Push buttons are rectangular buttons, labeled with text, that can initiate an action when the user selects the button. On some graphics devices, a push button has a three-dimensional appearance.

Push buttons can perform three types of actions:

- execute a single command
- request more information from the user in order to complete the task identified by the button text
- display a list or menu of commands.

Using the Push Button Class

Assume you want an entry called TENNIS.FRAME to execute when a push button, PBUTTON1, is selected. This requires the following steps:

1. In the Value on Selection window, enter tennis.frame in the Value field.
2. Include the following code in your SCL program:

   ```
   PBUTTON1:
   call display(pbutton1);
   return;
   ```

Alternatively, you can accomplish the same thing by accessing the Command Processing Window and assigning `afappl c=sasuser.test.tennis.frame` to the Execute SAS commands on selection field. In this case, the SCL code shown
above is not needed. When the push button is selected, the entry, 
SASUSER.TEST.TENNIS.FRAME, executes.

**Methods**

Methods specific to the Push Button class are described here. Inherited methods are 
described in the Object class and the Widget class.

**Dictionary**

---

### _getLabel

Returns the label assigned to a push button

---

**Syntax**

CALL NOTIFY (push-button-name, '_getLabel', label);

---

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>label</td>
<td>C</td>
<td>returns the label assigned to the push button</td>
</tr>
</tbody>
</table>

---

### _getText

Returns the character value assigned to a push button

Inherited from Widget

---

**Syntax**

CALL NOTIFY (push-button-name, '_getText', text-value);
Push Button Class

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>text-value</td>
<td>C</td>
<td>returns the character value assigned to the push button. This value is blank if the push button is not selected.</td>
</tr>
</tbody>
</table>

Details

If a push button has been assigned a numeric return value in the Value on Selection window, the _getText method causes a program halt. Use the _getValue method to return the value of a push button that has a numeric value.

Example

If the value \texttt{print} was assigned to the push button in the Value on Selection window, _getText returns the value \texttt{print} and assigns it to the variable TEXT:

\begin{verbatim}
length text $ 24;
call notify('pbutton1','_get_text_',text);
if (text='print') then
   _msg_='You have selected the print option.';
\end{verbatim}

_getValue

Returns the numeric value assigned to a push button

Inherited from Widget

Syntax

\begin{verbatim}
CALL NOTIFY (push-button-name, '_getValue', push-button-value);
\end{verbatim}

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>push-button-value</td>
<td>N</td>
<td>returns the numeric value assigned to the push button. This value is blank if the push button is not selected.</td>
</tr>
</tbody>
</table>

Details

If a push button has been assigned a character return value in the Value on Selection window, the _getValue method causes a program halt. Use the _getText method to return the value of a push button that has a character value.

Example

If the value 4 was assigned to the push button in the Value on Selection window, _getValue returns the value 4 and assigns it to the variable VALUE:

\begin{verbatim}
call notify('pbutton1','_get_value_',value);
if (value=4) then
\end{verbatim}
Chapter 85

_msg_ = ‘You have made reservations for four.’;

_setColor

Assigns a color to a push button label

Restrictions  Some systems do not support using methods to change the color of a push button label.

Syntax

CALL NOTIFY (push-button-name, '_setColor', label-color);

Argument Type Description

| label-color | C | specifies the color for the push button label.

_setLabel

Assigns a label to a push button

Syntax

CALL NOTIFY (push-button-name, '_setLabel', label);

Argument Type Description

| label | C | specifies the label to assign

Details

_setLabel assigns the value specified in label as the new label for the push button. If the push button does not have a return value, the new label also becomes the new return value.