Surpass and XLWT Manual

Ana Nelson

March 19, 2009

Contents

Chapter 1

Installation and Hello World

1.1 Hello World

Let's do a minimal "Hello World" script. We'll need to take care of any imports, initialize a Workbook object, create a Worksheet within the workbook, then write some text. Here's how.

1.1.1 Surpass

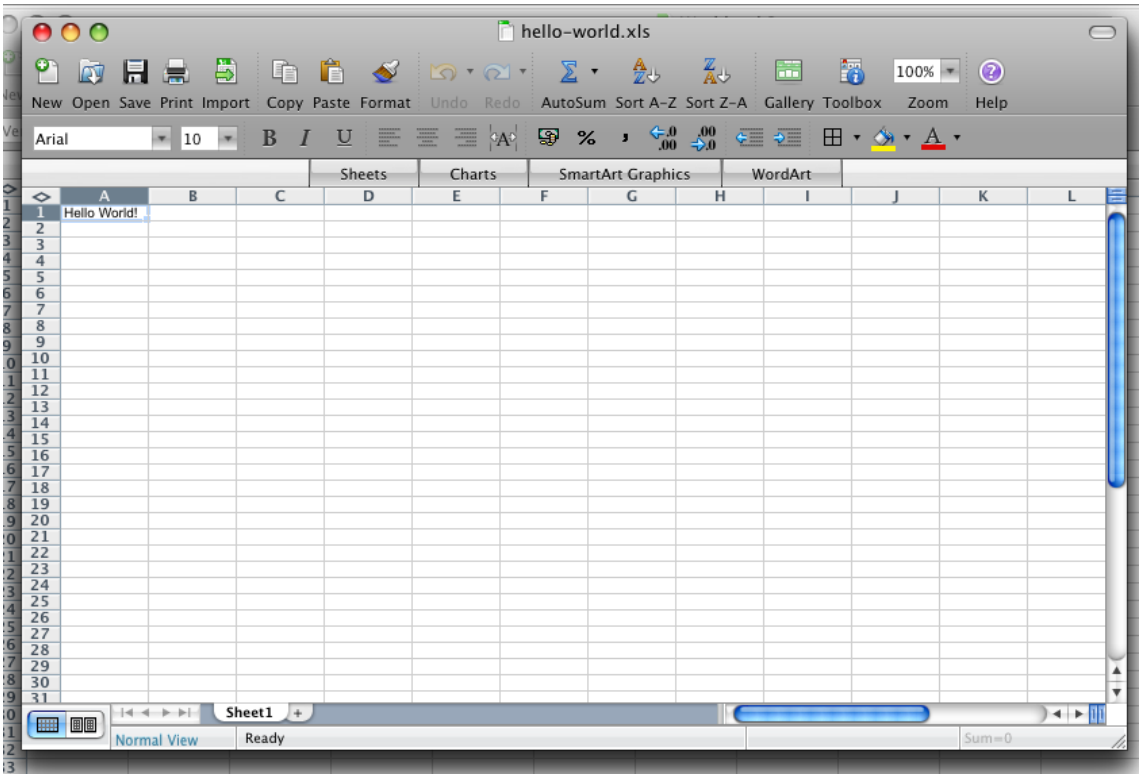
```
require 'rubygems'
require 'surpass'

book = Workbook.new
sheet = book.add_sheet

sheet.write(0, 0, "Hello World!")

book.save("output/examples/hello-world.xls")

```

1.1.2 XLWT

```
from xlwt import *

book = Workbook()
sheet = book.add_sheet("Sheet1")

sheet.write(0, 0, "Hello World!")

book.save("output/examples/hello-world-python.xls")

```

1.1.3 Result

And, here's how it looks.


hello-world.xls
New
Open
Save
Print
Import
Copy
Paste
Format
Undo
Redo
AutoSum
Sort A–Z
Sort Z–A
Gallery
Toolbox
100%
Zoom
Help
Arial
10
Sheets
Charts
SmartArt Graphics
WordArt
Hello World!
Sheet1
Normal View
Ready
Sum=0

