



SilverDev

by Experia

SilverDev Designer Tutorial

*Graphical RPG Applications Development Studio
On AS/400 – IBM i*



Creation of a SilverDev application

1.	Introduction	3
2.	Select a context	3
3.	Form Creation	4
4.	Database Wizard	6
5.	Add a Pop-Up	7
6.	Saving the screen	9
7.	Compiling the screen	9
8.	RPG source	10
9.	Additional Tools	12
10.	Whole source	13
11.	Compiling the RPG source	14
12.	Create a CLP program	15
13.	Making an application available	15
14.	For more information	17
14.1.	Programming guide	17
14.2.	Wizard	17

1. Introduction

This tutorial explains how to create “manually” a graphical application in RPG with SilverDev’s IDE : Designer Studio.

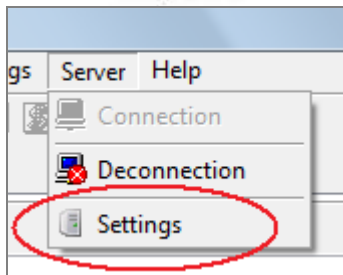
You have to install SilverDev on your server. You can download a free 60-day trial on www.silverdev.com



*If you want to learn how to create an application with the Wizard, see the document named “**Wizard Tutorial**” available on this webpage : <http://www.silverdev.com/test-silverdev.php>*



Designer.exe



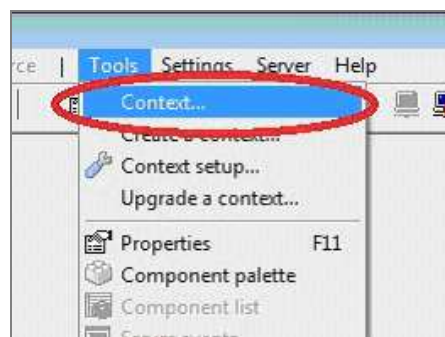
Run the Designer.exe program in the installation directory.

If you did not enter your login information during the installation step, you can do it now using the “Server/Configuration” menu then follow the instructions in the [QuickStart](#) chapter
(download it here : <http://www.silverdev.com/test-silverdev.php>)

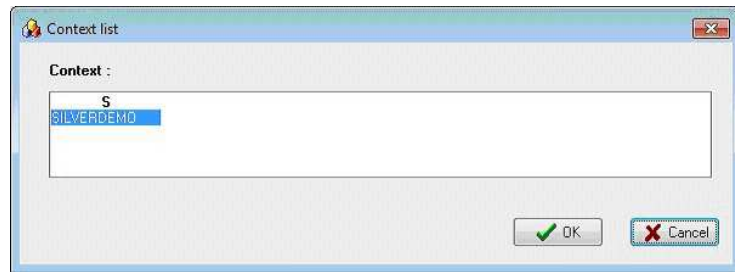
2. Select a context

1

Use the “Tools/Context...” menu to display the context list.



- 2 If you have selected the silverdemo option when installing SilverDev, the silverdemo context is displayed in the list.
Click the OK button.



- 3 The list displayed shows the programs provided as examples.

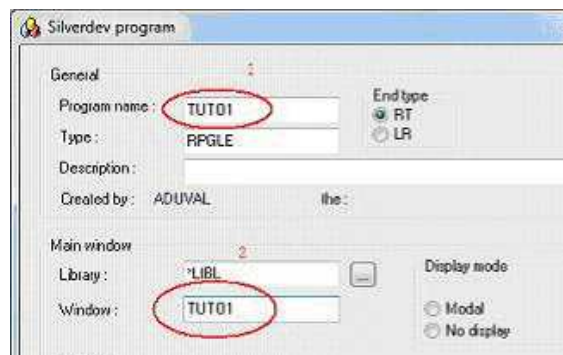
Note : You can see/modify any demo program source by double clicking on an item in the list.

Context					
SILVERDEMO					
Program	Description	Window	Window library	Display mode	End type
SDADDGRP	Ajout d'un groupe	SDADDGRP	*LIBL	*MODAL	RPGL
SDCHGCFG		SDCHGCFG	*LIBL	*NOMODAL	RPGL
SDDMACT	Activités AS/400	SDDMACT1	*LIBL		RPGL
SDDMAUT	Auteurs	SDDMAUT	*LIBL	*NOMODAL	RPGL
SDDMCALC		SDDMCALC	*LIBL	*NOMODAL	RPGL
SDDMCHART		SDDMCHART	*LIBL		RPGL
SDDMCOLOR		SDDMCOLOR	*LIBL		RPGL
SDDMDOCK1		SDDMDOCK1	*LIBL	*NOMODAL	RPGL
SDDMDOCK2		SDDMDOCK2	*LIBL	*NOMODAL	RPGL
SDDMDROP		SDDMDROP	*LIBL		RPGL
SDDMGRAPHE	Graphiques	SDDMGRAPHE	*LIBL		RPGL
SDDMLANG	Application multilingues	SDDMLANG	*LIBL	*NOMODAL	RPGL
SDDMLIB		SDDMLIB	*LIBL		RPGL
SDDMLIST		SDDMLIST	*LIBL	*NOSHOW	RPGL
SDDMLIV	gestion des livres	SDDMLIV1	*LIBL		RPGL
SDDMLIVSTA	Statistiques livres	SDDMLIVSTA	*LIBL		RPGL

3. Form Creation

- 1 Use the  button to add a program.

