

1 Introduction

(1) The original technique to execute an arbitrary command:

```
\immediate\write18{node
  "\coffeexelatexroute"
  "\currfilepath"
  "helo"
  "readers (one)"
  > /tmp/temp.dat}\input{/tmp/temp.dat}
```

(2) With ugly details largely hidden, the `\exec{}` command is still fully general:

```
\exec{node
  "\coffeexelatexroute"
  "\currfilepath"
  "helo"
  "readers (two)"}
```

(3) `\noderunscript{}` will execute NodeJS code that adheres to the call convention established by *CoffeeX_qL^AT_EX*:

```
\noderunscript
  {\coffeexelatexroute}
  {\currfilepath}
  {helo}
  {readers (three)}
```

(4) Like the previous example, but with standard values assumed as shown above. This is the form that you will want to use most of the time:

```
\noderun{helo}{readers (four)}
```

Outputs:

```
Hello, readers (one)!
Hello, readers (two)!
Hello, readers (three)!
Hello, readers (four)!
```

2 Geometry

Use geometry data from aux file to render a table of layout dimensions into the document; note the we could have used the `\auxgeo` command anywhere in the document and that this currently only works for documents with a single, constant layout.

Also note we're using a dash instead of an underscore here—in \TeX , underscores are special, so we conveniently allow dashes to make things easier. The *CoffeeX_g* \LaTeX command `show-geometry` does not take arguments, which is why the second pair of braces has been left empty:

Command:

```
\noderun{show-geometry}{}{}
```

Output:

firstlinev	10.03 mm
footskip	10.58 mm
headheight	4.23 mm
headsep	8.82 mm
marginparsep	3.88 mm
marginparwidth	1.06 mm
paperheight	210.71 mm
paperwidth	148.50 mm
textheight	175.39 mm
textwidth	103.95 mm
topmargin	-28.50 mm
voffset	25.49 mm

3 Character Escaping

The *CoffeeX*_{TeX} command `show-special-chrs` demonstrates that it is easy to include TeX special characters in the return value. The simple rule is that whenever the output of a command is meant to be understood literally, it should be `@escaped`:

Command:

`\noderun{show-special-chrs}{}`

Output:

opening brace	{
closing brace	}
Dollar sign	\$
ampersand	&
hash	#
caret	^
underscore	_
wave	~
percent sign	%