Chapter 2

Writing Data

Chapter 3

Formatting

3.1 Reference

There are some rake tasks included with Surpass which provide some useful reference data.

```
rake -T | grep excel

rake excel:colors          # list all available colours
```

And since you are running these on a command line, you can save them or pipe them to other commands, as usual.

```
rake excel:colors | grep green

bright_green
dark_green
green
light_green
olive_green
sea_green
```

3.2 Formatting

The StyleFormat class is a wrapper for the various types of formatting you can apply to a cell. StyleFormat has attributes:

- number_format_string
- font
- alignment
- borders
- pattern
- protection

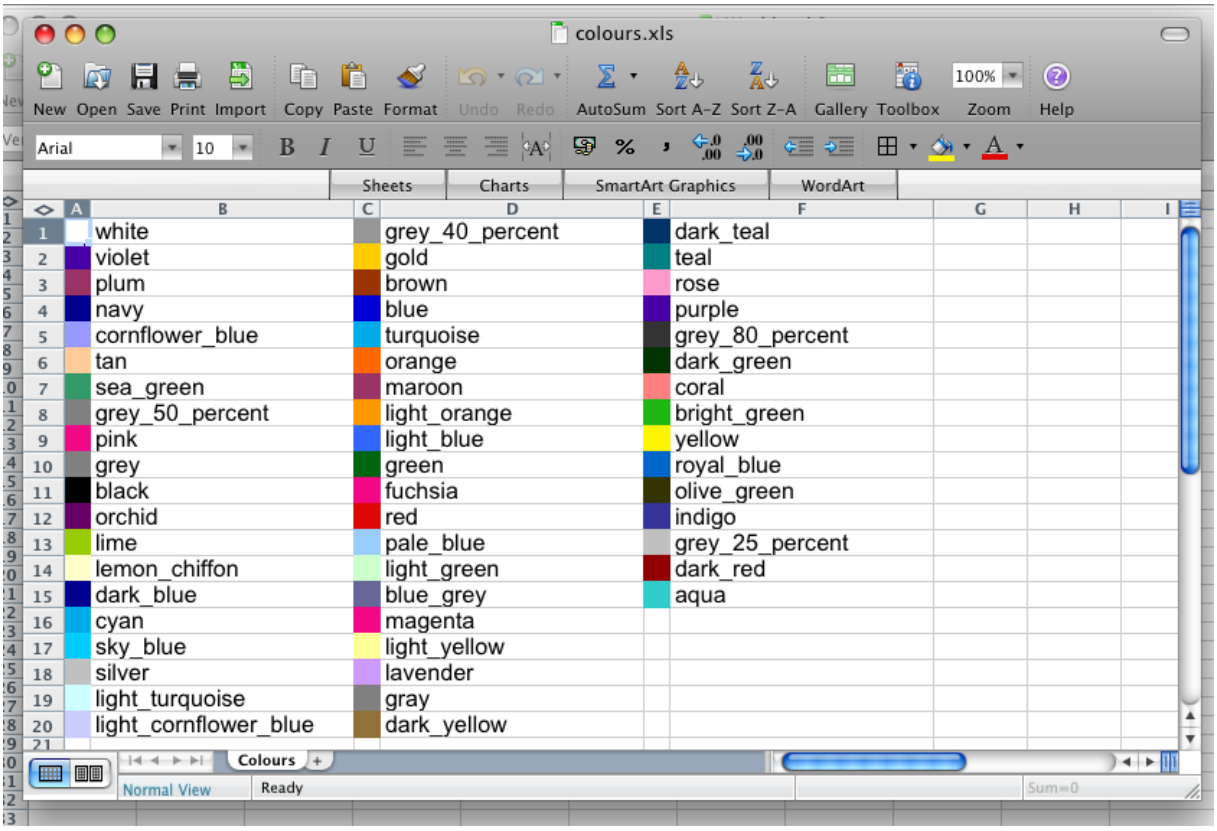
Each of these attributes has a corresponding class, and you can look in lib/formatting.rb for the source.