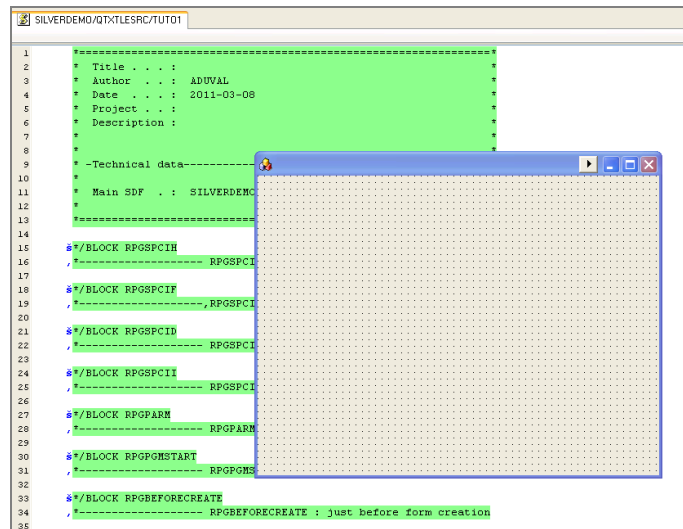
- 2 A form is displayed: fill in the name of the program that you want to create.



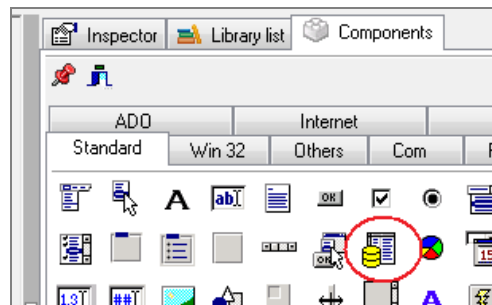
- 3 After clicking OK, the program appears in the list.

SDWRKGRP	Gestion des groupes	SDWRKGRP	*LIBL	*NOMODAL
SDWRKMBR	Gestion des membres d'un groupe	SDWRKMBR	*LIBL	*NOMODAL
TUT01		TUT01	*LIBL	*NOMODAL

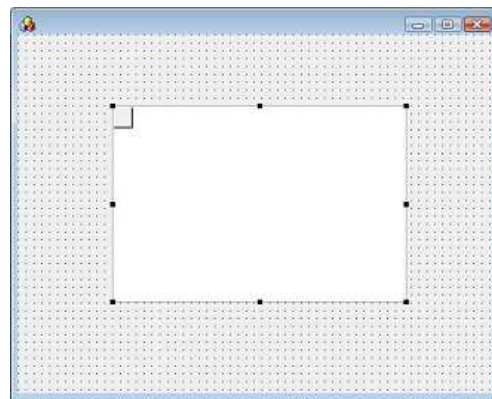
To access the program, double-click the line: the source and line will be displayed on the screen. The source file is already filled in.



Use the “Components” tool window to add a CSFL type component. Select the component from the palette and click the form under construction to place the component.



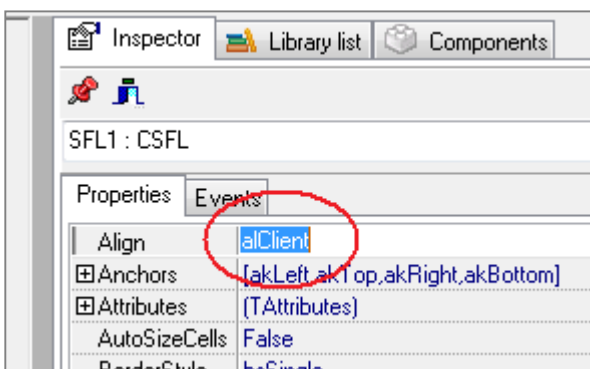
The component is now on the form, as shown on the right.



Use the “Inspector” tool window to modify the properties of the CSFL component.

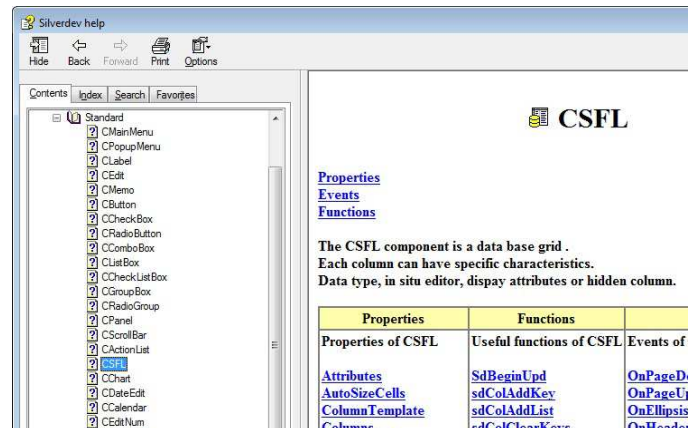
The component must be selected (surrounded by black squares).

Modify the Align property. Select the alClient value. This means that the component takes up the whole of its parent’s space (here, the parent is the form).



