup will display information to the user and wait for the operator to click the button.  Note:</b> The function <b>Close Pop-Up.vi</b> can be used to close this dialog early.  <b>Note:</b> The dialog vertical sizing will adjust for longer messages, so <b>test</b> all messages prior to deployment to make sure that scaling fits.  Inputs: -Dialog Title [String] - String variable to populate the dialog's title bar -Message [String] - String variable to populate Display -Theme [lvclass] - the theme for the style of the popup -Button Text [String] - String variable for defining the text to show on the button  <b> |  | N |
| Display |  | This is the override method required for each child of the pop up class. It is the pop up that will display to the user.  To extend this function, create a child class of the pop up class. Then create an override method for this VI. Display.vi</b> is a protected function, therefore it cannot be called in a main application. This VI must be called by a caller function. See one of the standard pop-ups for a setup example.  <b> |  | N |

**Plain - Two Button.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Two Button Popup (Plain) |  | The two button popup will display information to the user and wait for the operator to click the button to make a choice.  Note:</b> The function <b>Close Pop-Up.vi</b> can be used to close this dialog early.  <b>Note:</b> The dialog vertical sizing will adjust for longer messages, so <b>test</b> all messages prior to deployment to make sure that scaling fits.  Inputs: -Dialog Title [String] - String variable to populate the dialog's title bar -Message [String] - String Variable to populate Display -Theme [lvclass] - the theme for the style of the popup -Y-Text [String] - the button text for the <b>yes</b> button (right) -N-Text [String] - the button text for the <b>no</b> button (left)  Outputs: -Choice [Boolean] - true/false of user selection <b> |  | N |
| Display |  | This is the override method required for each child of the pop up class. It is the pop up that will display to the user.  To extend this function, create a child class of the pop up class. Then create an override method for this VI. Display.vi</b> is a protected function, therefore it cannot be called in a main application. This VI must be called by a caller function. See one of the standard pop-ups for a setup example.  <b> |  | N |

**Plain - Three Button.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Display Three Button Popup (Plain) |  | The one button popup will display information to the user and wait for the operator to click the button.  Note:</b> The function <b>Close Pop-Up.vi</b> can be used to close this dialog early.  <b>Note:</b> The dialog vertical sizing will adjust for longer messages, so <b>test</b> all messages prior to deployment to make sure that scaling fits.  Inputs: -Dialog Title [String] - String variable to populate the dialog's title bar -Message [String] - String Variable to populate Display -Theme [lvclass] - the theme for the style of the popup -Action 0 [String] - the button text for the <b>Action 0</b> button (far right) -Action 1 [String] - the button text for the <b>Action 1</b> button (middle) -Action 2 [String] - the button text for the <b>Action 2</b> button (far left)  Outputs: -Choice [U8] - user selection of button 0, 1, or 2  <b> |  | N |
| Display |  | This is the override method required for each child of the pop up class. It is the pop up that will display to the user.  To extend this function, create a child class of the pop up class. Then create an override method for this VI. Display.vi</b> is a protected function, therefore it cannot be called in a main application. This VI must be called by a caller function. See one of the standard pop-ups for a setup example.  <b> |  | N |

**Plain Parent.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Read Action 1 Text |  | No description found (add content in vi description) |  | N |
| Write Action 1 Text |  | No description found (add content in vi description) |  | N |
| Read Action 2 Text |  | No description found (add content in vi description) |  | N |
| Write Action 2 Text |  | No description found (add content in vi description) |  | N |
| Read Action 3 Text |  | No description found (add content in vi description) |  | N |
| Write Action 3 Text |  | No description found (add content in vi description) |  | N |
| Read Dialog Title |  | No description found (add content in vi description) |  | N |
| Write Dialog Title |  | No description found (add content in vi description) |  | N |
| Display |  | This is the override method required for each child of the pop up class. It is the pop up that will display to the user.  To extend this function, create a child class of the pop up class. Then create an override method for this VI. Display.vi</b> is a protected function, therefore it cannot be called in a main application. This VI must be called by a caller function. See one of the standard pop-ups for a setup example.  <b> |  | N |

