A step by step guide to display picture in screens and catching the mouse click event.

Introduction

A step by step guide on displaying pictures in the screen and catching the click event on the picture. Through the transaction SE78 we upload a picture to SAP. The picture uploaded is displayed in a screen and an information message is displayed on clicking that picture. All the steps and the coding required are demonstrated in a very simple way. A basic knowledge on Classes and Objects is preferable.

Step 1: Uploading picture using SE78

Run transaction SE78 and point to menu > Graphics > Import. Select the file and type a name as shown below. Here I have given the name CLICK_IMG for the picture.

Step 2: Creating a custom control in screen

Create a custom control in the screen. Here it is named as CLICK_IMG_CONTAINER.

Step 3: Coding for displaying the picture in screen.

In the PBO module of the screen, attach the following code. Please note that the picture name is CLICK_IMG and container name is CLICK_IMG_CONTAINER. Please pass your own object names to the method.

```
MODULE STATUS_1000 OUTPUT.
DATA: W_LINES TYPE I.
TYPES PICT_LINE(256) TYPE C.
DATA : CONTAINER TYPE REF TO CL_GUI_CUSTOM_CONTAINER,
      PICTURE TYPE REF TO CL_GUI_PICTURE,
      PICT_TAB TYPE TABLE OF PICT_LINE,
      URL(255) TYPE C.
DATA: GRAPHIC_URL(255).
DATA: BEGIN OF GRAPHIC_TABLE OCCURS 0,
      LINE(255) TYPE X,
      END OF GRAPHIC_TABLE.
DATA: L_GRAPHIC_CONV TYPE I.
DATA: L_GRAPHIC_OFFS TYPE I.
DATA: GRAPHIC_SIZE TYPE I.
DATA: L_GRAPHIC_XSTR TYPE XSTRING.
CALL METHOD cl_gui_cfw=>flush
EXCEPTIONS
  cntl_system_error = 1
  cntl_error = 2
  others = 3.
ENDIF.
CREATE OBJECT: CONTAINER EXPORTING CONTAINER_NAME = 'CLICK_IMG_CONTAINER',
      PICTURE EXPORTING PARENT = CONTAINER,
CALL METHOD CL_SSF_XSF_UTILITIES=>GET_BDS_GRAPHIC_AS_BMP
      EXPORTING
      P_OBJECT = 'GRAPHICS'
```
Step 4: Class definition and implementation for capturing click event.

Now we are going to add the coding for click event of picture. In the global part of the report program add the following code.

REPORT zaby_image_click.

CALL SCREEN 1000.

CLASS cl_picture_click DEFINITION.

PUBLIC SECTION.

METHODS: picture_yclick FOR EVENT picture_click of cl_gui_picture IMPORTING mouse_pos_x mouse_pos_y.ENDCLASS.

CLASS cl_picture_click IMPLEMENTATION.

METHOD: picture_yclick.

PERFORM onclick. "write the action of click in this form.ENDMETHOD

DATA: obj_cl_picture_class TYPE REF TO cl_picture_click.

DATA: it_event_picture TYPE tablen OF cntl_simple_event, wa_event_picture TYPE cntl_simple_event.

Step 5: Coding for action on click event
Now double-click on ‘PERFORM click’ that we just added in the class implementation and add the following code.

form ONCLICK.MESSAGE 'You have clicked on the image' TYPE 'I'.endform.

**Step 6: Coding for capturing click event**

Add another module in the screen PBO and add the following code to that module.

```plaintext
module CLICK_IMAGE_1000 output.
wa_event_picture-eventid = cl_gui_picture=>eventid_picture_click.
wa_event_picture-appl_event = 'X'.
APPEND wa_event_picture to it_event_picture.
CALL METHOD picture->set_registered_events
Exporting
events = it_event_picture
Exceptions
cntl_error = 1
cntl_system_error = 2
illegal_event_combination = 3
OTHERS = 4.
CREATE OBJECT obj_cl_picture_class.
SET HANDLER obj_cl_picture_class->picture_yclick FOR picture.
endmodule.
```

Now check, activate and execute. Click on the picture in the screen. A message will be displayed.
Quick Links

- Our Community
- Members
- SAP Mentors
- Show Your Community Pride
- Community Challenges
- SAP Community Network
- Success Story Center
- Online Users Record
- Hotly Discussed Topics
- Humorous Topics
- Great Contributors
- Wiki Contributors
- Second Life

Events

- Subscribers Contest
- On-line Meetups
- SAP Network Blogger Awards
- Community Events
- Community Photos
- SAP TechEd
- Second Life

Subscribe

- RSS 2.0
- ATOM FEED