Surpass and XLWT Manual

Ana Nelson

March 18, 2009

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 Surpass

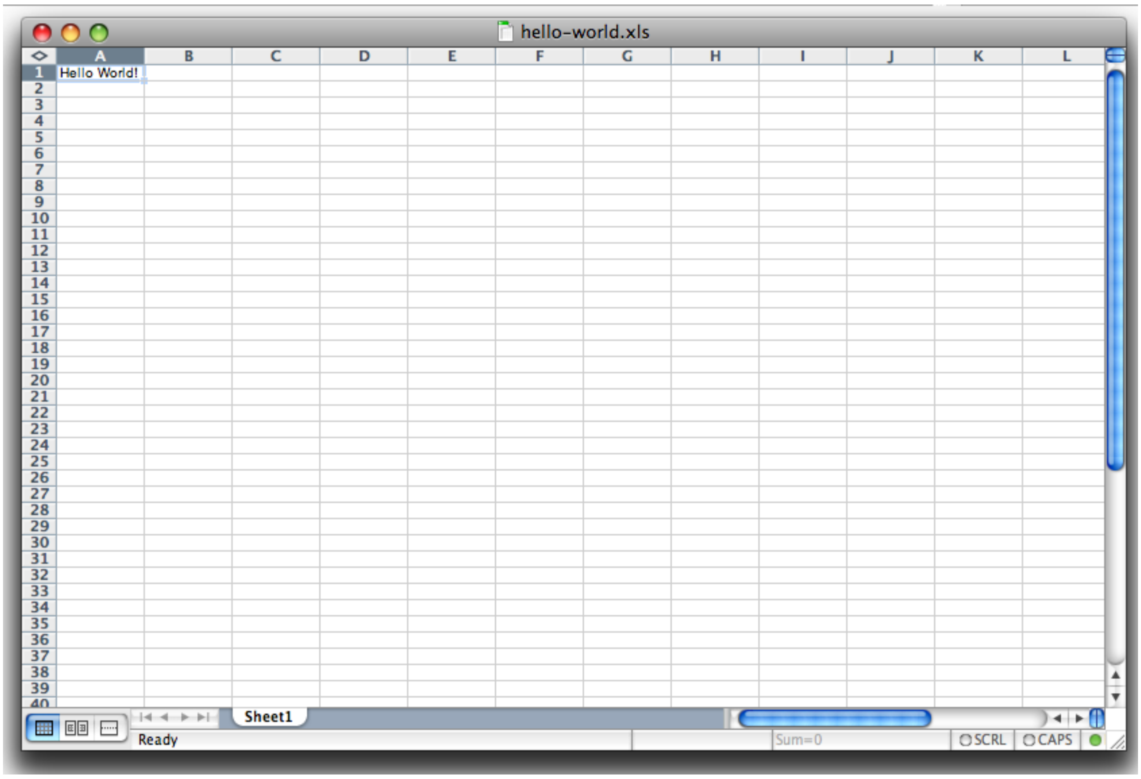
```
require 'rubygems'
require 'surpass'

book = Workbook.new
sheet = book.add_sheet

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

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

1.2 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")
```

1.3 Result

And, here's how it looks.



Hello World!

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 next section.