**Popup.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Read Image |  | This Data Member Access VI will read the Image from the object.  Outputs -Image [2D Image] - the image to display on the pop up |  | N |
| Write Image |  | This Data Member Access VI will write the Image to the object.  Inputs: -Image [2D Picture] - the image to write to the display pop up |  | N |
| Read Message |  | This Data Member Access VI will read the Label from the object.  Outputs -Label [string] - label string to write to the pop up display |  | N |
| Write Message |  | This Data Member Access VI will write the Label to the object.  Inputs: -Label [string] - the text to write to the display pop up |  | N |
| Read Choice |  | This Data Member Access VI will read the choice Boolean from the object.  Outputs -Choice [Boolean] - the user choice on a two button pop up |  | N |
| Write Choice |  | This Data Member Access VI will write the Choice Boolean to the object.  Inputs: -Choice [Boolean] - the value to write to the object |  | N |
| Read pre-empt |  | get the pre-emptible notifier reference |  | N |
| Write pre-empt |  | write the preemptible notifier |  | N |
| Read Theme |  | Read the theme of the pop up |  | N |
| Write Theme |  | Write the theme class for the pop up |  | N |
| Read Message Size |  | Read the size of the message |  | N |
| Write Message Size |  | Write the size of the message |  | N |
| Read Pic Width |  | Read the width of the custom picture |  | N |
| Write Pic Width |  | Write the custom picture width |  | N |
| Read Pic Height |  | Read the height of the custom picture |  | N |
| Write Pic Height |  | Write the custom picture height |  | N |
| Read Timeout |  | This Data Member Access VI will read the timeout Boolean from the object.  Outputs -Timeout [Boolean] - the timeout result |  | N |
| Write Timeout |  | This Data Member Access VI will write the timeout Boolean on the object.  Outputs -Timeout [Boolean] - the result of the timeout |  | N |
| Check pre-empt -- Boolean |  | check on the status of the pre-empt notifier |  | N |
| Check pre-empt -- DBL |  | check on the status of the pre-empt notifier |  | N |
| Check pre-empt -- I32 |  | check on the status of the pre-empt notifier |  | N |
| Check pre-empt -- String |  | check on the status of the pre-empt notifier |  | N |
| Check pre-empt |  | check on the status of the pre-empt notifier |  | N |
| Close pre-empt |  | get the pre-emptible notifier reference |  | N |
| Create pre-empt |  | get the pre-emptible notifier reference |  | N |
| Close FP |  | Will close the front panel from the screen for the input VI Reference - should be used in all pop-up types that configure all the front panel BEFORE showing it when it is called so that the user doesn't see jitter |  | N |
| Configure FP-Message-Image |  | Configure the front panel size, the message, and the image for a custom pop up type. |  | N |
| Convert Enum to Image |  | This VI will take the input ENUM typedef for the Graphic chosen and will get its image data using the LabVIEW PNG files.  Inputs: -Graphics [typedef combobox] - the image to look up -background [2d picture] - the background color for the image  Outputs: -Image [2D Image] - the image data |  | N |
| Format Message - Embedded |  | This VI will check for embedded tags within a found tag in the loop searching for the content. |  | N |
| Format Message - Strip Tags |  | Remove the formatting tags from the string input - will strip off the first one so that string can continue to be processed for more tags. |  | N |
| LAVA.move picture |  | adapted from: https://lavag.org/topic/17138-2d-picture-translation-movement/#comment-105597 |  | N |
| LAVA.Parse drawing operation |  | Adapted from : https://lavag.org/topic/17138-2d-picture-translation-movement/#comment-105597 |  | N |
| Move Button |  | move the button on the screen to the correct location based on the FP size |  | N |
| Shift Input |  | move the input down to the correct position and shift the front panel size and location for input button if needed |  | N |
| Template Pattern - Caller |  |  |  | N |
| Template Pattern - Display |  | A "Template Pattern" VI that will call the display child for the given pop-up type and set up/tear down the pre-empt messaging |  | N |
| Update FP and Show |  | turn back on the front panel updates and show the FP on the screen - this should be done after all moving parts have been placed. this will be used for custom pop ups when the image size is unknown and the locations move. |  | N |
| Custom - Resize FP |  | customize the front panel by setting up all the positioning based on the size of the panel and delay inputs for now so that the shifting on screen will not be distracting for the user |  | N |
| Custom - Resize Pic and String |  | update the sizing of the button and the image based on the calculated inputs and with padding |  | N |
| Display |  | This is the override method required for each child of the pop up class. It is the pop up that will display to the user.  To extend this function, create a child class of the pop up class. Then create an override method for this VI. Display.vi</b> is a protected function, therefore it cannot be called in a main application. This VI must be called by a caller function. See one of the standard pop-ups for a setup example.  <b> |  | N |
| Close Pop-Up |  | This VI can be used to close the pop-up early before the user has finished interacting or it has completed its action.  For example, a timeout dialog can be implemented with an approximate time for an action, but could be closed early if the action finishes before the dialog is complete. |  | N |
| Close Pop-Up (Two Button Choice) |  | This VI can be used to close the pop-up early before the user has finished interacting.  Use this VI to also send a Boolean value - this is read when using the two-button dialog pop-up types (timeout or standard). This will make the selection for the user based on whatever the developer has decided. |  | N |
| Close Pop-Up (I32) |  | This VI can be used to close the pop-up early before the user has finished interacting.  Use this VI to also send an integer value - this is read when using the Input (I32) pop-up types. |  | N |
| Close Pop-Up (DBL) |  | This VI can be used to close the pop-up early before the user has finished interacting.  Use this VI to also send a DBL value - this is read when using the Input (DBL) pop-up types. |  | N |
| Close Pop-Up (String) |  | This VI can be used to close the pop-up early before the user has finished interacting.  Use this VI to also send a string value - this is read when using the Input (String) pop-up types. |  | N |
| Format Message |  | This VI applies special formatting to message strings. The tags are listed below and their purpose:  < b> bold < /b> - without spaces; used so tags will show in context help <i> italic </i> <strong> bold - strong </strong> <em> italic - emphasis </em> <del> strikethrough </del> <ins> underline </ins> <font rgb(#,#,#)> RGB-based color of text </font> |  | N |