8

To obtain help regarding the CSFL component, select the component and press F1 to display the on-line help:

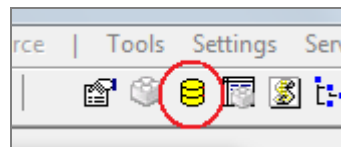


4. Database Wizard

To add columns to the CSFL component, you can double-click the component, but here, we will use a tool to import columns from a database file.

1

Use the icon indicated below to display the “Database wizard” tool window.



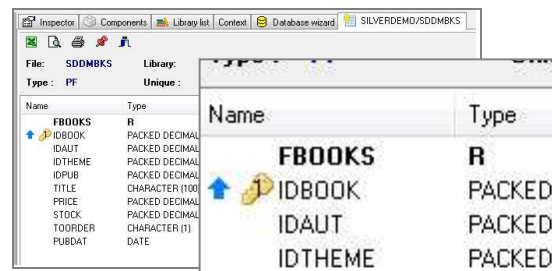
2

In the “Database wizard” tool window, use the search tab. Search for the SDDMBKS file (press Enter to start the search). The list of corresponding files on the System i is displayed. Double-click the name of a file to display its characteristics.

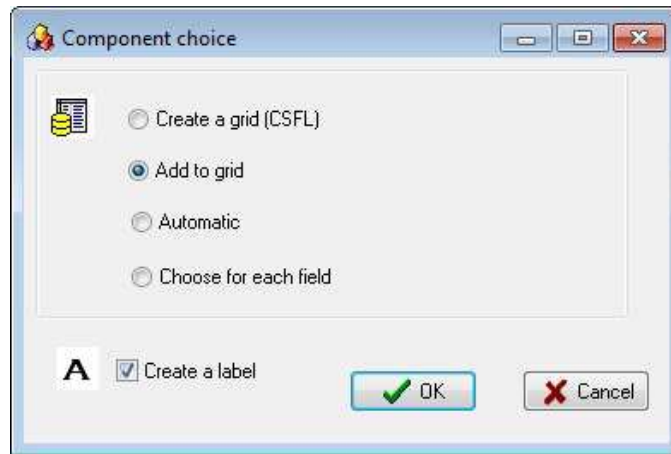


3

A new window opens: select the IDBOOK, TITLE and PRICE fields, and drag to the form under construction.



4 Another window opens, asking what you want to do. Leave the default selection and click OK.

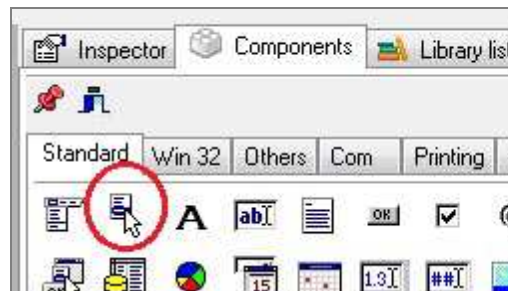


5 Three columns have been created in the CSFL component. The properties of these columns have been modified according to the file's fields. You can change the properties of these columns if you want.

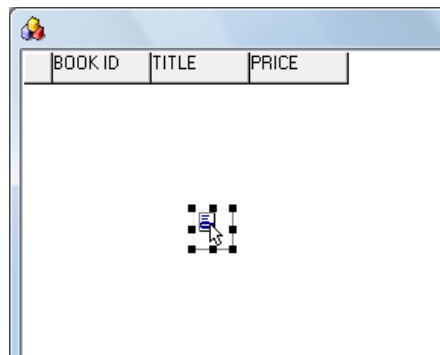


5. Add a Pop-Up

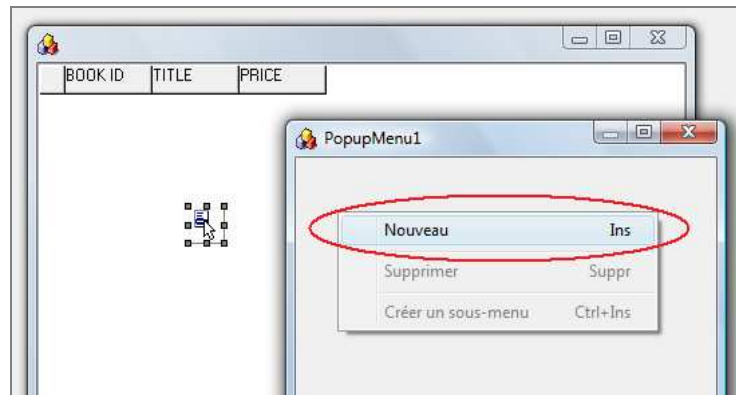
1 In the "Components" tool window, select the CPopupMenu component and place it on the form.



2 CpopupMenu is represented by a square. Double-click the square to add elements to the popup menu. A window appears.



3 Right click to open a menu.

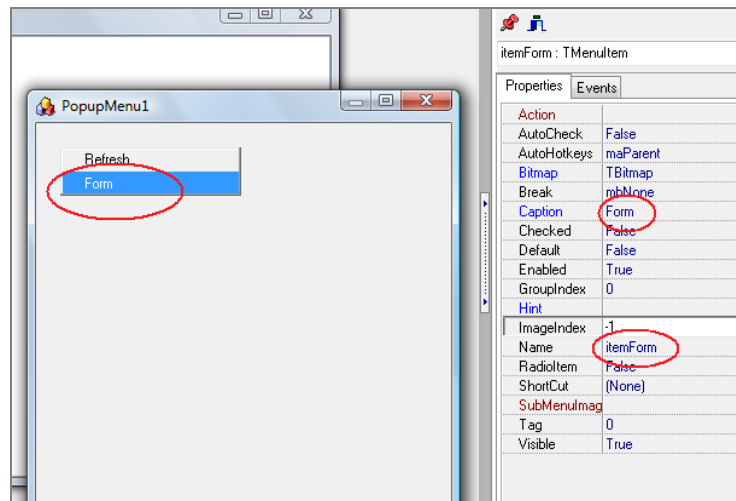


Using the “New” menu, create two menu elements.