- none
- thin
- medium
- dashed
- dotted
- thick

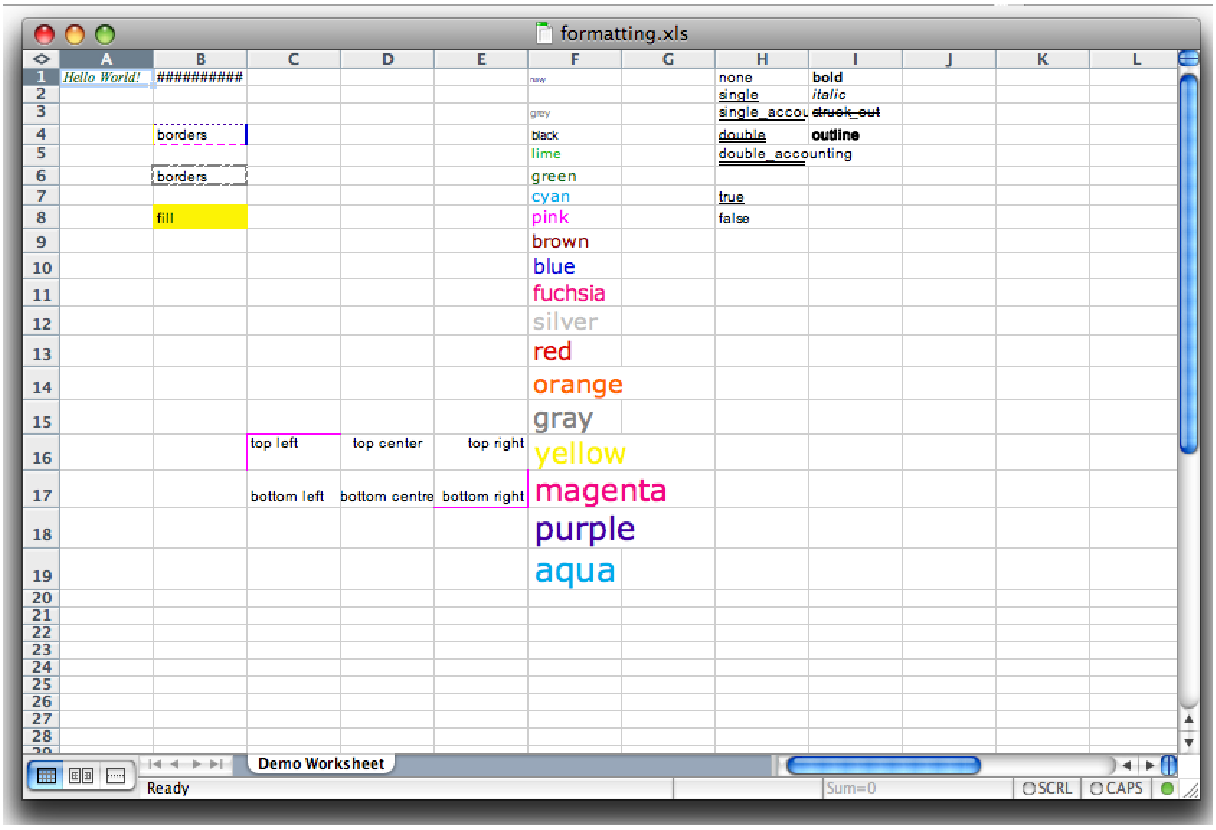
- double
- hair
- medium-dashed
- thin-dash-dotted
- medium-dash-dotted
- thin-dash-dot-dotted
- medium-dash-dot-dotted
- slanted-medium-dash-dotted

2.1 Surpass

```ruby
# require ’rubygems’
# require ’surpass’
require "../lib/surpass"

book = Workbook.new(__FILE__.gsub(/rb$/, ’xls’).gsub(’content/’, ’output/’)) # You can pass a filena
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

```

```ruby
# Font bold, italic, strikethrough, outline are simple booleans.
[: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 => 'thin yellow',
  :border_top => 'dotted purple',
  :border_bottom => 'dashed pink'
)

# 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

	A	B	C	D	E	F	G	H	I	J	K	L
1	*Hello World!*	###########				navy		none	**bold**			
2								single	*italic*			
3						grey		single_accou	~~struck_out~~			
4		borders				black		double	**outline**			
5						lime		double_accounting				
6		borders				green						
7						cyan		true				
8		fill				pink		false				
9						brown						
10						blue						
11						fuchsia						
12						silver						
13						red						
14						orange						
15						gray						
16			top left	top center	top right	yellow						
17			bottom left	bottom centre	bottom right	magenta						
18						purple						
19						aqua						
20												
21												
22												
23												
24												
25												
26												
27												
28												