**Theme.lvclass**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Connector Pane** | **Description** | **S.** | **U.T.** |
| Apply Style - button |  | Apply the theme style to a button (colors) |  | N |
| Apply Style - input |  | Apply the theme style to an input control (double/integer digital numveric or string) |  | N |
| Apply Style - Simple Dialog |  | Apply the style to the message pop up : format the header, background, and message box |  | N |
| Apply Style |  | Apply the style to the message pop up : format the header, background, and message box |  | N |
| Set picture background |  | Set the background of a picture to the specified color |  | N |
| Construct Theme |  | Construct a theme object from an input of all of the type defs |  | N |
| Load Material Dialog |  | Load a theme from a material file |  | N |
| Load Theme - TSC Dark |  | Load the default dark theme for pop ups |  | N |
| Load Theme - TSC Light |  | Load the default light theme for pop ups |  | N |
| Load Theme |  | Load a theme from file |  | N |
| Pick Button Hover |  | perform a calculation to lighten or darken the button to give a hover contrast |  | N |
| Pick Text Color |  | perform a calculation to determine which font color should be used for a given background color |  | N |
| priv-default theme |  | Load the default theme for pop ups |  | N |
| Read Theme |  | Read theme data from an object |  | N |
| Reset Theme |  | Reset the global theme for the application - this unloads the global theme that will be used and uses the default theme or input from a VI |  | N |
| Save Theme |  | Save the theme data |  | N |
| Temp - Change Font |  | No description found (add content in vi description) |  | N |

**Classes**