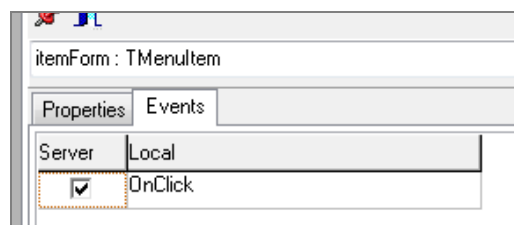
Use the Property Inspector to change the properties of these two menu elements.

4 Call the first menu element itemRefresh and the second one itemForm.

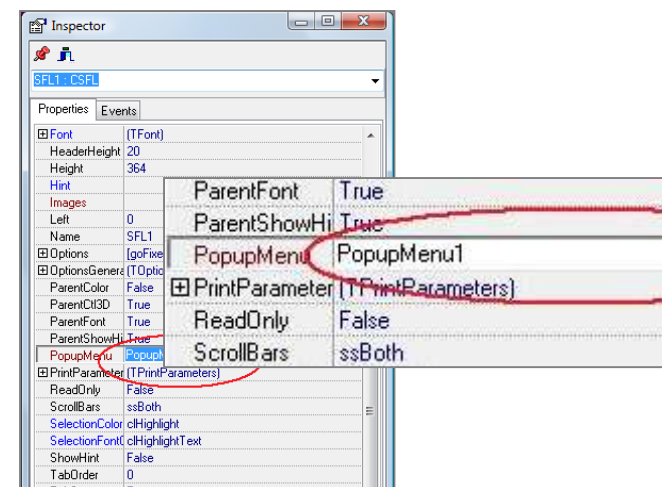
Set the Caption properties of these two elements to “Refresh” and “Form” respectively.



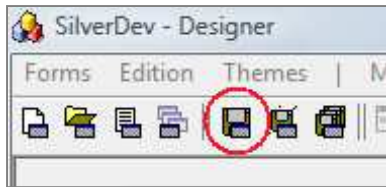
5 In the Property Inspector, select the Events tab and tick the OnClick events for each of the menu's two elements.



6 Select the CSFL component again and change its PopupMenu property. Select the PopupMenu1 value. This means that when a right click is made on the CSFL component, popupmenu1 will be displayed.



6. Saving the screen

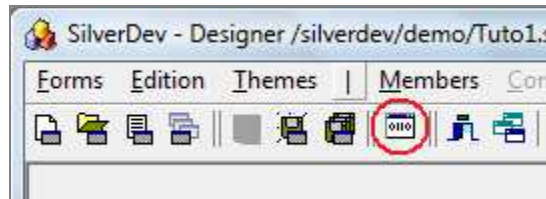


Save the screen source by clicking this icon

7. Compiling the screen

1

Compile the screen by clicking this icon:



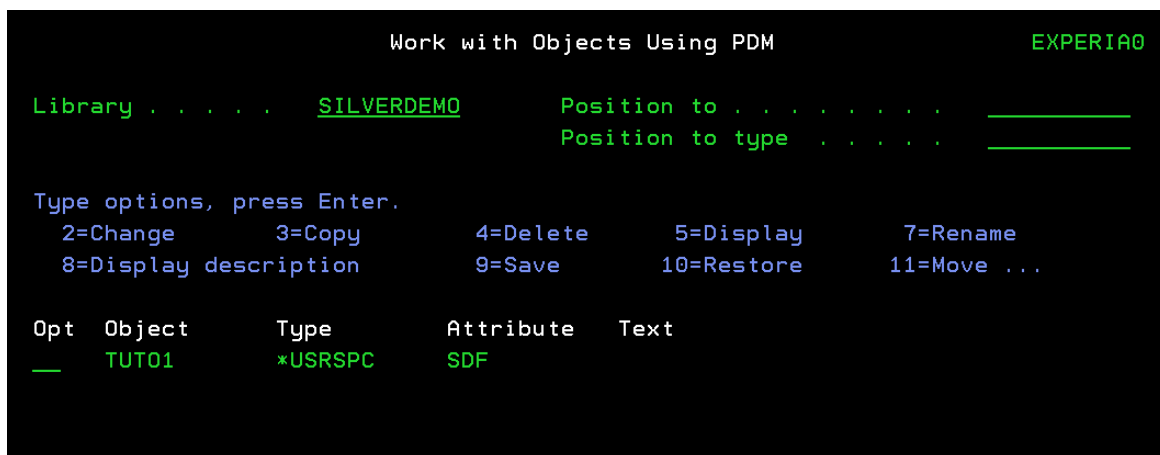
2

Choose the name TUTO1 in the SILVERDEMO library for the compiled object.