There are two basic ways to set formatting options. You can pass a hash with formatting options when you initialize a new StyleFormat instance, or you can set individual attributes of the formatting classes. You can combine both approaches. Both of these are demonstrated in the examples in this section.

3.2.1 Specifying Colours

Here is a list of available colours:

```
aqua                    grey_25_percent         orchid
black                   grey_40_percent         pale_blue
blue                    grey_50_percent         pink
blue_grey               grey_80_percent         plum
bright_green            indigo                  purple
brown                   lavender                red
coral                   lemon_chiffon           rose
cornflower_blue         light_blue              royal_blue
cyan                    light_cornflower_blue   sea_green
dark_blue               light_green             silver
dark_green              light_orange            sky_blue
dark_red                light_turquoise         tan
dark_teal               light_yellow            teal
dark_yellow             lime                    turquoise
fuchsia                 magenta                 violet
gold                    maroon                  white
gray                    navy                    yellow
green                   olive_green
grey                    orange
```



You can refer to any of these colours by name.

3.2.2 Border Formats

Here is a list of available border line types:

```
none
```

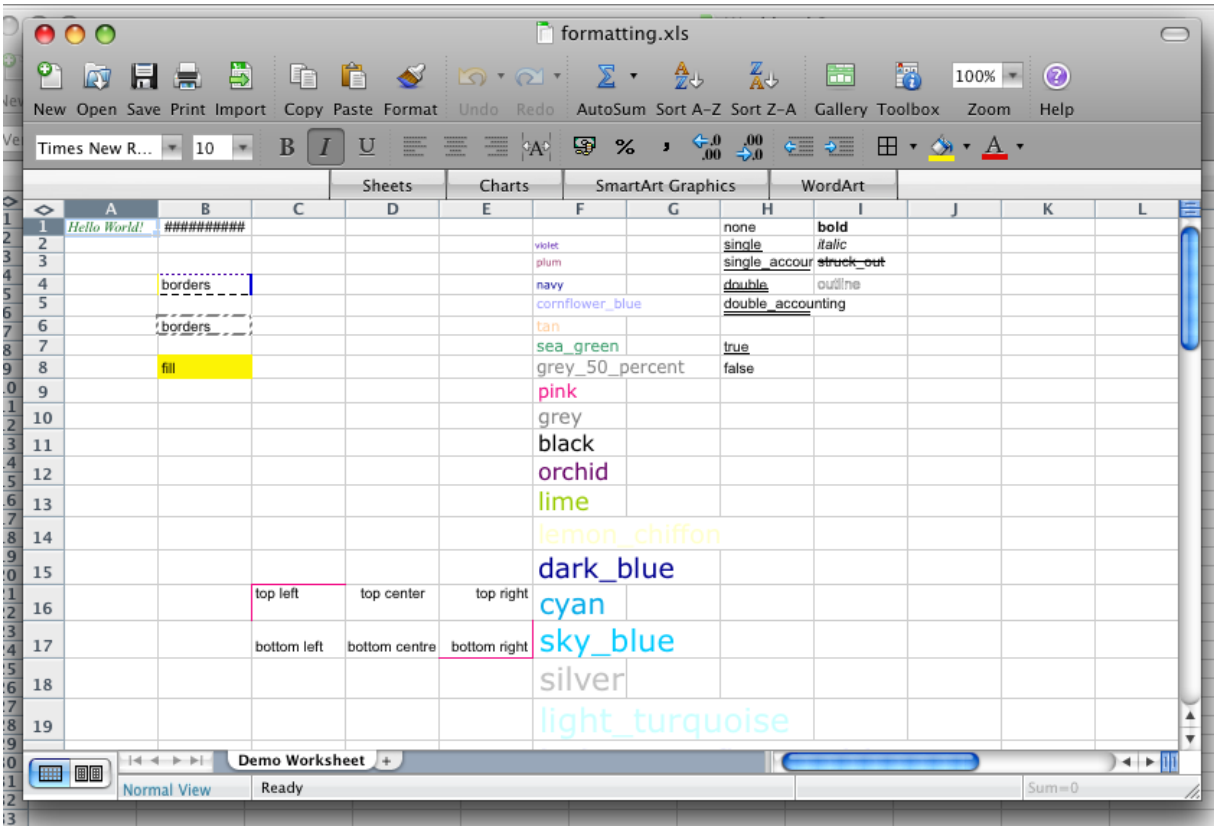
```
thin
medium
dashed
dotted
thick
double
hair
medium-dashed
thin-dash-dotted
medium-dash-dotted
thin-dash-dot-dotted
medium-dash-dot-dotted
slanted-medium-dash-dotted
```

