

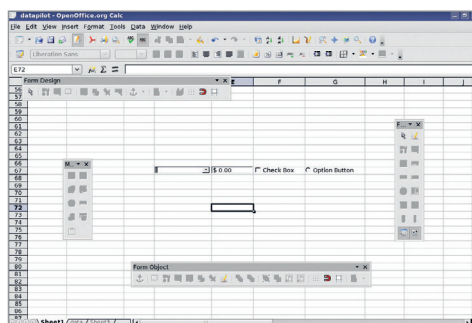
Using form controls for data entry

Form controls are usually associated with web work, but they can be useful in Calc too.

Earlier on in the *Calc* tutorial on the DataPilot, we made much of the fact that that particular tool could find the discrete values in a table of data and make sense of them. In our fictitious expenses document, we could enter anything, and invent new people and types of expense as we went along. In some instances that might be a good thing, but clearly for a real expenses record, you don't want users entering bogus categories and people to attribute their spending to!

The data validation tool in *Calc* is pretty comprehensive and robust. But that only solves half the problem. Although it prevents bogus data from being entered, it doesn't actually help the user input the data. They might struggle with their spelling, or not quite remember what categories are available. A simple form control can save time

inputting data, make sure the results are valid, and the data can still be used in complex calculations (even the DataPilot) without having to resort to filing it all away in a database. In fact, you don't need a database to use form controls at all!



A range of form controls are available in Calc.

Form controls

If you are familiar with the internet at all, you have come across forms. These are the pesky data grabbing areas of websites that ask for your name, address and whether you don't not want to not have no spam not sent to you.

Form controls in *OOo* follow along similar lines: they are rather like the GUI elements of software that allow the user to change settings and enter data. Normally forms input directly to a web-based script or a database, but with the power of *Calc*, there is no need for a database to get some use out of forms. For more information on using forms and databases, turn to p60; p76 and p83 are also loaded with information on how to integrate *Base* databases with the rest of *OOo*.

Calc: set up form controls

	Date	Type	Cost	Person
1	01/09/2007	Train	£56.00	Effy
2	01/09/2007	Food	£122.00	Paul
3	01/09/2007	Drink	£33.00	Mike
4	01/09/2007	Drink	£45.00	Graham
5	01/09/2007	Drink	£66.00	Effy
6	01/09/2007	Drink	£66.00	Paul
7	02/09/2007	Taxi	£23.00	Paul
8	02/09/2007	Train	£28.00	Mike
9	02/09/2007	Taxi	£123.00	Graham
10	02/09/2007	Food	£66.00	Matt
11	02/09/2007	Train	£56.00	Matt
12	02/09/2007	Drink	£88.00	Paul
13	03/09/2007	Taxi	£53.00	Graham
14	03/09/2007	Train	£122.00	Effy
15	03/09/2007	Drink	£126.00	Mike
16	05/09/2007	Train	£165.00	Nick
17	05/09/2007	Taxi	£86.00	Mike
18	05/09/2007	Food	£65.00	Matt
19	05/09/2007	Train	£57.00	Effy
20	05/09/2007	Food	£55.00	Paul
21	06/09/2007	Taxi	£45.00	Graham
22	06/09/2007	Drink	£33.00	Mike
23	06/09/2007	Train	£78.00	Mike
24	07/09/2007	Drink	£88.00	Effy
25	07/09/2007	Train	£69.00	Paul
26	07/09/2007	Taxi	£69.00	Effy
27	07/09/2007	Food	£83.00	Graham
28	07/09/2007	Drink	£44.00	Nick
29	08/09/2007	Food	£24.00	Nick
30	08/09/2007	Drink	£48.00	Graham
31	08/09/2007	Food	£22.00	Paul

1 Raw data

This is the data we had originally for our expenses project with DataPilot on page 46. At least two of the columns use a small set of values. Data like this is ideal for turning into a form, which will make input faster and less error-prone. The first task is to identify what the range of possible values could be. The expenses types are Food, Drink, Trains and Taxis.

	A	B	C
1	Food	Effy	
2	Drink	Mike	
3	Taxi	Paul	
4	Train	Matt	
5		Graham	
6		Nick	
7			
8			
9			

2 Listbox element

Right-click on the Sheet2 tab at the bottom of the screen and select Rename from the menu. Call the sheet **data**. Now click on the tab, and in the range A1: A4, enter the values we worked out from the previous step. These values will form the basis of our List Box form element that we will create. The format and style of these items doesn't matter as the form control will ignore it.

Project 1: form controls in Calc

01/09/2007	Food	£122.00	Paul
01/09/2007	Drink	£33.00	Mike
01/09/2007	Drink	£45.00	Graham
01/09/2007	Drink	£66.00	Effy
02/09/2007	Taxi	£23.00	Paul
02/09/2007	Train	£28.00	Mike
02/09/2007	Taxi	£123.00	Graham
02/09/2007	Food	£66.00	Matt
02/09/2007	Train	£56.00	Matt
02/09/2007	Drink	£88.00	Paul
03/09/2007	Taxi	£53.00	Graham
03/09/2007	Train	£122.00	Effy
03/09/2007	Drink	£126.00	Mike
05/09/2007	Train	£165.00	Nick
05/09/2007	Taxi	£86.00	Mike
05/09/2007	Food	£65.00	Matt
05/09/2007	Train	£57.00	Effy
05/09/2007	Food	£55.00	Paul
06/09/2007	Taxi	£45.00	Graham
06/09/2007	Drink	£33.00	Mike

3 Select toolbars

Switch back to the original sheet now. Before we create any form controls, we will need the relevant toolbars. There are actually several of them for form design, and it can be a bit confusing because some of the controls appear on more than one toolbar. Select View > Toolbar > Form Controls. One of the most important icons to get familiar with is the icon on the top right of the toolbar which switches the form design mode on and off (otherwise you can't click on them to make them work).

4 Testing, testing

Select the List Box control and drag out the shape of the control (you can move it later on). The list will appear, but it is blank. Right-click on it and choose Control from the menu to bring up the dialogue. Click on the Data tab, and in the bottom text box, enter the range for our data we defined, **data.A1:A4**. Now click on the General tab, scroll down, click on the button next to Default selection, and choose a default from the list. Switch Design Mode off in order to test out the lister.

5 Integraton

The form control List Box is now working, but there is still one problem – the value it contains is stuck in the listbox, so how do we actually use it for anything? There is another property of the List Box, and that is the target cell field. Specify a cell here, and the selected value will appear within it. If you are really clever, you can place the List Box directly over this cell, so it looks to all the world as if the List Box value is part of the spreadsheet. Then you can copy and paste the lister (though you will have to manually edit the target for each).

6 It's a date!

There is at least one more column of data that can be converted into a form control – the date. Although this isn't a small number of regular values, dates are well supported by the form controls, and it is fairly trivial to add this too. You don't need to go overboard with the use of form objects, but they can be very useful switches in all kinds of spreadsheet projects. **LXF**