As you can see below, the screen is a *USRSPC type object.

3



8. RPG source

1

In the RPGSPCIF block reserved for file declarations, declare the SDDMBKS file.

```

17
18  S*/BLOCK RPGSPCIF
19  -----,RPGSPCIF : Files declarations (F Spec.)
20  F SDDMBKS  if  | | | | | k disk  |
21
22  S*/BLOCK RPGSPCID
23  ----- RPGSPCID : Data descriptions (D Spec.)
24

```



Comment 1: to move from the screen to the source or from the source to the screen, press F10.

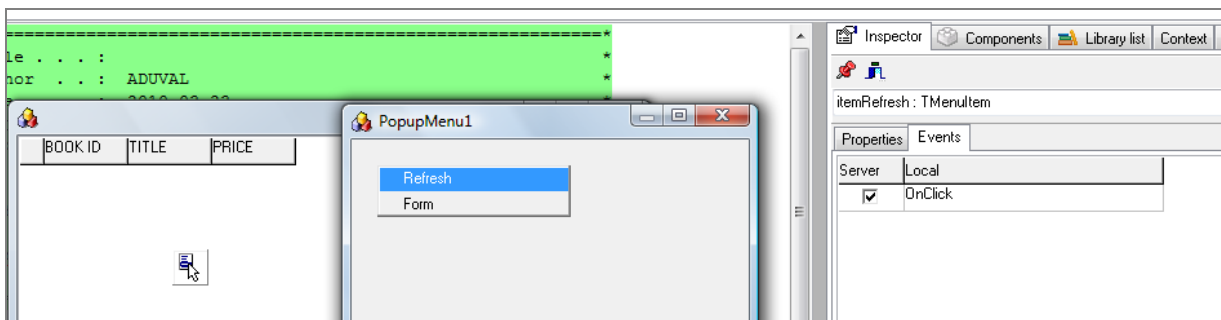
2

Comment 2: to display the ruler, press F8.

Comment 3: you can drag the file name from the Database wizard.

Double-click the CPopupMenu component and tick the OnClick event of the itemRefresh element.

3



Double click the OnClick line: the cursor is now in the source in the itemRefresh_Onclick event.

4

```

56
57  ~*/EVENT itemRefresh_OnClick
58  ,
59  , * Description :
60  ,
61  D Parameters      ds              based(pevtinf)
62  D Win              5u 0
63  D Evt              48a
64  ,

```

Enter the code as shown below:

5

```
SILVERDEMO/QTXTLESRC/TUT01 *
56  /*/EVENT itemRefresh_OnClick
57  /
58  /* Description :
59  /
60  D Parameters      ds          based(pevtnf)
61  D Win              5u 0
62  D Evt              48a
63  /
64  D row              s          10u 0 inz(0)
65  c                  callp      sdClear(F1:'SFL1')
66  C      *loval      setll      sddmbks
67  c                  read       sddmbks
68  c                  dow        not %eof(sddmbks)
69  c                  eval       row = row + 1
70  c                  callp      sdSetCell(F1:'SFL1':'Title':row:title)
71  c                  callp      sdSetCellNum(F1:'SFL1':'Price':row:price)
72  c                  callp      sdSetCellNum(F1:'SFL1':'idBook':
73  c                  row:idBook)
74  c                  read       sddmbks
75  c                  enddo
```

In the itemRefresh_OnClick event, we will read the sddmbks file and fill in the CSFL component.

Do the same for the itemForm_OnClick event and enter this code:

6

```
76  /*/EVENT itemForm_OnClick
77  /
78  /* Description :
79  /
80  D Parameters      ds          based(pevtnf)
81  D Win              5u 0
82  D Evt              48a
83  /
84  D modRes           s          10u 0
85  D row              s          10i 0
86  c                  eval       row = sdGetInt(F1:'SFL1':'RowSelected')
87  c                  if         row <= 0
88  c                  return
89  c                  endif
90  c                  eval       idBook = sdGetCellNum(F1:'SFL1':'idBook':
91  c                  row)
92  c                  call       'SDDMBKS2'
93  C                  parm              modRes
94  C                  parm              idBook
95  c                  if         modRes = mrOk
96  c                  callp      itemRefresh_OnClick
97  c                  endif
```

In the itemForm_OnClick event, we will call the SDDMLIV2 program, which is provided.