3.2.3 Surpass

```ruby
require 'rubygems'
require 'surpass'

book = Workbook.new('output/examples/formatting.xls') # You can pass a filename here too.
sheet = book.add_sheet("Demo Worksheet") # You can name your worksheets.

# Let's set up some formatting.

# Remember to use Excel-style formatting directives, not sprintf.
date_format = StyleFormat.new(:number_format_string => "DDDD DD MMM YYYY")

fancy_format = StyleFormat.new(
  :font_name => 'Times New Roman',
  :font_colour => 'green',
  :font_italic => true
)

sheet.write(0, 0, "Hello World!", fancy_format)
sheet.write(0, 1, Date.today, date_format)

# You can also set up formatting by passing attributes directly to the constituents of StyleFormat

# Font colours.
Formatting::COLOURS.keys.each_with_index do |c, i|
  format = StyleFormat.new
  format.font.name = 'Verdana'
  format.font.color = c
  format.font.size = i + 5
  sheet.write(i, 5, c, format)
end

# Font underlining.

[:none, :single, :single_accounting, :double, :double_accounting, nil, true, false].each_with_index
  format = StyleFormat.new
  format.font.underline = u
  sheet.write(i, 7, u.to_s, format)
end

# Font bold, italic, strikethrough, outline are simple booleans.
```

```ruby
[:bold, :italic, :struck_out, :outline].each_with_index do |s, i|
  attribute = ("font_" + s.to_s).to_sym
  sheet.write(i, 8, s.to_s, StyleFormat.new(attribute => true))
end

# Cell alignment.
sheet.write(15, 2, "top left", :text_align => 'top left',
  :border_top => 'pink',
  :border_left => 'pink'
)
sheet.write(15, 3, "top center", :text_align => 'top center')
sheet.write(15, 4, "top right", :text_align => 'top right')
sheet.write(16, 2, "bottom left", :text_align => 'bottom left')
sheet.write(16, 3, "bottom centre", :text_align => 'bottom centre')
sheet.write(16, 4, "bottom right", :text_align => 'bottom right',
  :border_bottom => 'pink',
  :border_right => 'pink'
)


# Borders
sheet.write(3, 1, "borders",
  :border_right => 'medium blue',
  :border_left => 'yellow', # thin by default
  :border_top => 'dotted purple',
  :border_bottom => 'dashed' # black by default
)

# Or the hash-free option.
crazy_border_format = StyleFormat.new
crazy_border_format.borders.all = 'slanted-medium-dash-dotted grey'

sheet.write(5, 1, "borders", crazy_border_format)

sheet.write(7, 1, "fill", :fill_color => 'yellow')

book.save

```

And, here's how it looks.


formatting.xls
New Open Save Print Import Copy Paste Format Undo Redo AutoSum Sort A-Z Sort Z-A Gallery Toolbox Zoom Help
Times New R... 10
Sheets Charts SmartArt Graphics WordArt
Hello World!
##########
borders
borders
fill
top left
top center
top right
bottom left
bottom centre
bottom right
violet
plum
navy
cornflower_blue
tan
sea_green
grey_50_percent
pink
grey
black
orchid
lime
lemon_chiffon
dark_blue
cyan
sky_blue
silver
light_turquoise
none
single
single_accour
double
double_accounting
bold
italic
struck_out
outline
true
false
Demo Worksheet
Normal View
Ready
Sum=0