Table of Contents

Sometimes you want to show dialogues instead of workscreens because a dialog is a simple popup that shows details about a selected record or requests input from the user. Another reason could be that a workscreen shows different dialogues on button clicks.

It's very easy to show a dialog in your application or workscreen. Use the dialog class and use it as follows:

```
//dialog content
UIFormLayout folContent = new UIFormLayout();

UIPanel panContent = new UIPanel(folContent);
panContent.add(new UILabel("Name"));
panContent.add(new UITextField(), folContent.getConstraints(1, 0, -1, 0));
panContent.add(new UILabel("Value"));
panContent.add(new UITextField(), folContent.getConstraints(1, 1, -1, 1));

//dialog with Ok button
Dialog dlg = new Dialog(panContent);
dlg.setTitle("Dialog test");
dlg.setModal(true);
//show dialog as frame
Dialog.openInternalFrame(getApplication());
```

The dialog itself is a content and can be used without internal frames as well. It depends on your IApplication implementation: if you use an internal frame or the content itself.

The class supports OK and cancel buttons and allows user-defined buttons instead of default buttons.

The title is an optional attribute. If you don't set the title, the name of the dialog will be used. The dialog itself has a default preferred size. You should change the default setting if you need a different size.

The above dialog looks like this:

With Ok and cancel:

Here's an example dialog:

```
private UINumberField nfValue;

public void doShowDialog()
{
    nfValue = new UINumberField();
    nfValue.setPrecision(3);
    nfValue.setScale(0);

    UIFormLayout flDialog = new UIFormLayout();
```

```
    UIPanel panDialog = new UIPanel(flDialog);
    panDialog.add(new UILabel("Value"), flDialog.getConstraints(0, 0));
    panDialog.add(nfValue, flDialog.getConstraints(1, 0, -1, 0));
    panDialog.setPreferredSize(200, 40);

    Dialog dialog = new Dialog(panDialog);
    dialog.setModal(true);
    dialog.setTitle("Enter a numeric value");
    dialog.setPreferredSize(null);
    dialog.getOkButton().eventAction().addListener(this::doAfterDialog);

    dialog.open(getApplication());
}

public void doAfterDialog()
{
    showInformation("Value = " + nfValue.getValue());
}
```

The dialog can be opened by button press or another action. It opens a simple dialog which shows a label and a number field. The number can be max. 3 digits. If you press OK button in the dialog, an information message with the entered number will be shown.

From:
https://doc.sibvisions.com/ - **Documentation**

Permanent link:
https://doc.sibvisions.com/jvx/client/gui/custom_dialog

Last update: **2025/08/20 10:01**