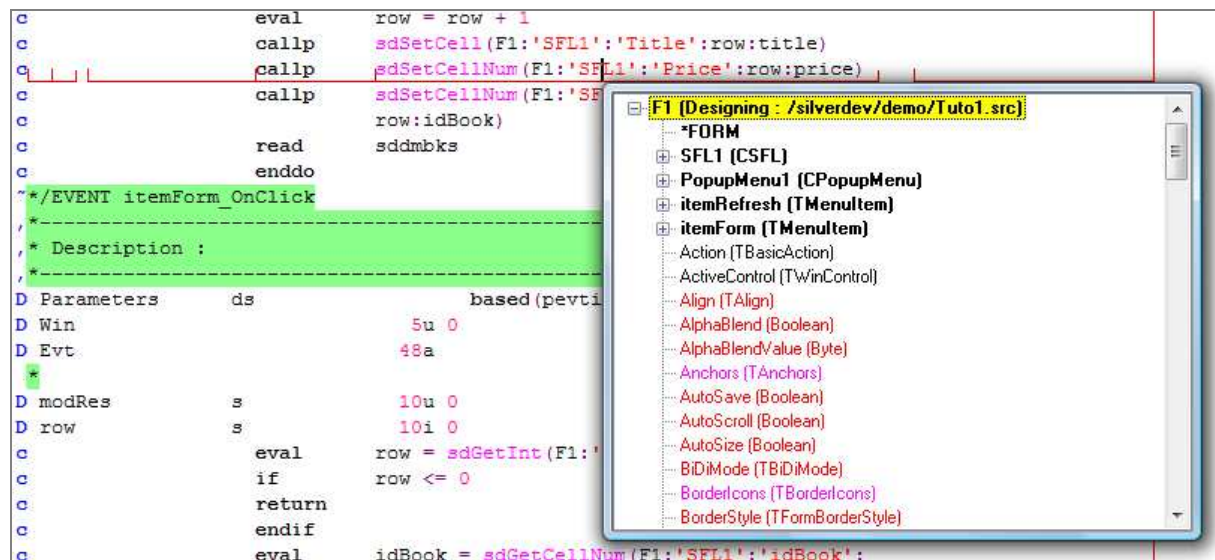
In the example, the words shown in purple are functions provided with SilverDev. They enable the components to be manipulated via the RPG program.

To see the list of Silverdev functions, press **Ctrl+Space** to display the list of functions corresponding to the first letters of the word you are typing in.

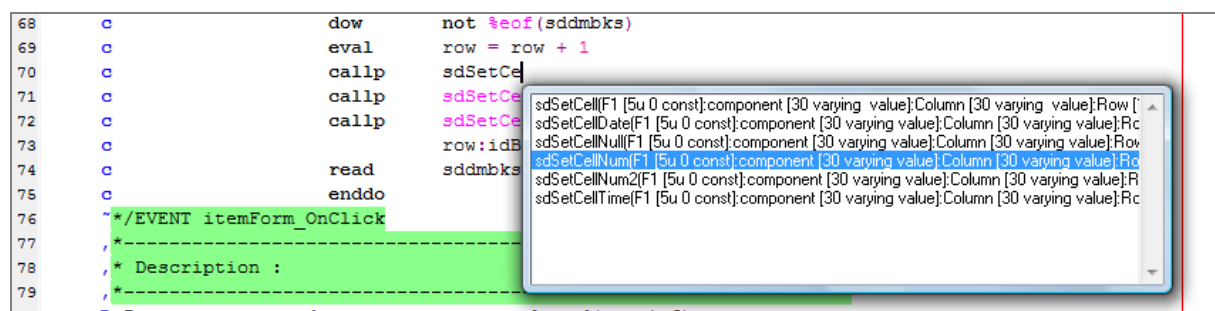
9. Additional Tools

Press F2: to display the list of the window's components and properties in a tree structure

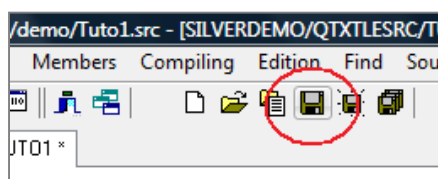
Press Enter: to insert a component or property name



Press F4: to see a format prompt window



Press F8: to display the cloning of the current line



Save the modifications made to the source by clicking the icon shown below or via the "Members/Save member" menu.

10. Whole source

```

=====
* Title . . . :
* Author . . : ADUVAL
* Date . . . : 2011-03-08
* Project . . :
* Description :
*
*
* -Technical data-----
*
* Main SDF . . : *LIBL/TUTO1
*
=====
§*/BLOCK RPGSPCIH
,----- RPGSPCIH : H specifications (Heading)
§*/BLOCK RPGSPCIF
,-----,RPGSPCIF : Files declarations (F Spec.)
FSDDMBKS if e k disk
§*/BLOCK RPGSPCID
,----- RPGSPCID : Data descriptions (D Spec.)
§*/BLOCK RPGSPCII
,----- RPGSPCII : Input descriptions (I Spec.)
§*/BLOCK RPGPARM
,----- RPGPARM : Program parameters and declarations (KLIST, DEFINE, ...)
§*/BLOCK RPGPGMSTART
,----- RPGPGMSTART : Beginning of Initializations procedure
§*/BLOCK RPGBEFORECREATE
,----- RPGBEFORECREATE : just before form creation
§*/BLOCK RPGAFTERCREATE
,----- RPGAFTERCREATE : After form creation
§*/BLOCK RPGBEFORESHOW
,----- RPGBEFORESHOW: Before show(F1)
§*/BLOCK RPGAFTERSHOW
,----- RPGAFTERSHOW : After show(F1)
§*/BLOCK RPGPROCDEF
,----- RPGPROCDEF : User procedures
§*/BLOCK RPGSR
,----- RPGSR : Sub Routine
§*/BLOCK RPGTABLE
,----- RPGTABLE : Compile time array data (**)

~/EVENT itemRefresh_OnClick
,-----
, Description :
,-----
D Parameters ds based(pevtinf)
D Win 5u 0
D Evt 48a
,
D row s 10u 0
C callp sdClear(F1:'SFL1')
C *loval settl sddmbks
C read sddmbks
C dow not %eof(sddmbks)
C eval row = row + 1
C callp sdSetCell(F1:'SFL1':'Title':Row:Title)
C callp sdSetCellNum(F1:'SFL1':'Price':Row:price)
C callp sdSetCellNum(F1:'SFL1':'idbook':Row:idbook)
C read sddmbks
C enddo

~/EVENT itemForm_OnClick
,-----
, Description :
,-----
D Parameters ds based(pevtinf)
D Win 5u 0
D Evt 48a
,

