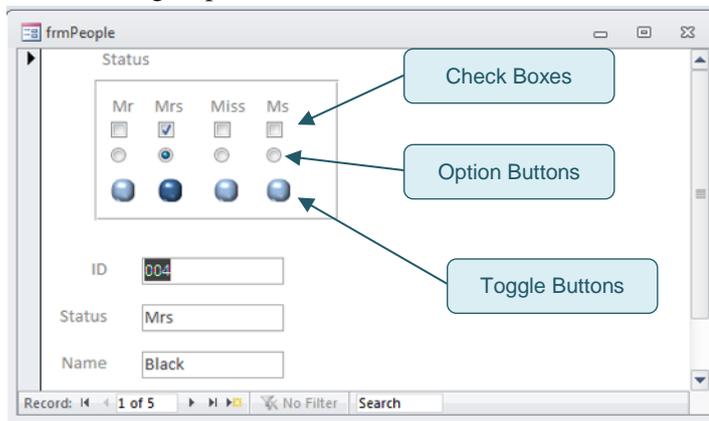


Introduction

Option groups allow us to select one of a set of values. They appear on a form as a group of check boxes, option buttons or radio buttons.

The screenshot below shows three different option groups and a textboxes on the same form.

Each of the groups is bound to the same data source (*Status*), so each is displaying the same value (*Mrs*). Each group is made up of a different type of control.



Ticking a different check box, pressing a different toggle button or clicking a different option button would all change the data stored in *Status* to a new value (as would typing a new value into the textbox).

There is however one problem with option groups. They can only handle integer (whole number) values. This means that if an option group is used

on a form to enter or display data in a table, the data can only be a whole number.

Suppose we wanted to display marital status (Mr, Mrs, Miss or Ms) using an option group. A tick box would be ideal for this. But the values we want to store are text, not integers so we cannot use an option group directly in this situation.

There are various ways we can get round this. One way would be to link the option group to a Visual Basic procedure rather than to the data. The procedure would convert the underlying text value to a number to display in the option group. If we wanted changes to the option group to be stored then another procedure would be needed to react to a change in the option group value and store the correct text in the underlying table.

A simpler method, which the sample solution uses, is to store the numeric value in the table. The option group can then interact with this value directly. A second table, linked to the first, could then be used to hold the corresponding text values. This is probably easier to understand with an example.

In our example we will store marital status in a table and use an option box on a form to enter and display it.