

# Rmarkdown Basics

Eric Odongo

2025-04-12

## Introduction

Rmarkdown uses markdown language to help researchers or anyone really to mix their code and prose to form a unifying report. The report can be in multiple formats like word, html, pdf or even slides. For pdf reports you will need latex installed on your system. The easiest way of installing latex is via using the `tinytex` package by Yihui Xie the creator of `knitr` package.

Simple type the below code in the console

```
# install.packages("tinytex")
# tinytex::install_tinytex()

#if you want to install a specific latex package that type the below code in the console

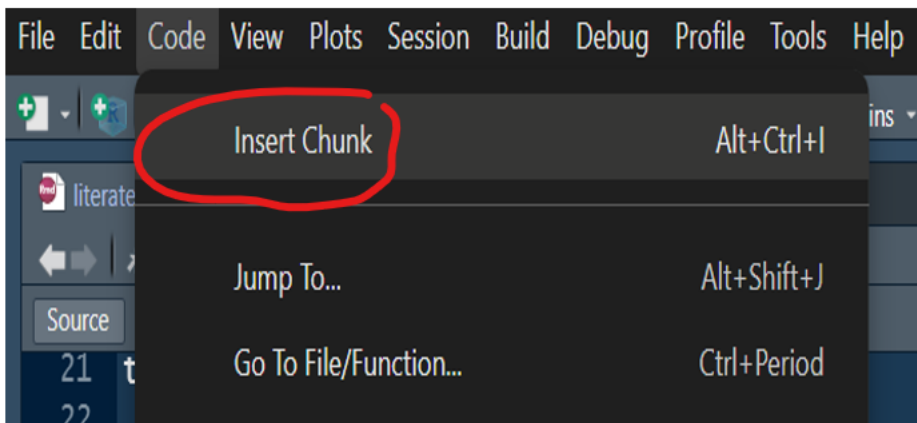
#tinytex::tlmgr_install("grfext") # to install grfext package
tinytex::tlmgr_install("tabularray.sty")

tinytex::reinstall_tinytex(repository = "illinois")
```

Titles, sections and subsections all start with a `#` and the depth level is determined by the number of `#`s. For bold text simply surround your text with double asterisks like this **some bold text** and the italics surround your text with single asterisks like this *this is some italicized text*. Underline text using pandoc (the program that converts Rmd files into final output) syntax as shown below;

This is some underlined text in a R Markdown document

To use R code inside your document we use code chunks. This can be inserted using the keyboard shortcut `Ctrl + Alt + I` or using the menu bar



Each chunk can have a label, though the label is optional. The chunk also contains various optional settings that determine what you want to do with the chunk, for example do you want the code to be displayed in your report or only the output. To use this feature just specify `echo = FALSE` or `echo = TRUE` which is the default by the way.

The chunk below will not display the R code in the report, only the output will be displayed.

```
## [1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
```

There are several settings that you can explore on your own as an exercise.

To use inline code chunk write your code between two back ticks as in this example; 32 .

To write a footnote use the following syntax;

This sentence has a footnote.<sup>1</sup>

## Tables in R markdown documents

```
library(flextable)
```

```
## Warning: package 'flextable' was built under R version 4.4.3
```

```
my_table <- head(mtcars)
```

```
flextable(my_table) |>  
  set_caption(caption = "Head of mtcars dataset") |>  
  theme_booktabs()
```

```
## Warning: fonts used in 'flextable' are ignored because the 'pdflatex' engine is  
## used and not 'xelatex' or 'lualatex'. You can avoid this warning by using the  
## 'set_flextable_defaults(fonts_ignore=TRUE)' command or use a compatible engine  
## by defining 'latex_engine: xelatex' in the YAML header of the R Markdown  
## document.
```

Table 1: Head of mtcars dataset

mpg	cyl	disp	hp	drat	wt	qsec	vs	am
21.0	6	160	110	3.90	2.620	16.46	0	1
21.0	6	160	110	3.90	2.875	17.02	0	1
22.8	4	108	93	3.85	2.320	18.61	1	1
21.4	6	258	110	3.08	3.215	19.44	1	0
18.7	8	360	175	3.15	3.440	17.02	0	0
18.1	6	225	105	2.76	3.460	20.22	1	0

---

<sup>1</sup>This is the footnote

	Model 1	Model 2
(Intercept)	26.585 (1.425)	30.099 (1.634)
hp	-0.059 (0.008)	-0.068 (0.010)
am	5.277 (1.080)	
Num.Obs.	32	32
R2	0.782	0.602
R2 Adj.	0.767	0.589
AIC	164.0	181.2
BIC	169.9	185.6
Log.Lik.	-78.003	-87.619
F	52.024	45.460
RMSE	2.77	3.74

```
library(modelsummary) # good for regression and summary tables
```

```
## Warning: package 'modelsummary' was built under R version 4.4.3
```

```
model_1 <- lm(mpg ~ hp + am, data = mtcars)
model_2 <- lm(mpg ~ hp, data = mtcars)
models <- list("Model 1" = model_1, "Model 2" = model_2)

modelsummary(models)
```