```

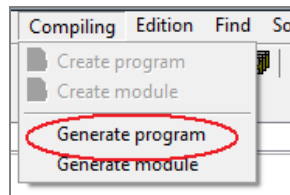
```

D modres      s      10u 0
D row         s      10i 0
c
c      eval      row = sdGetInt(F1:'SFL1':'RowSelected')
c      if        row <= 0
c      return
c      endif
c      eval      idbook = sdGetCellNum(F1:'SFL1':'idbook':Row)
c      call      'SDDMBKS2'
c      parm      modRes
c      parm      idbook
c      if        modRes = mrOk
c      callp     itemRefresh_OnClick
c      endif

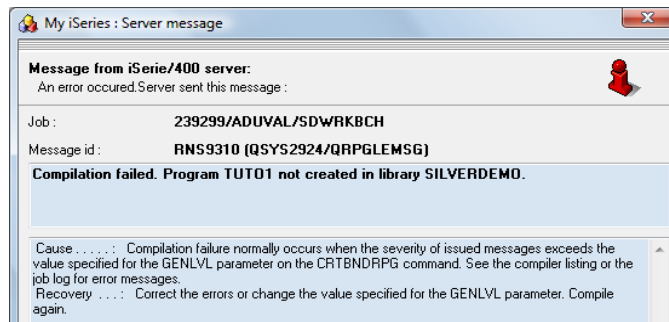
```

11. Compiling the RPG source

- 1 Compile the program using the “Compilation/Generate program” menu.



- 2 If an error occurs during compilation, the following dialogue box will be displayed.

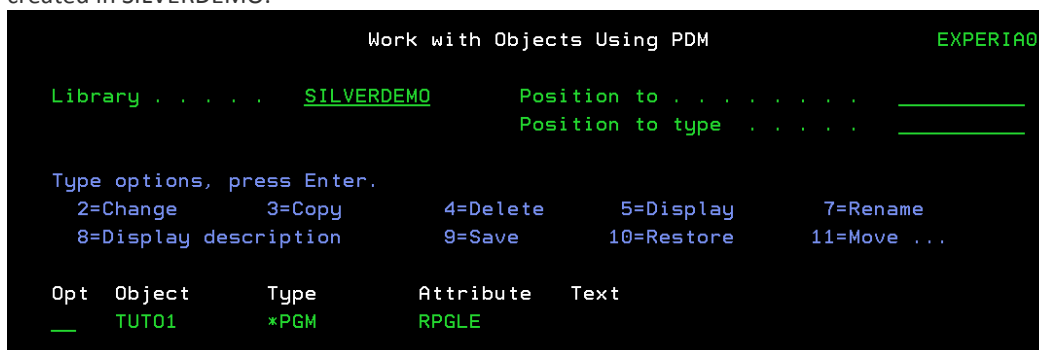


- 3 The compilation results will be displayed in a new window. Compilation errors are shown in bold.

Compilation results				
Id	Line	Column	Level	Message
RNS9308	0	0	50	Compilation stopped. Severity 30 errors found in program.
RNF5410	71	1	30	The prototype for the call is not defined.
RNF7030	71	68	30	The name or indicator TITE is not defined.
RNF7503	71	68	30	Expression contains an operand that is not c

Double click a line to access the location in the source. When there are no more errors, the program will be created in SILVERDEMO.

- 4



12. Create a CLP program

Now, we are going to create a CL CTUTO1 program which will add SILVERDEMO to the library list and run the TUTO1 program.

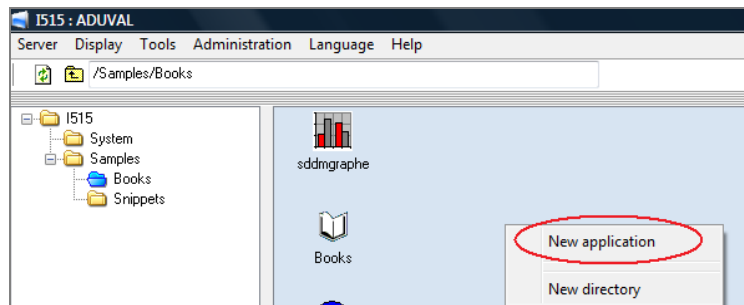
```
Columns . . . : 1 71          Edit          SILVERDEMO/QCLSRC
SEU==> _____ CTUT01
***** Beginning of data *****
0001.00          PGM
0002.00          ADDLIB  LIB(SILVERDEMO) POSITION(*AFTER QTEMP)
0003.00          MONMSG  MSGID(CPF0000)
0004.00          CALL    PGM(TUTO1)
0005.00          ENDPGM
***** End of data *****
```

13. Making an application available

To make the application available, you must have administrator rights. Check that your profile is in the SILVERDEV/PSVDADM file. Log on to MyDesk, and right click in a directory. Choose the "New application" menu.

