

Convert strings to formatted numerical values

Introduction

In the creation of an Excel workbook, sometimes a column contains numerical values stored as strings. For example, a percent column can display percents as strings. This poses a problem if you want to use these values for equations, or if you want to clean up the display of the file, as Excel will mark these strings with a green corner indicating that they can be displayed as numerals, along with a warning message, "The number in this cell is stored as text or preceded by an apostrophe."

ExcelApplication allows you to change the NumberFormat of a column so that it will automatically format numerals. This alone is not enough to change the strings into formatted numerals, though, as you must trim any characters off the strings and convert them to doubles to be handled correctly by Excel.

Hence, converting the values for an entire column requires three steps:

1. Gather the values from the column into a collection and convert them to numeric values.
2. Apply the desired Number Format to the styles of column.
3. Cycle through the collection and add the values back to each cell of the column.

The following sample method takes a column of strings of the form "00%" and display them as percentages with no decimals.

Code

```
private static void ConvertToPercent(Workbook wb, Worksheet ws, int Column, int
FirstRow)
{
    //--- String object to hold the original value
    String value;

    //--- Separator to parse the original string
    String[] separator = { "%" };

    //--- ArrayList to hold the values
    ArrayList values = new ArrayList();

    //--- Cycle from the first row of data to the
    //--- last populated cell
    for (int i = FirstRow; i < ws.PopulatedCells.RowCount; i++)
    {
        //--- Check for null values
        if (ws[i, Column].Value != null)
        {
            //--- Grab the original string value for each cell
            value = ws[i, Column].Value.ToString();

            //--- Remove the percentage sign to
            //--- return the percentage, a number
            //--- represented by a string
            value = value.Split(separator,
StringSplitOptions.None)[0];

            //--- Convert the number to a double
            //--- (dividing by 100 because of
            //--- the percentage conversion)
            //--- and add to the ArrayList
            values.Add((Convert.ToDouble(value) / 100));
        }
    }
}
```

```
}

//--- Create a new style to apply the new NumberFormat
Style style = wb.CreateStyle();

//--- Set the number format to percentage
style.NumberFormat = "0%";

//--- Apply the style change to the column's cells
ws.GetColumnProperties(Column).ApplyStyle(style);

//--- Create an index for the ArrayList in the next loop
int index = 0;

//--- Cycle from the first row of data to the
//--- last populated cell
for (int i = FirstRow; i < ws.PopulatedCells.RowCount; i++)
{
    //--- Handle for null values
    if (ws[i, Column].Value != null)
    {
        //--- Put the values back in as doubles
        //--- and raise the index
        ws[i, Column].Value = values[index];
        index++;
    }
}
```

}	}
---	---