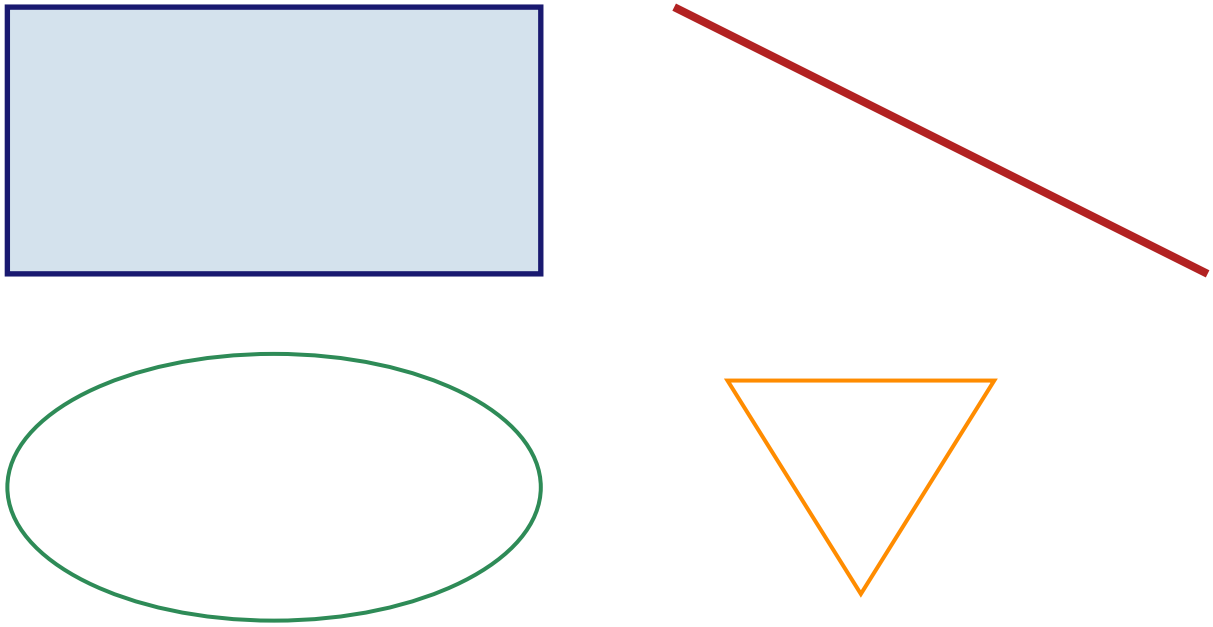
The page being read

A deliberately short page, so that the operators it produces fit on the next one and can be read against the calls that made them. Six calls: a string, a rectangle with a pen and a brush, a line, an ellipse, a path and a second string in another colour.



Six calls, and the operators overleaf

What the page is made of

The previous page came back as 52 operators. The first sixty are below, each with its operands, in the order they were written. An operator's name is the PDF one rather than the drawing call's - re for a rectangle, S to stroke, f to fill, B to do both, Tj to show text.

```
q
q
BT
rg    0 0 0
gs    /GS0
Tf    /F0 16
Td    50 782
Tj    (38 bytes)
Tf    /F1 9
Td    0 -28.148
Tj    (248 bytes)
Td    0 -10.349
Tj    (256 bytes)
ET
w     2
J     0
j     0
d     [0 items] 0
RG    0.098 0.098 0.439
gs    /GS1
rg    0.275 0.51 0.706
gs    /GS2
re    50 592 200 100
B
w     3
RG    0.698 0.133 0.133
m     300 692
l     500 592
S
w     1.5
RG    0.18 0.545 0.341
m     250 512
c     250 484.386 205.229 462 150 462
c     94.772 462 50 484.386 50 512
c     50 539.614 94.772 562 150 562
c     205.229 562 250 539.614 250 512
h
S
RG    1 0.549 0
m     320 552
l     420 552
l     370 472
h
S
BT
rg    0.698 0.133 0.133
gs    /GS0
Td    50 432
Tj    (74 bytes)
ET
Q
Q
```

By operator

The same content counted rather than listed, which is usually the more useful view: it says at a glance whether a page is mostly text, mostly paths or mostly graphics state, and a page that is unexpectedly large usually says so here.

4	`c`	curve to
4	`gs`	apply an extended graphics state
4	`RG`	set a stroking colour
4	`Td`	move the text position
4	`Tj`	show text
3	`l`	line to
3	`m`	move to
3	`rg`	set a non-stroking colour
3	`S`	stroke
3	`w`	set the line width
2	`BT`	begin text
2	`ET`	end text
2	`h`	close the figure
2	`q`	save the graphics state
2	`Q`	restore it
2	`Tf`	set font and size
1	`B`	fill and stroke
1	`d`	set the dash
1	`j`	set the line join
1	`J`	set the line cap
1	`re`	add a rectangle to the path

Why the text is not readable

The operands of a Tj are shown above as a byte count rather than as words, and that is not the reader being coy. Fonts here are embedded as Identity-H by default, so a show-text operator carries two-byte glyph identifiers rather than characters - the face's own numbering, which means nothing without the font's tables. Setting XPdfFontOptions.WinAnsiDefault writes readable string literals instead, at the cost of the characters WinAnsi cannot represent. The Unicode demo is where that trade is laid out.

Reading content back is how the test suite checks what the renderer wrote without rasterizing anything - four of its helpers do exactly this. It is also the quickest answer to "why is my page blank": a page whose operators are there but whose coordinates are wrong looks identical to one that drew nothing, and only one of the two says so here.