1

Connectez vous ensuite à **MyDesk**, et effectuez un click droit dans un répertoire. Choisissez le menu « Nouvelle application ».



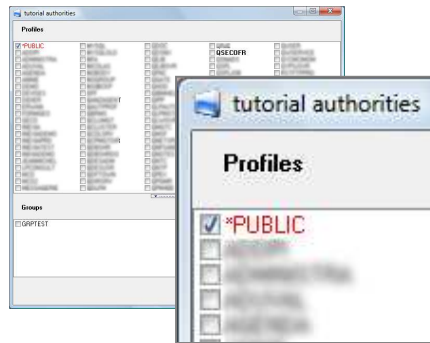
2

Fill in the fields as shown here:

Title:	tutorial
app file name:	tutorial
Command:	call silverdemo/ctuto1
Description:	
<input type="checkbox"/> Only one instance	
Icon:	
Internal icon	
Order:	-1

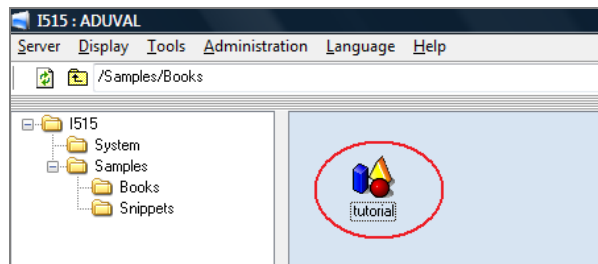
3

When you click OK, a rights management window will be displayed. Click either *PUBLIC to enable everyone to see the application or a defined profile list.



4

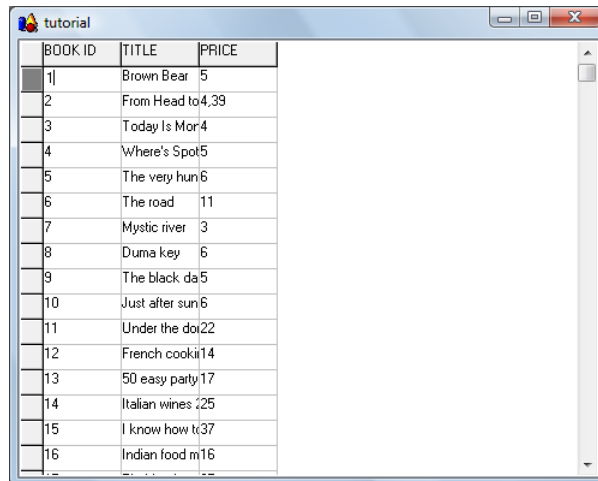
An icon is now shown in MyDesk. The application is available. Double click the icon to start the application that we have just created.



5

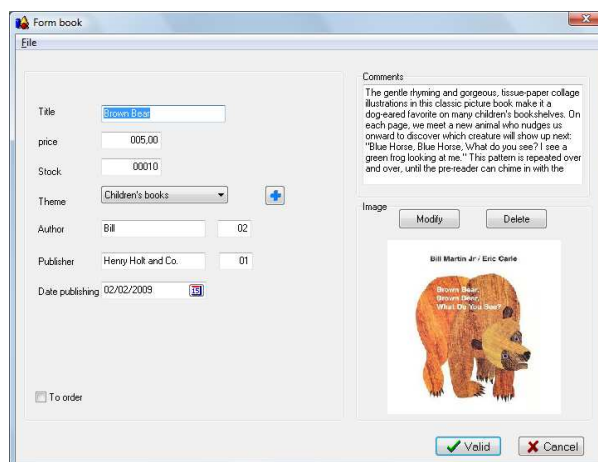
Once the application is running, right click on the form and click "Refresh".

Right click one of the lines, click "Form".



6

Result!



For a more detailed example, you can view the SDDMLIV1 and SDDMBKS2 programs.

14. For more information

14.1. *Programming guide*

Detailed documentation on the SilverDev product is available in the "Programming guide.pdf" file. This file is on the installation cd-rom as well as in the archive file that you have downloaded.

14.2. *Wizard*

Since the V3R5 release, Silverdev allows to create programs directly from the database. To learn how to use the wizard, read the [Wizard_tutorial.pdf](#) document available on this webpage : <http://www.silverdev.com/test-silverdev.php>

Copyright Information.

The information contained in this document may be modified without notification and may not be considered binding in any manner upon EXPERIA Europe. Software packages are supplied under licence or confidentiality agreement. The software package may only be used, copied or reproduced onto any support in compliance with the terms of the said licence or confidentiality agreement. The buyer may only make copies for backup or archiving purposes.

No part of the manual or software package may be reproduced or transmitted via any means, electronic or otherwise, including photocopying, recording or any other storage, processing or information retrieval process for any purpose other than the buyer's personal use without express, written permission from EXPERIA Europe.

IBM, AS/400, iSeries, System i, i5, Power i are registered trademarks belonging to International Business Machines Corporation.

Windows is a registered trademark belonging to Microsoft.

All other products are registered trademarks belonging to their respective companies.



EXPERIA Europe

4, rue L.Beridot
Les jardins d'Epione
F-38500 VOIRON

www.experia.com
www.silverdev.com

Contact Us:

Information info@experia.com

Technical Support support@silverdev.com

Phone +33 (0)4 76 67 07 70