

# Use custom fonts in a PDF file compiled from Rmarkdown

## 1. Run capabilities() in Console.

If **cairo** shows up in the output as TRUE, you are ready to proceed!

```
> capabilities()
      jpeg      png      tiff      tcltk
      TRUE      TRUE      TRUE      TRUE

      NLS      Rprof      profmem      cairo
      TRUE      TRUE      TRUE      TRUE
```

## 2. In YAML, add the following:

```
header-includes:
  - \usepackage[default]{sourcesanspro}
  - \usepackage[T1]{fontenc}
mainfont: sourcesanspro
```

Does not work if you omit header-includes:.

In this example, sourcesanspro font is used.

## 3. Notes

*All these three formats work!*

```
mainfont: sourcesanspro
mainfont: SourceSansPro
mainfont: Source Sans Pro
```

There is no need to set a *latex\_engine*, as the RStudio's (Sweave) default, *pdfLaTeX* is used.

It is important to place the output argument at the end of YAML if this is the **first time** you are attempting to test particular options for a resulting PDF file.

```
output: pdf_document
```

By doing so, *tinytex* in RStudio will attempt to install any additional LaTeX packages needed for a particular Rmarkdown file before compiling as a PDF document.

Once this is successful, you may move the output: pdf\_document before the header-includestop: arguments. This is to make the resulting document immediately apparent. *Now, you are ready to customise your font!*

## References

1. See [Cecina Babich Morrow's page] <https://babichmorrowc.github.io/post/changing-fonts-in-rmarkdown-pdfs/>
2. For a discussion on `fontenc` and the usage of pdfLaTeX, see [A discussion on fontenc], <https://tex.stackexchange.com/questions/664/why-should-i-use-usepackage1fontenc>