

iText とか Inline::Java とか

iText は、Java で PDF を操作するためのライブラリ。

<http://www.lowagie.com/iText/>

ロゴが「で Text」に見えてしまう ...

で、ふと perl から使えたら便利かなと、

Inline::Java で Perl から Java の日付関連クラスを使用するを参考に (といつかそのまま)

iText を Inline::Java つかって perl から使ってみた。動いた。

```
#!/usr/bin/perl
use strict;

use Inline

    Java => 'STUDY',
    STUDY => [qw(
        java.io.FileOutputStream
        java.io.IOException
        com.lowagie.text.Document
        com.lowagie.text.DocumentException
        com.lowagie.text.Paragraph
        com.lowagie.text.pdf.PdfWriter
    )];

my $doc = com::lowagie::text::Document->new();
my $fos = java::io::FileOutputStream->new("test.pdf");
my $writer = com::lowagie::text::pdf::PdfWriter->getInstance($doc, $fos);
$doc->open();
$doc->add(com::lowagie::text::Paragraph->new("Hello World"));
$doc->close();
```

とここまでやってから、perl には、PDFJ があるよなあと思い出したり。

しかし、Inline::Java って便利だな。Perl 書けないけど。

これで COINS とかいじると、今より楽に実験できるようになるかな？

CPAN の設定をやりなおす

最初から設定をやり直したいときは、cpan のシェルを起動して

```
o conf init
```

とする

JRuby と iText

JRuby が 1.0.0RC1 になった。

というわけで、iText を JRuby で。

```
include Java

require 'java'

import java.io.FileOutputStream
import java.io.IOException
import com.lowagie.text.Document
import com.lowagie.text.DocumentException
import com.lowagie.text.Paragraph
import com.lowagie.text.pdf.PdfWriter

begin
  doc = Document.new();
  fos = FileOutputStream.new("test.pdf");
  writer = PdfWriter.getInstance(doc, fos);
```

```
doc.open();
doc.add(Paragraph.new("Hello World"));
doc.close();
rescue DocumentException => e
  puts e
rescue IOError => e
  puts e
end
```

これは、便利 !!