GUI Technical Manual

Ben Pasero

OpenFrame is the driver class for the entire program and starts the GUI. The OpenFrame JFrame simply displays a frame and takes user input (row and column size) and creates a spreadsheet based on those inputs. OpenFrame also contains code to prevent the user for entering anything but an integer (input is checked with a regular expression representing a valid integer).

CellTable is the frame instantiated by OpenFrame. The CellTable frames main purpose is to hold the Jtable at the heart of the GUI. The Jtable is created with an underlying dataModel. The datamodel is a class with a 2d array of type Object at is core, as well as various getters and setters. In this case the array is of Cell objects.

In CellTables constructor we see mostly typical swing related code, but the RowNumberTable class needs explaining. By default, a JTable cannot display row numbers. To get around this, the general solution is to create another JTable next to the main table with only one column and an equal number of rows with every cell displaying a row number. The cells are also specially rendered to look similar to the column headers.

The final component of the GUI is the CellEditor class. In the JTable, the data being edited is a cell formula while the data being displayed is a cell value. This makes things tricky. To overcome this problem a custom cell editor for Cell objects must be used. Typically customs cell editors are used for table cells that display things other than text (most commonly other swing visual components). In the case of this program we want to edit cell formulas and calculate the spreadsheet in order after. The getTableCellEditorComponent method allows us to get a reference to the cell being edited and display its formula to the user. The getCellEditorValue method is used to set the formula of the referenced cell and to recalculate the spreadsheet. Error checking for cycles is also done here.